MUSICHE LIQUIDE

LIQUID MUSIC



XX COLLOQUIO DI INFORMATICA MUSICALE
20TH COLLOQUIUM ON MUSIC INFORMATICS

Conservatorio Santa Cecilia
Roma, 20–22 ottobre 2014
Rome, October 20-22, 2014

AA.VV.

MUSICHE LIQUIDE / *LIQUID MUSIC*
XX Colloquio di Informatica Musicale
20th Colloquium on Music Informatics

A cura di:
Editors:
Michele Geronazzo, Simone Spagnol

© 2014 AIMI – Associazione Informatica Musicale Italiana
I diritti dei singoli articoli rimangono agli autori
The copyrights of the single articles remain with the authors

ISBN 9788890341335
Per gentile collaborazione di Davide Rocchesso e Stefano Delle Monache
Courtesy of Davide Rocchesso and Stefano Delle Monache

Pubblicato da:
Publisher:
Università IUAV di Venezia

In co-edizione con:
Co-published with:
Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Padova

Immagini di copertina:
Cover images:
"Disastro Ecologico" by Alberto Seveso, licensed under CC BY-NC-ND 3.0
http://creativecommons.org/licenses/by-nc-nd/3.0/



Quest'opera è sotto licenza Creative Commons Attribuzione - Non opere derivate 4.0 Unported License.
Copia della licenza è disponibile presso http://creativecommons.org/licenses/by-nd/4.0/.
This work is licensed under the Creative Commons Attribution-NoDerivs 4.0 Unported License.
To view a copy of this license, visit http://creativecommons.org/licenses/by-nd/4.0/.

XX Colloquio di Informatica Musicale

20th Colloquium on Music Informatics

MUSICHE LIQUIDE

LIQUID MUSIC

Roma, 20–22 Ottobre 2014

Rome, October 20-22, 2014

AIMI

Conservatorio Santa Cecilia

In collaborazione con / *In collaboration with*:

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
Università degli Studi di Padova

UNIVERSITA' degli STUDI di ROMA TOR VERGATA

SAPIENZA UNIVERSITÀ DI ROMA

AUDIO ES Audio Engineering Society *Italian Section*

CEMAT Ente di Promozione Music Art Technology

Centro Ricerche Musicali

FONDAZIONE ISABELLA SCELSI

RADIO CEMAT

MINISTERO DELL'ISTRUZIONE, DELL'UNIVERSITÀ E DELLA RICERCA

ROMA CAPITALE

AIMI – Associazione di Informatica Musicale Italiana
www.aimi-musica.org

Conservatorio di Musica "S.Cecilia", Roma
Dipartimento di Nuove Tecnologie e Linguaggi Musicali
www.conservatoriosantacecilia.it

In collaborazione con / *In collaboration with*

Università degli Studi di Padova
Dipartimento di Ingegneria dell'Informazione
www.dei.unipd.it

Università di Roma Tor Vergata
Facoltà di Ingegneria
www.ing.uniroma2.it

Sapienza Università di Roma
Dipartimento di Ingegneria dell'Informazione, Elettronica e Telecomunicazioni
web.uniroma1.it/dip_diet/

Audio Engineering Society Italia
www.aesitalia.org

Federazione CEMAT
www.federazionecemat.it

Centro Ricerche Musicali
www.crm-music.it

Fondazione Isabella Scelsi
www.scelsi.it

Con il patrocinio di / *Endorser*

Ministero dell'Istruzione, dell'Università e della Ricerca

Comune di Roma

Media partner

Radio CEMAT
http://www.radiocemat.org

XX COLLOQUIO DI INFORMATICA MUSICALE

L'Associazione di Informatica Musicale Italiana (AIMI), fondata nel 1981, è punto di riferimento nel panorama italiano per musicisti e ricercatori interessati al rapporto tra musica, scienza e tecnologia. L'AIMI, che vanta tra i suoi associati musicisti e ricercatori di rilevanza internazionale, patrocina con cadenza biennale i Colloqui di Informatica Musicale (CIM), che rappresentano il più atteso momento di discussione scientifica e di confronto musicale della comunità italiana di informatica musicale.

Il Colloquio di Informatica Musicale costituisce il punto d'incontro tra gli scienziati, gli artisti e i musicologi che si dedicano secondo le rispettive metodologie all'interazione musica-tecnologia e ha come scopo la divulgazione delle loro ricerche sulle applicazioni artistico-musicali delle tecnologie informatiche, nei risvolti accademici, artistici, culturali e pedagogici.

Come di consueto, il programma del Colloquio include comunicazioni scientifiche, attività didattiche e produzioni artistiche. Le comunicazioni scientifiche/artistiche sono state selezionate mediante una Call for papers e i lavori musicali e artistici mediante una Call for scores/works; tutte le proposte sono state vagliate dal Comitato scientifico e selezione musicale.

MUSICHE LIQUIDE

Nel 2014 il Colloquio di Informatica Musicale giunge alla ventesima edizione. Questo traguardo rende ineludibile uno sguardo al passato e una riflessione sulle mutazioni epocali in atto. L'uscita dalla società industriale, l'entrata nell'era dell'informazione descritta così minuziosamente da Manuel Castells, la nuova fase della modernità che Zygmunt Bauman ha caratterizzato con l'aggettivo "liquida" (contrapponendola cosi` a quella "solida", sistemica e tangibile), non può lasciare impassibile la creazione musicale e in particolar modo quella creazione che si lega indissolubilmente alla tecnologia e all'innovazione strumentale. L'informatica musicale è attore affatto secondario in questo contesto, e questo XX Colloquio di Informatica Musicale intende così costituirsi come momento di riflessione sulle mutazioni in corso.

20th COLLOQUIUM ON MUSIC INFORMATICS

The Italian Association of Music Informatics (AIMI), founded in 1981, is a reference point for Italian musicians and researchers who are interested in the relationships between music, science and technology. The AIMI, which counts internationally renowned musicians and researchers among its affiliates, organizes every two years the Colloquium on Music Informatics, representing the most important moment of scientific discussion and musical debate in the Italian community of music informatics.

The Colloquium on Music Informatics is a touch point for scientists, artists and musicologists who dedicate their methodologies to the interaction between music and technology, and serves the dissemination of their research on the artistic-musical applications of information technology in all its academic, artistic, cultural and pedagogical facets.

As it is customary, the program of the Colloquium includes scientific communications, seminal activities and artistic productions. The scientific/artistic communications have been selected trough a Call for papers, and the musical and artistic works through a parallel Call for scores/works; all submissions have been reviewed by the Scientific and Music Committee.

LIQUID MUSIC

The Colloquium on Music Informatics reaches in 2014 its twentieth edition. This landmark strongly suggests a critical look at our past and a reflection over the far-reaching changes which are currently taking place in our society. The end of the industrial society, the beginning of the information era so accurately described by Manuel Castells, the new phase of modernity characterized by Zygmunt Bauman with the adjective "liquid" (in contrast with the "solid" one, so tangible and systemic), cannot leave music creation indifferent, in particular when it is strongly connected to technology and instrumental innovation. As a matter of fact, music creation has brought a fundamental contribution to the transition towards our information society. Thus, computer music (a discipline now known as "Sound and Music Computing") is one of the leading actors in this context, and this twentieth Colloquium intends to be an opportunity of reflection on its current developments.

COMITATO SCIENTIFICO E DI SELEZIONE MUSICALE
SCIENTIFIC AND MUSIC PROGRAM COMMITTEE

Andrea Agostini, HES-SO Genève – Conservatorio di Cuneo
Federico Avanzini, Università di Padova
Stefano Baldan, IUAV University of Venice
Stefano Bassanese, Conservatorio di Torino
Nicola Bernardini, Conservatorio S.Cecilia Roma
Laura Bianchini, Centro Ricerche Musicali Roma
Roberto Bresin, KTH Royal Institute of Technology
Nicola Buso, Conservatorio di Trieste
Sergio Canazza, Università di Padova
Luigi Ceccarelli, Conservatorio di Perugia
Giovanni Cospito, Conservatorio G.Verdi Milano
Giovanni Costantini, Università di Roma Tor Vergata
Riccardo Dapelo, Conservatorio di Genova
Maria Cristina De Amicis, Conservatorio Alfredo Casella l'Aquila
Amalia de Gotzen, Aalborg University in Copenhagen
Giovanni De Poli, Università di Padova
Stefano Delle Monache, IUAV University of Venice
Agostino Di Scipio, Conservatorio Alfredo Casella l'Aquila
Roberto Doati, Conservatorio di Genova
Carlo Drioli, Università di Udine
Federico Fontana, Università di Udine
Francesco Galante, Conservatorio di Cosenza
Michele Geronazzo, Università di Padova
Daniele Ghisi, HES-SO Genève – Conservatorio di Torino
Marco Giordano, Conservatorio Alfredo Casella l'Aquila
Giorgio Klauer, Conservatorio di Padova
Silvia Lanzalone, Conservatorio di Salerno
Enrico Marchetto, Institut de Recherche et Création Acoustique/Musique
Davide Andrea Mauro, IUAV University of Venice
Marco Momi, Conservatorio di Sassari
Giorgio Nottoli, Università di Roma Tor Vergata
Paolo Pachini, Conservatorio di Trieste
Lorenzo Pagliei, Conservatorio di Vicenza – Institut de Recherche et Création Acoustique/Musique
Stefano Papetti, Zürcher Hochschule der Künste
Alessandra Carlotta Pellegrini, Fondazione Isabella Scelsi Roma
Pietro Polotti, Conservatorio di Trieste
Davide Rocchesso, IUAV University of Venice
Antonio Rodà, Università di Padova
Sylviane Sapir, Conservatorio di Como
Franco Sbacco, Conservatorio S.Cecilia Roma
Federico Scalas, Conservatorio S.Cecilia Roma
Domenico Sciajno, Conservatorio di Torino
Stefania Serafin, Aalborg University in Copenhagen
Simone Spagnol, Università di Padova
Gianni Trovalusci, Federazione CEMAT Roma
Aurelio Uncini, Sapienza Università di Roma
Andrea Valle, Università di Torino
Gianluca Verlingieri, Conservatorio di Cuneo
Alvise Vidolin, Università di Padova
Gualtiero Volpe, Università di Genova
Laura Zattra, Università di Padova

Coordinamento generale
General chairs
Michelangelo Lupone, Nicola Bernardini

Coordinamento programma musicale
Music program chairs
Stefano Bassanese, Laura Bianchini, Fabio Cifariello Ciardi

Coordinamento programma scientifico
Scientific program chair
Federico Avanzini

Curatori degli atti
Editors
Michele Geronazzo, Simone Spagnol

Staff tecnico
Technical staff
Paolo Gatti, Luana Lunetta, Massimo Massimi, Giuseppe Silvi, Anna Terzaroli

Programma musicale

Musical program

Concerto 1: Trigger(ed)

Curatrice: Laura Bianchini

SILVIA LANZALONE – Èleghos

per flauto aumentato, tubo risonante ed elettronica — flauto – Gianni Trovalusci

MARCO FERRAZZA – Radiale

acusmatico

AGOSTINO DI SCIPIO – Studio sul rumore di fondo, nel tratto vocale (Ecosistemico udibile n.3b)

per tre voci femminili — voci – Concetta Cucchiarelli, Virginia Guidi e Angelina Yershova

FEED-BAND (V. De Luca, F. Quercia, S. Disanto, A. Scarcia, G. Scarola, F. Scagliola)

A Sad Song: Feedback, indeterminazione e forma aperta di una performance

ALBA FRANCESCA BATTISTA, MATTEO NICOLETTI, GIORGIA BONINI e VITTORIO CASTELNUOVO

A contemporary Dawn: a wor(l)d soundscape

installazione elettroacustica interattiva

Concerto 2: Inside out – Outside in

Curatore: Fabio Cifariello Ciardi

NICOLA CASETTA – Piano Simulacrum

acusmatico

JULIAN SCORDATO – Constellations

audiovisual

LUCA RICHELLI – Keep going again

acusmatico

DANIELE POZZI, MATTEO MARZANO e ALESSANDRO SETTENVINI – Zero Termico

audiovisual

ALBERTO NOVELLO – hBar - Unpredictable interaction

flauto, video — flauto – Gianni Trovalusci

Concerto 3: Stati di aggregazione (solido-liquido-gassoso)

Curatore: Stefano Bassanese

ROBERTO VETRANO – NOOS 1

audiovisual

FRANCO DEGRASSI – Experiences a1

acusmatico

CARLO CICERI – NOOS 2

audiovisual

CARLO BARBAGALLO – Dal sottosuolo

voce, live electronics — voce – Annalisa Pascai Saiu

ANTONIO SCARCIA – Barcarola

acusmatico

MAURA CAPUZZO – NOOS 3

audiovisual

Indice dei contributi scientifici

Table of contents

Sessione 1: Sistemi per la Composizione Assistita

Chair: Sylviane Sapir

STEFANO SILVESTRI

1 Modelli di sistemi complessi ed elaborazione numerica di segnali caotici in SuperCollider.

FABIO DE SANCTIS DE BENEDICTIS

7 Dall'analisi musicale alla composizione e formalizzazione algoritimica: esempi applicativi con PWGL.

FRANCESCO BIANCHI e ANDREA AGOSTINI

11 Algoritmi Genetici per la composizione assistita in ambiente Real-time.

ANDREA AGOSTINI e DANIELE GHISI

17 Cage: una libreria di alto livello per la composizione assistita da computer in tempo reale.

Sessione 2: Conservazione del patrimonio musicale

Chair: Gualtiero Volpe

SERGIO CAVALIERE, ANDREA ARCELLA, NADIA POLVERINO e STEFANO SILVESTRI

23 Bringing back to life ancient music scores: the case of music in the 1799 Neapolitan revolution.

NICCOLÒ PRETTO e SERGIO CANAZZA

29 Rewind: simulazione di un'esperienza d'ascolto storicamente fedele di dischi fonografici digitalizzati.

Sessione 3: Analisi, sintesi, rendering del suono

Chair: Federico Fontana

GIACOMO VALENTI, ANDREA VALLE e ANTONIO SERVETTI

35 Permutation synthesis.

GIOVANNI COSTANTINI, MASSIMILIANO TODISCO, DANIELE CASALI e RENZO PERFETTI

41 Automatic piano music transcription system.

SANDRO SCAIELLA, SIMONE SPAGNOL, MICHELE GERONAZZO e FEDERICO AVANZINI

47 Valutazione parametrica di un modello strutturale di orecchio esterno per il rendering binaurale del suono.

Sessione 4: Analisi e ricostruzione di opere elettroacustiche

Chair: Giorgio Klauer

MASSIMO VITO AVANTAGGIATO

53 Donatoni: verso una rivalutazione di Quartetto III.

CARLO BARBAGALLO

59 Ricostruendo Nono: da Die Ermittlung a Ricorda cosa ti hanno fatto in Auschwitz.

Sessione 5: Generazione automatica di contenuti audio e musicali

Chair: Stefano Bassanese

MARINOS KOUTSOMICHALIS e ANDREA VALLE

65 SoundScapeGenerator: soundscape modelling and simulation.

ALESSIO GABRIELE

71 LAdOp1: definizione e computazione degli adattamenti in opere d'arte intermediali adattive.

Sessione 6: Storia ed estetica della musica elettroacustica

Chair: Alessandra Carlotta Pellegrini

LUCA DE SIENA

77 La Messa Elettroacustica: Sull'utilizzo dell'Ordinarium Missae in alcune composizioni elettroacustiche.

ALEX DI NUNZIO

83 Musica (pre) informatica negli Stati Uniti.

LUCA DE SIENA

89 La musica elettroacustica e l'animazione astratta - Problematiche estetiche, modelli analitici e sviluppi futuri.

Sessione 7: Sistemi per l'apprendimento aumentato

Chair: Stefano Delle Monache

MARCELLA MANDANICI, LEONARDO AMICO, ANTONIO RODÀ e SERGIO CANAZZA

95 Conoscere l'armonia tonale nell'ambiente interattivo "Harmonic Walk".

GIACOMO LEPRI, CORRADO CANEPA, GUALTIERO VOLPE, SIMONE GHISIO e ANTONIO CAMURRI

101 Crafting sounds with The Potter.

MICHELE GERONAZZO, LUCA BRAYDA, ALBERTO BEDIN e FEDERICO AVANZINI

107 Audio 3D e ancoraggio sonoro per l'esplorazione multimodale di ambienti virtuali.

Sessione 8: Interazione e controllo

Chair: Marcella Mandanici

YURI DE PRÀ, FEDERICO FONTANA e FAUSTO SPOTO

113 Confronto tra sistemi di rilevamento del gesto basati su sensori a infrarossi o ultrasuoni per applicazioni di pianoforte virtuale.

ANDREA VALLE e SIMONE PAPPALARDO

118 The Gate Modulator. An experiment in digitally-controlled analog synthesis.

Sessione 9: Modelli della voce e imitazioni vocali

Chair: Pietro Polotti

DAVIDE ROCCHESSO e DAVIDE ANDREA MAURO

124 Self-organizing the space of vocal imitations.

CONCETTA CUCCHIARELLI

128 Riflessioni su un nuovo modello vocale.

STEFANO BALDAN, STEFANO DELLE MONACHE e LUCA COMANDUCCI

134 His engine's voice: towards a vocal sketching tool for synthetic engine sounds.

Poster Session 1

LUCA RICHELLI

140 La Libreria OpenMusic OMChroma - Documentazione online.

ELISABETTA CAPURSO

144 Per un'analisi di Improvviso Statico di Carmine Emanuele Cella.

MARCO MARINONI

149 The feature extraction based hypersampler in Il grifo nelle perle nere: a bridge between player and instrument paradigm.

155 The hyperviolin in Dalla Sua Orbita.

LUCA RICHELLI e SERGIO CANAZZA

161 PianoSpace: modelli morfologici virtuali e mappe compositive.

VALERIO DE BONIS e GIULIO COLANGELO

164 Un nuovo approccio alla forma compositiva nelle installazioni sonore: [RE]BO[U]NDS e la composizione in tempo reale.

Poster Session 2

ANNALISA METUS e GIORGIO KLAUER

170 A Sonic Art Book (Little Red Riding Hood).

STEFANO LOMBARDI VALLAURI

175 Some difficulties inherent in the project of an artificial musical experience.

MARTINO SAROLLI

177 VID/AO (Video Imaging Digital/Analog Operator).

GIULIANO ANZANI

179 nVoices: Gendy e il live electronics.

GIOVANNI COSTANTINI, MASSIMILIANO TODISCO, DANIELE CASALI e GIORGIO NOTTOLI

184 A kinematic sensor to expressively control musical processing.

ANNA TERZAROLI

190 “Pitch-class set”, Forte e Drobish.

MODELLI DI SISTEMI COMPLESSI ED ELABORAZIONE NUMERICA DI SEGNALI CAOTICI IN SUPERCOLLIDER

Stefano Silvestri
Conservatorio di Napoli
stefanodrive_1@libero.it

SOMMARIO

Nel presente lavoro viene descritta l'implementazione algoritmica di alcuni modelli di sistemi dinamici complessi in domino discreto e vengono presentate particolari applicazioni dei sistemi non-lineari nel contesto dell'elaborazione numerica dei segnali musicali in *real time*. Lo studio riguarda l'applicazione, in linguaggio SuperCollider (SC), delle equazioni ricorsive che, in certe regioni del loro *spazio dei parametri*, producono traiettorie caotiche nello *spazio delle fasi* del sistema. Si tratta di comportamenti autonomi tempo-varianti che descrivono, in dominio sonoro, evoluzioni dinamiche interessanti sotto l'aspetto musicale generativo.

1. INTRODUZIONE

L'applicazione musicale dei sistemi dinamici complessi è una prassi consolidatasi nell'ambito della composizione algoritmica generativa [1, 2, 3]. L'origine di questo orientamento è rintracciabile in un "*approccio musicale sistemico*" [4], un paradigma che abbraccia i concetti fondamentali della teoria generale dei sistemi applicandoli alla formalizzazione e generalizzazione dei costrutti musicali [5]. Un processo musicale può essere definito in termini di un ente o *sistema*, inteso come complesso risultante dall'interazione di più parti, caratterizzato da input/output e da una funzione di trasferimento specifica. In base ai termini della sua espressione il sistema può essere di tipo *lineare* o *non-lineare*, in tal sede verranno considerati alcuni semplici sistemi non-lineari. Un'altra distinzione utile è quella tra *sistemi statici* e *dinamici*. Nella trattazione l'interesse è rivolto alla seconda categoria in quanto, tenendo conto del cambiamento di stato in funzione del tempo, è indicata per la rappresentazione di fenomeni musicali.
È noto che Iannis Xenakis (1922-2001), pioniere della composizione algoritmica [6], ha avviato il processo di applicazione dei modelli di fenomeni naturali all'ambito della sintesi sonora e composizione strumentale. Modelli formali di questo tipo intervengono inoltre sul piano del *continuo ordine-caos* [7], dove gli oggetti musicali corrispondenti vengono considerati in termini di prevedibilità e imprevedibilità, ordine e disordine.

Copyright: © 2014 Stefano Silvestri. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Dal pensiero musicale stocastico si è giunti al concetto di "evoluzione sistemica": i sistemi dinamici non-lineari della teoria del caos deterministico sono stati usati nei processi di sintesi del suono, a livello microstrutturale, e come procedure di trattamento simbolico, a livello macrostrutturale [8]. L'idea è che: "*sistemi dinamici perfettamente deterministici e apparentemente semplici manifestano comportamenti di enorme complessità, spesso irriducibili e caotici [...] esibendo forme di organizzazione molto particolari*"[1]. Altre direzioni recenti sono più vicine alla scienza della complessità e riguardano un approccio di tipo *eco-sistemico*, "*riflettente una dinamica di auto-eco-organizzazione*" [9, 10], dove l'attenzione è rivolta soprattutto alle *relazioni* e le *interazioni* che si instaurano tra elementi interni/esterni al sistema. In quest'ultimo caso si ha solitamente una relazione input-output inversa, il sistema è autoregolatore e prevede meccanismi di *feedback negativo*. In tal sede ci si occupa di quei sistemi in cui la relazione di input-output è diretta e sono presenti meccanismi di *feedback positivo* che permettono comportamenti tendenti all'*auto-organizzazione* [11].

2. COMPLESSITÀ ED EMERGENZA

Nell'indagine musicale basata sul *feedback* e sulla scienza della complessità, l'attenzione rivolta al suono si traduce nella valutazione dell'*omeostasi* e dell'*omeoresi* degli elementi sonori, nel primo caso si tratta della naturale tendenza del sistema alla stabilità, nel secondo si tratta della tendenza verso comportamenti evolutivi. Altra valutazione è quella del livello di *entropia* e *sintropia*, dove con il primo termine si intende una misura del rumore mentre con il secondo il grado di informazione. Inoltre è possibile considerare: i possibili stati di *equilibrio*, la risposta di *stabilità*, la *metastabilità*, la *stazionarietà*, ovvero tutti aspetti del comportamento dei sistemi dinamici complessi.
La necessità di coinvolgere, in ambito musicale, i modelli di sistemi complessi, nasce dall'intento di ridurre la pratica dell'*agire il suono* alla definizione di proprietà di basso-livello che divengono responsabili di comportamenti emergenti posti ad alto-livello. L'*emergenza* di un sistema è l'azione del manifestare autonomamente pro-

[1] Ibid. p.338.

prietà nuove, non deducibili dalle caratteristiche dei singoli componenti. Fra questi elementi si instaurano rapporti di *interazione*, relazioni talvolta bidirezionali che stanno alla base della *sinergia*. Si afferma quindi che il risultato complessivo del sistema è "più, o anche meno, della somma delle singole parti" [12, 13], e questa stessa idea, riferita alla strutturazione del suono, è alla base della ricerca dell'*emergenza sonologica* [14].
Il suono risultante dai *sistemi caotici*[2] è emergente in quanto evento imprevedibile che non nasce da un progetto del pensiero compositivo, viceversa è causato dall'elevata *sensibilità alle condizioni iniziali* e dalla non-linearità dei termini che caratterizzano il sistema stesso. I segnali *caotici*, le dinamiche di *turbolenza*, gli stati di *auto-dis/organizzazione*, le *orbite* in *equilibrio/dis-equilibrio*, sono causate da interazioni locali: si parla di *Local Activity* (attività locale), ovvero la caratteristica essenziale che rende un certo processo caotico, imprevedibile e creativo [15].

3. EQUAZIONI DIFFERENZIALI IN FORMA DI E.D.F. NON LINEARI

Le equazioni differenziali esaminate in tal sede sono state ricondotte in forme dirette di equazioni alle differenze finite (e.d.f.). Ai fini dell'implementazione algoritmica si è trattato di sostituire l'espressione con la relativa versione tempo-discreta, indicando i termini di input/output sulla base di stati attuali e precedenti del sistema. Essendo presenti inoltre stati di uscita reintrodotti all'ingresso dell'equazione stessa, il procedimento di calcolo delle orbite risulta di tipo ricorsivo, in altri termini le struttura complessiva è riconducibile idealmente a un sistema di tipo IIR (Infinite Impulse Response).
Un'equazione differenziale coinvolge una funzione di una variabile come incognita $y(x)$ e le sue derivate $y'...y^n$, ad es. nel caso delle equazioni differenziali di primo ordine risulta $F(x,y,y')$. Un esempio generale è dato dalla semplice relazione in (1), che viene risolta in base al *teorema fondamentale del calcolo integrale*[3], pertanto ogni soluzione risulta nella forma (2):

$$y' = g(x) \tag{1}$$

$$y(x) = y_0 + G(x) \tag{2}$$

con y_0 costante e $G(x)$ primitiva[4] di $g(x)$. La (2) è un'espressione che sarà di riferimento per le implementazioni successive.

Le equazioni differenziali possono essere scritte come un sistema di equazioni che legano il tasso di variazione di un certo numero di incognite alle loro derivate [16] e per trovare le rispettive soluzioni numeriche è opportuno definire delle condizioni iniziali. Il seguente esempio mostra che, a partire da una semplice equazione differenziale, in cui Y è una variabile che cambia nel tempo t in modo continuo, è possibile trovare una soluzione numerica tale che Y assume la forma di un'equazione differenza[5]:

$$\frac{dY}{dt} = \sin(Y) \tag{3}$$

$$Y_{n+1} = Y_n + \sin(Y_n)\Delta t \tag{4}$$

L'equazione differenziale in forma esplicita (3) ha corrispondenza con la relativa versione alle differenze finite (4). Per trovare una soluzione numerica è opportuno stabilire una condizione iniziale Y_0 e scegliere un valore definito per Δt, quindi si può procedere al calcolo della funzione in modo iterativo. Risolvere (o integrare) un'equazione differenziale di ordine n significa ricercare tutte le funzioni incognite $y = f(x)$ tali che $F(x, f(x), f'(x),..., f^n(x)) = 0$, si ha quindi che il grafico è relativo a un insieme di funzioni ottenute al variare dei parametri del sistema. Per la versione DSP (Digital Signal Processing) del modello iterativo in (4) è opportuno eseguire il calcolo della funzione su ogni campione. Nei linguaggi di elaborazione musicale in real-time, come SuperCollider, bisogna considerare la grandezza del blocco di trasferimento dell'audio tra le unità di generazione sonora: per lavorare al *livello del campione* (*sample level*), secondo la *scala temporale musicale* [17], occorre rendere unitario il *blocksize*. In altre parole per $N = 1$ si può accedere effettivamente al campione precedente [16, 18]. Dall'equazione alle differenze del primo ordine nella variabile Y (4) si ottiene la forma utile per l'implementazione algoritmica:

$$y[n] = x[n] * \sin(y[n-1]) + y[n-1] \tag{5}$$

Dove $y[n]$ è il segnale di uscita del sistema, $y[n-1]$ è il segnale di uscita ritardato di un campione e reintrodotto all'ingresso dell'equazione, ovvero $Y(z)z^{-1}$ nel dominio della frequenza, mentre $x[n]$ indica il segnale di ingresso, per il quale è opportuno scegliere alcune condizioni iniziali.
Modelli di questo tipo producono in uscita sequenze di valori che possono essere sfruttati a vari livelli della progettazione sonora, ad es. per la sintesi di campioni o per l'elaborazione con strutture di controllo. Tra queste

[2] Lo studio dell'emergenza riguarda la *scienza della complessità*, l'*auto-organizzazione*, l'*autopoiesi* e l'*adattività*. L'emergenza è correlata inoltre al risultato dei sistemi dinamici sviluppati dalla *geometria frattale* e dalla *teoria del caos*.

[3] Il teorema pone in stretta relazione il concetto di derivata al concetto di integrale e permette di determinare il valore dell'integrale definito tramite una semplice differenza.

[4] Il calcolo integrale consiste, conoscendo la derivata, nel ritrovare la funzione primitiva da cui proviene la derivata. La derivata di $G(x)$ è pari a $g(x)$, pertanto l'operazione di risalire dalla derivata alla sua primitiva è detta integrazione; ad es. la funzione primitiva che ha per derivata $y' = 6x$ è $y = 3x^2$.

[5] Ibid. p.1.

possibilità, di notevole interesse risulta la combinazione dei sistemi di equazioni non-lineari con equazioni alle differenze finite di filtri lineari IIR e FIR (Finite Impulse Response), componendo mappe numeriche che godono delle proprietà dei sistemi complessi. Spesso tali strutture trovano impiego nei modelli fisici strumentali, per la simulazione dei moti di turbolenza dei transienti e delle meta-oscillazioni del segnale musicale [19].

3.1 Modello di Lotka-Volterra

Come prima applicazione si propone il modello *preda-predatore* (1925-26), descritto dalle equazioni differenziali non-lineari di I ordine di Lotka-volterra:

$$\begin{aligned} \frac{dx}{dt} &= (a - by)x \\ \frac{dy}{dt} &= (cx - d)y \end{aligned} \tag{6}$$

Si tratta di un esempio classico di *sistema dinamico adattivo*, molto sfruttato anche nell'ambito della musica interattiva e generativa [20]. Esso rappresenta il caso più semplice di interazione tra due *agenti*: la popolazione di *prede* è inversamente proporzionale all'accrescimento dei *predatori* presenti sul territorio, viceversa la popolazione di predatori è direttamente proporzionale alla presenza di prede. Nella (6) il primo termine indica la popolazione delle prede e il secondo la popolazione dei predatori, i coefficienti a, b sono detti rispettivamente *tasso di crescita* e *mortalità* della preda, mentre c, d rappresentano il tasso di crescita e mortalità dei predatori. In base alla (6) è possibile riscrivere il sistema come in (7), quindi è possibile giungere alla forma alle differenze in (8):

$$\begin{cases} x(t + \Delta t) = (a - by)x\Delta t + x \\ y(t + \Delta t) = (-c + dx)y\Delta t + y \end{cases} \tag{7}$$

$$\begin{aligned} x[n] &= x[n-1] + (((a - (b * y[n-1])) * x[n-1]) * dt) \\ y[n] &= y[n-1] + (((-c + (d * x[n-1])) * y[n-1]) * dt) \end{aligned} \tag{8}$$

Per realizzare il processo ricorsivo in SC si possono sfruttare la riduzione del blocksize e un bus locale con espansione multicanale:

```
Server.local.options.blockSize_(1); //N=1

SynthDef(\lvm, //Lotka-Volterra model (LVM)
{
  arg a=0.5471, b=0.01281, c=0.8439,  d=0.0266,
  gx=0.5, dt = 0.01,   riscal=0.99, limit=0,
  comb=0, del=1,       cutoff=2e4;
  var x0=20, y0=20, y1, x1;
  var zdelrd = Array.newClear(2), normal_sig,
      limit_sig;
  zdelrd = [x0, y0];//initial state condition
  zdelrd= ~delta.ar(1)*gx + LocalIn.ar(2);
   //del = 10 -> 10/100=0.1s max

#x0,y0=Select.ar(comb,[zdelrd,DelayL.ar(zdelrd,
0.1,del/100)]);
x1 = x0 + (((a - (b * y0)) * x0) * dt);
y1 = y0 + ((((-1*c) + (d * x0)) * y0) *dt);
  normal_sig=[x1, y1]*riscal;
limit_sig=Clip.ar(LeakDC.ar(normal_sig,  1.07),
-10, 10);
#x1,   y1   =   Select.ar(limit,   [normal_sig,
limit_sig,     LPF.ar(limit_sig, cutoff)]);

LocalOut.ar([x1,  y1]);//write  recursive  delay
z^-1 on localbus
//#x1, y1 = ([x1, y1]*2)-1;//yn ->[-1, 1]
  y1.poll;//[a, b, c, d, gx, dt].poll;
  [x1, y1].scope("Phase space");//xy mode
  Out.ar(0, [x1, y1]);
}).send(s);
```

Alcuni accorgimenti previsti nell'algoritmo permettono di risolvere il problema dell'instabilità strutturale[6] che, in questo modello, si manifesta soprattutto per brusche variazioni del parametro evolutivo.

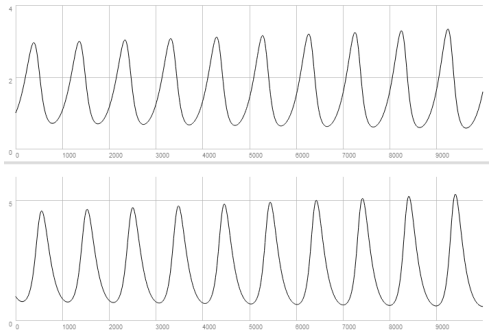


Figura 1. Oscillazioni emergenti nel modello preda-predatore

La fig.1 mostra un risultato in SC della sequenza temporale dei campioni di uscita quando all'ingresso si pone il segnale di stimolo $\delta[n]$. Un'implementazione della stessa iterazione numerica in linguaggio Processing[7] ha permesso di valutare lo sviluppo delle orbite nel cosiddetto *spazio delle fasi* del sistema dinamico:

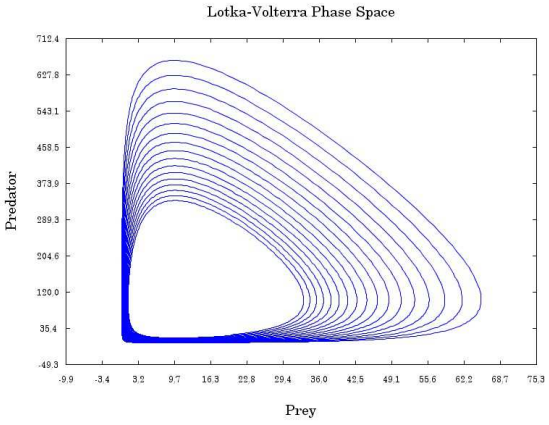


Figura 2. Spazio delle fasi del modello preda-predatore, per $\Delta t = 1000$, $a = 0.5471$, $b = 0.001281$, $c = 0.8439$, $d = 0.0266$.

[6] A causa di questa instabilità, piccole perturbazioni causate anche da errori numerici e imprecisioni durante l'integrazione, introducono deviazioni irreversibili nelle traiettorie risultanti.

[7] Si tratta del progetto di una serie di classi per Processing dedicate alla visualizzazione di funzioni. Queste classi sono state scritte allo scopo di realizzare i grafici risultanti dall'iterazione di funzioni non-lineari. E' in corso di sviluppo una libreria per Processing in grado di fornire funzionalità sullo stile di Matlab e Matplotlib di Python.

3.2 Caso di Filtered Logistic Map come sistema IIR

Si propone di seguito l'implementazione algoritmica della *mappa logistica filtrata* (*Filtered Logistic Map*) [21], ovvero un caso specifico di *mappa filtrata*[8] che sfrutta la combinazione della *mappa logistica* con e.d.f. di sistemi LTI (lineari tempo-invarianti).
L'*equazione logistica* è un modello di sistema dinamico che consiste in una funzione quadratica di tipo monoparametrico (equazione differenziale di Verhulst, 1838), nel 1950 è stata usata in ecologia e biologia come modello matematico per descrivere le dinamiche di accrescimento di una popolazione in presenza di risorse limitate (modello di May, 1976). La *funzione logistica*, secondo il modello di Verhulst, viene formalmente espressa in dominio continuo attraverso l'equazione differenziale:

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right) \tag{9}$$

La (9) indica l'accrescimento/declino della popolazione P al tempo t, con r detto *fattore di crescita* [3, p.86] e K detto *capacità di carico* (*carrying capacity*), termine asintotico della popolazione definito dalle risorse ambientali disponibili. Il rapporto x/K rappresenta la *competizione intraspecifica*, un concetto che in ecologia indica la concorrenza tra individui della stessa specie. È possibile riscrivere l'espressione (9) in forma di mappa polinomiale logistica (modello di May):

$$y(t+1) = ry(t)\left[1 - y(t)\right] \tag{10}$$

Il polinomio della mappa logistica è un caso particolare di funzione quadratica: $F(x) = rx(1-x) = r\left(x - x^2\right)$, con punti fissi $\left(0, \frac{r-1}{r}\right)$ [21,22]. L'iterazione numerica in Processing mostra l'insorgere del caos per certi valori dello spazio dei parametri del sistema:

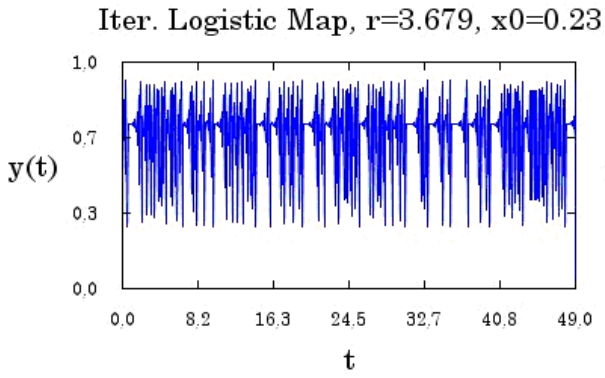


Figura 3. Comportamento caotico della mappa logistica

È possibile verificare inoltre alcune proprietà frattali di *auto-similarità* delle *finestre periodiche*[9] presenti nella *mappa di biforcazione* del sistema:

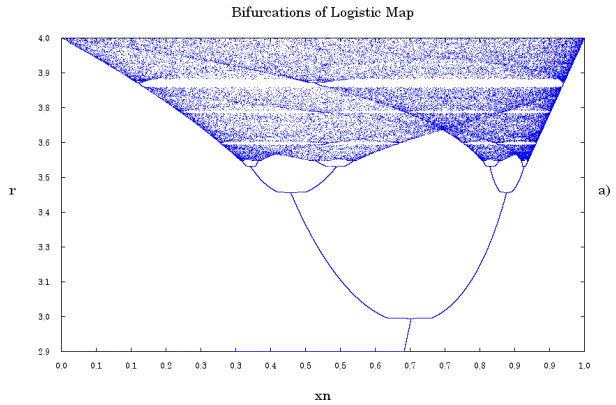


Figura 4. Mappa di biforcazione della funzione logistica

Il grafico di Fig.4 mostra il comportamento complesso del sistema dinamico non-lineare ed è stato ricavato iterando la mappa logistica al variare del parametro r. Per considerare la mappa logistica come filtro numerico è possibile ricorrere, in base all'equazione generale dei sistemi LTI, alla formulazione generale dei sistemi di tipo caotico [21]:

$$\begin{aligned} x_{n+1} &= f\left(y_n\right) \\ y_n &= \sum_{k=0}^{M} a_k x_{n-k} - \sum_{k=1}^{N} b_k y_{n-k} \end{aligned} \tag{11}$$

Dove $f : \mathbb{R} \to \mathbb{R}$ è una funzione non-lineare e $a_k, b_k \in \mathbb{R}$ sono i coefficienti del sistema, rispettivamente del segnale di ingresso $x[n]$ e di quello di uscita $y[n]$. Le mappe filtrate sono sistemi autonomi ed i filtri sono sempre situati nel percorso di feedback del sistema[10]. Sostituendo, nel modello di Verhulst, le operazioni di integrazione con quelle di iterazione $F^n(x_0)$ si ottiene la relativa forma di e.d.f.:

$$y[n] = r * y[n-1] * k - y[n-1] \tag{12}$$

Nel caso più semplice l'espressione della *mappa logistica filtrata* (FLM) è data combinando la (12) con la e.d.f. di un sistema FIR di tipo passa-basso (PB) di I ordine:

$$\begin{cases} x[n] = r * y[n-1] * 1 - y[n-1] \\ y[n] = a * x[n] + (1-a) * x[n-1] \end{cases} \tag{13}$$

[8] Si tratta di un modello di sistema definito da una mappa iterata usata come filtro numerico. A tal proposito si sottolinea che nei modelli in cui risulta un ritardo non unitario z^{-N}, le relative e.d.f. possono essere considerate come caso particolare di filtri non-lineari di tipo COMB.

[9] Regioni di minore densità in cui si replicano le caratteristiche del pattern iniziale, possono essere anche definite come "micro-interruzioni di instabilità".

[10] Ibid. p.1.

Dove le condizioni iniziali del sistema sono scelte in $[0,1]\times[0,1]$; il filtro PB è $y[n]=a_0x[n]+a_1x[n-1]$, di coefficienti $a=a_0;(1-a)=a_1$ tale che $a\in[0,1]$ e con risposta in frequenza $H(e^{j\omega})=a+(1-a)e^{-j\omega}$. Sfruttare un filtro qualsiasi di ordine superiore corrisponde a rendere la mappa stessa di ordine superiore [21]. L'algoritmo che implementa la (13) è il seguente:

```
( //FLM IIR explicit form
s.waitForBoot { SynthDef(\flm,
{
        arg r=0, a=0, gx=1;
        var yz, x1, y1, y0=0, x0=0;
        var zdelrd;
        zdelrd=LocalIn.ar(2);
        yz = ~delta.ar(1)*gx + zdelrd;
        #y0, x0 = yz;
        x1 = r * y0 * (1 - y0);
        y1 = a*x1 + ((1 - a)*x0);
        LocalOut.ar([y1, x1]);//write recursive
        #x0, x1, y1 = ([x0, x1, y1]*2)-1;
        x1.poll; [x1, x0].scope("Phase space");
        Out.ar(0, [x1, y1]); //xy mode
}).send(s);
}; )
```

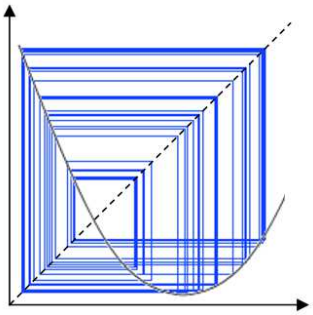


Figura 5. Diagramma a scala nello spazio degli stati della *filtered logistic map*, per $r=3.62$

Di notevole interesse risultano il *diagramma a scala* (fig.5), ottenuto in SC escludendo il contributo del FIR con $a=0$, e la risposta in frequenza del sistema al variare del parametro r, ottenendo il cosiddetto *spettro di biforcazione*[11] [21]:

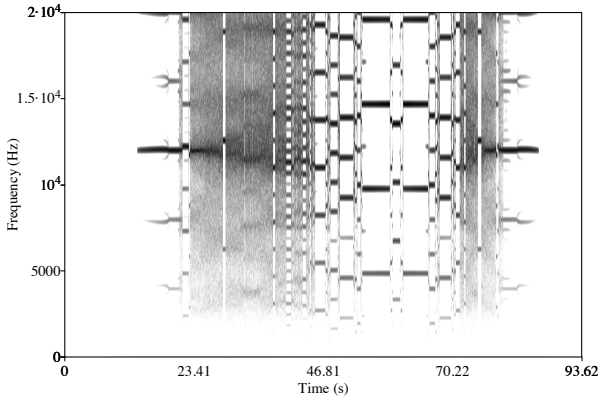


Figura 6. Spettro di biforcazione della mappa logistica filtrata, con escursione $1.5<r\le 5$.

[11] Nei punti in cui l'orbita è periodica vengono percepite altezze toniche più o meno stabili, mentre oltre il *margine del caos* si tende al rumore non uniforme. Dall'orbita caotica per *r>3.57* possono tal volta emergere microfluttuazioni pseudo-periodiche e intermittenze.

3.3 Sistema di equazioni di Lorenz

I segnali caotici possono essere sintetizzati trovando soluzioni numeriche ad equazioni differenziali di ordine superiore. Il sistema di equazioni di Lorenz[12] è un modello composto da tre equazioni differenziali ordinarie che si presta efficacemente per la generazione di segnali caotici in dominio discreto [16, p.1-2]:

$$\begin{cases}\dfrac{dX}{dt}=\sigma(Y-X) & \forall t\ge 0\\ \dfrac{dY}{dt}=X(\rho-Z)-Y & \forall t\ge 0\\ \dfrac{dZ}{dt}=XY-\beta Z & \forall t\ge 0\end{cases} \tag{14}$$

Le incognite X,Y,Z rappresentano le tre variabili di stato del sistema, mentre $\sigma,\rho,\beta\in\mathbb{R}$ sono i parametri di controllo per la modifica degli andamenti nello spazio degli stati. In tal sede si inizializza lo spazio dei parametri in regioni di stabilità, quali ad es. $\sigma=10;$ $\rho=22;\beta=8/3$ e si procede alla scoperta del *cammino evolutivo*, e dei relativi *attrattori*, variando di poco ogni valore. La forma di e.d.f. derivante dalla (14) è:

$$\begin{cases}x[n]=x[n-1]+(\sigma*y[n-1]-\sigma*x[n-1])*dt\\ y[n]=y[n-1]+(-x[n-1]*z[n-1]+\rho*x[n-1]*y[n-1])*dt\\ z[n]=z[n-1]+(x[n-1]*y[n-1]-\beta*z[n-1])*dt\end{cases} \tag{15}$$

```
(
s.waitForBoot {
/*----delta di dirac----*/
SynthDef(\delta,
{
      Out.ar(~delta.index, PlayBuf.ar(1,
            ~tab.bufnum, 1, doneAction: 2))
}).send(s);

SynthDef(\lorenzSystem,    //Lorenz model
{       //coefficients of model
        arg sgm = 10.0, rho= 22, bta = 8.0/3.0,
            dt = 0.01;
        var x0= -7.5, y0= -3.6, z0=30.0, x1,
            y1, z1;//init previous and actual
        var zdelrd, rscl_x, rscl_y, rscl_z;
        zdelrd= ~delta.ar(1)+LocalIn.ar(3);
        #x0, y0, z0= zdelrd;
        x1= x0 +( (sgm * (y0 - x0)) * dt);
        y1= y0 +( (x0 * (rho - z0) - y0) * dt);
        z1= z0 +( ((x0 * y0)-(bta * z0)) * dt);
        LocalOut.ar([x1, y1, z1]);
        rscl_x=x1/20; rscl_y=y1/20;
        rscl_z=z1/20; x1.poll;
[rscl_x, z0/20].scope("Lorenz Attractor");
        FreqScope.new(600, 500, 0);
        Out.ar(0, [rscl_x, rscl_z]);
}).send(s) })
```

Come per gli algoritmi precedenti, la condizione x_0 della (15) è la sequenza canonica $\delta[n]$ fornita nella `SynthDef(\delta)`. Di seguito si mostrano alcuni risultati di elaborazione in SC:

[12] Questo modello matematico è considerato come l'archetipo per il problema delle previsioni del tempo.

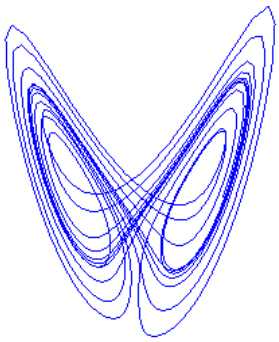


Figura 7. *Attrattore di Lorenz* ottenuto nello spazio dei parametri per $\sigma = 6.24$, $\rho = 41.21$, $\beta = 2.67$ $dt = .01$

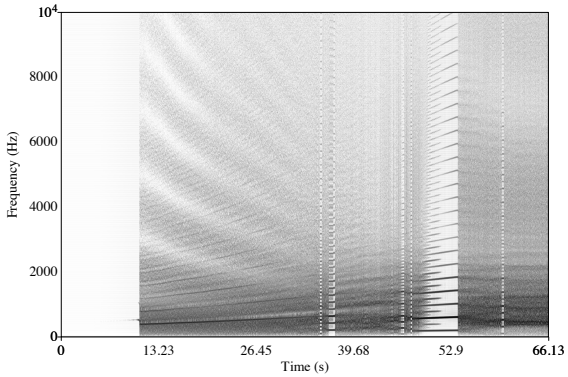


Figura 8. Spettrogramma del cammino evolutivo ottenuto variando $12.58 \le \rho \le 50$ e con coefficienti costanti $\sigma = 10, \beta = 2.67, dt = .01$

4. CONCLUSIONI

Oltre i modelli proposti è stata studiata l'implementazione numerica dell'oscillatore complesso di Chua [4]. Numerosi altri esperimenti possono essere effettuati con le tecniche di feedback di uno o più campioni. Con equazioni riconducibili a COMB e All-Pass è possibile ridurre la frequenza base f_b del sistema. Sviluppi successivi prevedono l'uso di questi modelli in meccanismi di *feedback auto-regolato* e in strutture FDN (Feedback Delay Network). Seguendo la "prassi musicale sistemica": specifiche configurazioni musicali possono, in alcune condizioni, essere ricondotte ad interazioni tra unità irriducibili di un sistema dinamico, inoltre, sistemi non-lineari di questo tipo possono consentire cammini evolutivi *emergenti* e *complessi* sul piano sonoro-musicale.

5. BIBLIOGRAFIA

[1] R. E. Miranda, J. A. Biles, *Evolutionary Computer Music*, Springer, 2007.

[2] R. E. Miranda, *Composing Music with Computers*, Music Technology series, Focal Press, 2001.

[3] G. Nierhaus, *Algorithmic Composition*, Springer Wien-NewYork, 2009.

[4] S. Silvestri, Approccio Musicale Sistemico basato su modelli numerici di sistemi dinamici complessi, tesi di II livello in Musica Elettronica e Nuove Tecnologie, conservatorio di Napoli, 2013.

[5] G. Haus, *Elementi di Informatica Musicale*. Gruppo editoriale Jackson, 1984.

[6] P. Hoffmann, Music Out of Nothing? A Rigorous Approach to Algorithmic Composition by Iannis Xenakis, Universität Berlin, 2009.

[7] T. Bolognesi, Composizione musicale assistita da elaboratore. Quaderni del Dipartimento di Matematica dell'Università del Salento, p.85, 1981.

[8] A. Di Scipio, Caos Deterministico, Composizione e Sintesi del Suono, *Atti del CIM IX*, pp.337-350, Genova, 1991.

[9] A. Di Scipio, Emergenza di Suono, Suono di Emergenza. Intellectica, Vol. 48, pp.2-20, 2008.

[10] A. Di Scipio, Sound is The Interface: from interactive to ecosystemic signal processing. Organised Sound. p.269–277, Cambridge, 2003.

[11] F. Heylighen, The Science of Self Organization and Adaptivity, Center "Leo Apostel", Free University of Brussels, Belgium, 2001.

[12] F. A. De Toni, L. Comello, L. Ioan, *Auto-organizzazioni*. Marsilio Editori, p.22, 2011.

[13] E. Morin, Restricted complexity, general complexity, *Histoiré Ecole des Hautes´ Etudes en Sciences Sociale*, CNRS , p.7, 2006.

[14] A. Di Scipio, Note sulla Nozione di "Emergenza Sonologica" L'Aquila, 2006.

[15] http://galileo.cincom.unical.it/CAOS_SEMPLICE/Intervista.html, Intervista al Professor Leon Chua, 31/05/2014.

[16] S. Yadegari, Chaotic Signal Synthesis with Real-Time control, *6th int. conference on digital audio effects* (dafx-03), Cap.2, 2003.

[17] C. Roads, *Microsound*, The MIT Press, 2004.

[18] H. Nishino, On Conceptual Misfits in Computer Music Programming. *Journal of The Japanese Society for Sonic Arts*, vol.1 no.1 pp.17–21, 2011.

[19] X. Rodet, C. Vergez, Nonlinear Dynamics in Physical Models. *Computer Music Journal* © MIT, 23:3, pp.18–34, 1999.

[20] A. Eldridge, Collaborating with the Behaving Machine. University of Sussex, par.5.4.1, 2008.

[21] R. Holopainen, Logistic Map with a First Order Filter. World Scientific Publishing, p.1, p.8, 2010.

[22] S. Silvestri, Elaborazione numerica di segnali caotici mediante algoritmo della mappa logistica filtrata. Corso di teoria del suono (Acustica, Psicoacustica, Ecologia), p.3, 2013.

DALL'ANALISI MUSICALE ALLA COMPOSIZIONE E FORMALIZZAZIONE ALGORITMICA: ESEMPI APPLICATIVI CON PWGL

Fabio De Sanctis De Benedictis
ISSM "P. Mascagni" - Livorno
fdesanctis@teletu.it

SOMMARIO

L'analisi della musica è una disciplina intermedia tra l'estetica musicale e la teoria della composizione. Rispetto a quest'ultima l'analisi è strumento utile per estrapolare i processi compositivi e trarne un possibile algoritmo. Questo articolo propone, senza pretesa di esaustività, degli esempi applicativi con PWGL, passando in rassegna alcune delle tecniche del Novecento. Le conseguenze e gli ulteriori possibili sviluppi di un lavoro qui appena impostato possono essere molteplici: la conservazione e la trasmissione di un patrimonio storico e musicale che in alcuni casi potrebbe non avere la diffusione che merita o addirittura andare perso; un diverso approccio alla didattica della composizione; una maggiore consapevolezza dell'evoluzione del pensiero e delle tecniche compositive dell'ultimo secolo; un potenziamento del fare compositivo individuale, con maggiore possibilità di concentrazione sui processi; l'interazione, attraverso opportune librerie e *patch*, con altri *software*.

1. INTRODUZIONE

Analizzare un'opera significa distinguere le parti e le loro relazioni. [1] In questo senso l'analista affronta l'opera intera per giungere ai suoi materiali primi, percorrendo un tragitto sotto certi aspetti inverso a quello del compositore, che può iniziare da pochi elementi per giungere all'opera completa. A ragione quindi questa disciplina si colloca tra l'estetica e la teoria della composizione. [2] L'analisi musicale può dunque individuare le tecniche compositive adottate e i processi seguiti, e con un *software* di composizione algoritmica è possibile codificare il tutto in *patch* riutilizzabili o personalizzabili. Seguono alcuni esempi realizzati con PWGL, con riferimento ad alcune delle tecniche compositive del Novecento. In conclusione alcune riflessioni sulla valenza e ulteriori sviluppi possibili di un lavoro come quello qui discusso. Non possiamo in questa sede mostrare un numero maggiore di esempi. Dalla pagina web: https://sites.google.com/site/fdsdbmascagni/conferenze-e-seminari è possibile scaricare una versione ampliata di questo scritto e le *patch* di PWGL, onde poterne comprendere e verificare più direttamente il funzionamento e l'utilizzo. [3]

2. ESEMPI APPLICATIVI CON PWGL

2.1 Premessa

Come noto, PWGL è un *software* di composizione algoritmica e di sintesi audio basato su *Common Lisp* e *OpenGL*, derivato da *PatchWork* e mantenuto attualmente da Mika Kuuskankare e altri presso l'Accademia Sibelius di Helsinki. [4] *Software* di questa categoria permettono la creazione di algoritmi ponendo e collegando determinati oggetti all'interno di una finestra. La scelta di PWGL, invece che OpenMusic, non escludendo a breve/medio termine la traduzione delle *patch* da un ambiente all'altro, permette l'utilizzo di una interfaccia grafica e di una concezione complessiva più moderna, che sotto certi aspetti semplifica e velocizza il lavoro. La composizione assistita al calcolatore può essere classificata secondo diversi criteri. Carlos Agon [5] distingue due differenti approcci: costruttivo quando il *software* è usato principalmente per lo sviluppo del materiale compositivo; dichiarativo quando l'elaboratore viene istruito per restituire un brano completo (o una sezione di esso) secondo le caratteristiche richieste. A queste due categorie corrispondono le divisioni in CAC e CGM (*Computer Aided Composition* e *Computer Generated Music*) immediatamente prima proposte da Miller Puckette nello stesso testo. Il secondo volume della medesima raccolta [6] contiene una prefazione di David Cope con una classificazione più dettagliata, in CGS, CGA, CGC, rispettivamente *Computer Generated Sounds*, *Computer Generated Assistance* e *Computer Generated Composition*, con ulteriori sotto-categorie. [7] Senza entrare troppo in merito

[1] Sul concetto di forma ci basiamo principalmente su quanto esposto in [1], p. 209 e segg. e in [2] p. 1.

[2] Si veda [2], p. 2.

Copyright: ©2014 Fabio De Sanctis De Benedictis. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

[3] Le tecniche compositive prese in esame nella versione *online* globalmente afferiscono alla Dodecafonia secondo Babbitt, alla composizione seriale e post-seriale (Messiaen, Boulez, Manzoni, Berio, Maderna, Xenakis), alla musica spettrale, alla *Pitch-class Set Theory*, sia per l'aspetto analitico che compositivo, ai *costraints*.

[4] http://www2.siba.fi/PWGL/index.html

[5] [3] p. V.

[6] [4] pp. X-XV.

[7] Sui rapporti invece tra analisi musicale e programmi di composizione algoritmica, al di là di quanto illustrato nei tutorial e negli help di PWGL e OpenMusic, in rapporto alle rispettive funzionalità di analisi musicale, e in alcune pubblicazioni (si vedano ad esempio [5, 6, 7, 8, 9]), non sembra apparire sufficiente letteratura che segua l'impostazione qui proposta.

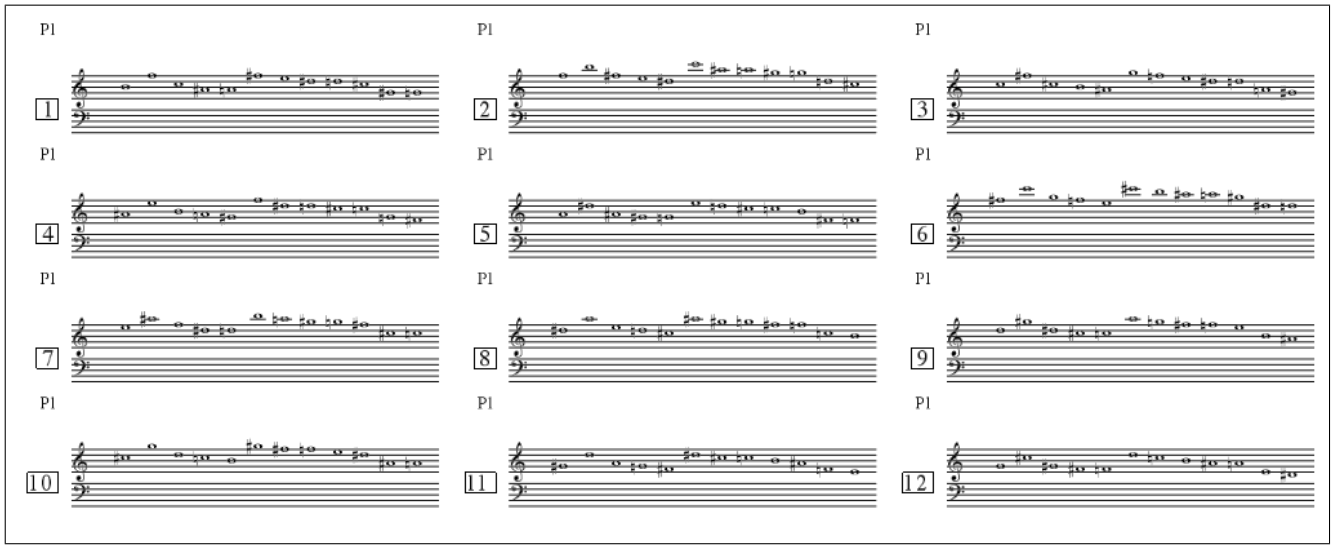


Figura 1. Le 12 trasposizioni del Retrogrado Inverso.

alla questione, che ci porterebbe fuori dai limiti di questo saggio, si può anticipare che gli esempi sviluppati ricadono in parte nella categoria CAC, ovvero CGA, e parzialmente nella classificazione CGM (ossia CGC). [8] Procediamo con alcuni modelli applicativi.

2.2 La dodecafonia secondo Babbitt

Preliminare alla composizione dodecafonica è lo sviluppo della serie originale secondo i noti procedimenti di inversione, retrogradazione e retrogrado dell'inversione, più tutte le 12 trasposizioni per categoria, per un totale di 48 serie. Le trasposizioni possono essere effettuate cromaticamente, [9] oppure possiamo trasporre le serie secondo l'ordine stabilito dalla successione delle altezze nella prima forma di ogni specie, come fa Boulez nelle sue *Structures Ia*. [10] È possibile creare una *patch* di PWGL che automatizzi la realizzazione delle 48 trasposizioni canoniche, ognuna regolata dall'ordine delle altezze della prima serie di ogni tipologia.

Con un oggetto Chord-editor si imposta la serie Originale. Dall'uscita *pitches* del Chord-editor abbiamo i numeri di altezza MIDI della serie impostata, ovvero la lista (75 74 69 68 67 66 64 73 72 70 65 71). Si crea la forma inversa: l'oggetto x->dx trasforma le altezze in intervalli, in semitoni, (-1 -5 -1 -1 -1 -2 9 -1 -2 -5 -6), poi invertiti moltiplicandoli per -1. Il risultato deve essere inviato all'ingresso destro dell'oggetto dx->x, che riconverte gli intervalli in note, a partire dalla prima nota della serie Originale, selezionata con l'oggetto first e inviata all'ingresso sinistro di dx->x. Si genera poi il Retrogrado, semplicemente rovesciando l'ordine delle altezze originali con l'oggetto reverse. Per la produzione del Retrogrado Inverso, si unificano i due procedimenti precedenti. Per il motore di trasposizione si procede come segue: x->dx converte in intervalli le altezze di ogni forma seriale; la coppia di oggetti pwgl-enum e pwgl-map genera un *loop* per cui, a partire da ogni nota della serie, a turno se ne genera un'altra dagli intervalli immessi, riconvertiti poi in altezze dall'oggetto dx->x posto tra pwgl-enum e pwgl-map; si ottengono così le dodici trasposizioni, ognuna con inizio, in ordine, dalle note della serie di riferimento. Ad esempio, nel caso della serie originale, pwgl-enum preleva la prima altezza midi (75) e la stabilisce come punto di partenza per produrre una serie con gli intervalli introdotti; poi si preleva la seconda altezza (74) e si ripete il procedimento sino alla fine, restituendo il tutto come una lista di dodici serie, visualizzabile valutando e aprendo i Chord-Editor corrispondenti. [11] Nella Fig. 1 il risultato relativo alle 12 trasposizioni del Retrogrado Inverso.

Babbitt, in un suo famoso articolo, [12] considera la Dodecafonia matematicamente come un gruppo, manifestando un passaggio significativo di concezione, dall'approccio tematico di Schönberg al suo, di carattere permutativo. Lo scopo è quello di investigare alla ricerca di opportune trasposizioni che permettano, in qualsiasi serie che presenti intervalli ricorrenti al proprio interno, di riottenerli nella nuova trasposizione in punti differenti, ma con le stesse note, definite invarianti. Questo consente una maggiore coerenza dovuta alla migliore riconoscibilità, garantita dalle medesime altezze, soprattutto se poste nello stesso registro. Anche in questo caso è possibile creare una *patch* apposita.

Prendiamo ad esempio la serie dell'op. 25 di Schönberg. La possiamo dividere in bicordi con l'oggetto group-list con argomento 2, e calcolare la differenza intervallare tra le note di ognuno di essi. Il primo, il penultimo e l'ultimo bicordo risulteranno identici, presentando lo stesso risultato. Corrispondono rispettivamente ai semitoni mi-fa, si-do e la-la diesis della serie. L'utente attraverso l'oggetto posn-

[8] Come detto, una maggiore varietà di esempi (CAC e CGC), è visionabile nella versione *online* di questo scritto. Ugualmente ad essa rimandiamo per le figure complete delle *patch* cui si fa riferimento e i file di PWGL relativi agli esempi affrontati.

[9] È ad esempio il modo di procedere di Dallapiccola, come di recente osservato nell'analisi di alcuni suoi manoscritti conservati presso il Gabinetto Vieusseux di Firenze.

[10] Vedi [10].

[11] La valutazione avviene, in PWGL come in OpenMusic, selezionando l'oggetto della *patch* e digitando la lettera "v" sulla tastiera del computer.

[12] L'articolo in questione è [11].

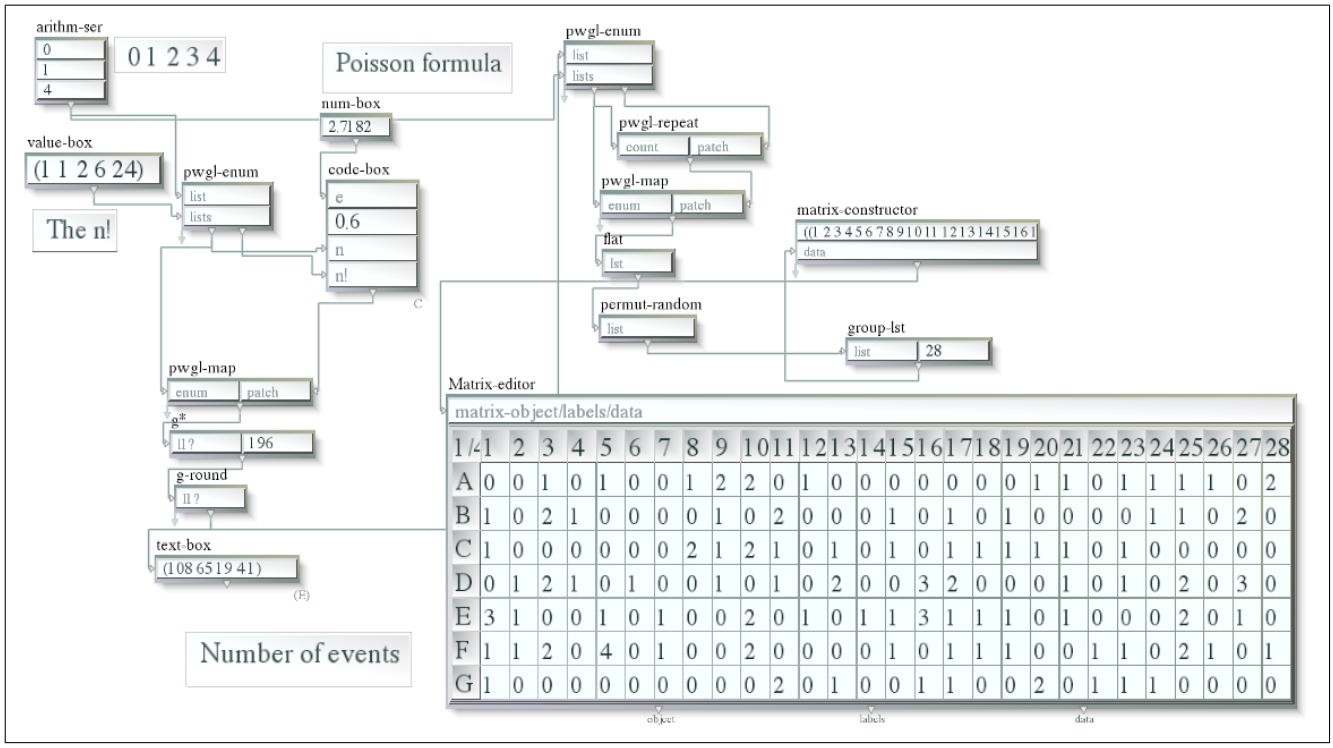


Figura 2. Ricostruzione semplificata della matrice poissoniana di *Achorripsis* di Xenakis

match seleziona gli opportuni bicordi, finalmente valuta il tutto e ottiene il risultato. La *patch* puó essere opportunamente strutturata per operare le trasposizioni necessarie affinché nel primo caso il bicordo si-do appaia in prima posizione al posto del mi-fa della serie originale. [13] Nel secondo caso mi-fa va al posto detenuto da si-do nell'Originale. Abbiamo quindi la conservazione delle invarianti attraverso le opportune trasposizioni illustrate da Babbitt.

2.3 Tecniche compositive seriali e post-seriali

Boulez prende spunto da *Mode de valeurs et d'intensités* di Messiaen, di cui utilizza la prima serie di altezze, per la creazione di *Structures Ia*, per due pianoforti, una delle opere seriali per eccellenza. Il motore principale di sviluppo del materiale è dato da due matrici quadrate numeriche, desunte dalle trasposizioni della serie originale e inversa, ai cui numeri vengono associati valori parametrici specifici per le durate, le dinamiche e gli attacchi, ottenendone le serie corrispondenti. Possiamo ricostruire il processo compositivo legato alla creazione delle trasposizioni della serie originale, alla creazione della matrice numerica, [14] alla generazione delle serie ritmiche con unità la biscroma e all'associazione di ogni serie di altezze con una serie di durate. [15]

Da un Chord-Editor si estraggono le classi di altezze (numeri di nota MIDI mod12, complessivamente da 0 a 11) più 1 (numeri da 1 a 12) di tutte le trasposizioni. La coppia pwgl-enum/map associa ogni cifra al valore percentuale 12.5 (tramite un oggetto gpercentage), corrispondente alla biscroma, generando così 12 serie ritmiche. Ciascuna di queste viene associata con ognuna delle serie di altezze attraverso un oggetto Multi-Score-PMC. Risulta immediato comprendere che, cambiando la serie dodecafonica originale e valutando nuovamente la *patch*, si ottengono all'istante le serie parametriche corrispondenti al nuovo assetto. In linea generale questo vale per tutte le *patch*.

Terminiamo con un esempio preso da Xenakis. Orcalli [16] mostra nel suo testo alcune tecniche di composizione di *Achorripsis* che portano alla compilazione della nota tabella che stabilisce il numero di eventi per ogni strato timbrico in base alla cifra rappresentata in ogni cella. La *patch* in Fig. 2 formalizza una parte del procedimento, per cui ad ogni valutazione abbiamo una nuova versione della matrice con una distribuzione di tipo Poissoniano. Nella parte in alto a sinistra troviamo la sezione dedicata al computo relativo alla funzione di Poisson. Dentro l'oggetto code-box un semplice codice in lisp permette tale calcolo: `(* (expt e (* -1 d)) (/ (expt d n) n!))`, dove `e` è il numero di Nepero, e `d` la densità media, qui col valore di 0.6. La prima coppia pwgl-enum/map applica la formula di Poisson ad ogni cifra, nell'ordine, da 0 a 4, ottenendo il numero complessivo di presenze di ogni numero. Il resto della *patch* si occupa in sintesi di generare materialmente le occorrenze numeriche, randomizzarne la distribuzione con l'oggetto permut-random e produrre la

[13] Le trasposizioni vanno effettuate per l'intervallo relativo alla distanza tra i bicordi selezionati e il suo complementare.

[14] Per essere precisi, diversamente da Boulez, che indicizza ogni nota sulla base della sua posizione nella serie originale, in questo caso i numeri sono generati dal numero di nota MIDI mod12, ossia dalla classe di altezza corrispondente ad ogni nota, più uno, per evitare il numero zero che sarebbe ovviamente inutilizzabile per la generazione della durata corrispondente.

[15] Boulez in realtà procede a differenti tipi di associazione tra serie di altezze e di durate, scartando l'accostamento qui realizzato.

[16] Si veda [12] pp. 47-65.

matrice finale.

3. CONCLUSIONI

L'uso di un *software* per la composizione algoritmica ha conseguenze e vantaggi notevoli, ma occorre premettere che nel campo della composizione, a nostro avviso, il suo utilizzo senza un tirocinio anche di tipo tradizionale (carta e matita, per intenderci), può rappresentare un rischio, poiché può ridursi ad un mero vuoto gioco di strutture senza intenzionalità compositiva. La tecnica, l'uso di determinati materiali, le scelte compositive, hanno valenze espressive che occorre avere ben interiorizzato per superarle e usarle in maniera adeguata e consapevole. Tuttavia possiamo condividere quello che afferma Donatoni relativamente al fatto che «l'inventività assopita» possa aver bisogno di una disciplina per risvegliarsi, [17] disciplina nella quale la tecnica compositiva può trovare una sua funzione maieutica, così come un sussidio informatico quale PWGL. Ciò premesso, gli sviluppi e gli esiti possono essere molti: un nuovo approccio alla didattica della composizione, per cui le tecniche, acquisite e interiorizzate, sono usate delegando all'elaboratore elettronico il compito gravoso della proliferazione del materiale, permettendoci così una maggiore concentrazione sulle idee e sui processi; l'eventualità di suscitare nuovi stimoli attraverso l'interazione con il *software*; la possibilità di formalizzare le tecniche compositive storiche, per una loro migliore conservazione e diffusione; una maggiore consapevolezza relativa all'evoluzione del pensiero compositivo del Novecento e oltre; un potenziamento dei procedimenti creativi, svincolandoci parzialmente dalla fatica dello sviluppo del materiale; l'interazione, mediante apposite librerie, con altri programmi, in particolare, ma non solo, per quanto riguarda la musica elettronica. [18]

4. BIBLIOGRAFIA

[1] L. B. Meyer, *Emozione e significato nella musica*. Il Mulino, Bologna, 1992.

[2] I. Bent, *Analisi musicale*. EDT, Torino, 1990.

[3] C. Agon, G. Assayag, and J. Bresson, *The OM Composer's Book. Volume One*. Editions Delatour France/Ircam, Parigi, 2006.

[4] J. Bresson, C. Agon, and G. Assayag, *The OM Composer's Book. Volume Two*. Editions Delatour France/Ircam, Parigi, 2008.

[5] M. Kuuskankare and M. Laurson, "Survey of music analysis and visualization tools in PWGL," in *Proceedings of International Computer Music Conference*, pp. 372–375, 2008.

[6] J. Bresson and C. Pérez-Sancho, "New Framework for Score Segmentation and Analysis in OpenMusic," in *Proceedings of the 9th Sound and Music Computing Conference*, pp. 506–513, 2012.

[7] Y.-K. Ahn, C. Agon, and M. Andreatta, ""Structures Ia pour deux pianos" by Boulez: towards creative analysis using OpenMusic and Rubato," in *Mathematics and Computation in Music*, pp. 412–418, 2009.

[8] M. Laurson, M. Kuuskankare, and K. Kuitunen, "Introduction to Computer-Assisted Music Analysis In PWGL," in *Sound and Music Computing '05*, 2005.

[9] M. Giommoni, *Gli algoritmi della musica*. Cleup sc, Padova, 2011.

[10] G. Ligeti, "Pierre Boulez," in *die Reihe, n. 4*, pp. 36–62, 1958.

[11] M. Babbitt, "Twelve-Tone Invariants as Compositional Determinants," in *Musical Quaterly, n. 46*, pp. 246–259, 1955.

[12] A. Orcalli, *Fenomenologia della musica sperimentale*. Sonus Edizioni, Potenza, 1993.

[13] F. Donatoni, *Il sigaro di Armando*. Spirali, Milano, 1982.

[17] Si veda [13], p. 83

[18] Non abbiamo accennato alle funzioni audio di PWGL, essendo queste al di fuori della presente trattazione. I formati di esportazione XML e MIDI delle partiture permettono una migliore interazione con programmi di scrittura musicale. La libreria pwgl2bach permette l'importazione e l'esportazione in questo formato, utilizzabile anche in Max attraverso la libreria bach. È stata infine da poco rilasciata, al momento in cui scriviamo, PWCsound, la libreria di Giorgio Zucco, dedicata all'interazione di PWGL con Csound.

ALGORITMI GENETICI PER LA COMPOSIZIONE ASSISTITA IN AMBIENTE *REAL TIME*

Francesco Bianchi
Conservatorio "G. Verdi" di Torino
frabianchi72@gmail.com

Andrea Agostini
Conservatorio "G. F. Ghedini" di Cuneo
and.agos@gmail.com

SOMMARIO

Nel panorama della ricerca euristica trovano spazio alcuni algoritmi ispirati a processi naturali. A questa categoria appartengono anche gli Algoritmi Genetici, che si ispirano ai meccanismi dell'evoluzione naturale per la risoluzione di un'ampia gamma di problemi. Grazie alla relativa semplicità di implementazione e all'efficienza il loro uso si è molto diffuso negli ultimi 30 anni in ambiti come l'elettronica, la biologia e l'economia. Dopo un breve *excursus* storico sull'evoluzione di questi algoritmi, e una disamina di alcune loro applicazioni in contesti musicali, il presente lavoro ha il duplice obiettivo di presentare la libreria *bios*, un insieme di *external* per il software Max che implementano alcune categorie di algoritmi genetici, e di descrivere con una serie di esempi alcune metodologie e strategie del loro impiego nell'ambito della composizione assistita.

1. INTRODUZIONE

Negli ultimi decenni del secolo scorso sono state sviluppate numerose famiglie di algoritmi ispirati a processi naturali, come le Reti Neurali, la *Swarm Intelligence*, l'*Ant Colony Optimization* e gli Algoritmi Genetici. Tutti questi procedimenti appartengono alla classe di algoritmi di ricerca euristica, che tenta di trovare soluzioni seguendo indicazioni momentanee che si sviluppano durante la ricerca stessa. Le tecniche euristiche sono spesso utilizzate nella soluzione di problemi per i quali una ricerca deterministica non porterebbe a risultati in tempi accettabili.

Inventati a metà degli anni '70, gli Algoritmi Genetici si ispirano all'adattamento e alla selezione naturale imitandone le caratteristiche di base, i processi di funzionamento e mutuandone la terminologia. Il primo algoritmo genetico, definito ormai come classico, compare in un testo dedicato all'argomento da John H. Holland nel 1975[1]. Il suo meccanismo si basa sulla generazione pseudocasuale di un insieme di possibili soluzioni a un problema, che vengono valutate una ad una; le migliori sono quindi selezionate e ricombinate fra loro al fine di formarne di nuove e sempre migliori. Il ciclo ricomincia finché non viene trovata una soluzione accettabile. Come tutti i metodi euristici, un algoritmo genetico non dà garanzia di convergenza verso la soluzione ottima. Diverse tecniche sono quindi state proposte per ovviare a questo problema, o almeno limitarne l'impatto.

Prescindendo dall'obiettivo finale di un algoritmo, che è quello di fornire una soluzione, ciò che rende interessante un algoritmo genetico è il percorso di ricerca della soluzione che, in base all'architettura con cui è stato progettato, può assumere svariate configurazioni, accomunate dall'assenza di linearità. Anche nel caso dei percorsi più fluidi, che potremmo definire di tipo *morphing*, le approssimazioni successive con cui si converge verso la soluzione non seguono un ordine prevedibile. Questa assenza di linearità è probabilmente l'elemento che rende più appetibile l'uso di tali algoritmi nella composizione assistita non solo al fine di trovare una soluzione, ma in funzione della progettazione di processi basati sui percorsi di ricerca.

Gli oggetti che verranno esaminati nel presente lavoro fanno parte di una libreria di *external* [1] per Max [2] (chiamato fino al 2011 Max/MSP) che implementano un insieme di algoritmi dedicati alla Vita Artificiale. La libreria, un *work in progress* attualmente in fase di sviluppo che prende il nome di *bios* [3], comprenderà anche oggetti per la manipolazione e generazione di Automi Cellulari, *Swarm Intelligence*, Biomorfi e altri algoritmi, più alcune *utility* per la gestione di liste numeriche. Al momento della scrittura di questo testo sono disponibili gli oggetti che implementano due tipi di algoritmi genetici (da ora in poi GA): GA classico con codifica binaria (*bios.BGA*) e intera (*bios.IGA*), e GA combinatorio (*bios.PGA* e *bios.PGAinternal*) che opera sulle permutazioni di una sequenza.

2. ANTECEDENTI STORICI

Probabilmente la prima esperienza relativa all'uso di procedimenti genetici nell'ambito della composizione algoritmica risale al 1991, quando Andrew Horner e David Goldberg[2] hanno descritto la possibilità di manipolare materiale melodico attraverso una tecnica definita dagli autori "thematic bridging", che consiste nel modificare un *pattern* iniziale attraverso un piccolo insieme di operazioni elementari (rotazione di un segmento melodico, mutazione o eliminazione di una nota); il *pattern* risultante viene quindi confrontato con un *target* per ottenere un valore di *fitness*, cioè un numero che esprima il livello di somiglianza con il *target* stesso. Attraverso l'iterazione di questo processo e la selezione e ricombinazione ripetuta dei *pattern* con più alti

Copyright: ©2014 Francesco Bianchi, Andrea Agostini. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

[1] moduli addizionali programmati in linguaggio C/C++
[2] www.cycling74.com
[3] http://bioslib.blogspot.it/

valori di *fitness*, si costruisce una transizione melodica non lineare dal *pattern* originale al *target* stabilito.

I lavori di Andrew Gartland-Jones[3] trattano la questione più importante che investe l'uso degli algoritmi genetici nell'ambito della composizione assistita: il processo trasformazionale e la sua somiglianza con il processo compositivo. A questo proposito Gartland-Jones[3, p. 2-3]:

> A commonly used compositional process may be described as taking an existing musical idea and changing it in some way. Musicians in various styles and genres may follow the process differently, [...] but what is most often practiced is taking an existing idea and mutating it to provide a new idea. [...] With the core elements of GA's being mutation (including cross-over) and fitness assessment, there appears to be a strong correlation with at least some aspects of the human process of generating musical material.

Questi lavori sono accomunati dall'idea di *sottomettere* i classici operatori di un GA a esigenze specifiche del contesto compositivo (altezze, durate o punti d'attacco) rendendo inutile l'uso di uno stesso algoritmo per esigenze diverse. Non si tratta quindi di algoritmi astratti, di uso generale: ognuno di essi è pensato per essere applicato a uno specifico parametro musicale. Ciò si riflette anche nel tipo di tecniche trasformazionali utilizzate, che appaiono tipiche di procedimenti impiegati da compositori che lavorano con un approccio combinatorio al materiale musicale, e sostanzialmente indipendenti dal loro inserimento in un GA. In altre parole, l'azione è volta a compiere un processo di emulazione di un pensiero compositivo relativamente tradizionale.

Il compositore e artista mediale americano Gary Lee Nelson[4] a metà degli anni '90 usa invece gli algoritmi genetici per far evolvere *pattern* ritmici senza adattare gli operatori di *crossover* e *mutazione* alle esigenze del processo, ma usando la più classica delle codifiche dei GA e compiendo così quello che si potrebbe definire un processo di sonificazione.

In anni più recenti sono stati sviluppati sistemi interattivi in tempo reale (IGA, *Interactive Genetic Algorithms*). Al Biles[5] ha creato GenJam, un generatore di improvvisazioni jazzistiche basato su GA, che produce frammenti melodici valutati da un utente che assegna loro i valori di *fitness*. Successivamente lo stesso Al Biles, con Peter G. Anderson and Laura W. Loggi[6] hanno implementato un sistema dove la funzione di *fitness* è realizzata da una rete neurale che viene addestrata tramite analisi statistiche dei frammenti melodici prodotti. Interattivo è anche Vox Populi, di J. Manzolli[7], che genera strutture musicali in tempo reale, valutate da una funzione di *fitness* automatica che può anche essere influenzata dall'intervento dell'utente.

Infine vale la pena evidenziare il lavoro di Palle Dahlstedt e Matts Nordahl, citati da Nierhaus[8, p. 182-183], che usano i GA per creare mondi artificiali (*Living Melodies*) di creature *soniche* che possono muoversi, interagire fra loro, mangiare e cercare dei compagni con cui accoppiarsi. Gli individui sono caratterizzati da un doppio cromosoma, una parte del quale è costituita da una melodia e l'altra dalle informazioni sulle operazioni da svolgere.

3. TERMINOLOGIA E ALGORITMO

Gli elementi costitutivi di un GA sono i *cromosomi*, detti anche individui, ognuno dei quali rappresenta una possibile soluzione al problema. Un cromosoma è una sequenza di *geni*, ognuno dei quali può assumere un determinato *allele*, cioè un valore a seconda del tipo di codifica del GA. Nel caso di un GA binario un gene avrà sempre e solo alleli pari a zero o uno. La posizione di un gene nella sequenza cromosomica è detta *locus*. Un insieme di cromosomi costituisce una popolazione di individui.

Ogni cromosoma viene selezionato tramite una funzione di valutazione che si occupa di attribuire a ogni individuo un punteggio di *fitness* che tipicamente assume un valore compreso fra 0 e 1. Gli individui migliori vengono ricombinati fra loro per dare vita a nuovi cromosomi. Tale processo viene definito *crossover*. I nuovi individui generati sostituiscono gli individui peggiori della popolazione dando vita a una nuova generazione. Al fine di evitare il ristagno del materiale genetico, un ulteriore operatore, detto di *mutazione*, si occupa di cambiare casualmente l'allele di uno o più geni di un individuo con una probabilità predefinita.

I passi per implementare un GA classico sono i seguenti:

1. Generazione della popolazione iniziale;
2. Calcolo del valore di *fitness* di ogni individuo;
3. Selezione degli individui con *fitness* migliore;
4. Applicazione dell'operatore di *crossover* agli individui migliori;
5. Applicazione dell'operatore di mutazione ai nuovi individui generati;
6. Costituzione della nuova generazione;
7. Se la soluzione ottenuta non è soddisfacente ricominciare dal punto 2, altrimenti uscire.

4. ALGORITMI GENETICI IN *BIOS*

In tutti gli oggetti della libreria *bios*, la metà migliore della popolazione viene mantenuta nella generazione successiva, mentre l'altra metà viene sostituita con nuovi individui. I diversi oggetti differiscono invece nella natura dei cromosomi: negli algoritmi a codifica binaria e intera le sequenze sono stringhe generate pseudocasualmente, mentre negli algoritmi combinatori sono permutazioni (senza ripetizioni) di un numero dato di elementi. Ogni individuo contiene le informazioni necessarie a rappresentare una o più variabili del problema.

4.1 Codifica a numeri interi

Nel caso della rappresentazione con interi, i parametri necessari sono:

1. Dimensione della popolazione (multiplo di 4);
2. Numero di variabili;

3. Minimo e massimo valore di una variabile.

Ogni gene del cromosoma rappresenta una variabile, definita come un intero compreso fra un minimo e un massimo. Ad esempio, nel cromosoma $[44, 30, 61, 72, 4]$, il primo *locus* contiene la variabile con valore 44, il secondo 30 e così via.

E' possibile impostare uno fra quattro tipi di *crossover*:

1. Single Point Crossover;
2. Two-Point Crossover;
3. Uniform Crossover;
4. Average Crossover.

Con il *single-point crossover* viene scelto casualmente un punto (*splitpoint*) del cromosoma, dopo il quale avviene lo scambio del materiale genetico. I due nuovi individui avranno una sezione di cromosoma di un genitore e, a partire dallo *splitpoint*, una sezione dell'altro. Nel caso del *two-point crossover*, dati due individui genitori P_1 e P_2, si scelgono casualmente due *splitpoint*. Il segmento compreso fra i due punti viene invertito fra i due genitori, il resto rimane invariato. Questo processo produrrà due cromosomi figli (O_1 e O_2):

P_1:	31	19	|	61	72	4	|	12	70
P_2:	*30*	*22*	|	*19*	*40*	*34*	|	*18*	*1*
O_1:	31	19	|	*19*	*40*	*34*	|	12	70
O_2:	*30*	*22*	|	61	72	4	|	*18*	*1*

L'*uniform crossover* invece fa uso di una *maschera di crossover* (M) casuale, generata prima di ogni accoppiamento. Si tratta di una stringa binaria di lunghezza pari al cromosoma. In corrispondenza di un 1 le variabili vengono scambiate, altrimenti non c'è scambio:

P_1:	31	19	61	72	4	12	70
P_2:	*30*	*22*	*19*	*40*	*34*	*18*	*1*
M:	0	0	1	0	1	1	0
O_1:	31	19	*19*	72	*34*	*18*	70
O_2:	*30*	*22*	61	*40*	4	12	*1*

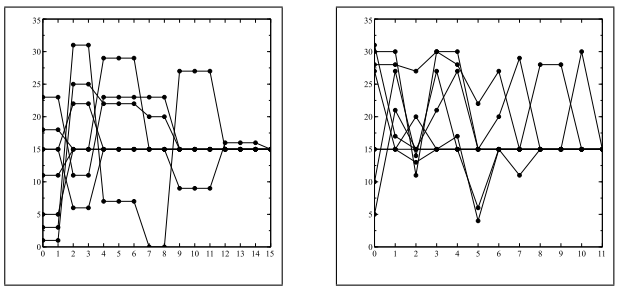
L'operatore di *average crossover* produce due cromosomi, ottenuti il primo dalla media aritmetica delle variabili dei genitori, l'altro dalla media geometrica.

L'operatore di mutazione infine cambia il valore di una o più variabili secondo una probabilità predefinita espressa in percentuale.

4.2 Codifica binaria

Nel caso della rappresentazione binaria, i parametri sono:

1. Dimensione della popolazione (multiplo di 4);
2. Numero di variabili;



(a) GA binario (b) GA intero

Figura 1: Evidenza del comportamento di algoritmi con codifiche diverse. L'ascissa rappresenta le generazioni, l'ordinata il valore assunto da una variabile.

3. Numero di bit necessari a rappresentare ogni variabile.

Un cromosoma è una sequenza di bit; ogni variabile è rappresentata da una porzione di tale sequenza. Nell'esempio seguente il cromosoma ha una dimensione di 24 bit, divisi fra tre variabili di 8 bit:

$$\underbrace{11111111}_{255}\underbrace{01001101}_{77}\underbrace{10001100}_{140}$$

Gli operatori di *crossover* sono i primi tre della codifica con interi (*single point*, *two-point*, *uniform*), mentre l'operatore di mutazione varia uno o più bit di un cromosoma secondo una probabilità prestabilita.

4.3 Differenze di comportamento fra codifiche

La scelta di implementare due algoritmi con codifiche diverse è stata motivata dal fatto che, sebbene entrambi possano trovare una soluzione in tempi accettabili, i profili comportamentali che essi esibiscono nelle generazioni transitorie sono molto diversi. Il GA binario tende a far evolvere l'individuo migliore di ogni generazione con gradualità, mantenendo molti valori in comune con il migliore individuo della generazione precedente. D'altra parte la codifica intera genera individui migliori che possono cambiare repentinamente da una generazione all'altra, pur mantenendo la tendenza a migliorare il valore di fitness. Graficamente (figura 1) il GA binario mostra un andamento a scalini, mentre quello a codifica intera è spigoloso. In funzione della ricerca della soluzione questa differenza è poco rilevante, ma in termini di uso delle transizioni per fini musicali può essere importante.

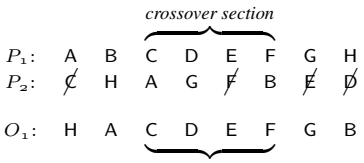
4.4 Algoritmo combinatorio

Il terzo tipo di algoritmo trae ispirazione da Whitley[9], che descrive un sistema per implementare algoritmi genetici per problemi permutativi. I parametri richiesti sono:

1. Dimensione della popolazione (multiplo di 4);
2. Numero di variabili.

Ogni individuo è una permutazione di una sequenza ordinata che va da 0 a $n-1$ dove n è il numero delle variabili. Può essere comodo pensare a un cromosoma come a una sequenza di indici di una tabella.

L'operatore di *crossover*, per evitare ripetizioni all'interno del cromosoma, funziona secondo un meccanismo peculiare: dati due genitori P_1 e P_2, a turno uno di essi svolge la funzione di *cut string*, l'altro quella di *filler string*. Dalla *cut string* si sceglie la sezione di cromosoma (*crossover section*) da trasmettere al figlio, mentre dalla *filler string* si prendono gli elementi restanti e si inseriscono nel nuovo individuo in ordine di apparizione. Successivamente si compie la stessa operazione invertendo i ruoli. Ad esempio:



L'operatore di mutazione per questo algoritmo interviene sostituendo un intero individuo con una nuova permutazione.

5. LA FUNZIONE DI FITNESS

5.1 Implementazione

Tutti i passi dell'algoritmo sono inglobati all'interno di un oggetto che svolge le operazioni di calcolo dopo aver ricevuto alcuni parametri necessari, che si limitano a definire il tipo di *crossover* e la probabilità di mutazione, senza cambiare sostanzialmente l'architettura dell'algoritmo stesso. La funzione di *fitness* è invece diversa di volta in volta. L'algoritmo non sa come manipolare le variabili e quali calcoli effettuare per attribuire agli individui il valore di *fitness* appropriato. E' stato quindi necessario implementare un sistema che permettesse di gestire la funzione all'esterno dell'oggetto e che reinviasse all'oggetto i valori della stessa. A questo scopo è stato adottato il design pattern chiamato *lambda loop*, progettato e impiegato in vari oggetti della libreria *bach* da Andrea Agostini e Daniele Ghisi[10].

Tutti gli oggetti della libreria hanno un *inlet* e un *outlet* (quelli posti più a destra) che svolgono questo compito. L'*outlet* resituisce un cromosoma per permettere alla funzione di valutarlo, mentre l'*inlet* riceve a sua volta il valore di *fitness* appena assegnato.

5.2 Progettazione della funzione di fitness

L'obiettivo della funzione di *fitness* è quello di giudicare la prestazione di un individuo rispetto al problema che si vuole risolvere. È composta da alcuni parametri che si ritengono importanti ai fini della soluzione del problema e attribuisce ad ogni individuo un valore numerico (convenzionalmente compreso fra 0 e 1) proporzionale alla bontà della soluzione offerta. Per essere ben progettata, una funzione di *fitness* deve acquisire le variabili inglobate nei cromosomi, elaborarle e infine restituire un valore di valutazione adeguato.

Le implementazioni più semplici di funzioni di *fitness* hanno luogo quando il problema richiede di raggiungere un obiettivo prefissato. In questo caso è sufficiente stabilire a priori la lista di valori da raggiungere e confrontarla con gli individui della popolazione di ogni generazione. Questo tipo di funzione è detta *Target Fitness Function* o TFF.

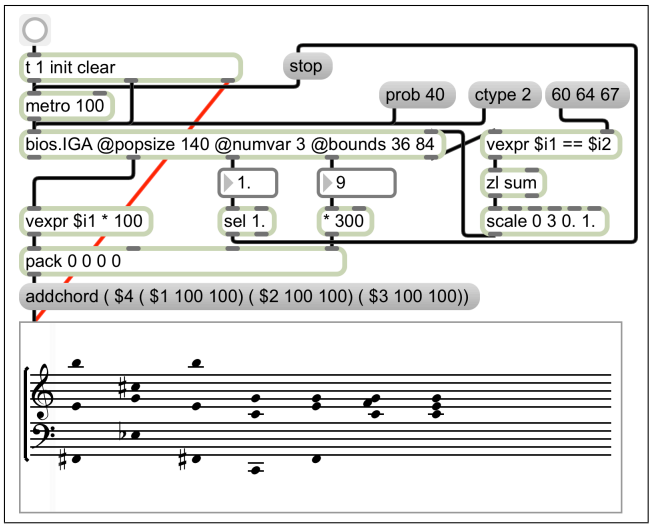


Figura 2: Algoritmo di ricerca di un accordo con valutazione binaria.

Il primo esempio usa l'algoritmo genetico per trovare l'accordo di do maggiore in posizione $\frac{5}{3}$ a partire dal do centrale del pianoforte (nota MIDI 60) [4] .

L'esempio in figura 2 mostra la *patch* di Max, con codifica a numeri interi. La dimensione della popolazione è pari a 140 individui, il numero delle variabili è 3 mentre il dominio di ricerca è [36,84], cioè da due ottave sotto a due ottave sopra il do centrale. La scelta della dimensione della popolazione dipende dalla quantità di variabili del problema e dalla velocità con cui si vuole ottenere una soluzione. Non esistono formule per stabilirne l'adeguata dimensione, quindi spesso è necessario affidarsi a prove empiriche.

La funzione di *fitness* non fa altro che confrontare la lista in entrata con la lista *target*. Per ogni valore 'giusto' si assegna 1, altrimenti zero.

Questo genere di valutazione è detto 'binario' ed è quello più diffuso in letteratura, anche se ha lo svantaggio di attribuire lo stesso punteggio a tutti i valori sbagliati. Con una valutazione 'continua' invece è possibile attribuire ad ogni variabile un valore in base al grado di prossimità con il valore *target*. Come mostrato in figura 3 il valore di *fitness* varia in base alla prossimità delle note ai propri valori *target*. Il percorso accordale quindi è sensibilmente diverso e privilegia gli accordi che ruotano attorno a quello di arrivo.

Un'altra categoria di funzioni, definita *Constraints Fitness Function* o CFF, valuta un individuo in base alla rispondenza dello stesso a determinati requisiti, o vincoli. Piuttosto che confrontare la lista di variabili con la lista *target* questo tipo di funzione verifica che le variabili e le relazioni fra le variabili possiedano determinati requisiti e rientrino in un determinato dominio. Il prossimo esempio (figura 4) mostra una funzione che valuta un individuo al fine di trovare una sequenza di 15 intervalli che formino un

[4] Per la grafica notazionale è stata usata la libreria *bach: automated composer's helper* per il software Max, sviluppata da Andrea Agostini e Daniele Ghisi.

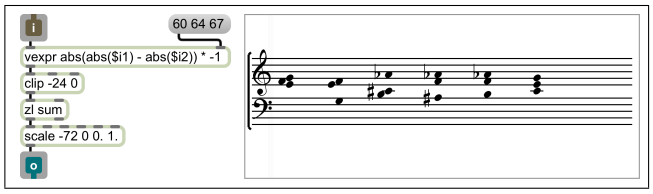


Figura 3: Funzione con valutazione continua. Per il resto l'algoritmo è lo stesso dell'esempio in figura 2.

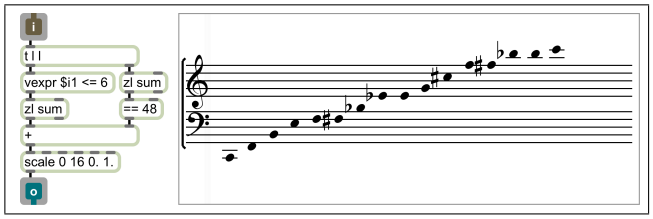


Figura 4: *Constraints Fitness Function* e relativo risultato.

arpeggio di quattro ottave esatte. Inoltre ogni intervallo non deve superare l'estensione di sei semitoni.

Questo approccio consente di definire funzioni per realizzare ricerche di soluzioni complesse, come la generazione di melodie con determinati profili, l'armonizzazione o la generazione di spettri armonici. A titolo di esempio, la figura 5 mostra una parte della funzione di *fitness* per generare un *bicinium* in contrappunto severo, costituita da una sezione per analizzare il *canto dato* e da alcune *subpatch* che si occupano di verificare il soddisfacimento dei vincoli.

6. ALGORITMO COMBINATORIO

L'oggetto *bios.PGA* implementa l'algoritmo genetico combinatorio. Un problema classico che può essere risolto con tale oggetto è la costruzione di una serie dodecafonica omni-intervallare, cioè contenente tutti gli intervalli possibili all'interno di un'ottava. Si tratta di un problema per cui sono note soluzioni estremamente più efficaci rispetto a un algoritmo genetico. Il vantaggio di quest'ultimo è la semplicità della funzione di *fitness*: premesso che le variabili rappresentino le altezze, è sufficiente calcolare gli intervalli assoluti fra le variabili e verificare che tale lista, dopo aver eliminato i doppioni, risulti avere lunghezza pari a 11 (figura 6).

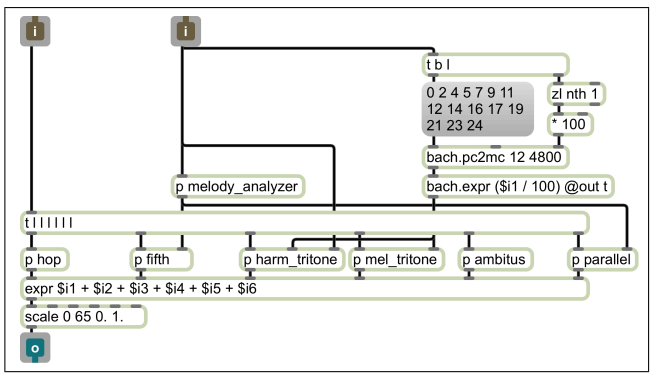


Figura 5: Funzione per generare un *bicinium*.

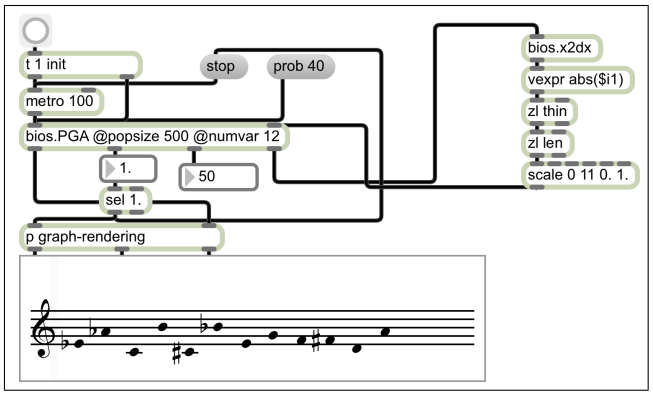


Figura 6: GA combinatorio. L'oggetto *bios.x2dx* restituisce una lista composta dalle distanze fra gli elementi della lista in entrata.

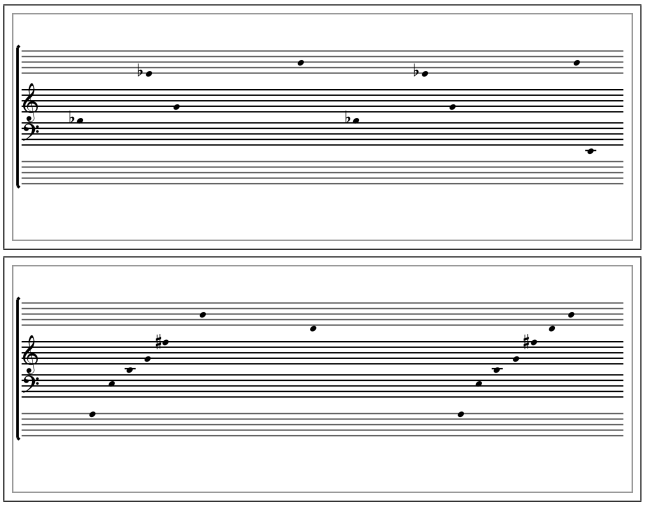


Figura 7: *Incipit* e sezione finale di Thirty.

Per generalizzare l'uso dell'algoritmo combinatorio, è possibile interpretare le variabili come indici di un vettore che può contenere qualunque cosa. Nella sezione seguente c'è la descrizione di come l'algoritmo genetico combinatorio sia stato usato adottando questa stategia per costruire la forma complessiva di un brano.

Rispetto agli altri, l'algoritmo combinatorio restituisce sequenze meno interessanti di passaggi intermedi: esso infatti si limita a manipolare permutazioni senza ripetizioni degli stessi elementi. Per questa ragione è stato anche implementato in una versione *internal*, che restituisce solo la soluzione finale in caso di successo.

7. UN ESEMPIO COMPOSITIVO

Il brano per Disklavier *Thirty* di Francesco Bianchi è stato composto usando algoritmi genetici per generare le due figure principali: arpeggio e accordo. Per la generazione degli arpeggi è stato usato un GA per le altezze e uno per i punti di attacco. Entrambi hanno una *Target Fitness Function*. L'algoritmo per le altezze cerca un aggregato costituito da 7 suoni ([33, 52, 60, 78, 78, 86, 95] in valori MIDI) mentre i punti di attacco sono definiti da una lista di posizioni ([0, 200, 400, 600, 800, 1000, 1200] in millisecondi).

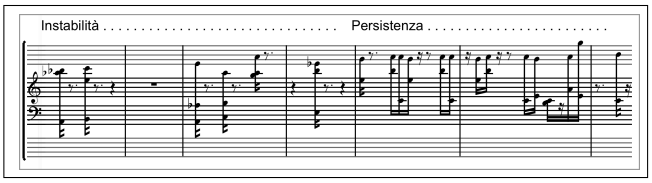


Figura 8: Sezione di *Thirty* con il primo ciclo di alternanze fra zone di instabilità e zone di persistenza armonica.

La generazione delle figure accordali è stata realizzata tramite un uso meno convenzionale di un GA. Tutti gli oggetti di *bios* accettano il messaggio *totrefresh*, che consente di effettuare un *reset* totale della popolazione ogni n generazioni. L'uso convenzionale di questo messaggo permette di evitare che l'algoritmo ristagni e converga verso soluzioni inaccettabili. Nel brano in questione il messaggio *totrefresh* viene usato a prescindere dal raggiungimento dell'obiettivo: data la semplicità della funzione di *fitness*, ci si aspetta che il GA converga verso una soluzione entro poche generazioni, per poi tornare in uno stato di 'caos genetico' (nel momento del *refresh* della popolazione). Questa tecnica ha permesso di generare alternanza fra zone di instabilità e zone di persistenza armonica. È utile notare che la funzione di *fitness* per questo algoritmo valuta gli accordi non secondo le altezze assolute che li costituiscono, ma in base all'appartenenza delle stesse a determinate *pitch-class*: quindi di volta in volta l'individuo con *fitness* massima cambia generando zone di persistenza armonica, ma non di fissità delle altezze (figura 8).

Al termine della creazione del materiale è stata creata una lista con gli eventi del gesto 'arpeggio' seguiti da quelli del gesto 'accordo'. Un GA combinatorio ha generato a partire da questa lista una permutazione con determinati requisiti. Tale permutazione ha fornito la base per la distribuzione di tutto il materiale nella *timeline* del brano.

8. CONCLUSIONI E LAVORI FUTURI

Le precedenti implementazioni di GA in campo musicale e la libreria *bios*, differiscono per l'architettura del sistema e per l'orientamento. I primi sono pensati per scopi predefiniti e inglobano all'interno dell'algoritmo la funzione (o un set di funzioni) di *fitness*, mentre *bios* è un sistema dall'orientamento più generale, in cui la funzione di *fitness* si può progettare in base all'esigenza. Per tali ragioni la comparazione con gli altri sistemi in termini di prestazioni è strettamente dipendente dalla complessità della funzione. Un software come Vox Populi ha mostrato notevole efficacia per gli scopi per cui è stato progettato, ma *bios* si è mostrato più versatile e in grado di affrontare un insieme eterogeneo di problemi, a scapito, talvolta, dell'efficienza di fronte a funzioni di *fitness* molto complesse e di popolazioni grandi (oltre 3200 individui). Uno dei prossimi obiettivi è proprio una generale ottimizzazione dell'algoritmo in modo da ridurre i consumi di CPU e i tempi di calcolo.

Allo stato attuale è in lavorazione un nuovo oggetto, che implementa un GA binario con popolazione clonata, cioè formata da individui identici (*bios.cloneBGA*), che permette all'utente di inserire a piacimento l'individuo di partenza al fine di realizzare interpolazioni complesse fra individui.

Per i GA binari e interi i tipi di operatori di *crossover* e di mutazione sono quelli classici. Sarà quindi interessante implementare nuovi tipi di operatori al fine di raffinare i percorsi di ricerca in funzione di specifiche esigenze musicali. Completata la sezione sugli Algoritmi Genetici la libreria verrà arricchita con un insieme di oggetti che implementano gli Automi Cellulari. Il primo di tali oggetti, *bios.ECA*, è già in fase di *testing*. A seguire un insieme di algoritmi per gli *L-Systems* e la *Swarm Intelligence*.

La versione per Max (a 64 bit) esiste attualmente solo per Mac OSX, mentre alcuni oggetti sono stati portati su Pure Data [5] su Mac OSX e GNU/Linux. Nelle prossime settimane il *porting* per Pure Data verrà completato e la libreria verrà compilata anche su Windows (sia per Max che per Pure Data).

9. BIBLIOGRAFIA

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[2] A. Horner and D. Goldberg, "Genetic algorithms and computer-assisted music composition," in *Proceedings of the 1991 International Computer Music Conference*, San Francisco, 1991.

[3] A. Gartland-Jones, "Can a genetic algorithm think like a composer?," in *Proceedings of 5th International Conference on Generative Art*, Milano, 2002.

[4] G. L. Nelson, *Sonomorphs: An Application of Genetic Algorithms to the Growth and Development of Musical Organisms*. unpublished paper, Oberlin (USA): Conservatory of Music, 1995.

[5] J. A. Biles, "GenJam: a genetic algorithm for generating jazz solos," in *Proceedings of the 1994 International Computer Music Conference*, San Francisco, 1994.

[6] J. Biles, P. Anderson, and L. Loggi, "A neural network fitness function for a musical IGA," in *Anderson PG, Warwick K (eds) IIA '96/SOCO '96*, pp. B39–B44, ICSC, Millet, Alberta, 1996.

[7] J. Manzolli, F. Moroni, R. V. Zuben, and R. Gudwin, "An evolutionary approach applied to algorithmic composition," in *Proceedings of SBC99 - XIX National Congress of the Computation*, Rio de Janeiro, 1999.

[8] G. Nierhaus, *Algorithmic Composition: Paradigms of Automated Music Generation*. Springer, 2009.

[9] D. Whitley and N. Yoo, *Modeling Simple Genetic Algorithms for Permutation Problems*. Computer Science Departement, Colorado State University, 1995.

[10] A. Agostini and D. Ghisi, "*bach*: una libreria per la composizione assistita in Max," in *Proceedings of 19th CIM*, (Trieste, Italy), pp. 150–157, 2012.

[5] `http://puredata.info`

CAGE: UNA LIBRERIA DI ALTO LIVELLO PER LA COMPOSIZIONE ASSISTITA DA COMPUTER IN TEMPO REALE

Andrea Agostini
HES-SO, Ginevra
Conservatorio di Cuneo
and.agos@gmail.com

Éric Daubresse
HES-SO, Ginevra
eric.daubresse@hesge.ch

Daniele Ghisi
HES-SO, Ginevra
danieleghisi@gmail.com

SOMMARIO

Questo articolo è un'introduzione a *cage*, una libreria per l'ambiente Max [1] costituita da moduli di alto livello per la composizione assistita da computer (CAC). La libreria, che al momento è ancora nella fase denominata *alpha*, è composta da un insieme di strumenti che facilitano la manipolazione dei dati musicali simbolici e che risolvono una serie di problemi tipici della CAC, come ad esempio generazione di note o di profili melodici, interpolazione armonica o ritmica, operazioni ispirate a processi tipici del *digital signal processing*, automi cellulari e L-sistemi, moduli per la *musical set theory*, strumenti per la generazione e la gestione di partiture. Questo progetto, supportato dalla *Haute École de Musique* di Ginevra, ha una connotazione spiccatamente pedagogica: tutti i moduli nella libreria sono astrazioni, che si prestano quindi ad essere facilmente analizzate e modificate.

1. INTRODUZIONE

Questo articolo descrive alcuni dei concetti principali e dei componenti della libreria *cage* [2] per Max, contenente diversi moduli di alto livello per la composizione assistita da computer (CAC). In questo articolo, versione italiana di [1], completiamo la visione d'insieme della libreria e forniamo una panoramica sui suoi obiettivi.

cage è interamente basata sulla libreria *bach: automated composer's helper*, che è sviluppata da due degli autori [2, 3]. *bach* è una libreria di circa 200 oggetti e astrazioni finalizzata a dotare i Max un insieme di 'primitive' per la manipolazione di dati simbolici musicali, insieme ad alcune interfacce grafiche per la loro rappresentazione e manipolazione. All'interno di *bach*, i dati sono rappresentati costantemente attraverso usi specializzati di una generica struttura chiamata *llll* ('Lisp-like linked list') che è essenzialmente, come l'acronimo suggerisce, una struttura ad albero nella forma di una lista annidata, ispirata direttamente dal linguaggio di programmazione Lisp.

[1] *http://cycling74.com*
[2] *www.bachproject.net/cage*

Copyright: ©2014 Andrea Agostini, Éric Daubresse, Daniele Ghisi. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Di conseguenza, i moduli di *bach* sono per lo più strumenti per la manipolazione a basso livello di *llll* (per esempio per compiere rotazioni, sostituzioni, inversioni...) o per operazioni più complesse ma essenzialmente basilari, come la risoluzione di problemi a vincoli o la quantizzazione ritmica.

Diversamente da *bach*, i moduli di *cage* compiono in generale operazioni di più alto livello, con una connotazione più musicale che tecnica (ad esempio: generazione di materiale melodico, o calcolo di una modulazione di frequenza simbolica). In ogni caso, *cage* eredita da *bach* alcuni meccanismi e principi fondamentali, come ad esempio il fatto che la comunicazione tra moduli differenti avviene, la maggior parte delle volte, attraverso *llll*.

Due criteri principali hanno presieduto all'ideazione e allo sviluppo della libreria.

Il primo è l'idea che si trova all'origine di *cage*: la volontà di costruire una libreria di moduli pronti all'uso, che implementino un certo numero di processi di CAC largamente utilizzati. Di conseguenza, una parte della libreria è ispirata apertamente a librerie già esistenti per altri programmi (in particolare le librerie *Profile* [4] ed *Esquisse* [5, 6] per *Patchwork*, portate successivamente in *OpenMusic*); un'altra parte della libreria è invece esplicitamente dedicata a problemi tipicamente associati con interazioni in tempo reale (come ad esempio *cage.granulate*, il motore di granulazione simbolica).

Il secondo criterio è una forte connotazione pedagogica [3]: tutti i moduli della libreria sono astrazioni, che si prestano quindi ad essere facilmente analizzate e modificate. Non è difficile, per l'utilizzatore che voglia imparare a trattare i dati musicali, copiare, modificare o ritoccare le patch per le proprie necessità. In quest'ottica, tutti gli oggetti della libreria sono intrinsecamente *open source*: sebbene ciascun processo implementato sia concepito per un utilizzo sostanzialmente standard, l'utilizzatore avanzato potrà partire facilmente da queste astrazioni e modificarne il comportamento. Questa vocazione pedagogica è completata dal fatto che la libreria sarà meticolosamente documentata tramite file di aiuto, fogli di riferimento e una collezione di tutorials.

[3] La libreria *cage* è supportata dalla HES-SO.

2. UN APPROCCIO IN TEMPO REALE ALLA COMPOSIZIONE ASSISTITA DA COMPUTER

Il paradigma del tempo reale influenza profondamente la natura stessa del processo compositivo. Per esempio, compositori che lavorano nel dominio della musica elettroacustica spesso hanno bisogno che il computer si adatti immediatamente a ogni cambiamento di parametri. Similmente, compositori che lavorano con dati simbolici potrebbero volere che il computer si adatti nel più breve tempo possibile alla nuova configurazione di dati. Il paradigma che soggiace a *cage* è lo stesso che ha improntato la libreria *bach*: creare e modificare dati simbolici non è necessariamente un'attività fuori dal tempo, ma segue il flusso temporale compositivo, e si adatta ad esso (si veda anche [3, 7, 8]).

3. COMPOSIZIONE DELLA LIBRERIA

La libreria *cage* è composta da diverse famiglie di moduli, che descriveremo più dettagliatamente nei prossimi paragrafi, non prima di aver però sottolineato che il progetto non ha ambizioni di completezza. La composizione assistita da computer è un dominio molto vasto, e le pratiche sono solitamente personali: le specificità del singolo compositore spesso prevalgono sulla messa a punto di procedimenti standardizzati. In ogni caso, ci è sembrato di poter identificare, come linea guida per la formazione della libreria, alcune tipologie generale di approccio e alcune operazioni specifiche di uso ormai comune. Tentiamo di esemplificare almeno alcune di queste operazioni, sperando che il nostro lavoro sia utile anche ai compositori che volessero partire da esse per implementare i propri processi ed operazioni individuali.

3.1 Generazione di altezze

La prima famiglia di moduli che analizziamo è dedicata alla generazione di altezze, secondo diversi criteri: *cage.scale* e *cage.arpeggio* generano, rispettivamente, scale ed arpeggi all'interno di un'estensione data. Si può specificare il tipo di accordo o di scala sia tramite un nome simbolico (p.e. **F#m**, **ReM**), sia tramite un pattern di *midicents*[4] . I nomi di scale e accordi possono anche contenere alterazioni quartitonali. *cage.harmser* genera invece serie armoniche, a partire da una fondamentale data, con un fattore di distorsione opzionale.

Mentre per i moduli appena citati la generazione di altezze dà come risultato l'intera scala, arpeggio o serie, altri moduli si occupano invece di generare le altezze ad una ad una, in tempo reale: *cage.noterandom* genera aleatoriamente altezze, a partire da un insieme di altezze dato (prendendo anche in considerazione pesi di probabilità opzionali, che possono essere a loro volta generati ad esempio attraverso *cage.weightbuilder*); *cage.notewalk* genera un cammino aleatorio all'interno di un insieme di altezze, data una lista di passi permessi. In entrambi i casi il risultato dell'operazione è progettato per essere utilizzato in combinazione con *bach.transcribe*, che trascriverà in tempo reale il flusso di note in entrata. Inoltre l'elemento selezionato volta per volta può essere convalidato o scartato dall'utente tramite un *lambda loop*.[5]

3.2 Generazione e trattamento di profili melodici

Una famiglia di moduli è specificatamente dedicata alla generazione e al trattamento di profili melodici, in maniera esplicitamente ispirata dalla libreria *Profile* di OpenMusic e PatchWork [4]. Una funzione può essere convertita in una sequenza di altezze (un profilo melodico) grazie a *cage.profile.gen*. Questo profilo può quindi essere modificato in varie maniere: può essere compresso o espanso (con *cage.profile.stretch*), invertito (con *cage.profile.mirror*), approssimato a una griglia armonica o a una scala (con *cage.profile.snap*), costretto all'interno di una certa estensione (con *cage.profile.rectify*), perturbato aleatoriamente (con *cage.profile.perturb*) o filtrato (con *cage.profile.filter*). Il filtraggio del profilo avviene tramite l'applicazione di un filtro mediano, medio o personalizzato - quest'ultimo è definibile dall'utente tramite un *lambda loop* (si veda la Fig. 1).

3.3 Processi ispirati da pratiche elettroacustiche

cage contiene un gruppo di moduli dedicati all'emulazione simbolica di processi appartenenti al dominio della sintesi audio o del *digital signal processing*.

cage.freqshift permette la trasposizione di materiale linearmente sull'asse delle frequenze, come in un modulatore ad anello a banda laterale singola. A causa della stretta relazione tra i due processi, *cage.pitchshift* viene considerato come appartenente alla stessa categoria, benché un'operazione di *pitch shifting* applicata alla notazione musicale corrisponda a una semplice trasposizione.

cage.rm e *cage.fm* calcolano rispettivamente modulazioni ad anello e modulazioni in frequenza. L'idea alla base di tali tecniche, largamente utilizzate da compositori legati allo spettralismo, è la seguente: partendo da due accordi (un accordo 'portante' e uno 'modulante'), le cui note sono assimilate a semplici onde sinusoidali, viene calcolato lo spettro ottenuto modulando reciprocamente questi due gruppi di sinusoidi. Ogni componente dello spettro risultante è quindi rappresentato come una nota nell'accordo risultante. Questa operazione richiede un insieme di approssimazioni e compromessi che possono rendere il risultato significativamente diverso dal comportamento corrispondente in un contesto di trattamento audio: in ogni caso, si tratta di un approccio efficace per generare ricche

[4] *cage*, come *bach*, adotta la convenzione di *Patchwork* e *OpenMusic* di esprimere le altezze e gli intervalli in midicents, ovvero centesimi di nota MIDI.

[5] Un *lambda loop* in *bach* e *cage* è una configurazione di feedback simbolico: i moduli che supportano tale comportamento hanno uno o più *'lambda'* outlet dedicati, che forniscono i dati che devono essere accettati o modificati; tali dati sono quindi processati in una specifica sezione della patch, e il risultato è reimmesso in un *'lambda'* inlet dedicato del modulo originale. Questa configurazione è utilizzata all'interno di *bach* per definire comportamenti personalizzati per operazioni specifiche come ad esempio un criterio di ordinamento, o un processo che debba essere applicato a ogni elemento di una *llll*. Il nome *'lambda'* fa riferimento al fatto che questa configurazione, in un certo senso, permette di passare una sezione di patch come pseudo-argomento di un modulo. Si tratta tuttavia di una mera allusione: nel processo non entrano in gioco né lambda calcolo né funzioni interpretate.

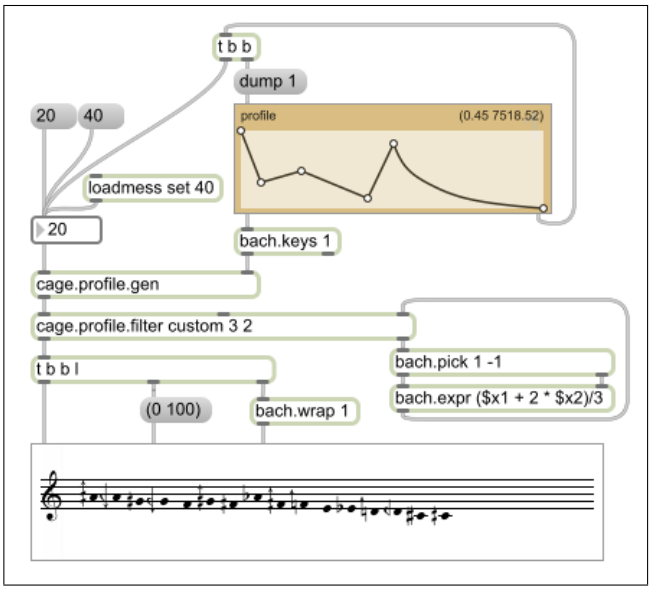


Figura 1. Un profilo melodico è costruito da una funzione definita in un oggetto *bach.slot* e campionata su 20 punti. Il profilo è quindi filtrato da un processo espresso tramite un *lambda loop*, che opera su finestre di tre note; ogni finestra è sostituita con un singolo valore, la media del primo ed ultimo elemento della finestra stessa, pesata con i pesi $(1, 2)$. Questo processo di filtraggio è ripetuto due volte. Si può notare che, a causa del finestraggio, il risultato contiene quattro note in meno del campionamento originale.

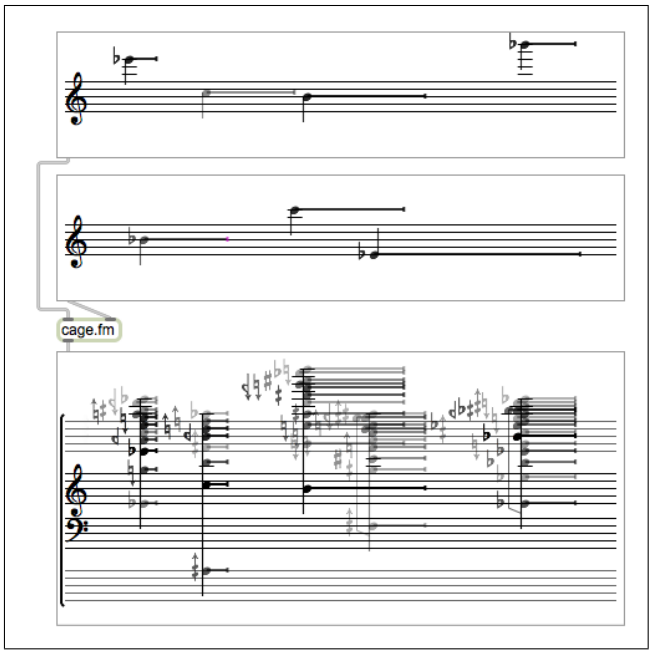


Figura 2. Un esempio di modulazione di frequenza di due partiture, ottenuta attraverso l'astrazione *cage.fm*. La partitura 'portante' e 'modulante' sono in alto, il risultato è in basso. La velocità MIDI delle note (trattata come ampiezza delle corrispettive componenti sinusoidali) è rappresentata in scala di grigi.

famiglie armoniche a partire da materiali semplici, e da ciò discende il suo interesse compositivo.

Benché *cage.rm* e *cage.fm* si ispirino esplicitamente alla libreria *Esquisse* [5, 6] per *OpenMusic*, il loro paradigma operativo e alcuni dettagli computazionali differiscono. In particolare, essendo concepiti per lavorare tenendo conto del parametro temporale, questi due moduli possono accettare non solo semplici accordi, ma anche partiture che rappresentano variazioni di accordi 'portanti' e 'modulanti' nel tempo. In questo caso, l'esito sarà una nuova partitura contenente il risultato di tali variazioni nel tempo (si veda la Fig. 2). Per quanto concerne l'effettivo calcolo interno, i due moduli compiono una stima delle opposizioni di fase generate dalle modulazioni, e cancellano le componenti interessate da tale opposizione, diversamente da quanto accade nella libreria *Esquisse*. Per questa ragione, i risultati dello stesso processo applicato nei due ambienti possono essere significativamente diversi.

cage.virtfun restituisce una o più stime della fondamentale virtuale di un accordo, che si può percepire, ad esempio, dopo l'applicazione di un processo di waveshaping. L'implementazione è estremamente semplice: si percorre la serie di subarmoniche della nota più grave dell'accordo, finché non viene trovata una frequenza le cui armoniche approssimano tutte le note dell'accordo, entro una certa tolleranza. *cage.virtfun* può essere anche applicato a sequenze di accordi nel tempo; in questo caso, il risultato sarà la sequenza delle fondamentali virtuali. D'altro canto, l'operazione numerica compiuta da *cage.virtfun* ha un ambito di applicazione più ampio: può essere considerata come il calcolo di un massimo comun denominatore approssimato di un insieme di numeri. Come tale, è richiamato ad esempio da *cage.accrall*, per stabilire un'unità minimale ritmica 'ragionevole' all'interno di una partitura in notazione proporzionale.

cage.delay e *cage.looper* estendono il concetto di linea di ritardo con feedback nel dominio simbolico. Il loro obiettivo è creare strutture ripetitive in cui il materiale possa essere alterato a ogni passaggio attraverso un *lambda loop*. La differenza tra i due moduli risiede nell'unità musicale che viene passata attraverso il *lambda loop*: un singolo accordo nel caso di *cage.delay*, un'intera sezione di partitura nel caso di *cage.looper*. In entrambi i casi, il tempo di ritardo può essere cambiato a ogni ripetizione. In linea di principio non c'è limitazione alla ricchezza dei processi che possono essere applicati al materiale nel *lambda loop*: il risultato musicale può essere quindi ben più complesso di una semplice iterazione.

cage.cascade~ e *cage.pitchfilter* estendono il principio di filtraggio al dominio simbolico. Il primo applica a una partitura una catena di filtri a due poli e due zeri, analogamente agli oggetti Max *biquad~* e *cascade~*, emulando la risposta in frequenza di un filtro digitale con risposta all'impulso infinita. Il secondo opera direttamente sulle altezze, anziché sulle frequenze, applicando a una partitura un filtro definito da una funzione, ad esempio tramite un oggetto *function* o *bach.slot*. In entrambi i casi, la velocità MIDI di ogni nota è modificata a seconda della risposta del filtro, e le note con velocità al di sotto di una certa soglia sono rimosse. È anche possibile interpolare tra differenti configurazioni di filtri nel tempo (si veda la Fig. 3).

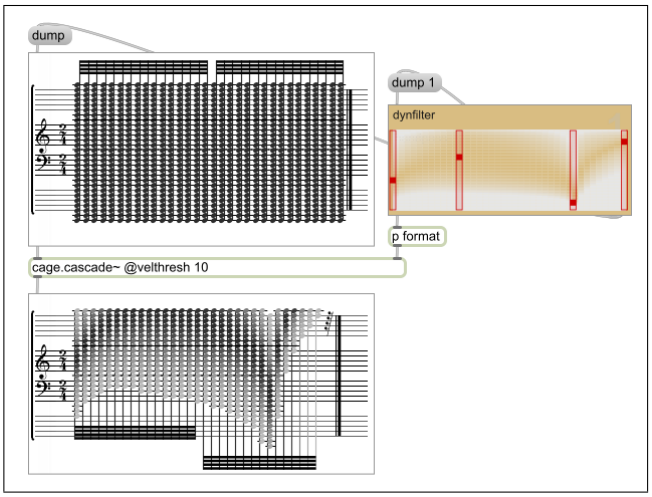


Figura 3. Un esempio di filtraggio dinamico di una partitura ottenuto attraverso *cage.cascade*$\sim$ pilotato da uno slot di tipo *dynfilter* in un oggetto *bach.slot*. Ogniqualvolta i parametri del filtro sono modificati tramite l'interfaccia, il risultato è aggiornato in tempo reale.

cage.granulate è un motore di granulazione simbolica. I parametri della granulazione sono gli stessi del corrispondente processo elettroacustico: l'intervallo di tempo tra due grani, la durata di ogni grano, la regione di partitura da cui i grani devono essere estratti. L'altezza e la durata dei singoli grani può essere modificata. Basandosi su questi parametri, *cage.granulate* riempie in tempo reale un oggetto *bach.roll* collegato al suo outlet.

3.4 Interpolazione armonica e ritmica, formalizzazione dell'agogica

L'astrazione *cage.chordinterp* opera un'interpolazione lineare all'interno di un insieme di accordi, attraverso l'assegnazione di differenti pesi per ciascuno di essi. Analogamente, un'interpolazione ritmica può essere ottenuta tramite il modulo *cage.rhythminterp*.

cage.timewarp, d'altro canto, opera una distorsione temporale di una partitura, ottenuta tramite una funzione (definita tramite *lambda loop*) da applicarsi a ogni onset di ogni evento discreto nella partitura. Tra le altre cose, questo meccanismo è abbastanza flessibile da permettere di ottenere qualsiasi forma di rallentando o accelerando, una volta definita la funzione appropriata - un compito che è facilitato dall'astrazione *cage.accrall*, che permette di esprimere l'agogica attraverso un insieme di parametri di alto livello, come la durata totale del processo o la velocità iniziale o finale.

3.5 Automi cellulari, L-sistemi, ecc.

L'astrazione *cage.chain* implementa automi cellulari monodimensionali e L-sistemi. Tale modulo opera riscritture di una data lista, a seconda di un insieme di regole definite dall'utente attraverso un messaggio o un *lambda loop*. Le sostituzioni possono avvenire sui singoli elementi (ad esempio: una certa lettera o nota viene sostituita da una lista di lettere o note), o su sequenze di elementi aventi

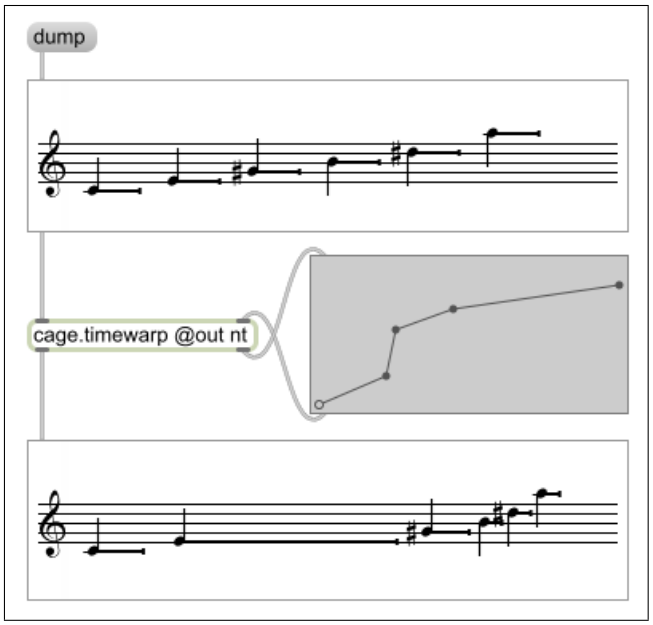


Figura 4. Un esempio di distorsione temporale ottenuta grazie a *cage.timewarp*. La funzione nel *lambda loop* associa il tempo nella partitura originale (in alto), rappresentato sull'asse x, al tempo nella partitura risultante (in basso), rappresentato sull'asse y.

lunghezza prefissata (ad esempio: ogni coppia di elementi è rimpiazzata da una certa altra sequenza); in quest'ultimo caso, *cage.chain* gestirà il comportamento ai bordi a seconda dei valori di alcuni specifici attributi (*pad*, *align*). In sintesi, questo modulo facilita la generazione di automi cellulari a dimensione 1, o di frattali ottenuti per sostituzione.

cage.life si occupa invece di automi cellulari a due dimensioni (il più famoso dei quali è il 'gioco della vita' di John Conway). Le regole per questi automi sono definite tramite *lambda loop*. La grandezza delle sottomatrici su cui dev'essere operata la sostituzione può essere definita dall'utente.

Un'astrazione strettamente connessa alle due precedenti è *cage.lombricus*, che implementa un meccanismo per costruire sistemi generativi basati su regole. Questo modulo accetta una serie di elementi iniziali, raggruppati in famiglie, con un peso associato ad ogni famiglia. Il compito dell'astrazione è creare una sequenza di un numero arbitrario di elementi, facendo in modo che il numero relativo di occorrenze di elementi di ogni famiglia rispetti la distribuzione dei pesi associati alle famiglie. Il *lambda loop* dell'astrazione fornisce le proposte di elementi che andrebbero incatenati alla sequenza esistente, insieme all'intera sequenza costruita fino a quel punto; ogni proposta può essere rifiutata o accettata con un certo punteggio assegnato in base a regole personalizzabili: tra gli elementi accettati, il 'vincitore' sarà scelto in base al punteggio e al peso della famiglia cui appartiene. Se a un certo punto non viene trovato alcun elemento idoneo, l'astrazione è in grado di tornare sui propri passi e sostituire un elemento scelto precedentemente con uno diverso, avente punteggio più basso ma potenzialmente in grado di permettere la

costruzione di una catena più lunga. È necessario sottolineare che gli elementi non devono necessariamente essere copiati letteralmente nella sequenza risultante: ad esempio, l'utente può voler dare al sistema un insieme di intervalli come elementi iniziali, e ottenere una sequenza melodica come risultato del processo: la sostituzione può avvenire all'interno del *lambda loop* descritto in precedenza. In sintesi, il meccanismo soggiacente a *cage.lombricus* condivide alcune caratteristiche con automi cellulari e L-sistemi da un lato (in particolare un comportamento basato su regole di sostituzione costruttive) e con la programmazione a vincoli dall'altro (la capacità di prendere decisioni basate su pesi, e la possibilità per l'algoritmo di tornare sui suoi passi), senza appartenere strettamente ad alcuna delle due categorie. Sebbene questo meccanismo possa apparire complicato, un'analisi accurata delle pratiche compositive nostre e di altri compositori (primo fra tutti Michaël Jarrell) suggerisce che tale procedimento possa essere ben adatto a modellizzare una grande quantità di tecniche di formalizzazione musicali.

3.6 Strumenti per la *set theory*

Un gruppo di moduli di *cage* si occupa di rappresentazioni tipiche della *set theory* musicale, ossia della teoria matematica, elaborata in origine negli Stati Uniti a partire dagli anni Sessanta, per categorizzare gli oggetti musicali (agglomerati armonici in primis) e descriverne le relazioni. Due moduli di *cage*, *cage.chroma.topcset* e *cage.chroma.frompcset*, operano conversioni tra insiemi di *pitch class* e vettori di *chroma* (si veda [9]); altri due moduli, *cage.chroma.tocentroid* e *cage.chroma.fromcentroid*, operano conversioni tra i vettori di *croma* e centroidi tonali, ottenuti tramite la trasformata descritta da Harte and Sandler in [10]. Il calcolo del centroide tonale a partire dal vettore di *chroma* implica una perdita d'informazione; la conversione inversa quindi non è univoca: l'algoritmo restituisce un singolo vettore di *chroma*, tra tutti quelli aventi il vettore in input come centroide.

3.7 Partiture

cage contiene un gruppo di moduli per il trattamento globale di partiture: *cage.rollinterp* interpola tra il contenuto di due oggetti *bach.roll*, a seconda di una curva di interpolazione o di un singolo valore (nel caso di interpolazione statica). *cage.envelopes* aiuta a sincronizzare una famiglia di funzioni alla durata totale della partitura, rendendo più agevole la modifica in tempo reale della partitura a seconda dei valori delle funzioni a ogni istante. *cage.scissors* divide la partitura contenuta in un oggetto *bach.roll* in base a punti di divisione verticali (tempo) e orizzontali (voci); l'astrazione restituisce quindi una matrice contenente in ogni cella il corrispettivo estratto di partitura. *cage.glue* compie l'operazione inversa: riempie un singolo *bach.roll* riassemblando una matrice contenente partiture più piccole, in base alla disposizione verticale (tempo) e orizzontale (voci) implicita nella matrice stessa o definita esplicitamente. *cage.ezptrack* parte da una sequenza di accordi e tenta di ricostruire delle voci musicali, in maniera simile a quanto gli algoritmi di *partial tracking* fanno con l'analisi dei dati armonici.

3.8 Supporto per file SDIF

Una famiglia di moduli di *cage* è progettata per facilitare la lettura e la scrittura di file di analisi in formato SDIF [11, 12]. Questa famiglia contiene sottofamiglie per alcuni dei più comuni tipi di analisi e descrittori, in particolare frequenza fondamentale, picchi, *partial tracking*, *markers*.

A partire dalla versione 0.7.4, *bach* supporta la lettura e la scrittura di file SDIF attraverso gli oggetti *bach.readsdif*, uno strumento di basso livello che legge tutta l'informazione contenuta in un file SDIF e la converte in una *llll*, e *bach.writesdif*, che permette di scrivere file in formato SDIF a partire dalla loro rappresentazione come *llll*. Questa rappresentazione è completa: se colleghiamo l'output di *bach.readsdif* all'input di *bach.writesdif* otteniamo un file SDIF equivalente, quando non identico, all'originale. D'altro canto, questa completezza rende la manipolazione della rappresentazione stessa più farraginosa.

Per questa ragione *cage* annovera un insieme di moduli che implementano operazioni basilari sul contenuto dei file SDIF. Alcuni moduli convertono i dati SDIF in una sintassi direttamente comprensibile da *bach.roll*, per esempio *cage.sdif.ptrack.toroll* (si veda la Fig. 5). Altre astrazioni ristrutturano i dati di analisi in una forma più accessibile. A titolo di esempio, *cage.sdif.fzero.unpack* filtra i *frame* di tipo 1FQ0 (stime di frequenze fondamentali) e restituisce gli onset, le frequenze, i valori di confidenza, i punteggi e le ampiezze dai suoi diversi outlet sotto forma di *llll* strutturate con una sottolista per ogni *stream* SDIF. Due astrazioni gestiscono le rappresentazioni di *partial tracking* (*cage.sdif.ptrack.resolve* e *cage.sdif.ptrack.assemble*), permettendo la conversione tra una rappresentazione strutturata per istanti di tempo a una rappresentazione strutturata per indici delle voci.

In generale, non abbiamo considerato come scenario di utilizzo frequente la scrittura di un file SDIF a partire da una partitura, con una sola importante eccezione: i *marker*. Per questa ragione, la sola astrazione che fornisce una traduzione diretta da un oggetto di notazione a una *llll* contenente l'informazione SDIF è *cage.sdif.markers.fromroll*, che esporta tutti i *marker* di un oggetto *bach.roll*, ciascuno con la propria posizione nel tempo e il proprio nome.

3.9 *Rendering* audio

In aggiunta ai moduli di composizione assistita precedentemente elencati, *cage* contiene un insieme di utilità che mirano a rendere più semplice la prototipazione e la verifica delle soluzioni musicali. In particolare, due moduli di *cage* si occupano di *rendering* audio di partiture *bach*: *cage.ezaddsynth~* (un motore di sintesi additiva elementare) e *cage.ezseq~* (un campionatore elementare). Come *bach.ezmidiplay*, entrambi sono progettati per essere connessi direttamente all'uscita 'playout' di *bach.roll* o *bach.score*.

Il motore di sintesi additiva risponde all'esigenza di un *rendering* audio essenziale, che possa ovviare alle limitazioni degli strumenti MIDI: ciò può essere utile ad esem-

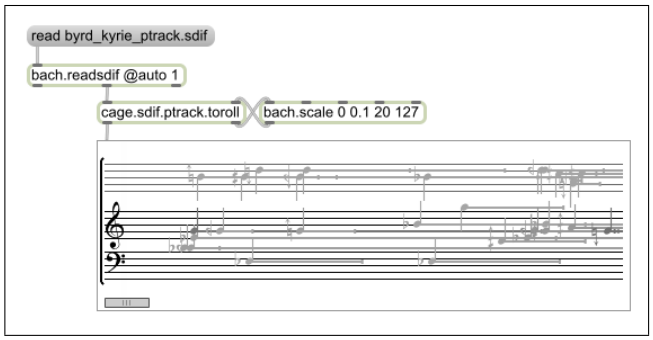


Figura 5. Un'analisi di *partial tracking* contenuta in un file SDIF è importata in un *bach.roll* attraverso *cage.sdif.ptrack.toroll*. Il *lambda loop* è usato per definire una mappatura personalizzata sulle velocità MIDI (se un *lambda loop* non è definito, l'astrazione utilizza una mappatura di default).

pio quando si lavora con griglie microtonali non standard, o quando gli inviluppi di ampiezza e panning o i glissandi non possono essere trascurati. Gli inviluppi devono essere definiti all'interno di *slot*. [6]

Il campionatore risponde all'esigenza di utilizzare *bach.roll* e *bach.score* come '*sequencer* aumentati': *cage.ezseq∼* considera il nome del campione da utilizzare, gli inviluppi di ampiezza, il panning, la velocità di riproduzione, eventuali filtri audio e il punto di inizio all'interno del campione (tutti definiti all'interno di *slot*). *cage.ezseq∼* è inoltre capace di pre-caricare in memoria i file audio, se una data cartella è assegnata. Se richiesto, *cage.ezseq∼* può trasportare ogni campione senza alterazione temporale (attraverso l'oggetto Max *gizmo∼*), in dipendenza dall'altezza della nota associata.

4. CONCLUSIONI

Nel momento in cui questo articolo viene scritto, una versione 'alpha' della libreria è disponibile [7] : non tutte le funzionalità descritte sono implementate, e la documentazione non è ancora completa. Ciononostante, la maggior parte dei moduli è già perfettamente utilizzabile. La prima versione completa della libreria sarà resa pubblica in ottobre 2014, all'occasione di una presentazione che avrà luogo a Ginevra, e sarà distribuita gratuitamente. A partire dall'anno accademico 2014-2015, *cage* sarà insegnata all'interno dei corsi di composizione e musica elettronica della *Haute École de Musique* di Ginevra, e in altre istituzioni partner.

4.1 Ringraziamenti

cage è un progetto di ricerca che nasce all'interno del centro di musica elettroacustica della *Haute École de Musique* di Ginevra, supportato dalla divisione musica e arte della *Haute École Specialisée* della Svizzera occidentale. Il nome *cage*, che nel contesto della denominazione anglosassone delle note rappresenta la famosa espansione di *bach*, è anche un acronimo che riconosce questo supporto: *composition assistée Genève*.

[6] Gli *slot* sono contenitori per metadati di diversa natura, associabili individualmente a ogni singola nota (si veda [2]).

[7] Si può ottenere la libreria dal sito www.bachproject.net/cage

5. BIBLIOGRAFIA

[1] A. Agostini, E. Daubresse, and D. Ghisi, "cage: a high-level library for real-time computer-aided composition," in *Proceedings of the International Computer Music Conference (ICMC 2014)*, (Athens, Greece), 2014, to appear.

[2] A. Agostini and D. Ghisi, "*bach*: an environment for computer-aided composition in Max," in *Proceedings of the International Computer Music Conference (ICMC 2012)*, (Ljubljana, Slovenia), pp. 373–378, 2012.

[3] A. Agostini and D. Ghisi, "Real-time computer-aided composition with *bach*," *Contemporary Music Review*, no. 32 (1), pp. 41–48, 2013.

[4] M. Malt and J. B. Schilingi, "Profile - libreria per il controllo del profilo melodico per Patchwork," in *Proceedings of the XI Colloquio di Informatica Musicale (CIM)*, (Bologna, Italia), pp. 237–238, 1995.

[5] J. Fineberg, "Esquisse - library-reference manual (code de Tristan Murail, J. Duthen and C. Rueda)," 1993.

[6] R. Hirs and B. G. editors, *Contemporary compositional techniques and OpenMusic*. Delatour/Ircam, 2009.

[7] A. Cont, *Modeling Musical Anticipation*. PhD thesis, University of Paris 6 and University of California in San Diego, 2008.

[8] M. Puckette, "A divide between 'compositional' and 'performative' aspects of Pd," in *Proceedings of the First Internation Pd Convention*, (Graz, Austria), 2004.

[9] M. Müller, *Information Retrieval for Music and Motion*. Springer Verlag, 2007.

[10] C. Harte, M. S, and M. Gasser, "Detecting harmonic change in musical audio," in *In Proceedings of Audio and Music Computing for Multimedia Workshop*, 2006.

[11] M. Wright, R. Dudas, S. Khoury, R. Wang, and D. Zicarelli, "Supporting the sound description interchange format in the max/msp environment," in *Proceedings of the International Computer Music Conference*, 1999.

[12] M. Wright, A. Chaudhary, A. Freed, S. Khoury, and D. Wessel, "Audio applications of the sound description interchange format standard," in *Proceedings of the Audio Engineering Society 107th Convention*, 1999.

BRINGING BACK TO LIFE ANCIENT MUSIC SCORES: THE CASE OF MUSIC IN THE 1799 NEAPOLITAN REVOLUTION

Sergio Cavaliere
Dipartimento di Fisica
Università di Napoli
Federico II
cavalier@unina.it

Andrea Arcella
Conservatorio di Napoli
San Pietro a Majella
andrea.arcella@gmail.com

Nadia Polverino
Corso di Informatica Università di Napoli Federico
adina.linux@gmail.com

Stefano Silvestri
Conservatorio di Napoli
San Pietro a Majella
stefanodrive_1@libero.it

ABSTRACT

This paper describes an ongoing project, which, based on ancient archive material and scores, abounding in the music library of the *Conservatorio di Musica di Napoli*[1], *San Pietro a Majella*, tries to bring to new life some interesting handwritten and printed musical material. The project aims at enhancing fruition of ancient scores, multiplying modalities and means to enjoy ancient music but, at the same time, it promotes some interesting musicological and even historical research. All these goals are performed by means of the use of multimedia tools, including web technology, mobiles, tablets, smart tags and similar. The project may offer tools and materials to be used in different directions, ranging from smart guides for music museums, to musical education and more general interactive music listening.

1. INTRODUCTION

Advances in technologies, as well as a renewed interest in enriching music listening and education, advances in the techniques to represent music and sounds, open the way to a multiplicity of approaches which may drastically enhance music fruition, music education, museum visits and similar. We are referring to the approaches developed in the WEDELMUSIC initiative [1], but also to the 1599 IEEE standard developed in Milan at LIM (see [2, 3] and bibliography therein), and the german *Probado Music* project [4, 5].

2. THE STARTING STORY

Starting story for our reconstruction is the complex matter, actually only partly clear, of the music production devoted by Domenico Cimarosa to the 1799 Neapolitan Revolution.

Copyright: ©2014 Sergio Cavaliere, Andrea Arcella, Nadia Polverino, Stefano Silvestri. This is an open-access article dis-tributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and repro-duction in any medium, provided the original author and source are credited.

This revolution is an important episode of European and Italian history rich of aspects that, in some manner, have deeply contributed to some features of our history. This is true in particular for South Italy where it contributed to shape relationships among different social classes, state, clergy, and even religion, affecting advancements in culture, science, music, even as far as regards actual times; we may state after Croce *"…it gave the first seed of Unity of Italy"* [6].

From a musicological view-point we may share the evaluation made by Renato Di Benedetto (see Rao [7]): we cannot give deep political meaning to episodes that involved dramatically musicians such as Cimarosa, Paisiello or also the young pupils of the Music Conservatories, the latter undergoing terrible trials as *rei di stato* and often death penalty. A musician in XVIII century mostly conceived its activity as a '*service*' to rich '*customers*' without any ideological involvement. However direct participation to political events was a real and dramatic fact, which, after all, reveals widespread awareness, in the wide and active music community, of the new ideas circulating all over Europe; ideas that, actually, were funding the new world. This is the main idea to be communicated by means of our research.

2.1 The ceremony for burning flags and the hymn

Crucial episode of the whole story is the ceremony for burning flags, held around the tree of freedom in *Largo di palazzo Nazionale* in Naples, on Sunday 19 may 1799, or better, *8 fiorile 1799*, in the new language of the Revolution, based on the calendar introduced by the French Revolution [8]. The date comes few months from the foundation of the Republic – January 21 - and just few days before the end of the Republic, june 21, with the entrance of the troops of Cardinale Ruffo and the surrender of the republicans in the Neapolitan Castles. In this occasion "*young pupils of the Conservatorio di Musica sing a patriotic hymn, poetry by the well-known Citizen and Poet Luigi Rossi, and composition by Maestro Cimarosa*" (see *il Monitore Napoletano* journal directed by one of the leading personalities of the revolution, Eleonora Pimentel Fonseca, who paid with her life her role in the revolution [8]).

[1] http://sanpietroamajella.it/index.php, (20/09/2014).

2.2 Involvment of Cimarosa

This episode was the starting point of the dramatic story that since then hit Cimarosa, until his death, in Venice, on 11 January 1801. This episode was very emblematic, but, as results from official documentation, Cimarosa composed also other songs and hymns for the Revolution [9] and took part in a *Committee for the theatres*. After the restoration of king Ferdinando IV, when thousands of so called "*giacobini*" where imprisoned, many of them scientists, economists, best part of the Neapolitan intellectual class, often well known all over Europe, judged by a special military court, *Giunta di Stato*, and brutally decapitated or hanged. All Europe astonished assisted to a true massacre. Cimarosa was not at all touched, until the end of the year. He, trying to hide his recent republican past, wrote a *Cantata a 3 Voci* [10] and published it with the title of *Maestro di Cappella*, dedicating it to the king. This however had opposite outcome: on 9 December 1799 Cimarosa was imprisoned for a long four months period in the prison of *Santa Maria Apparente*, in terrible conditions, as usual in that regime. He was then released, by intercession of some of his supporter [11], and also thanks to his well known musical activity. He was "*advised or better obliged*" to go in exile (oral testimony by the son Paolo Cimarosa, himself a musician [12]). Cimarosa went to Venice where he could rely on the friendship of artists or old pupils or venetian aristocrats; there he intended to write the opera commissioned by La Fenice, *Artemisia,* that unfortunately he had not the chance to see performed on stage.

2.3 Death of Cimarosa

Few months later Cimarosa died at age 52, on 11 January 1801. In order to solve doubts on the death of the musician, a physician, three months later, was called to sign a death certificate explaining the medical reasons of his death (attached in [12]).

3. THE CASE OF THE "INNO NAZIONALE" BY CIMAROSA

Our research deals with the Hymn that Cimarosa, as reported above, composed for the *Neapolitan Republic*. We will see that the hymn is not clearly identified, neither verses nor score; on the contrary, many different attribution were made during the long time elapsed since that distant vicissitudes. The material, tools and instruments set up in our research deal with the presentation and the visual and aural display of the complex matter of the hymn. We will therefore describe in short the complicated vicissitudes of the proposed attributions. First notices of the hymn are known from contemporary reports. We read in the *Monitore Napoletano* [8]: "*n. 30 del Monitore Napolitano SEXTIDI 6. PRATILE ANNO VII DELLA LIBERTA"* (SABBATO 25. MAGGIO 1799), "*On Sunday* (it is the preceding Sunday, 19 May 1799) *finally was held the public feast for the burning of flags won in various action against the insurgents and, for the first time were deployed the flags donated by the Government to Guardia Nazionale"*.
The report continues in issue 31: "*young pupils of the 'Conservatorio di Musica' began singing a patriotic hymn, poetry by the well known Citizen and Poet Luigi Rossi, and composition by Maestro Cimarosa; then it was to be read a public allocution to People, but general transport prevented it: general joy became then more than 'ebrietà, became sacred orgasm and furor. People approached signs and flags to break them, more, to tear them*".

Another testimony is given in [13], pp.143-144: "Sunday 19 of May". "In that occasion they have been sung some hymns composed one by Vincenzo Mundo, the other by Eugenio Palumbo and put in music by Cimmarosa and Paisiello. Probably I will write in the margin their words…", what the author unfortunately never did. Here Mundo is the well known poet Vincenzo Monti author of many poems such as the well known Inno alla Libertà, while Eugenio Palumbo was probably erroneously reported instead than Rossi. Besides the above, some documents also exist, in the form of 'pagine sparse' or 'fogli', printed at that times in order to disseminate hymns, allocutions, odes and similar to the general public.

3.1 Damnatio memoriae

The above then are direct witnesses of the facts regarding the hymn by Cimarosa and Rossi. In the following years a curtain of silence fell in the music community on the dramatic story of Cimarosa and the revolution: most authors commenting on the life and music production by Cimarosa, celebrate the Genius but ignore completely the 1799 parenthesis in his life and of course also the controversial story of his hymns.

3.2 Historians begin to tell the whole story

In his monumental work [14], the historian Carlo Botta, in book XVIII, dealing with the year 1799, describes the dramatic climate of that year in Italy. Regarding Cimarosa, "*when Naples was taken by assassins hired by Ruffo, first his houses were ransacked, moreover his gravicembalo* (klavicord*), happy source of many beautiful chants, was drawn out of the windows to break on hard flints, then he was put in prison, where he stayed for an heavy four months period"* [14]. The episode later will be shown as unfaithful; moreover the Cembalo is still stored in the Museum of the *Conservatorio di Napoli* [15] (see Figure 1).

3.3 Early vicissitudes of the hymn

In the tradition the hymn known as *Marcia reale dei Borboni* was believed to be the celebrated hymn composed by Cimarosa for the *Repubblica di Napoli* [12]. In 1868 Giuseppe Orlandi found, among his family papers, the so called *Bell'Italia,* an autographed score by Cimarosa, and he donated it to the *Conservatorio of Napoli*. Saverio Mercadante at that times Director of the *Conservatorio* agreed with the donor on the fact that this was the real hymn of the revolution. Florimo too, musi-

cologist and archivist of the Conservatorio, agreed on the same hypothesis [12].

Later on in 1888 Rocco [16] contested the above hypothesis. He reported an oral testimony on the hymn according to which verses were different from the manuscript and the melody was used later by Bellini in his opera *Capuleti e Montecchi* for the *aria*: *La tremenda ultrice spade,* premiered a La Fenice in Venice on march 11/1830[2].

The author claims that the hymn stored in the Conservatorio seems an "*inno sanfedista*" [16]: the author of the poems, Vincenzo de Mattei, belonged to the "*sanfedisti army*"[3]. Moreove Rocco found *I dritti dell'uomo*, '*canzonetta'* (short song) by citizen Luigi Rossi, beginning with the strophe recalled by the above cited oral testimony, and identified it as the true hymn.

More recently, Benedetto Croce in 1891 [17] contests both the attribution by Orlando and Florimo, and the proposal by Rocco: "*they actually are just 'canzonette', not hymns"*. Croce in turn advances a further hypothesis, the final, in his judgment; he found the patriotic hymn in a collection of *"fogli volanti"* (single papers) stored in *Bibl. Della Società Stor. Nap*., with the title: *Inno patriottico del cittadino Luigi Rossi per lo bruciamento delle Imagini dei Tiranni, posto in musica dal Cimmarosa, da cantarsi nella festa de' 30 fiorile sotto l'albero della libertò avanti al Palazzo Nazionale*". Few years later, in 1930, Ettore Santagata again reports, in a guide of the Conservatorio di Napoli, the verses of the "*Bell'Italia*" as "*Inno Rivoluzionario autografo di D. Cimarosa, scritto nel 1799, per ordine dei signori della Repubblica Partenopea (7 maggio - 13 giugno 1799), su parole di Luigi Rossi*" [18].

Battaglini in 1974 contends the attribution made by Croce and supports as a candidate hymn on of the songs discarded by Croce, *Vanno al foco le infami bandiere* [19].

3.4 Recent times

In 1999, in the second centennial of the revolution, Roberto De Simone premiered in Teatro San Carlo in Naples on 8 January 1999 [20, 21, 22] the "*Oratorio drammatico Eleonora*": the hymn adopted here was "*I diritti dell'uomo*" the one proposed by Rocco more than 100 years before. In the volume published for the second centennial Stefano Innamorati [21], on the ground of thorough research, carried on also by De Simone, states c briefly that, on the ground of factual elements and documents, nothing definite may be told on the matter, except that each opinion and each proposal has its own consistency in partial documents.

He proposes also the hypothesis, very plausible indeed, that, in many cases, hymn and songs may be taken by already known melodies and arias, circumstance that make possible for large popular public to acquire and then sing new poems in a short time.

[2] https://www.youtube.com/watch?v=nSbd98f3UxI, (20/09/2014).

[3] The name of the army fighting against the revolution and of the related movement for the restoration of the king.

The question is not at all closed: Innamorati concludes with words which may be easily shared by us: "*And, who knows, may in the future from a remote cleft of some library it will come out the Hymn of the Neapolitan Republic*". This is also our hope.

4. SYNCHRONIZING MULTIMEDIA MATERIAL

A very efficient way to "read" ancient music scores and give them life again, is that of synchronizing performance of the score with the printed score itself, eventually in its manuscript and original form, in order to highlight pitches and parts while listening. This is a way of promoting a deeper acquaintance of the pieces under exam, also in order to fill the gap between simply listening to a piece of music and the more complex listening while reading a score. This kind of active listening gives the experience of a deeper understanding and fruition of music, for general public and also for music students or people with only partial ability to read music.

Synchronized annotation may include emphasizing single pitches while they are sounding, or music phrases, or accompanying poems, but also related images or other kind of multimedia document which we believe to be relevant for this kind of augmented listening [5][23]; all these annotations are to be properly synchronized.

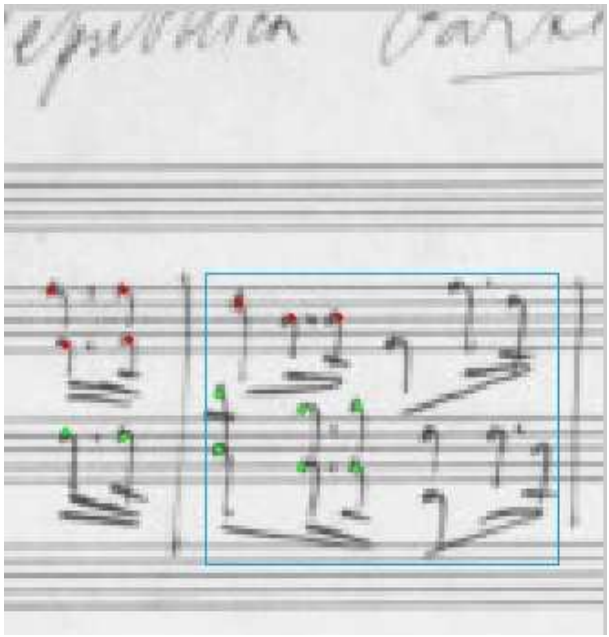


Figure 2. Real time score following on a manuscript by Cimarosa - Conservatorio di Napoli, Rari 1.6.7(29).

The above approach may be easily coupled to interactive listening, providing the possibility to choose the source score, printed or manuscript, may be in different versions or arrangements, to choose the proper instrumentation, if this choice is allowed, isolate eventually single parts to be listened to separately, or properly enhanced, such as the bass line or the accompaniment.

Also *spatialization* may be used, eventually simulating different listening ambients, camera concert, large cathedral or other, proper from an historical viewpoint.

In Figure 3 another way of listening is shown, where note events are displayed in real time on the waveform with their time domain occurrence.

In the last example pitches are displayed on the frequency spectrogram as in Figure 4, showing also their position on the superimposed pentagram (both violin and bass key); this visualization highlights the real frequency of the notes, with the related logarithmic spacing, very useful for didactical viewpoint. Same visualization may be done using a psychoacoustic scale, such as the *mel* scale.

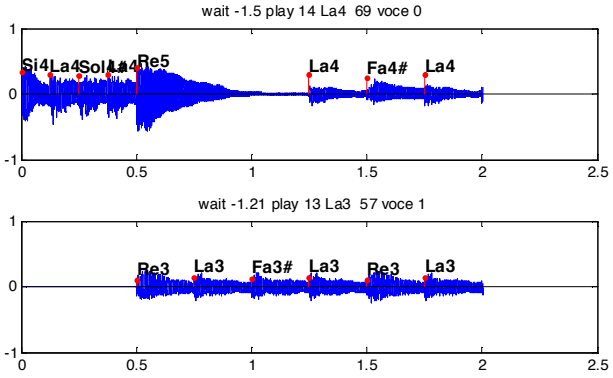


Figure 3. Real time events on the waveform.

4.1 A large palette of tools and instruments

As already pointed out research from the computer music community has developed a large realm of instruments for all the tasks involved in the annotation and listening process.

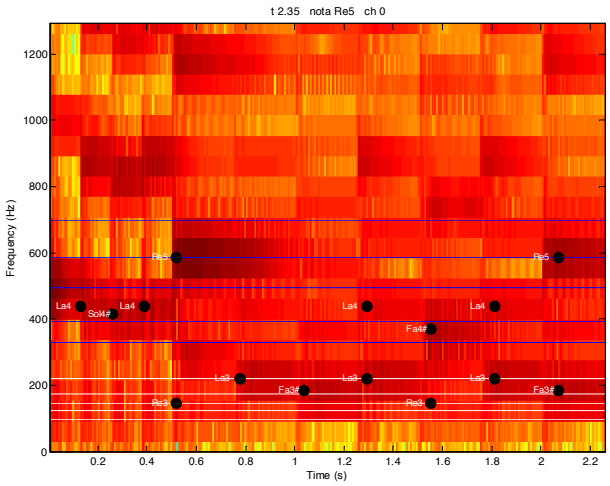


Figure 4. Real time events on the spectrogram and superimposed pentagrams.

The computer music community has for years developed actions and initiatives in the field of Music Information Retrieval and Music related DSP, with the idea in mind of searching for new tools for active listening and fruition of music, but also for carrying on musicological research in new ways. We refer for example to the DAFx conferences, or to the initiatives taken by *The International Society for Music Information Retrieval* ISMIR.

A useful tool used in our project is *Alignmidi* [24], a project by *LabRosa*, at Columbia University[4] useful when real music execution must be aligned to its source midi file, in order to modify properly times in the latter; the midi transcription is assumed as an approximate description to be improved by means of modifications of the time information.

The *Alignmidi* tool is based on the use of similarity matrix between frames of the spectrogram of the waveform pair to be aligned; dynamic programming is then used for the alignment procedure [24, 25]. Similar solutions are also devised in the literature, based on the Chroma representation [26, 27] used to build the similarity matrix, and the on-line DTW (Dynamical Time Warping) algorithm for live tracking of musical performances [28, 29]. We used the *Alignmidi* tool in order to provide timing information for pitches.

Another tool used in our project is the *Sonic Visualiser*, developed in the Centre for Digital Music of the Queen Mary University of London [30]. This open source program allows easily representing audio signals and some useful parameters like instantaneous energy values and onset time instants. This time information may be used in order to compute exact time instants of each incoming pitch. In our project we used this information in order to verify and correct timing information obtained by means of *Alignmidi*. Also in the Sonic Visualiser tool it might be used the VAMP plugin by Dixon, which, using the time-warping algorithm [29] proposes alignment between a wave file and its midi resinthesis, thus giving the exact time information useful for score following. The same approach may be adopted using the stand alone MATCH alignment tool [29].

Last tool used in our project is the *Miditoolbox* for Matlab [31]; this tool was used in order to read midifiles and build the proper time ordered list of events to be annotated, which, together with occurrence times, is the framework of our data structure.

4.2 An audio editor to align printed scores or midi score to handwritten or ancient scores

The procedure used in order to obtain all information useful to play multimedia material in the above described methods, is that of attaching to each multimedia file another file containing synchronization information. The form of this information is just an ordered list of events with their time of occurrence and with the coordinates for the graphic annotation on the score, printed or manuscripts or whatsoever. Suppose that we want to set up the synchronization/annotation file for a manuscript score, which is actually our prevalent case. We use an editor, purposely written in Matlab, see Figure 5, which loads and shows at one time both the scores, the

[4] http://labrosa.ee.columbia.edu/matlab/alignmidi/, (20/09/2014).

manuscript score and the one obtained from the midi file; our editor precedes one step at a time along the midi file showing actual tones on the printed score and asking the user to point to the corresponding pitch on the manuscript; a third window shows the pentagram and pitches in exam, one after the other. The midi file is thus annotated to a target file where successive events are listed in one with the ideal timing (that contained in the source midi file) and the x y coordinates of pitches, to be highlighted during listening, both for the manuscript and the pretty-print version of the score.

Final step is that of correcting midi times with real timing of the performance under annotation, which may differ even drastically from the ideal midi timing.
This kind of time alignment is performed by means of the tools described above.

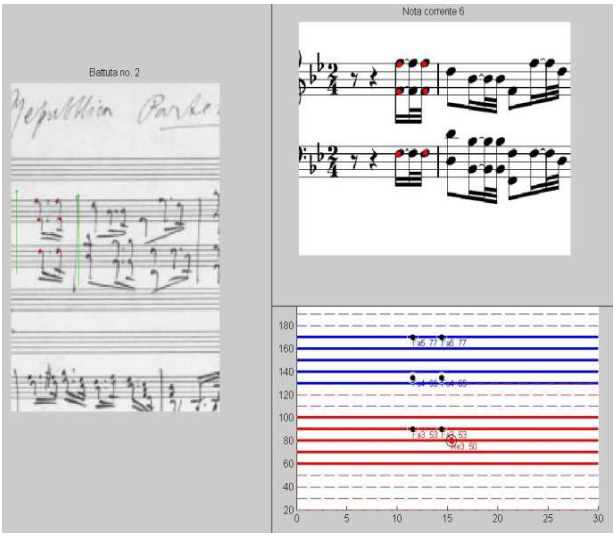


Figure 5. The tree windows of the manuscript editor: source manuscript, pretty printed version and pitches from the midi file on the pentagram.

The whole procedure moreover is allowed to eventually neglect some event if this cannot be easily identified: during listening few errors may be tolerated without substantial loss of information. For the above purpose we set a program which allows identifying the time position of the event, moving a sliding window along time to the position in which the event will occur, on the basis of the midi timing: the operator has just to choose the exact point, on a time line where energy of the signal is represented; this trace is easily provided by the above mentioned tools *Sonic Visualiser* and *Alignmidi*.

These steps, as may be seen, are highly interactive and rely on good musical skills; as a matter of fact in the current literature doesn't seem to exist reliable tools to perform the task in a completely automatic way. Advances in this field may help in the future; any way, in our case, the task may be performed off-line and usually for short music excerpts.

5. SOME USES OF THE PROPOSED TOOLS

The tools indicated and the criteria and methods presented in this paper may be used for different purposes. Main example is museum applications: using a tablet or mobile a visiting scholar/or general public may identify (may be using pattern recognition from the camera images or wifi-tags, smart tags or other similar technology) a piece displayed, say an ancient score, or a music instrument and get it immediately sounding with an ancient performance. The listener can select individual traces for listening and displaying on the score, but also pertinent images or musicological annotations or whatsoever, or even spatializing it in a historically pertinent virtual listening room. The resulting experience will be greatly enhanced and enriched by the connected multimedia information and by interactivity in listening. This approach will be used in a project set up by the authors in the DATABENC initiative in the context of museum applications (http://www.databenc.it/).

A further output of work is the rich collection of multimedia information organized in the website *Cimarosa e la rivoluzione napoletana del 1799*[5], where all the source material is provided and possibly expanded.

Another output of the same research will be didactical modules realized on the same subject in the form of *apps* proposing some aspect of the story described above.

Finally, in order to underline the sense of our work we will close the paper quoting from Eleonora Pimentel Fonseca her last sentence: "*Forsan et haec olim meminisse iuvabit*" (Perhaps the time may come when these events will be useful to remember).

**Translations from Italian language were made by the authors*

6. REFERENCES

[1] WEDELMUSIC Web DELivering of MUSIC scoresftp://ftp.cordis.europa.eu/pub/ist/docs/ka3/iep_wedel.pdf.

[2] L. B. Denis, G. M. Haus, Music Navigation with Symbols and Layers: "Toward Content Browsing with IEEE 1599 XML Encoding", *Wiley-IEEE Computer Society Press*-ISBN: 978-0-470-59716-3, 2013.

[3] IEEE Recommended Practice for Defining a Commonly Acceptable Musical Application Using XML, IEEE, 2008.

[4] V. Thomas, D. Damm, C. Fremerey, M. Clausen, F. Kurth and M. Müller: "PROBADO Music: a Multimodal Online Music Library", *Proceedings of the 38th International Computer Music Conference*

[5] http://people.na.infn.it/~cavalier/Cimarosa ed il 1799/Cimarosa ed il 1799.htm, (20/09/2014).

(ICMC 2012) (pp. 289-292). Ann Arbor, MI: Michigan Publishing.

[5] D. Damm, C. Fremerey, V Thomas, M. Clausen, F. Kurt, M. Müller: "*A digital library framework for heterogeneous music collections: from document acquisition to cross-modal interaction*", J. Digit Libr (2012) 12:53–71, Springer.

[6] B. Croce: "*La Rivoluzione Napoletana del 1799*" - Bari Laterza 1948, page XII.

[7] A. M. Rao: "La Filosofia in soccorso dei Melodrammi: La riforma dell'Opera in musica nel Pensiero degl'illuministi Napoletani", *Proceedings of the Conference*: *Napoli 1799 fra storia e storiografia*, Vivarium, Napoli, 2002. pag.547-563.

[8] G. Giaccio: "*Monitore Napoletano*", Napoli 1799 and other reprinted editions, issue.30-31.

[9] Report of the Giunta di Governo to the King in: "De Cesare - *Aversa a Domenico Cimarosa*", p.123-128.

[10] "Cantata a 3 Voci con cori Per il Fausto ritorno Di Ferdinando IV-DG Re delle due Sicilie dopo l'ingresso delle vittoriose sue armi in Napoli Festeggiato dai Realisti di Chiaia_Musica il 23 settembre 1799 nella Chiesa della Vittoria". See at webpage: http://www.internetculturale.it/opencms/opencms/it/index.html, (20/09/2014).

[11] A. Lanfranchi Card, in Dizionario Biografico Treccani.

[12] F. Florimo: "*Cenno storico sulla scuola musicale di Napoli*", Napoli, Lorenzo Rocco, 1869, p.386, p.472, p.467-470.

[13] C. De Nicola: "*Diario napoletano : 1798-1825 Pt.1*", Napoli 1906, Società di Storia Patria.

[14] C. Botta: "*Storia d'Italia dal 1789 al 1814 sixt volume*", published on 1824, pp.176-177.

[15] Il Conservatorio di San Pietro a Majella: Electa, Napoli, 2008, p.67-72.

[16] E. Rocco: "*L'inno repubblicano del 1799 musicato dal Cimarosa*", Talk given at Accademia Pontaniana on 17 June 1888 with the title, p.182.

[17] B. Croce: "*I teatri di Napoli*", 1992. p.661n.

[18] E. Santagata: "*Il Museo storico musicale di S. Pietro a Majella*", Napoli, Giannini & Figli, 1930, card 393.

[19] Battaglini, Atti III: "Atti, leggi, proclami ed altre carte della Repubblica Napoletana", 1798-1799, Società Editrice Meridionale III , pag.488 doc. n. 1165.

[20] Eleonora, De Simone: youtube https://www. youtube.com/watch?v=p0TKcSvfYBk#t=38. And see: http://www.youtube.com/watch?v=34IB3GRsVPE#t=25, (20/09/2014).

[21] R. De Simone: "*La musica della Rivoluzione: dai canti politici all'Inno della Repubblica Napoletana*". *Napoli 1799*, Sorrento, Franco Di Mauro ed., 1999, pp.232-262.

[22] R. De Simone: "Eleonora, Oratorio - *Libretto di sala"*, San Carlo theatre in Napoli, January 1999.

[23] L. D. Baggi, M. G. Haus: "Music Navigation with Symbols and Layers: Toward Content Browsing" with *IEEE 1599 XML Encoding*, 2013, Wiley-IEEE Computer Society Press-ISBN: 978-0-470-59716-3.

[24] D. P. W. Ellis: "Aligning MIDI files to music audio", Web resource: http://labrosa.ee.columbia.edu/matlab/alignmidi/, 2013.

[25] R. Turetsky, D. Ellis. "Ground-Truth Transcriptions of Real Music from Force-Aligned MIDI Syntheses", *4th International Symposium on Music Information Retrieval ISMIR-03*, Baltimore, October 2003.

[26] Hu N. Dannenberg, R. G. Tzanetakis: "Polyphonic Audio Matching and Alignment for Music Retrieval", *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics* (WASPAA), 2003 p. 185-188.

[27] E. Dannenberg, R.B. and Hu, N. "Polyphonic: "Audio Matching for Score Following and Intelligent Audio Editors", *Proc. Of 2003 ICMC*, pp. 27-33, ICMA.

[28] F. S. Dixon: "Live Tracking of Musical Performances Using On-Line Time Warping", *DAFx'05, 8th International Conference on Digital Audio Effects*, 20-22 September 2005, Madrid, Spain.

[29] G. S. Dixon and G. Widmer: "MATCH: A Music Alignment Tool Chest", ISMIR 2005, *6th International Conference on Music Information Retrieval*, 11-15 September 2005, London, England.

[30] C. Cannam, C. Landone, M. Sandler: "Sonic Visualiser: An Open Source Application for Viewing, Analysing, and Annotating Music Audio Files", *Proceedings of the ACM Multimedia International Conference*, 2010.

[31] H. T. Eerola, P. Toiviainen: "MIR in Matlab: the MIDI TOOLBOX", ISMIR 2004. Manual at: https://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/miditoolbox/Manual.

REWIND: SIMULAZIONE DI UN'ESPERIENZA D'ASCOLTO STORICAMENTE FEDELE DI DISCHI FONOGRAFICI DIGITALIZZATI

Niccolò Pretto
Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Padova
prettoni@dei.unipd.it

Sergio Canazza
Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Padova
canazza@dei.unipd.it

SOMMARIO

Questo articolo presenta una virtualizzazione di un grammofono per la riproduzione di dischi fonografici digitalizzati, realizzata utilizzando tecnologia web. Il lavoro introduce il problema della conservazione attiva dei dispositivi per la riproduzione di documenti sonori, risaltando la centralità dell'aspetto filologico nella fruizione. La web-application riproduce fedelmente il funzionamento del grammofono, attraverso una interfaccia utente intuitiva che ne rappresenta graficamente tutte le componenti meccaniche. La descrizione delle tecnologie alla base dell'applicazione si sofferma sulle librerie *Web Audio Api*, fondamentali per ricostruire le peculiarità sonore che contraddistinguono i grammofoni per dischi in gommalacca. Nei primi anni del Novecento la mancanza di uno standard univoco per l'incisione di registrazioni fonografiche ha favorito la proliferazione di un enorme varietà di curve di equalizzazione e velocità di rotazione dei dischi: la virtualizzazione propone una soluzione flessibile per adattare in tempo reale i brani digitalizzati alle caratteristiche del supporto originale.

1. INTRODUZIONE

Le registrazioni audio rivestono un ruolo di primaria importanza come fonti documentarie per la ricerca in numerose discipline, quali musicologia, linguistica, storia e sociologia. Negli ultimi trent'anni, le risorse destinate alla conservazione e al restauro di questi documenti sono diventate sempre più consistenti. Il valore attribuito a questa parte del patrimonio culturale è cresciuto di pari passo con i dibattiti relativi alle metodologie e alle tecnologie da adottare per la loro conservazione. Proprio come accade per le fonti bibliografiche, è emersa la necessità di mettere al centro della discussione le problematiche filologiche, riguardanti, per esempio, l'autenticità e l'autorevolezza del documento [1].

Il problema relativo alla standardizzazione del processo di conservazione audio fu argomentato per la prima volta da William Storm nel 1980, che individuò "two legitimate directions", ovvero due tipologie di *re-recording* appropriate dal punto di vista archivistico. La prima definisce la copia "as the perpetuation of the sound of original recording as it was initially reproduced and heard by the people of the era" [2]. La seconda, invece, propone il raggiungimento di un obiettivo ancora più ambizioso: ottenere "the live sound of original performers" [2].

Con l'accettazione da parte della comunità scientifica del concetto "preserve the content, not the carrier" [1] e, quindi, con la presa di coscienza della necessità di digitalizzare i documenti sonori, il problema filologico è diventato sempre più rilevante. Il dibattito attuale, però, viene affrontato quasi esclusivamente dal punto di vista della conservazione e del restauro del documento, trascurando un aspetto essenziale: la fruizione. La maggior parte dei documenti sonori disponibili negli archivi e nelle varie biblioteche digitali, infatti, vengono riprodotti con applicazioni dedicate o commerciali "iTunes-like", che non pongono attenzione alle peculiarità dei dispositivi di riproduzione con cui i documenti erano riprodotti originariamente. Sono disponibili in commercio prodotti che riprendono alcune caratteristiche dei supporti di riproduzione, ma spesso sono ricostruzioni parziali e imprecise.

In controtendenza, questo articolo propone una soluzione software che mette in risalto il ruolo e le caratteristiche del supporto e dello strumento necessario per la sua riproduzione, proponendo una virtualizzazione di un grammofono per la fruizione di dischi fonografici "78 giri" digitalizzati.

Questo lavoro si colloca nell'ambito della conservazione attiva [3, 4, 5] dei supporti. Ogni documento sonoro, infatti, si può associare al supporto fisico su cui viene registrato e, quindi, al dispositivo che ne permette la riproduzione e che ne caratterizza l'esperienza di ascolto. Questi dispositivi, quali grammofoni o magnetofoni, proprio come i documenti sonori, sono destinati ad una rapida obsolescenza ed è di primaria importanza preservare le loro funzionalità e le loro peculiarità. L'articolo propone, quindi, la virtualizzazione dei dispositivi di riproduzione come l'equivalente del concetto di "copia di accesso" nel campo della conservazione dei documenti sonori. Questa deve presentare l'interfaccia dello strumento, fornendo all'utente le funzionalità, le gestualità e le sonorità tipiche dello stesso, nonché ricreare un'esperienza d'ascolto conforme alla prima "legitimate direction", ovvero al paradigma "the sound preservation of audio history" [2].

"Votre Faust" di Henri Pousseur rappresenta un esempio reale per rimarcare l'importanza della fruizione mediante

Copyright: ©2014 Niccolò Pretto, Sergio Canazza. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

uno strumento conforme al supporto e al dispositivo originali. L'opera del compositore belga rientra nelle categoria di opera "aperta", ovvero una composizione in cui è possibile alternare alcune sezioni, all'interno di certi vincoli, fornendo all'ascoltatore diverse versioni della stessa. La versione originale è stata distribuita nel 1973 con un box composto da 3 vinili e un gioco con le carte. Scegliendo casualmente alcune di queste carte, che corrispondono alle varie sezioni dell'opera, si può ottenere una tra le 30 versioni possibili della composizione. Grazie all'utilizzo dell'applicazione è possibile ricreare le condizioni di ascolto dell'epoca. Un *player* moderno permette un accesso alle varie sezioni veloce e precisa, ma non conforme al metodo di fruizione originale. La virtualizzazione, invece, prevede di spostarsi all'interno di un disco esclusivamente muovendo il braccio del grammofono, costringendo così l'utente ad ascoltare l'opera proprio come l'autore della composizione l'aveva ideata.

REWIND (Restoring the Experience: a Web-based Interface for accessiNg Digitized recordings) mette a disposizione dell'utente uno strumento addizionale, che, sebbene non storicamente fedele in relazione al grammofono originale, può risultare utile per lo studio e il perfezionamento della riproduzione di un brano. Si tratta della virtualizzazione del trasformatore di segnale Albiswerk modello 502/50, introdotto nel mercato nel 1957 e prodotto fino al 1983.

La sezione 2 offre una descrizione delle tecnologie web alla base della virtualizzazione; la sezione 3 presenta l'interfaccia utente che permette di interagire con l'applicazione, mentre la sezione 4 introduce il problema delle equalizzazioni e della velocità di rotazione e propone la soluzione sviluppata.

2. TECNOLOGIA WEB E GRAFO AUDIO

In accordo con il paradigma "distribution is preservation" [6], la rete risulta un ineguagliabile mezzo per la distribuzione dei contenuti sonori digitalizzati, sebbene necessiti di un grande lavoro di organizzazione dei contenuti. Sulla base di questo assunto, la scelta progettuale si è focalizzata sulle tecnologie web. L'applicativo si basa su una piattaforma LAMP (Linux, Apache, MySql, PHP); i linguaggi utilizzati nella realizzazione dell'applicazione sono HTML5, PHP, Javascript e CSS3. Per quanto riguarda Javascript, si sono rivelate essenziali le librerie *JQuery* e *Web Audio Api*: la prima per gli aspetti grafici e strutturali, per manipolare il *Document Object Model* (DOM) e per mantenere la compatibilità tra i browser; mentre la seconda consente di gestire componenti audio complesse. Quest'ultima permette, inoltre, di superare le limitazioni del tag HTML5 <*audio*> [7]: offre, infatti, la possibilità di sviluppare applicazioni audio complesse senza dover ricorrere a plugin come Flash e QuickTime.

Queste librerie sono costruite attorno al concetto di *context*, ovvero un grafo diretto di *nodi* audio, modificabile dinamicamente, che definisce il flusso audio dalla sorgente (solitamente un file audio) alla destinazione (nella maggior parte dei casi diffusori acustici o cuffie). Mentre il segnale

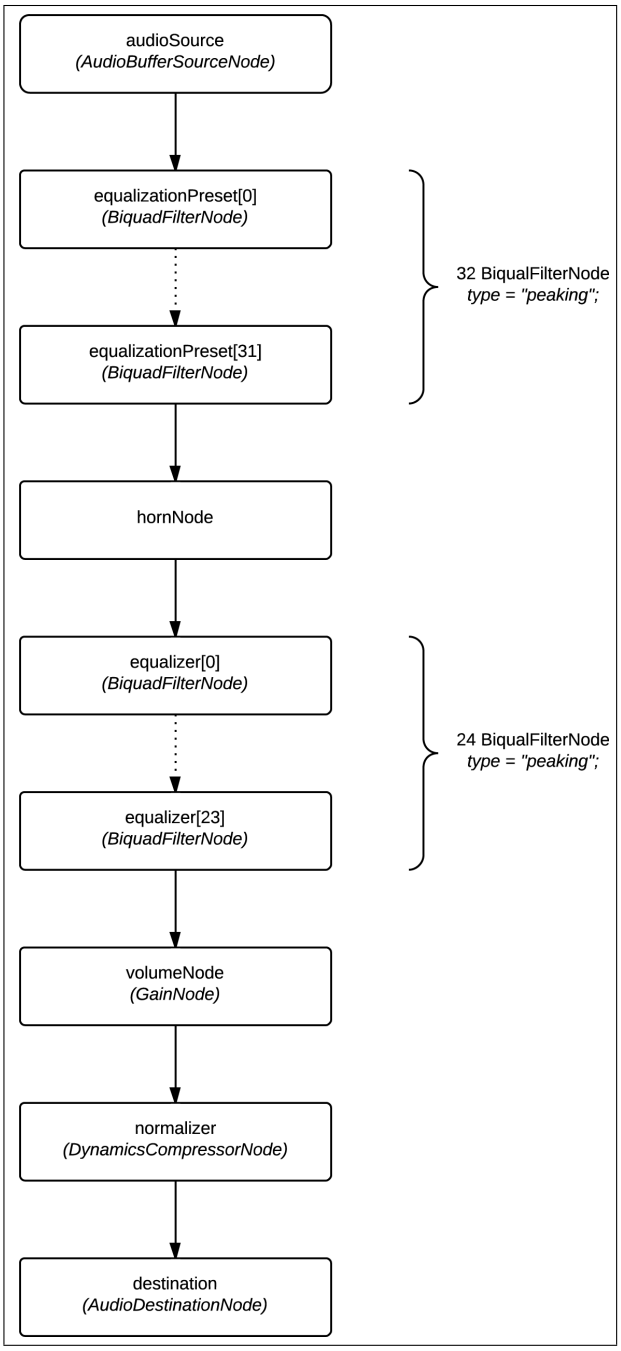


Figura 1. Grafo audio completo (*context*) utilizzato per la virtualizzazione del grammofono.

passa attraverso un nodo, le sue proprietà possono essere modificate e controllate. Le tipologie di nodi messi a disposizione dalla libreria per creare il grafo sono:

Source nodes Sorgenti audio come buffer, live audio inputs, <*audio*> tag, oscillatori o JS processors [1] ;

Modification nodes Filtri, convolvers, JS processors, panners, etc.;

Analysis nodes Nodi per l'analisi o JS processors;

[1] nodo di tipo *ScriptProcessorNode*, che permette allo sviluppatore di sintetizzare ed elaborare l'audio mediante algoritmi scritti in linguaggio Javascript.

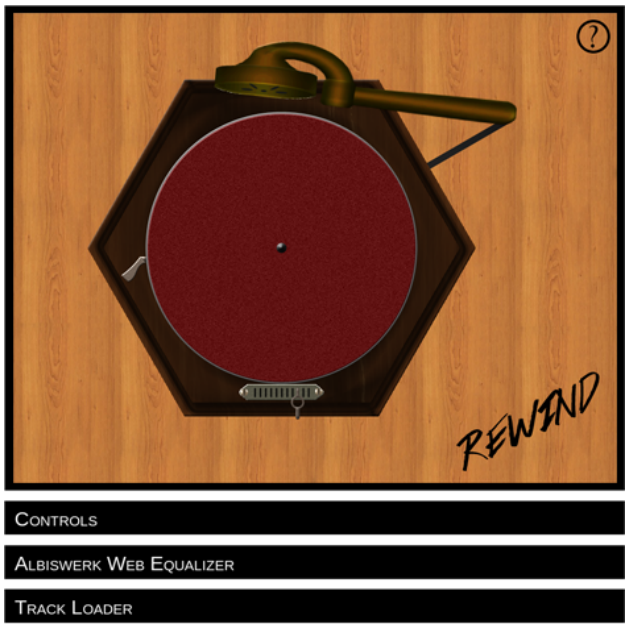


Figura 2. Grammofono prima del caricamento di un brano.

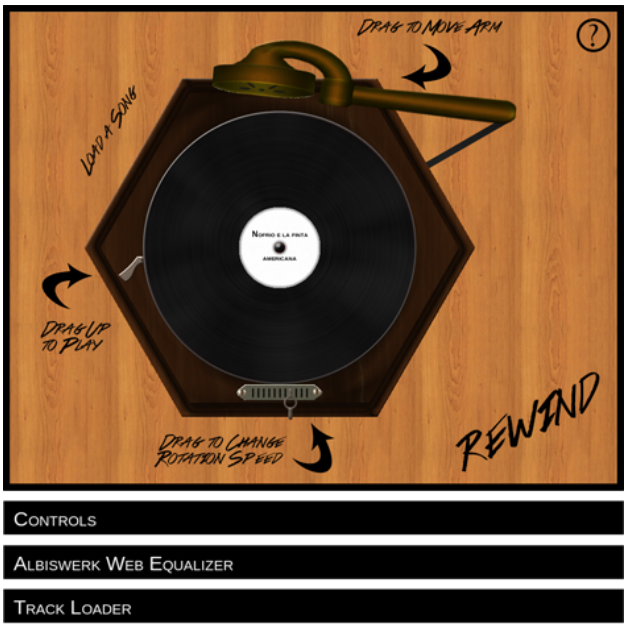


Figura 3. Grammofono dopo il caricamento di un brano.

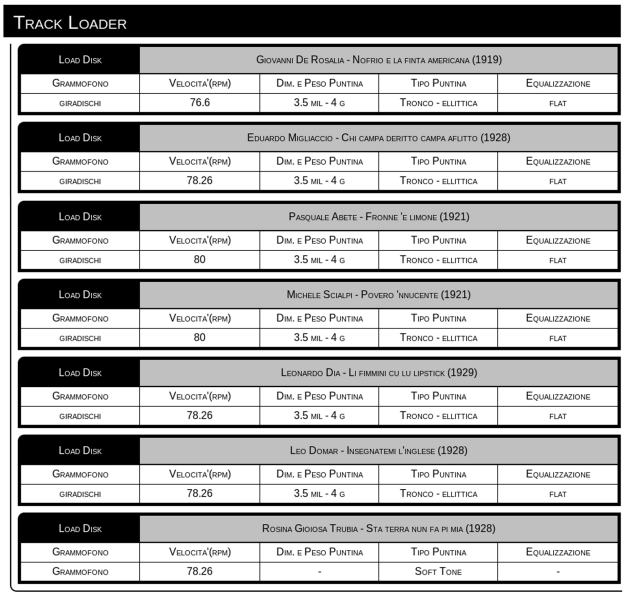


Figura 4. Track Loader per selezionare il brano da riprodurre.

Destination nodes Output audio e buffer per offline processing. [7]

La Figura 1 propone i nodi che compongono il grafo dell'applicazione, nel caso in cui tutte le funzionalità siano abilitate. Il primo nodo rappresenta la sorgente audio, contiene il buffer della traccia che viene riprodotta dal grammofono e un parametro per modificare la velocità di riproduzione. Il segnale passa attraverso una serie di 32 filtri low-order di tipologia *peaking* [8] che, tramite opportune configurazioni, permettono di ottenere le equalizzazioni. Questi sono contenuti in un vettore di nodi denominato *equalizationPreset* i cui indici vanno da 0 a 31 ([0], [1], ..., [31]). Il nodo seguente è fittizio ed è funzionale per gli sviluppi futuri, che prevedono di dotare l'applicazione di altre curve di equalizzazioni, derivanti dalla tipologia di "tromba" utilizzata per la riproduzione dal grammofono. La successiva serie di filtri è alla base dell'implementazione di uno strumento di equalizzazione per personalizzare il segnale mediante 24 filtri *peaking*, contenuti in un vettore denominato *equalizer*. Il nodo *volumeNode* consiste in un semplice *gain* che regola il guadagno applicato al segnale, che, collegato ad un <*range*> HTML5 permette all'utente di modificare il volume di riproduzione. Il grafo presenta poi un *DynamicsCompressorNode* [8], necessario per compensare le eccessive variazioni del volume percepito ed evitare possibili distorsioni del segnale, dovute all'introduzione delle curve di equalizzazione. L'ultimo nodo, *destination*, rappresenta ciò che viene ascoltato dall'utente; spesso viene considerato come il dispositivo di output collegato agli altoparlanti o alle cuffie [8].

3. INTERFACCIA UTENTE

L'applicazione cerca di riprodurre il più fedelmente possibile il funzionamento del grammofono e le sue modalità di utilizzo. Per fornire all'utente un'esperienza d'uso simile a quella reale è stata realizzata una *user interface* intuitiva che rappresenta nel dettaglio le varie componenti meccaniche del grammofono. Nelle Figure 2 e 3 si può vedere come l'applicazione sia composta da un *player* e da un menù a fisarmonica che contiene i controlli per la personalizzazione del segnale (*Controls*), l'equalizzatore Albiswerk (*Albiswerk Web Equalizer*) e una sezione per caricare il brano desiderato (*Track Loader*).

Dopo il caricamento della pagina web, l'applicazione si presenta come in Figura 2. Nella sezione *Track Loader*, l'utente può selezionare la traccia desiderata tra quelle proposte (Figura 4). I brani a disposizione provengono dalla collezione di Giuliana Fugazzotto [9]. Per ogni brano vie-

ne visualizzata una tabella contenente tutti i metadati dell'opera, recuperati tramite un'interrogazione del database dell'applicazione. Una volta caricata la traccia, vengono sbloccate le funzionalità del *player* e della sezione *controls*: l'applicazione si trova nello stato raffigurato dalla Figura 3.

Le principali componenti del *player* sono il braccio del grammofono, una levetta di azionamento e un cursore per la velocità di rotazione. È possibile spostare il braccio trascinandolo all'interno del disco: l'applicazione tiene traccia dello spostamento del mouse, calcola l'angolo di spostamento e ruota il braccio di conseguenza. Il movimento rotatorio che lo caratterizza durante la riproduzione è impostato mediante una trasformazione sviluppata con una funzione jQuery e CSS3.

La levetta di azionamento consiste in un tag <*range*> di HTML5, opportunamente personalizzato con fogli di stile, con ammette esclusivamente due valori: uno per avviare la rotazione del disco e quindi la riproduzione, l'altro per fermarla.

Il cursore per la velocità è anch'esso un <*range*>, ma permette di impostare valori di tra 70 e 80, corrispondenti al numero di rotazioni del disco al minuto. Nella sezione *Controls* vengono messi a disposizione, mediante 3 sottosezioni, i valori pre-impostati per le velocità standard più comuni (sottosezione *Rotation Preset*), uno strumento per le curve di equalizzazione e il trasformatore di segnale Albiswerk modello 502/50 (Figura 5). Quest'ultimo viene attivato aprendo il menù, e si disattiva solo se chiuso. L'interfaccia è composta da 24 <*range*>, opportunamente modificati con stili CSS3.

Data la varietà dei browser utilizzati per la navigazione, durante lo sviluppo si è posta una particolare attenzione alla compatibilità. Per il momento i browser desktop che permettono di utilizzare l'applicazione correttamente sono Chrome, Firefox e Safari. Tra i principali programmi per la navigazione, solo Internet Explorer non permette l'utilizzo del grammofono virtualizzato, poiché al momento non supporta le librerie *Web Audio Api*.

4. EQUALIZZAZIONI E VELOCITÀ DI ROTAZIONE

Le curve di equalizzazione, che vengono applicate in fase di riproduzione da un grammofono, ricadono nella categoria delle alterazioni del segnale che Schüller [10] ha denominato alterazioni "intenzionali". Seguendo le indicazioni fornite dalla tipologia di re-recording *Type B* di Schüller [10, 11], le tracce ri-mediate devono essere equalizzate per compensare le equalizzazioni adoperate nella fase di registrazione del supporto. Le informazioni riguardanti la curva di equalizzazione corretta, come quelle per l'esatta velocità di rotazione, spesso non sono agevolmente reperibili [9]. Bisogna considerare il fatto che, solo per i cosiddetti "78 giri", ci sono innumerevoli standard e curve di equalizzazioni che variano a seconda della casa discografica e dell'anno di produzione. Il grammofono virtualizzato risulta uno strumento adatto a questo studio grazie alla possibilità di personalizzare le curve in real-time.

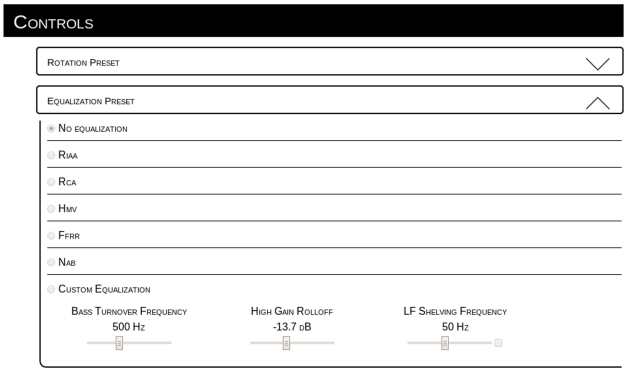


Figura 6. Sottosezione dedicata alla personalizzazione delle curve di equalizzazione.

A partire da una registrazione *flat*, quindi, senza alcuna compensazione, è possibile applicare una tra le più comuni curve di equalizzazione pre-caricate (*RIAA, RCA, HMV, FFRR, NAB*) oppure ricostruirne una, tarando opportuni parametri (vedi Figura 6).

Una curva di equalizzazione è caratterizzata da due punti di *turnover*, ovvero le frequenze al di sotto delle quali le case discografiche applicavano i filtri per ridurre il guadagno delle frequenze interessate, e che necessitano di essere compensate. Se ci si riferisce a basse o medie frequenze si parla di *bass turnover*, altrimenti, per medie o alte, *high turnover*. Quest'ultimo valore viene solitamente sostituito con un parametro denominato *rolloff*, ovvero la diminuzione di intensità (in dB) al punto di taglio delle alte frequenze (10 kHz); conoscendo la pendenza della curva (6 dB/ottava), infatti, è semplice individuare il punto di *turnover* [12]. In alcuni standard è previsto l'utilizzo di un ulteriore filtro *Low Frequency Shelving*, al fine di minimizzare i rumori delle basse frequenze (*rumble*, *hum*) introdotti dal sistema di registrazione.

Per impostare i valori dei filtri, e quindi ottenere la curva desiderata, l'applicazione mette a disposizione tre tag <*range*> di HTML5. Come già anticipato nella descrizione del *context*, vengono utilizzati 32 filtri di tipo *peaking*; questa tipologia di filtro consente di applicare un determinato *gain* a un range di frequenze, che viene definito da una frequenza centrale e un *Q factor* (in questo caso pari a 5.72, che corrisponde a una larghezza di banda di un quarto di ottava). L'algoritmo che modifica i guadagni dei filtri, per prima cosa si occupa di calcolare il punto di *high turnover* (h) a partire dal *rolloff* (r), attraverso la seguente formula:

$$h = \sqrt{\frac{10000^2}{10^{-(\frac{r}{10})} - 1}}$$

Il guadagno realativo ad ogni filtro, viene calcolato sommando il guadagno parziale derivante da ognuno dei tre punti di *turnover*, ottenuti con:

$$gainLF = 10\log_{10}(1 + \frac{l^2}{f^2})$$

$$gainHF = -10\log_{10}(1 + \frac{f^2}{h^2})$$

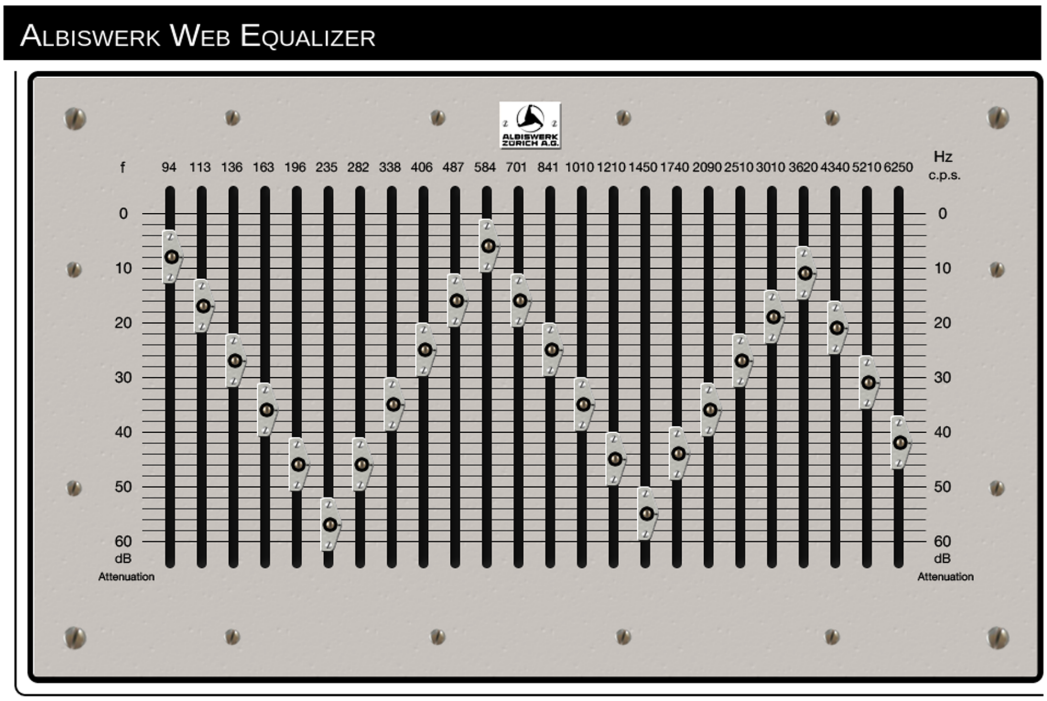


Figura 5. Virtualizzazione del trasformatore di segnale Albiswerk modello 502/50.

$$gainSHF = -10\log_{10}(1 + \frac{s^2}{f^2})$$

dove l è la frequenza del *bass turnover*, f la frequenza del nodo, h la frequenza del *treble turnover* e s frequenza del *low shelving turnover*.

L'algoritmo, successivamente, trova il valore massimo e calcola un'attenuazione, da applicare a tutti i filtri, per mantenere i guadagni al di sotto dei +10 dB, soglia al di sopra della quale c'è un forte rischio di distorsione del suono. Questa attenuazione viene compensata dal nodo *DynamicsCompressorNode*, che senza equalizzazioni è inizializzato a -20dB, mentre, a seconda della necessità, può essere incrementato fino a una soglia massima di 0 dB.

Come anticipato nell'introduzione, l'utente ha a disposizione anche la virtualizzazione del trasformatore di segnale Albiswerk modello 502/50: è stato riprodotto il banco di 24 filtri con larghezza di banda pari a un terzo di ottava, dando la possibilità di variare il guadagno da 0 a -60 dB, per le frequenze comprese tra gli intervalli centrati in 94 e 6250 Hz. L'equalizzatore è realizzato con una serie di 24 nodi *peaking*, opportunamente impostati con la frequenza centrale per ogni banda e un *Q factor* pari a 4.32.

La velocità di rotazione del disco deve coincidere con quella utilizzata in fase di incisione. L'unità adottata per riferirsi alla velocità di rotazione è *rpm*, acronimo di *revolutions per minute*. Tra il 1894 e il 1930 sono state adottate diverse velocità di incisioni dei dischi, da 70 a 90 rpm: ognuna di queste rappresentava un compromesso tra durata di riproduzione, qualità del segnale audio, possibilità tecnologiche e ottimizzazione dei costi di produzione. Lo standard 78 rpm fu introdotto nel 1912, ma divenne condiviso solo nei primi anni Trenta, con l'introduzione dei primi dispositivi muniti di motori elettrici sincroni [13, 9]. Per questo motivo per molto tempo le case discografiche adottarono svariati standard. Gli errori nel processo di rimediazione relativi alla velocità di rotazione impattano notevolmente sulla qualità della traccia digitalizzata, poiché ne modificano il *pitch*, compromettendone la fedeltà, e rendendola a tutti gli effetti inutilizzabile a fini informativi. Un esempio di digitalizzazione errata è la prima pubblicazione su CD-A delle registrazioni Savoy & Dial (anni Quaranta del secolo scorso) del sassofonista Charlie Parker, che proponeva alcune tracce digitalizzate a velocità troppo elevate, la cui esecuzione presentava una tonalità diversa da quella originale, mai utilizzata in quei suoi assoli.

I dispositivi dell'epoca permettevano di modificare la velocità di rotazione per adattarla alla tipologia di disco: pertanto è stata aggiunta questa funzionalità, sia sul grammofono mediante un cursore, sia con una sezione specifica che permette di impostare alcune delle velocità più comuni. Di default vengono messe a disposizione dell'utente le seguenti velocità: *Columbia* (70 rpm), *Victor e HMV* (71.29 rpm), *Acoustic Victor* (76.59 rpm), *Electronic Recording* (78,26 rpm) e *Acoustic Columbia e Vertical Recording* (80 rpm) [12] (Figura 7). Come per le equalizzazioni, l'applicazione può risultare un valido aiuto per un ascolto fedele dei brani.

Per una corretta riproduzione, l'applicazione richiede un valore della velocità, espresso in rpm, necessario come riferimento. Per cambiare la velocità di riproduzione, le librerie *Web Audio Api* mettono a disposizione la variabile *playbackRate* [8], contenuta nei nodi *AudioBufferSourceNode*. In base al valore percentuale passato a questo parametro, aumenta o diminuisce la velocità di riproduzione. In fase di caricamento del disco, viene inizializzato a 1 e associato alla velocità in giri al minuto passata in fase di

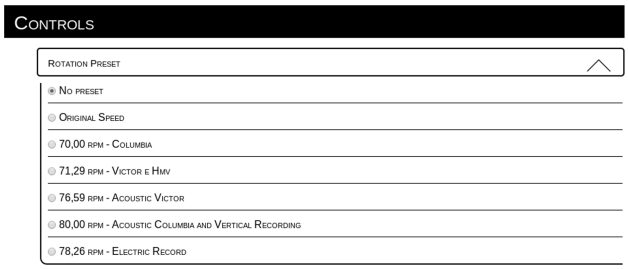


Figura 7. Sottosezione dedicata alla personalizzazione delle curve di equalizzazione.

caricamento del brano. Quando l'utente sposta il cursore del grammofono o seleziona una velocità pre-impostata attraverso un *radio button*, vengono rapportati il valore della velocità selezionata e quello originale. Sulla base del valore ottenuto, l'applicazione modifica il *playbackRate* e avvia i metodi predisposti per variare in tempo reale la velocità di rotazione del supporto fonografico virtuale, che dopo aver aggiornato lo stato del grammofono, agiscono sui fogli di stile (CSS3) relativi alle animazioni del braccio e del disco.

5. CONCLUSIONI E SVILUPPI FUTURI

Dal punto di vista della ricerca scientifica, REWIND (Restoring the Experience: a Web-based Interface for accessiNg Digitized recordings) è collocabile nell'ambito della conservazione attiva dei dispositivi di riproduzione. Il lavoro ha affrontato nel dettaglio diversi aspetti relativi alla fruizione, risaltandone la fruizione storicamente fedele. Grazie alle librerie *Web Audio Api* è stato possibile conciliare la versatilità e la facilità di accesso di internet con le funzionalità complesse in ambito audio tipiche di applicazioni desktop. Il risultato ottenuto è una virtualizzazione che ricostruisce fedelmente i meccanismi di un grammofono per dischi fonografici "78 giri". Allo stesso tempo l'applicazione riesce a essere adattabile ai vari standard di incisione dei dischi, grazie alla personalizzazione delle curve di equalizzazione e le velocità di rotazione. A ciò si aggiunge un altro importante punto di forza: la capacità di operare il tempo reale.

Le possibilità di utilizzo di questo strumento sono molteplici e spaziano dallo studio in ambito filologico e dei formati audio alla distribuzione e valorizzazione di beni culturali, in un'ottica di *edutainment*.

L'applicazione si apre a possibili miglioramenti e sviluppi futuri. Per rendere più verosimile il grammofono virtuale, sarebbe interessante sviluppare un modello più preciso dell'incisione del disco in gommalacca, che consentirebbe all'utente di spostarsi all'interno della traccia con maggior precisione, trascinando il braccio o ruotando il disco stesso. A questo potrebbe seguire una ricerca per ricreare equalizzazioni ancora più fedeli, mediante l'utilizzo di filtri personalizzati.

Dati i risultati ottenuti, questo lavoro potrebbe trovare la sua naturale prosecuzione nello sviluppo di altre applicazioni per la virtualizzazione di dispositivi diversi dal grammofono, come un magnetofono. Infine, data la diffusione di dispositivi *mobile*, risulterebbe molto interessante eseguire una operazione di *porting* dell'applicazione su piattaforme come iOS o Android, per valutare le potenzialità della soluzione utilizzando librerie audio più complete e *gesture* evolute.

6. BIBLIOGRAFIA

[1] F. Bressan and S. Canazza, "A Systematic Approach to the Preservation of Audio Documents: Methodology and Software Tools," *Journal of Electrical and Computer Engineering*, vol. 2013, p. 21.

[2] W. Storm, "The establishment of re-recording standards," *Phonografic Bulletin*, vol. 27, 1980.

[3] S. Canazza and A. Vidolin, "Preserving electroacustic music," *Journal of New Music Research*, vol. 30, no. 4, 2001.

[4] D. Schuller, "The Safeguarding of the Audio Heritage: Ethics, Principles and Preservation Strategy," tech. rep., 2005.

[5] S. Canazza and F. Avanzini, *Virtual analogue instruments: an approach to active preservation of the Studio di Fonologia Musicale*. Ricordi, July 2011.

[6] E. Cohen, "Preservation of Audio in Folk Heritage Collections in Crisis," *Council on Library and Information Resources*, 2001.

[7] B. Smus, *Web Audio API (Advanced Sound for Games and Interactive Apps)*. O'Reilly, 2013.

[8] *Web Audio API (W3C Working Draft)*, Apr. 2014.

[9] G. Fugazzotto and S. Canazza, *Sta terra nun fa pi mia: i dischi a 78 giri e la vita in America degli emigranti italiani del primo Novecento*. Nota, 2010.

[10] D. Schuller, "The ethics of preservation, restoration and re-issues of historical sound recordings," *Journal of Audio Engineering Society*, 1991.

[11] G. Boston, *Safeguarding the Documentary Heritage. A guide to Standards, Raccomanded Practices and Reference Literature Related to the Preservation of Documents of All Kinds.* UNESCO, 1988.

[12] G. Fugazzotto, *Il repertorio musicale dell'emigrazione italiana su dischi a 78 giri: la produzione made in USA dei primi trent'anni del Novecento*. PhD thesis, 2009.

[13] R. Warren, "Before the fine groove and stereo record and other innovations.," *Journal Audio Engineering Society*, vol. 25, no. 10/11, 1977.

PERMUTATION SYNTHESIS

Giacomo Valenti
Control and Computer Engineering Dept.
Politecnico di Torino
giacomo.valens@gmail.com

Andrea Valle
CIRMA-StudiUm
Università di Torino
andrea.valle@unito.it

Antonio Servetti
Control and Computer Engineering Dept.
Politecnico di Torino
antonio.servetti@polito.it

ABSTRACT

This paper introduces an experimental digital sound synthesis technique called *Permutation synthesis*, which focuses around creating new waveforms by moving groups of samples (*chunks*) of existing waves. Similarly to granular synthesis, permutation synthesis is a time-based technique, i.e. it operates directly on the discrete waveform: by means of varying the length of the chunks, several perceptual effects can be obtained. The most important parameter in permutation synthesis is the *permutation frequency*, which is inversely proportional to the chunk length; the resolution of this parameter is directly related to the sampling frequency (given the fact that a chunk is always an integer number of samples), thus a time-quantisation error is defined.
An algorithm for real-time permutation implemented as a SuperCollider plug-in is described, consisting of 3 UGens performing permutation synthesis in slightly different ways. The resulting signals are then analysed and their time and spectral effects are justified by defining formulas that analytically quantify the results.

1. PERMUTATION SYNTHESIS AND THE TIME-DOMAIN APPROACH

Permutation synthesis, i.e. a synthesis by permutation of samples, is a time-based technique. Historically, the most well-known synthesis techniques (from additive to subtractive to various members of the modulation family) have their roots in the spectral domain. The time-based family is younger in comparison, and typically many of the techniques belonging to this area share an experimental, empirical approach.
Among them, granular synthesis is perhaps the most notable: taking small grains of existing sounds, shaping them with an envelope and then scattering them in time and frequency with variable rates introduced a whole new perspective in the creation of sounds [1]. Permutation synthesis is similar to a particular variant of granular synthesis, the so-called time-granulation [1]: here grains are taken from one or more existing files, an envelope is applied, and then the grains are reproduced over time. If the source of the grain is a single audio stream, granulation results in scrambling parts of the same signal, which is the principle of permutation synthesis. However, most granulation approaches operate by applying an envelope, thus eliminating most of the discontinuities. Moreover, grains are typically scattered in time following some stochastic distributions. On the contrary, in permutation synthesis time discontinuities are the main feature, and the scrambling process is organised following a precise time-pattern.
Other techniques like synthesis by fragmentation and growth or PSOLA are based on the same principle: moving, replacing, repeating groups of samples whose duration is typically under the time resolution of the human ear, thus resulting in radically new sounds.
Not every newly produced sound must come from an existing one: techniques like dynamic stochastic synthesis and synthesis by instruction give the user full creativity by letting him or her start from scratch: by guiding the position of the next sample in the former case, and by using a logic gate paradigm in the latter (In general see [2],[3]).

Copyright: ©2014 Giacomo Valenti, Andrea Valle, Antonio Servetti. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

2. DESIGN

At its core, permutation synthesis is quite simple, designwise: an array of samples is given as an input to the permutation process, the array is then divided into chunks of a given dimension, the chunks are rearranged in a new order, and the rearranged array is the output (see Figure 1). This rearrangement often creates situations in which two samples that were meant to be away from each other are now adjacent, and this typically results in a waveform discontinuity. As already said, discontinuities are the heart of permutation synthesis: they both distort and enrich the resulting sound; by simply moving block of samples (chunks), a new waveform is created that retains some features of the original one and introduces new ones.
As the process operates in the digital domain, the requirements are minimal: a stream of samples and a buffer where to store them in order to retrieve them for rearranging. The parameters supplied by the user are permutation frequency (internally converted into the chunk size) and the pattern after which the chunks will be scrambled. An interesting side effect of the permutation technique is that the process does not affect global amplitude of the signal, as it simply re-organises the waveform. Thus, no digital distortion (in the sense of clipping) is introduced by the process, and the amplitude peak of the source signal is translated in the amplitude peak of the processed one.

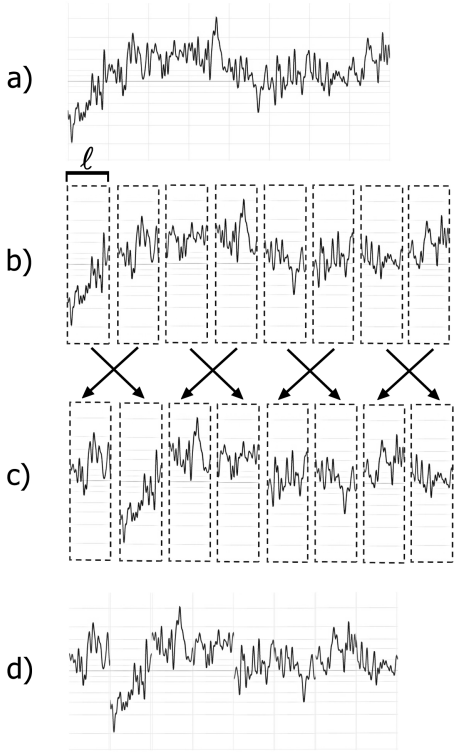


Figure 1. Permutation steps: a) the original audio source; b) division into chunks of length l; c) chunk scrambling; d) the permuted audio output.

2.1 Permutation frequency

Scrambling chunks is a trivial task, but indeed the time information related to a sample depends on the sampling frequency. A size of 50 samples does not provide any information on how much the chunk lasts; the chunk size parameter is then set implicitly by defining the permutation frequency (f_p) that indicates how often a new chunk occurs in time (to put it in another way: it is the discontinuity frequency). The size in samples of each chunk is then obtained by dividing the sampling frequency by f_p.

2.2 Time quantisation error

The conversion from frequency to chunk size leads indeed to non-integer values, that are not legal for the discrete domain. As a main concern is to maintain permutation frequency constant, all chunks are forced to have the same integer size by rounding. As an example, for $f_s = 48$ KHz and $f_p = 850$ Hz the chunk length is approximately 56.47 samples which gets rounded to 56. This means that different permutation frequencies could lead to the same rounded chunk size, which just in one case is the exact result of the division; in fact, the actual permutation frequency given by a 56 samples-long chunk is around 857 Hz. The difference between the expected f_p and the actual f_p can be seen as a time quantisation error.
The time quantisation error can be seen as an analogous in the time domain to digital quantisation error for amplitude, as it decreases with higher sampling frequencies. Moreover, for a given f_s it (globally) increases along with f_p (see Figure 3); time quantisation errors is more relevant with low sampling frequencies and/or high permutation frequencies, as shown in Figure 3.

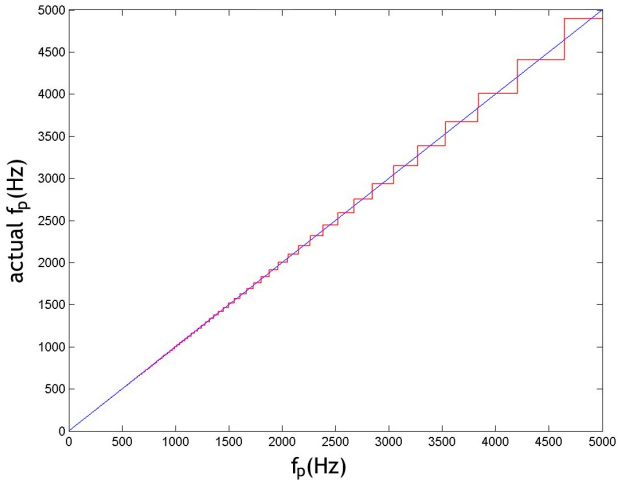


Figure 2. Desired f_p (straight line) versus actual f_p (stairs) given by a sampling frequency of 44100 Hz.

As f_p increases, the time quantisation error varies back and forth, always returning to zero but less and less frequently; however it is clear that its local (absolute) maximum increases with the permutation frequency and transversely decreases with higher sampling frequencies (see Figure 3).

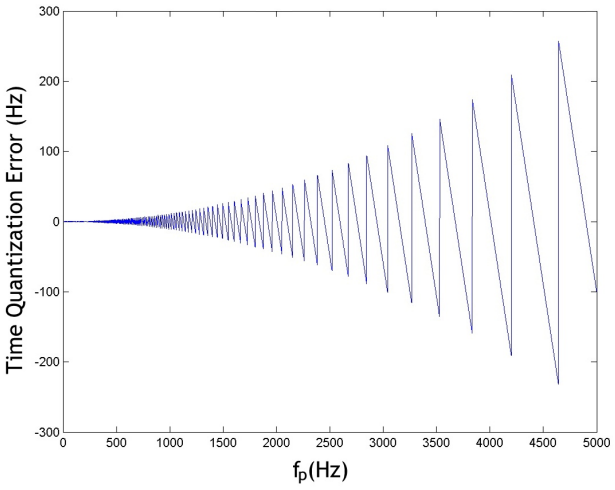


Figure 3. Time quantisation error for a sampling frequency of 44100 Hz.

2.3 Latency

If permutation synthesis is implemented in a real-time context, the presence of a buffer brings up a latency issue. An efficient indexing approach can be devised, in which samples are immediately written on the buffer according to the rearranged pattern, then the buffer is read from beginning to end and sent out to the audio stream. Changing the order

of the samples inevitably leads to store sample chunks that are going to be played later, waiting for incoming samples to be played first. This means that to further minimise latency the buffer could be read while it is being written on, but never in a way that would lead to reading a position of a sample that still has to happen, and this is highly dependent on the chosen pattern.

3. PERMUTATION RESULTING EFFECTS

Until now it has been simply stated that discontinuities are the heart of permutation synthesis because they enrich and distort the sound. A more analytical discussion follows.

3.1 Perceptual effects

In permutation synthesis the more interesting perceptual phenomena take place in the permutation range $[5, 30]$ msec, in which various possibilities occur, from a continuous perceptual scenario to a discrete one. Depending on the permutation frequency, four perceptual behaviours emerge:

- $2 < f_p < 20 \Rightarrow$ *arpeggio effect*: discontinuities are clearly audible as clicks and the chunk length is such that an entire note or syllable could be swapped with another.
- $20 < f_p < 100 \Rightarrow$ *modulation effect*: the timbre is affected and notes are distorted with a sort of tremolo effect, each discontinuity is perceived as a new attack phase of the note.
- $100 < f_p < 6000 \Rightarrow$ *harmonic distortion effect*: a sort of robotic buzz is audible, proportional to f_p, which affects the pitch of each note; discontinuities are no longer perceived.
- $f_p > 6000 \Rightarrow$ *dynamic distortion effect*: from this values onward, the permutation yields a similar effect to dynamic saturation filters even though not even a single sample amplitude has been changed.

The transition between subsequent behaviours is very smooth and strongly depends on the listener perception and the source sound, to the point that in some cases there is a large range of values in which two adjacent behaviours overlap.
However it is still worth noting that for sounds that have a periodic spectrum like pitched notes f_p highly affects the pitch when it is around the fundamental, whereas for percussive sounds the effect resembles a notch filter, flavoring sounds according to f_p.

3.2 Temporal effects

The temporal effects, or more properly the waveform effects, are of course more predictable even without actually hearing the permuted sounds. However, for the sake of readability of the resulting waveforms, from now on the simplest scenario has been chosen: a sinusoid as the source and an even-odd ([1,0]) pattern for the permutation. If the permutation frequency is set as an even multiple of the sinusoid frequency, we are in fact permuting inside the period, creating a new waveform with the same pitch as the original but a different timbre, as seen in figure 4. If the

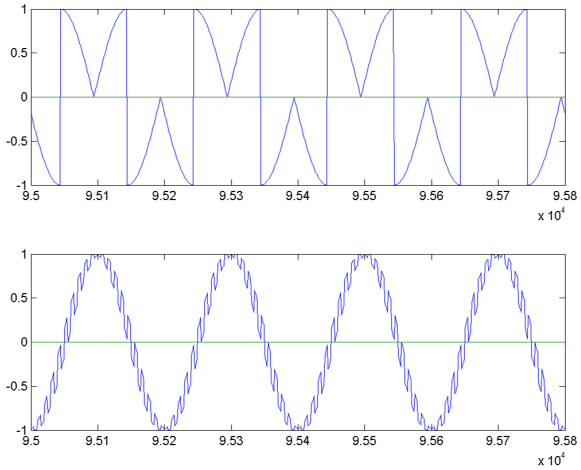


Figure 4. Permuting inside the period: with a sampling frequency of 44100 Hz a sinusoid at 220.5 Hz changes its waveform by being permuted with $f_p = 882$ Hz (top) and $f_p = 8820$ Hz (bottom).

permutation frequency is unrelated to the fundamental (for example by simply choosing a value which is not an integer multiple) a waveform whose real period is the least common multiple between the fundamental and f_p is obtained, but there can be still a noticeable pattern given by the periodic occurrence of discontinuities and the swapped portions of period of the original sinusoid, as seen in figure 5.

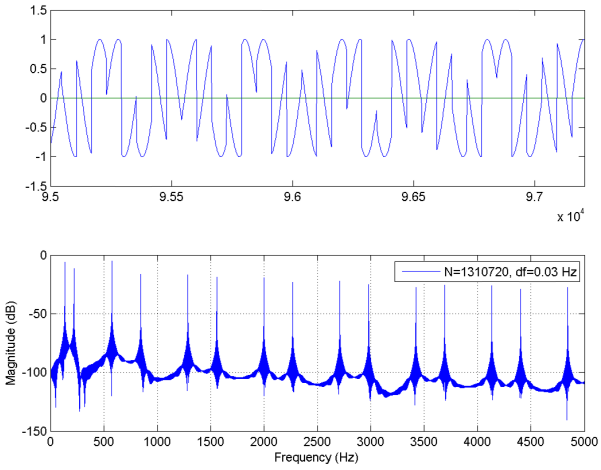


Figure 5. Waveform and spectrum of a 220.5 Hz sinusoid permuted with $f_p = 715$ Hz.

3.3 Spectral effects

The effects of permutation synthesis on the sound spectrum are here analyzed to provide a theoretical background to the perceptual features presented in Section 3.1. Here we keep using a simple waveform and an odd-even permutation pattern in order to have figures with an easily readable

spectrum and a simpler analytic demonstration of the phenomenon in section 4.
The spectrum in Figure 5 shows that, if a sinusoid is permuted with $f_p = 715$ Hz, then pairs of harmonic peaks are added to the fundamental frequency. These pairs of peaks appear near the odd harmonics of $f_p/2$ (which are not present) respectively at the frequencies $(2n+1)\frac{f_p}{2} - f$ and $(2n+1)\frac{f_p}{2} + f$. Obviously, those peaks that result in a negative frequency are aliased on the positive frequencies of the spectrum. Equation (1) represents this effect in an analytical form.

$$\sin(2\pi ft) \rightarrow \sin(2\pi ft) + \sum_{n=0}^{+\infty} \sin\left(2\pi\left[(2n+1)\frac{f_p}{2} - f\right]t\right) + \\ + \sin\left(2\pi\left[(2n+1)\frac{f_p}{2} + f\right]t\right) \quad (1)$$

4. ANALYTICAL DEMONSTRATION

If the permutation process is thought of as a kind of modulation of the source with a rectangular wave, then all the empirical observations presented in the previous sections can be justified by the following analytical demonstration. Let us rewrite the process of permuting a chunk of samples with frequency f_p as follows:

- take two copies of the source.
- apply a rectangular periodic windowing function (whose period is twice the chunk length) to the first source, obtaining the even chunks.
- apply the same windowing function, but phase-inverted, to the second source, obtaining the odd chunks.
- translate the first windowed copy (i.e., the odd chunks) forward and the second windowed copy (the even chunks) backward by an amount of samples equal to the chunk length.
- sum the two signals.

If we invert the roles of the two signals and we consider the windowing function as the carrier signal and the source as the modulator, the whole process can be seen as a ring modulation of an unipolar square wave with a 50% duty cycle.
This perspective justifies the position of the peak pairs, in fact, the modulated signal spectrum, i.e. the square wave, contains only components of odd-integer harmonics frequencies centered at integer multiples of its fundamental frequency $f_p/2$.
Moreover, we see the peak pairs and not the the square wave odd harmonics themselves because ring modulation suppresses the carrier signal and creates sidebands of the positive and negative modulator frequency centered around the carrier frequencies [2]. Lastly, the presence of the modulator original spectrum frequencies is due to the fact that the unipolar square wave is not zero mean and so it has a continuous component at 0 Hz that is suppressed, but produces the peak at $+f$. All of the above considerations were made concerning the odd-even pattern, but the same logic and steps apply to any other pattern with few minor changes: the windowing function would then be a sum of step functions which would be translated according to each index of the chosen pattern.

5. IMPLEMENTATION

Permutation synthesis has been proposed preliminary without any analytical treatment in [4], that introduces a Non Real-Time implementation in the SuperCollider language, based on swapping chunks of values from an array representing the signal. The implementation was intended as a proof of concept, is computationally inefficient and not suited for Real-Time. A general algorithm for real-time permutation synthesis is shown in Figure 6.

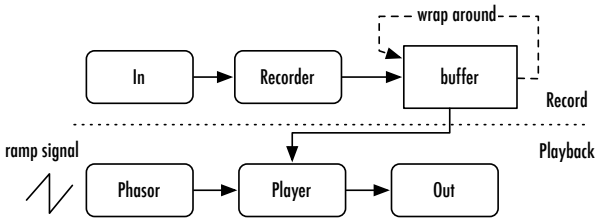


Figure 6. A general implementation for permutation synthesis.

The process is split into two parallel phases, Record and Playback. On top, a source (In) is recorded onto a recirculating buffer that wraps around overwriting old values at its beginning with new incoming ones. The second phase is Playback. Here a Player reads values from the buffer: its index must be provided by an opportune signal generator (Phasor) that is fed with a sawtooth-like signal. Buffer dimension is not particularly crucial if only it is able to store some periods of the incoming signal. This schema is apt to be implemented in various standard synthesis packages, such as Csound, PD, Max, SuperCollider, at the user level. However, in order to accurately check what happens at the sample level (typically not directly or easily accessible to the user in the aforementioned software packages), a real-time implementation of permutation synthesis has been written as a SuperCollider plug-in, i.e. as an extension of SuperCollider which consists of one or more UGens, a UGen being a basic specialised processing unit. By implementing a UGen, substantially a new audio primitive, also a greater efficiency is gained.
The SuperCollider environment [1] [5] has proven to be an apt choice as it allows to seamlessly bridge sound synthesis and algorithmic composition, thus providing also a high-level layer by which immediately experimenting with permutation synthesis in a musical context.
The SuperCollider architecture consists of two components: a server (named scsynth), dedicated to audio synthesis and processing, and a client (sclang), providing a language interface to the server. Every set of UGens must have an SC class on the language side that is bound to a C++ class, extending the server side [6].
The *PermUGens* plug-in is defined as a set of 3 different

[1] http://supercollider.sourceforge.net

UGens which allow the user to permute an existing audio source:

- **PermMod** swaps couples of chunks in an odd-even manner for a given f_p.
- **PermModArray** lets the user choose the number of chunks and the permuting pattern by providing an array of integers. As an example, the array $[1, 0, 2, 3]$ will swap the first two chunks while leaving the second two as they are.
- **PermModT** is similar to *PermMod* but adds a second frequency parameter whose ratio with f_p causes the introduction of a tail, a spare chunk with a different length, that is combined to the standard chunk.

The client side of each UGen is a simple SC class which defines the input parameters and does a first rough check of their validity. As an example, PermMod has an audio-rate (**ar*) method which takes an audio input (*in*), f_p (*freq*, set to 100 if not specified) and the standard SuperCollider *mul* and *add* parameters; if *in* is not audio-rate no information is even sent to the server module and no UGen is instantiated.

The server side is where audio processing really happens, written in C/C++ code. Inside the C++ class every UGen has several modules, the most important being the calculation functions to which the input audio samples are fed and the output is generated. Before talking into detail about the calculation function(s) of one of the PermUGens, it is worth noting that also the constructor has a very important role: for example in the case of *PermModArray* besides checking the validity of every input parameter and calculating the chunk length, it also calculates the jumping pattern from the array of integer fed by the user (more on that later); moreover it decides which calculation function to use and tests it by feeding it one single sample of audio. Since *PermModArray* is the most complete of the PermUGens, it is worth having a look at some extracts of the code to understand how it works.

In the actual implementation every PermUGen utilizes two buffers of the same size: while one is written the other is read and vice versa, therefore this is not the optimal setup latency-wise (see 2.3). Further versions of the algorithm should improve this feature.

As stated in section 2, every PermUGen relies on an indexing mechanism rather than actually moving the samples: this means that there is a single *read* variable that cycles throughout the buffers from beginning to end and back; the difference being that when it comes to the writing buffer *read* is summed to a *jump* amount in order to write samples according to the permuting pattern, while at the following cycle (which is when that buffer becomes the reading one) it gets read in sequential order as shown below:

```
if(swapped==false){

   swapbuf1[read+(pattern[index]*chunkl)]=in[i];
   out[i]=swapbuf2[read];
   }
else{

   swapbuf2[read+(pattern[index]*chunkl)]=in[i];
   out[i]=swapbuf1[read];
}
```

In the case of *PermModArray* the permuting pattern is inserted by the user when instantiating the UGen and it consists of an array of integers of length n: to be valid it must contain every cardinal number from 0 to $n-1$ (for example [2,1,3,0]); as anticipated in the previous paragraph, the *jumping pattern* (which is simply called *pattern[]* in the previous code extract) is calculated from the permuting one, and it is just another array of integers of the same size, but containing the *jump* amounts instead of the desired order of the chunks.

The most important thing when implementing a real-time synthesis UGen is allowing the user to change parameters on-the-fly. The rate at which this can occur is called *control rate* (**kr*) and its period corresponds to the audio buffer (typically 64 samples).

Since the chosen calculation function is called every control period, it also bears the responsibility of checking if a parameter has changed and to adjust the pipeline to reflect these changes. The algorithm allows to change f_p as well as the permuting pattern in real time and change the buffer size accordingly, that is, every time the new (actual) chunk length is different from the previous or the number of chunks has changed. For example if a value of the permuting pattern changes, the jumping pattern must be recalculated; whether f_p changes, the buffers have to be deallocated and reallocated and the chunk length recalculated for the next cycle. Here is an extract of the code triggered by a change of f_p:

```
if(newchunkl!=chunkl || numchunks!=newnumchunks){

    RTFree(unit->mWorld, unit->swapbuf1);
    RTFree(unit->mWorld, unit->swapbuf2);

    unit->chunkl = newchunkl;
    read = 0;
    index= 0;
```

6. EXAMPLES

The SuperCollider Permutation UGens can be used in various ways: they may serve as a distortion filter as well as a component of more complex UGen topologies of a virtual synthesizer (in SuperCollider called a *SynthDef*) [2] .

6.1 Basic usage

A basic test unit has been provided in the help files, that allows to experiment with permutation effect by controlling the signal and the permutation frequency by means of the mouse:

```
{PermMod.ar(SinOsc.ar(MouseY.kr(0,440)),
    MouseX.kr(500, 8800))}.play
```

In the above example the source audio fed to the UGen is a sinusoid whose frequency is changed in real time every control period *(*kr)* proportionally to the Y coordinate of the mouse; f_p is also changed the same way but proportionally to the X coordinate. With this configuration, every position of the arrow cursor on screen leads to a different distortion of a different note and the interesting part

[2] Some audio examples are available here: http://www.fonurgia.unito.it/wp/?page_id=606

is hearing the effects generated by the real-time variation of the aforementioned parameters. Figure 7 is a sonogram of a signal resulting from the previous code, in which the user is controlling via mouse the two frequencies. It clearly shows that spectral richness can be obtained also by a minimal example. Discrete steps in the spectrum depends indeed by the chunk integer size constraint.

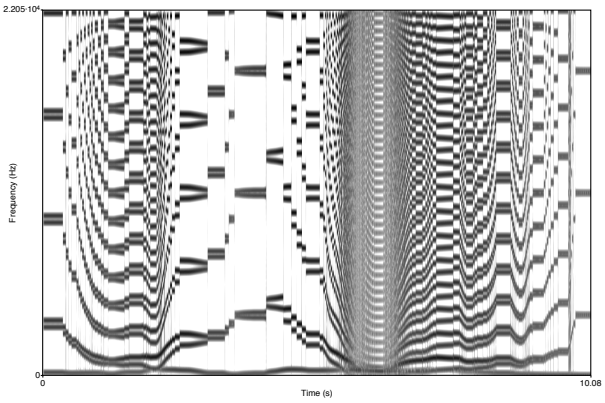


Figure 7. Effects of permutation on a sinusoid.

6.2 Wavetable permutation

In the example below, *PermMod* is a module of a more complex SynthDef that is capable of playing different notes by repitching an existing wave file. The pitch of each note is extracted and forced to chromatic scale by the *.cpsmidi* method, then this data is used to pilot the f_p parameter, therefore creating a new timbre.

```
c = Buffer.read(s, "C:/Folder/Shamisen-C4.wav");
SynthDef(\dist, { arg rate = 1;
    var sig = PlayBuf.ar(1, c, rate:rate,
    doneAction:2) ;
    var pt = Pitch.kr(sig)[0].cpsmidi ;
    Out.ar(0, PermMod.ar(
       sig,LFPulse.kr(
          1+(pt-36).midicps,0,0.5,
          pt.midicps+1,(pt+24).midicps)))
    }
).add ;
```

The example shows an application of permutation as a processing stage rather than as a pure synthesis technique.

6.3 Sequential Random Modulation

The next example consists of several stages of daisy chained modulations of a sine wave in which almost every signal and/or real-time parameter is driven by one or more of the three PermUGens. Since the code is too long to be included in the present paper, Figure 8 shows a diagram of its modules [3]. The main concept is that the last PerModArray UGen receives as its input a sinusoidal signal that has been already modulated by a previous PerModArray. This kind of iterated arrangement has been proposed many times for standard modulation techniques, in particular frequency and phase modulations[2] and shows how permutation is apt to be used as possible replacement in the same arrangements. An empirical investigation shows that the permutation algorithm introduces a DC bias, that in the example is removed as the final processing stage.

[3] We thank Marinos Koutsomichalis for providing the example and the diagram.

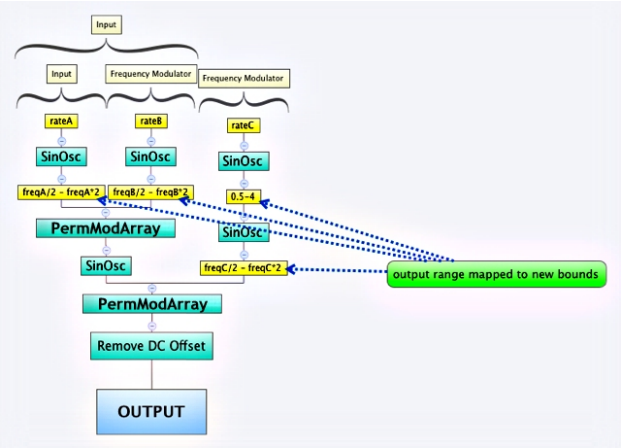


Figure 8. Modulation-Permutation diagram.

7. CONCLUSIONS

Permutation synthesis belongs to the family of non-standard synthesis technique. It does not refer to the electronic signal paradigm per se, rather it fully exploits the discrete nature of digital signals. It is a fairly simple method, indeed a purely digital member of the modulation family: nevertheless, it allows to obtain complex signals that show rich spectra but also specific temporal features, that depend on the time-domain operation at its basis. The permutation synthesis SuperCollider plug-ins have been compiled for Windows and Mac (but Linux porting is not an issue), and will be made available via the Quarks extension system in SuperCollider: user feedback is indeed relevant to assess the interest of the technique and to provide new examples of musical usage.

8. REFERENCES

[1] C. Roads, *Microsound*. Cambridge, Mass. and London: The MIT Press, 2001.

[2] C. Roads, *The Computer Music Tutorial*. MIT Pres, 1996.

[3] E. R. Miranda, *Computer Sound Design Synthesis techniques and programming*. Focal Press, 2nd Edition, 2002.

[4] A. Valle, *The SuperCollider Italian Manual*. CC Attribution-Non commercial-Share alike 2.5, available at www.cirma.unito.it/andrea/sc.html, 2008.

[5] S. Wilson, D. Cottle, and N. Collins, eds., *The SuperCollider Book*. Cambridge, Mass.: The MIT Press, 2011.

[6] D. Stowell, *The SuperCollider Book*, ch. Writing Unit Generator Plug-ins, pp. 691–720. Cambridge, Mass.: The MIT Press, 2011.

AUTOMATIC PIANO MUSIC TRANSCRIPTION SYSTEM

Giovanni Costantini
Dep. of Electronic Engineering
University of Rome "Tor Vergata"
Institute of Acoustics and Sensors
"Orso Mario Corbino"
costantini@uniroma2.it

Massimiliano Todisco
Daniele Casali
Dep. of Electronic Engineering
University of Rome "Tor Vergata"
massimiliano.todisco@uniroma2.it

Renzo Perfetti
Dep. of Electronic and
Information Engineering
University of Perugia
renzo.perfetti@unipg.it

ABSTRACT

Automatic music transcription is a challenging topic in audio signal processing. It consists in transforming the musical content of audio data into a symbolic notation. The system discussed in this paper takes as input the sound of a recorded polyphonic piano music and it produces conventional musical representation as output. For each note event two main characteristics are considered: the attack instant and the pitch. Onset detection is obtained through a time-frequency representation of the audio signal. Feature extraction is based on Sparse Nonnegative Matrix Factorization (SNMF) and Constant Q Transform (CQT), while note classification is based on Support Vector Machines (SVMs). Finally, to validate our method, we present a collection of experiments using a wide number of musical pieces of heterogeneous styles.

1. INTRODUCTION

Music transcription can be considered as one of the most demanding activities performed by our brain; not so many people are able to easily transcribe a musical score starting from audio listening, since the success of this operation depends on musical abilities, as well as on the knowledge of the mechanisms of sounds production, of musical theory and styles, and finally on musical experience and practice to listening. In fact, be necessary discern two cases in what the behavior of the automatic transcription systems is different: monophonic music, where notes are played one-by-one and polyphonic music, where two or several notes can be played simultaneously. Currently, automatic transcription of monophonic music is treated in time domain by means of zero-crossing or auto-correlation techniques and in frequency domain by means of Discrete Fourier Transform (DFT) or cepstrum. With these techniques, an excellent accuracy level has been achieved [1, 2]. Attempts in automatic transcription of polyphonic music have been much less successful; actually, the harmonic components of notes that simultaneously occur in polyphonic music significantly obfuscate automated transcription. The first algorithms were developed by Moorer [3] Piszczalski e Galler [4]. Moorer (1975) used comb filters and autocorrelation in order to perform transcription of very restricted duets.
The most important works in this research field is the Ryynanen and Klapuri transcription system [5] and the Sonic project [6] developed by Marolt, particularly this project makes use of classification-based approaches to transcription based on neural networks. The target of our work dealt with the problem of extracting musical content or a symbolic representation of musical notes, commonly called musical score, from audio data of polyphonic piano music. In this paper, an algorithm and model for automatic transcription of piano music are presented. The solution proposed is based on the onsets detection algorithm based on Short Time Fourier Transform (STFT) and a classification-based algorithm to identify the note pitch. In particular, we propose a supervised classification method that infers the correct note labels based only on training with tagged examples. This method performs polyphonic transcription via a system of Support Vector Machine (SVM) classifiers that have been trained starting from spectral features obtained by means of the well-known Constant-Q Transform (CQT). In a second experiment, we take into account the following assumption: melodic and harmonic musical structures depend on the method adopted by the composer; this means that every musical note is highly correlated to the previous note in the composition. Consequently, to improve classification results, we consider what happens before the onset; in particular, we introduce a short-term memory and an event-based memory. In a third experiment, we use as feature extraction a sparse signal representation based on Sparse Nonnegative Matrix Factorization (SNMF). The paper is organized as follows: in the following section the onset detection algorithm will be described; in the third section, the spectral features will be outlined; the fourth section will be devoted to the description of SNMF; in the fifth section, the classification method will be defined; in sixth and seventh sections, we present the systems and the results of several experiments involving polyphonic piano music. Some comments conclude the paper.

2. ONSET DETECTION

The aim of note onset detection is to find the starting time of each musical note. Several different methods have been proposed for performing this task [7, 8]. Our method is based on STFT and, notwithstanding its simplicity, it gives better or equal performance compared to other methods [7, 8].
Let us consider a discrete time-domain signal s(n), whose STFT is given by

$$S_k(m) = \sum_{n=mh}^{mh+N-1} w(n-mh)s(n)e^{-j\Omega_N k(n-mh)} \quad (1)$$

Copyright: © 2014 Costantini et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

where N is the window size, ΩN = 2π/N, h is the hop size, m = 0, 1, 2,…, M is the hop number, k = 0, 1,…, N-1 is the frequency bin index, w(n) is a finite-length sliding Hanning window and n is the summation variable. We obtain a time-frequency representation of the audio signal by means of spectral frames represented by the magnitude spectrum $|Sk(m)|$. The values $|Sk(m)|$ can be packed as columns into a non-negative L×M matrix, where M is the total number of spectra we computed and L = N/2+1 is the number of their frequencies. Afterwards, the rows of S are summed, and the following onset detection function is computed

$$f_{onset}(m) = \frac{f(m) - f(m-1)}{f(m)} \tag{2}$$

where $f(m) = \sum_{l=1}^{L} S(l,m)$ (3)

The peaks of the function fonset can be assumed to represent the times of note onsets. After peak picking, a threshold T is used to suppress spurious peaks; its value is obtained through a validation process as explained in Section 7. To demonstrate the performance of our onset detection method, let us show an example from real piano polyphonic music of Mozart's KV 333 Sonata in B-flat Major, Movement 3, sampled at 8 KHz and quantized with 16 bits. We will consider the second and third bar at 120 metronome beat. It is shown in Figure 1. We use a STFT with N = 512, an N-point Hanning window and hop size h = 256 corresponding to 32 milliseconds hop between successive frames. Figure 2 shows the onset detection function.

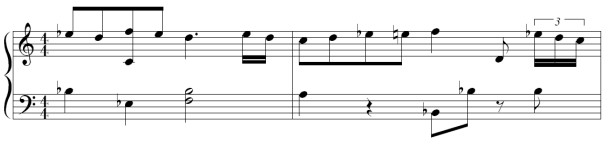


Figure 1. Musical score of Mozart's KV 333 Sonata in B-flat Major.

3. SPECTRAL FEATURES BASED ON COSTANT-Q TRASFORM

A frequency analysis is performed on notes played by piano, in order to detect the signal harmonics. Using the Fast Fourier Transform (FFT) the frequency resolution could be not sufficient. In fact, a FFT with 512 temporal samples x[n] on a sound recorded with the usual sampling rate (SR) of 44100 Hz, carries out a resolution of about 86.1 Hz between two FFT samples. This is not sufficient for low frequency notes, where the distance between two adjacent semitones is about 8 Hz (C3, 131 Hz and C#3, 139 Hz).

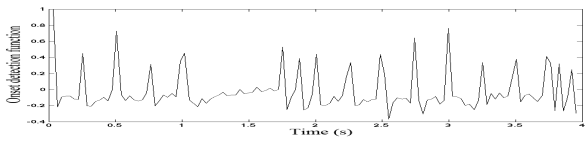


Figure 2. Onset detection function for the example in Figure 1.

The frequency resolution will get better if a higher number of temporal samples are used (with 8192 samples the resolution is of about 5.4 Hz), but that requires larger temporal windows for a fixed SR. In this case, the analysis of the instantaneous spectral information of the musical signal makes worse. To solve this problem, a Constant-Q Transform (CQT) [9] is used to detect the fundamental frequency of the note. Then, the upper harmonics may be individuated easily, as they are located at frequencies nearly multiples of the fundamental frequency. The Constant-Q Transform (CQT) has a logarithmic frequency scale, since a variable width window is used. It suits better for musical notes that are based on a logarithmic scale. The logarithmic frequency scale provides a constant frequency-to-resolution ratio for every bin

$$Q = \frac{f_k}{f_{k+1} - f_k} = \frac{1}{2^{1/b} - 1} \tag{4}$$

whereb is the number of bins per octave and k the frequency bin. If b=12, and by choosing a particular, then k is equal to the MIDI note number (as in the equal-tempered 12-tone-per-octave scale). There is an efficient version of the CQT that's based on the FFT and on some tricks, as shown in [10]. In our work, the processing phase starts in correspondence to a note onset. Notice that two or more notes belong to the same onset if these notes are played within 32ms. Firstly, the attack time of the note is discarded (in case of the piano, the longest attack time is equal to about 32ms). Then, after a Hanning windowing, as single CQT of the following 64ms of the audio note event is calculated. Figure 3 shows the complete process. All the audio files that we used have a sampling rate of 8 kHz. The spectral resolution is b=372, that means 31 CQT-bins per note, starting from note C0 (~ 32 Hz) up to note B6 (~ 3951 Hz). We obtain a spectral vector A composed by 2604 = 31 (CQT-bins) × 84 (musical notes). To reduce the size of the spectral vector, we operate a simple amplitude spectrum summation among the CQT-bin relative to the fundamental frequency of the considered musical note, the previous 15 CQT-bins and the subsequent 15 CQT-bins; then, we obtain a spectral vector B composed by 84 = 1 (CQT-bins) × 84 (musical notes). The scale of the values of the frequency bins is also logarithmic re-scaled into a range from 0 to 1. Figure 4 shows a feature vector obtained with this method.

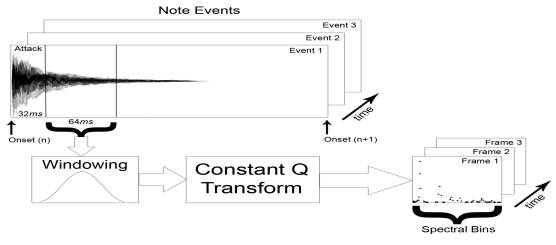


Figure 3. Spectral features extraction.

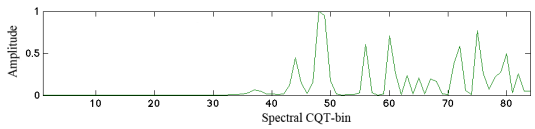


Figure 4. Feature spectral vector of MIDI note 38.

4. SPARSE NONNEGATIVE MATRIX FACTORIZATION (SNMF)

The problem addressed by NMF [11] is as follows: given a nonnegative n×m data matrix D, find nonnegative matrices W and H in order to approximate the original matrix

$$D \approx W \cdot H \tag{5}$$

The n×r matrix W contains the basis vectors and the r×m matrix H contains the coding vectors needed to properly approximate the columns of D as linear combinations of the columns of W. Usually, r is chosen so that (n+m)r<nm, thus resulting in a compressed version of the original data matrix. The elements of W and H can be estimated by minimizing the Frobenius norm

$$D\left(V\|W,H\right) = \|V - WH\|_F^2 \tag{6}$$

subject to $W, H \geq 0$

It may be advantageous to specify an additional constraint that will modify the representation in some way. One popular requirement is that the algorithm learns an over-complete basis by specifying additional constrain that the rows of H have a sparse activation for the basis contained in W. This means that the probability of two or more activation patterns being active simultaneously is low. Thus, sparse representations lend themselves to good separability.

A simple way to introduce a sparseness constraint on H is to replace (6) with the following function [12]:

$$G\left(V\|W,H\right) = D\left(V\|W,H\right) + \lambda \sum_{ij} H_{ij} \tag{7}$$

where the second term enforces sparsity by minimizing the L1-norm of H. The parameter λ controls the tradeoff between sparseness and accurate reconstruction. Function (7) is minimized by using the following update rules [12]:

$$W = W - \mu\left(W^T \cdot H - V\right)H^T$$
$$H = H.*\left(W^T \cdot V\right)./\left(W^T \cdot W \cdot H + \lambda\right) \tag{8}$$

where .* and ./ are element-wise multiplication and division, respectively and μ > 0 is a small positive real number. W and H are initialized with random positive values, and alternatively updated by rules (8) until the cost function does not significantly change.

5. MULTI-CLASS SVM CLASSIFIERS

A SVM identifies the optimal separating hyperplane (OSH) that maximizes the margin of separation between linearly separable points of two classes. The data points which lie closest to the OSH are called support vectors. It can be shown that the solution with maximum margin corresponds to the best generalization ability [13]. Linearly non-separable data points in input space can be mapped into a higher dimensional (possibly infinite dimensional) feature space through a nonlinear mapping function, so that the images of data points become almost linearly separable. The discriminant function of a SVM has the following expression

$$f(\mathbf{x}) = \sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \tag{9}$$

where xi is a support vector, K(xi, x) is the kernel function representing the inner product between xi and x in feature space, coefficients αi and b are obtained by solving a quadratic optimization problem in dual form [13]. Usually a soft-margin formulation is adopted where a certain amount of noise is tolerated in the training data. To this end, a user-defined constant C > 0 is introduced which controls the trade-off between the maximization of the margin and the minimization of classification errors on the training set [13]. The SVMs were implemented using the software SVMlight developed by Joachims [14]. A linear and a radial basis function (RBF) kernel was used

$$K\left(\mathbf{x}_i, \mathbf{x}_j\right) = \exp\left(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right), \quad \gamma > 0 \tag{10}$$

where γ describes the width of the Gaussian function. The selection of model parameters, C and γ, was performed using a grid-search on a validation set.

For note classification, we used the one-versus-all (OVA) approach, based on N SVMs, N being the number of classes. The ith SVM is trained using all the samples in the ith class with a positive class label and all the remaining samples with a negative class label.

6. TRANSCRIPTION SYSTEMS FOR POLYPHONIC PIANO MUSIC

In this Section, we present three different transcription systems for polyphonic piano music.

6.1 Asynchronous System

The asynchronous system is the basic system we develop. Feature extraction is based on Constant Q Transform (CQT) (see Section 3), while note classification is based on Support Vector Machines (SVMs). Figure 5 shows the complete asynchronous system. The output of the processing phase, including all the note onsets, is a matrix of 84 columns, corresponding to the CQT-bins, and a number of rows that's equal to the total number of note events in the MIDI file.

For sake of comparison with [15], both for training and test we used the same audio dataset, constituted by a rich collection of piano pieces of different musical styles.

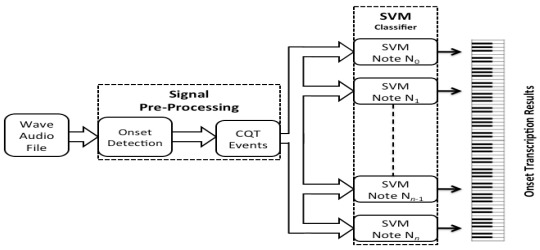


Figure 5. Asynchronous System

6.2 Memory based Asynchronous System

To improve classification results, firstly we consider what happens before the onset at time n, in particular, we introduce a short-term memory as follows: firstly, the segment of 32 ms preceding the onset time of the note is discarded, then, using Hanning windowing, the CQT on the previous 64 ms is computed. In a second experiment, we introduce one-event memory, and then we consider what happens at onset time n-1. Figures 6 and 7 show the spectral feature extraction, regarding short-term and one-event memory, respectively. The output of the processing phase, including all the note onsets, is a matrix of 168 = 84 × 2 columns, corresponding to the CQT-bins, and a number of rows that's equal to the total number of note events in the MIDI file. Consequently, to improve classification results, we also consider what happens before the onset at time n-1, in particular, we introduce two-events memory and three-events memory, and this means that we consider what happens at onset time n-2 and n-3, respectively.

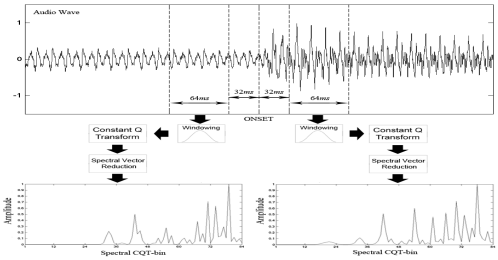


Figure 6. Short-time memory feature vector

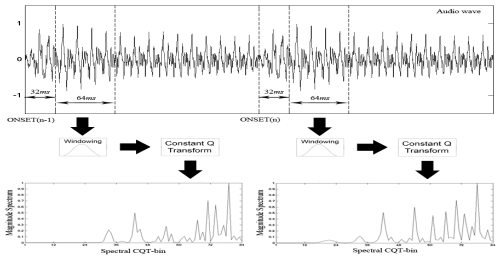


Figure 7. One-event memory feature vector

6.3 Sparse NMF based Asynchronous System

The method proposed is shown schematically in Figure 8. The lower side of the figure (8b) represents the polyphonic piano signal for testing. Its decomposition onto representative templates is provided a priori to the system as basis. These bases are learned off-line, as shown on the upper side of the figure (8a), and constitute the dictionary used during decomposition. The learning module aims at building a dictionary W of note templates onto which the polyphonic music signal is projected during the decomposition phase. The whole polyphonic sample of notes {N0,N1,…,Nn-1,Nn} that form the training dataset, from which the system learns characteristic basis template, is first processed in a short-time sound representation using the Constant Q Transform (CQT). The representations are stacked in a n matrices V{Ni}, for 0<i<n, where n is the note number and each column vj{Ni} is the sound representation of the jth time-frame. We then solve sparse NMF with V{Ni} and V{N0,…,Ni ,…,Nn}-{Ni}, this learning scheme gives two bases W{Ni} and W{N0,…,Ni ,…,Nn}-{Ni} that we stack them in columns to form the dictionary W, and two activations H{Ni} and H{N0,…,Ni ,…,Nn}-{Ni} for each note sample. The problem of the encoding phase is now to projecting the new music signal vjtest onto W. The problem is thus equivalent to a non-negative decomposition V ≈ WH where W is kept fixed and only H is learned. The learned vectors hj would then provide successive activations of the different notes in the music signal and can be classified by an SVM. Fig. 9 shows a schematic view of the transcription process.

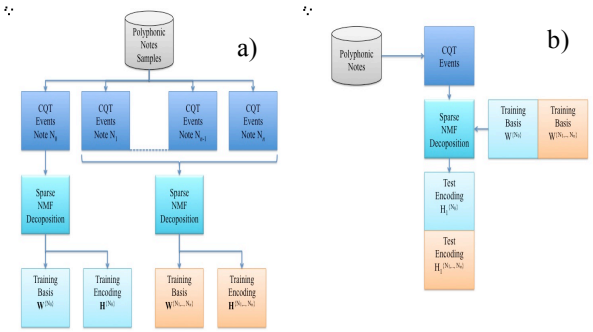


Figure 8. Learning polyphonic note N0 (a). Encoding polyphonic note N0 (b).

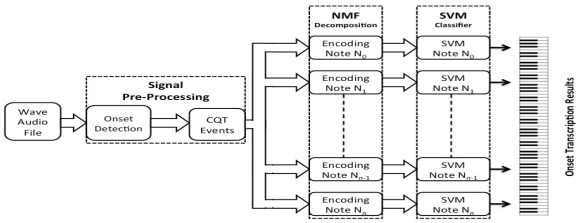


Figure 9. Sparse NMF based Asynchronous System

7. AUDIO DATA AND EXPERIMENTAL RESULTS

In this section, we report on the simulation results of our transcription system and compare them with some existing methods. The MIDI data used in the experiments were collected from the Classical Piano MIDI Page, http://www.piano-midi.de/. A list of pieces can be found in [15] (p. 8, Table 5). The 124 pieces dataset was randomly split into 87 training, 24 testing, and 13 vali-

dation pieces. The first minute from each song in the dataset was selected for experiments, which provided us with a total of 87 minutes of training audio, 24 minutes of testing audio, and 13 minutes of audio for parameter tuning (validation set). This amounted to 22680, 6142, and 3406 note onsets in the training, testing, and validation sets, respectively. As proposed in [15], we also investigate using live piano recordings to generate training, testing, and validation sets. Live piano recordings were made from a subset of the MIDI files using a Yamaha Disklavier playback grand piano. 20 training files and 10 testing files were randomly selected for recording. The MIDI file performances were recorded as monaural audio files at a sampling rate of 44.1 kHz. First, we performed a statistical evaluation of the performance of the onset detection method. The results are summarized by three statistics: the Precision, the Recall and the F-measure. Then we use a statistical evaluation of the performance of the musical notes classification by means of two metrics: the one proposed by Dixon [16], defined as Overall Accuracy = TP/(TP+FN+FP) where TP ("true positives") is the correct detections, FP is the number of false positives and is the number of FN false negatives, and the metric called *transcription error score* defined by NIST. In particular, NIST developed a metric that consists of a single error score which further breaks down into substitution errors (mislabelling an active voice), "miss" errors (when a voice is truly active but results in no transcript), and "false alarm" errors (when an active voice is reported without any underlying source). When running the onset detection algorithm we experimented with the threshold value for peak picking. We consider as correct the onset detected within 32 milliseconds of the ground-truth onset. The results reported here were obtained using the threshold value 0.01; it was chosen to maximize the F-measure value regarding the 13 pieces of validation dataset. Table I quantifies the performance of the method on the test set (including 6142 onsets). After detection of the note onsets, we have trained the SVMs on the 87+20 pieces of the training set and we have tested the systems on the 24+10 pieces of the test. The results are outlined in the following Tables.

Onset detection accuracy

Precision	*Recall*	*F-Measure*
96.9	95.7	96.3

Note onset transcription results on full synthesized audio test set [%]

	Acc	E_{tot}	E_{subs}	E_{miss}	E_{fa}
Asynchronous System	72.3	20.1	10.5	9.4	0.02

Note onset transcription results on full synthesized plus recorded test set [%]

	Acc	E_{tot}	E_{subs}	E_{miss}	E_{fa}
Asynchronous System	**68.0**	24.6	11.3	13.2	0.04
Poliner and Ellis	62.3	43.2	4.5	16.4	22.4
Ryynanen and Klapuri	56.8	46.0	6.2	25.3	14.4
Marolt	30.4	87.5	13.9	41.9	31.7

Dixon accuracy result of the memory based system [%]

SYSTEM	*Amplitude scale*	*SVM Kernel*	
		Linear	*RBF*
Short-term Memory Based Asynchronous System	*Linear*	70.4	71.9
	Logarithmic	74.7	**79.7**
One-event Memory Based Asynchronous System	*Logarithmic*	72.9	**83.2**
Two-event Memory Based Asynchronous System	*Logarithmic*	74.0	**77.3**
Three-event Memory Based Asynchronous System	*Logarithmic*	73.9	**78.2**

Dixon accuracy result of the Sparse NMF based system [%]

SYSTEM	*Amplitude scale*	*SVM Kernel*	
		Linear	*RBF*
Sparse NMF Based Asynchronous System	*Linear*	61.8	67.3
	Logarithmic	70.9	**75.1**

8. CONCLUSION

Automatic music transcription is a challenging topic in audio signal processing. It consists in transforming the musical content of audio data into a symbolic notation, commonly called musical score. In this paper a novel piano music transcription system based on the characterization of note events has been developed. The proposed approach exploits dynamic and sparse musical signal representation. The system takes as input the sound of a recorded polyphonic piano music and it produces conventional musical representation as output. As a result of the underlying procedures, for each event the proposed transcription system measures both the note onset and the note offset, the last with lower precision, because it has a lesser perceptual importance in comparison to the attack instant, and classify the note pitch. These informations can be directly used to reconstruct the MIDI piano roll and the symbolic representation of musical notes. Differently from previous approaches, an event-based analysis of errors has been pursued, since frame-level performance scores do not take into account the perceptual effects due to temporal position of missing or uncorrect detections. The proposed onset detection algorithm is based on Short-time Fourier Transform (STFT) combined with a suitable binary processing in order to improve the precision of onset measurement.
It has been shown that the onset detection algorithm is helpful in the determination of note attacks with modest computational cost and good accuracy. Regarding musical note transcription, the proposed method focuses on note events and their main characteristics: the attack instant, the pitch and the final instant. The main advantage is that the transcription results are more meaningful from a perceptual viewpoint with respect to frame-level statistics. Its main phases involve: (a) an onset detection algorithm providing sufficient precision on note attack instant measurement; (b) features extrac-

tion for each music event corresponding to a note onset; (c) recognition of note pitch. Features extraction is obtained by using the Constant Q Transform (CQT) and Sparse Non-negative Matrix Factorization (NMF), while note classification and offset detection are carried out using a bank of Support Vector Machine (SVM) with Linear kernel and Radial Basis Function (RBF) kernel. Musicological models appears to be a very powerful approach to automatic music transcription, therefore it has been taken into account the following assumption: melodic and harmonic musical structures depend on the method adopted by the composer; this means that every musical note is highly correlated to the previous note in the composition. Consequently, it has been found that the analysis of temporal musical structure with the introduction of a short-term memory or an event memory plays a pivotal role in the performance of the transcription system. It has been also found that a feature extraction based on Sparse NMF for template learning method is helpful to achieve better accuracy. The Sparse NMF learning method aims at building a dictionary of note templates onto which the polyphonic music signal is projected during the classification phase. These techniques already indicate that sparse coding is a powerful approach to automatic music. Finally, the performance of the proposed system has been compared to other algorithms for piano music transcription, and it has been tested over a set of classical piano music, concerning a publicly available dataset of synthesized MIDI files and piano recordings. The obtained results show that the proposed system outperforms state-of-the-art approaches, and are encouraging for further development.

9. ACKNOWLEDGEMENTS

The Master in SONIC ARTS of University of Rome Tor Vergata supports this research.

http://www.mastersonicarts.uniroma2.it

10. REFERENCES

[1] J. C. Brown, “Musical fundamental frequency tracking using a pattern recognition method”, *Journal of the Acoustical Society of America*, vol. 92, no. 3, 1992.

[2] J. C. Brown and B. Zhang, “Musical frequency tracking using the methods of conventional and narrowed autocorrelation”, *Journal of the Acoustical Society of America*, vol. 89, no. 5, 1991.

[3] Moorer, “On the Transcription of Musical Sound by Computer”. *Computer Music Journal*, Vol. 1, No. 4, Nov. 1977.

[4] M. Piszczalski and B. Galler, “Automatic Music Transcription”, *Computer Music Journal*, Vol. 1, No. 4, Nov. 1977.

[5] M. Ryynanen and A. Klapuri, “Polyphonic music transcription using note event modeling,” in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics* (WASPAA ’05), New Paltz, NY, USA, October 2005.

[6] M. Marolt, “A connectionist approach to automatic transcription of polyphonic piano music,” *IEEE Transactions on Multimedia*, vol. 6, no. 3, 2004.

[7] W.C. Lee, C.C. J. Kuo, “Musical onset detection based on adaptive linear prediction”, *IEEE International Conference on Multimedia and Expo*, ICME 2006, Toronto, Canada, pp. 957-960, 2006.

[8] G.P. Nava, H. Tanaka, I. Ide, “A convolutional-kernel based approach for note onset detection in piano-solo audio signals”, *Int. Symp. Musical Acoust. ISMA 2004*, Nara, Japan, pp. 289-292, 2004.

[9] J. C. Brown, “Calculation of a constant Q spectral transform”, *Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[10] J. C. Brown and M. S. Puckette, “An efficient algorithm for the calculation of a constant Q transform,” *Journal of the Acoustical Society of America*, vol. 92, no. 5, pp. 2698–2701, 1992.

[11] Cichocki, A., Zdunek, R., Phan, A. H., Amari, S., *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*, Wiley, 2009.

[12] Hoyer, P. O., “Non-negative Matrix Factorization with Sparseness Constraints”, *Journal of Machine Learning Research*, no. 5, pp. 1457–1469, 2004.

[13] J. Shawe-Taylor, N. Cristianini, *An Introduction to Support Vector Machines*, Cambridge University Press (2000).

[14] T. Joachims, *Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning*, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.

[15] G. Poliner and D. Ellis, “A Discriminative Model for Polyphonic Piano Transcription”, *EURASIP Journal of Advances in Signal Processing*, vol. 2007, Article ID 48317, pp. 1-9, 2007.

[16] S. Dixon, “On the computer recognition of solo piano music,” in *Proceedings of Australasian Computer Music Conference*, pp. 31–37, Brisbane, Australia, July 2000.

VALUTAZIONE PARAMETRICA DI UN MODELLO STRUTTURALE DI ORECCHIO ESTERNO PER IL RENDERING BINAURALE DEL SUONO

Sandro Scaiella, Simone Spagnol, Michele Geronazzo, Federico Avanzini
Università di Padova, Dipartimento di Ingegneria dell'Informazione
avanzini@dei.unipd.it

SOMMARIO

Gli effetti acustici cui è sottoposto il suono nel tragitto dalla sorgente alle orecchie dell'ascoltatore sono riassunti in una specifica funzione di trasferimento (Head-Related Transfer Function, HRTF), di cui la PRTF (Pinna-Related Transfer Function) rappresenta il solo contributo dato dal padiglione auricolare. La PRTF di un singolo ascoltatore può essere approssimata con un apposito modello sintetico che ne approssima i principali picchi e notch, i parametri dei quali sono solo in parte relazionati all'antropometria dell'ascoltatore stesso. Partendo da tale modello, l'articolo descrive un esperimento psicoacustico approntato con l'obiettivo di testare l'accuratezza (in termini di localizzazione verticale) del modello per sorgenti sonore sul piano mediano e per diverse combinazioni di parametri non relazionati all'antropometria. Si analizzano quindi i risultati relativi ai soggetti sperimentali e si ottiene la combinazione di parametri che meglio si adatta a una buona resa dell'elevazione della sorgente, ottenendo così utili indicazioni per la progettazione di modelli strutturali per la sintesi binaurale.

1. INTRODUZIONE

L'illusione che un dato suono sia posizionato in una determinata locazione virtuale rappresenta l'obiettivo principale di qualsiasi sistema di rendering di audio 3D. Tale obiettivo può essere raggiunto sia per mezzo di altoparlanti sia di cuffie. L'idea di base, sottesa al citato sistema, è quella di presentare ai due timpani dell'ascoltatore gli stessi segnali che produrrebbe una sorgente reale posizionata in quel particolare punto dello spazio di cui si vuol dare la percezione.

Il presente lavoro è dedicato allo studio della percezione dell'elevazione di sorgenti sonore localizzate sul piano mediano, ovvero equidistanti dalle due orecchie dell'ascoltatore. L'idea di fondo è quella di studiare e modellare un sistema che permetta, una volta ricavate alcune caratteristiche antropometriche relative all'ascoltatore, di ricostruire un paio di HRTF (*Head-Related Transfer Function*) individuali. Esse rappresentano dei filtri che modellano gli effetti sulla propagazione del suono dovuti alla testa, all'orecchio esterno, al busto e alle spalle. Questo lavoro rientra all'interno di tale ambito di studi, soffermandosi esclusivamente sul contributo dato dal padiglione auricolare (*Pinna-Related Transfer Function*, PRTF) i cui effetti, come vedremo nel corso della trattazione, permettono di discriminare l'elevazione di una sorgente sonora [1].

Al fine di valutare i parametri di un modello di PRTF sintetico non relazionati alle caratteristiche antropometriche dell'orecchio dell'ascoltatore, si è resa necessaria la progettazione e l'implementazione di un esperimento psicoacustico nel quale tali parametri potessero assumere diversi possibili valori. Tale sperimentazione permette un'analisi comparativa di suddetti parametri al fine di determinare la combinazione degli stessi in grado di rendere il più fedelmente possibile l'elevazione desiderata.

2. IL MODELLO STRUTTURALE

Secondo la cosiddetta *modellazione strutturale* [2], la HRTF misurata all'ingresso del canale uditivo può essere scomposta in contributi tra loro indipendenti, dovuti agli effetti di testa, padiglione auricolare (pinna) e busto/spalle sull'onda sonora. Possiamo quindi considerare la pinna come un oggetto indipendente dalle altre parti anatomiche e modellarne il contributo separatamente.

Tale risposta assume una particolare rilevanza in quanto tipicamente le differenze più marcate tra le HRTF di più soggetti sono principalmente imputabili alle diverse caratteristiche della pinna (forma, grandezza e orientazione), talmente varie da rendere la pinna oggetto di recenti studi di identificazione biometrica [3]. La pinna ha un ruolo fondamentale nel determinare la forma spettrale delle HRTF grazie a due fenomeni acustici principali: riflessioni (sui bordi principali) e risonanze (nelle cavità). Di conseguenza, la corrispondente PRTF presenta nella sua ampiezza una sequenza di amplificazioni (picchi) in corrispondenza delle frequenze di risonanza e di nette attenuazioni (*notch*) in corrispondenza delle frequenze in cui si ha massima interferenza distruttiva tra onde dirette e onde riflesse. È proprio la locazione spettrale di tali picchi e notch a costituire un indicatore fondamentale per la caratterizzazione della posizione spaziale della sorgente sonora, e in particolare della sua elevazione [4].

Nella letteratura dedicata troviamo diverse proposte modellistiche per rendere sinteticamente le PRTF [5, 6, 7]. Tuttavia, tali modelli soffrono di evidenti limiti: la presenza della sola componente riflettente [5], la validità in regioni spaziali eccessivamente ristrette [6], e/o l'assenza di

Copyright: ©2014 Sandro Scaiella, Simone Spagnol, Michele Geronazzo, Federico Avanzini. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

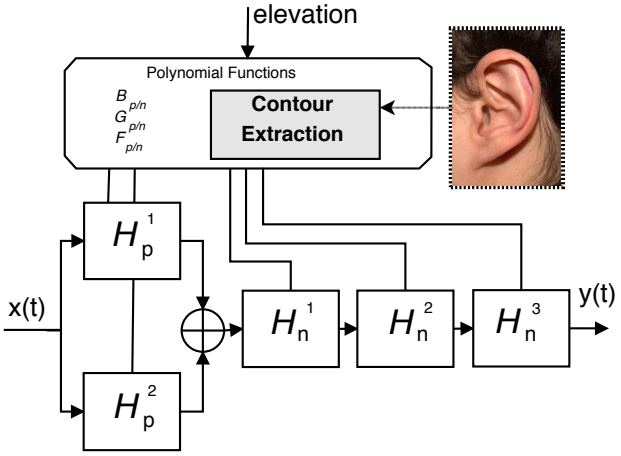


Figura 1. Il modello strutturale di pinna.

una parametrizzazione esplicita sull'antropometria dell'ascoltatore [7]. In un lavoro precedente [8] gli autori hanno proposto un modello strutturale di pinna costituito da due blocchi di filtri, il blocco *risonante* e il blocco *riflettente*. Il blocco risonante è costituito da due filtri *peak* del secondo ordine posti in parallelo; le uscite di tale blocco vengono sommate per essere inviate all'ingresso del blocco riflettente, costituito dalla cascata di 3 filtri *notch* del secondo ordine. Il modello è riportato schematicamente in Figura 1.

Nell'ambito dello stesso lavoro, gli autori hanno studiato la relazione tra le frequenze centrali dei notch presenti nelle PRTF e la geometria della pinna. A tal fine è stata utilizzata una procedura di *ray-tracing* su immagini 2D della pinna per mappare i punti di riflessione a una certa distanza dal punto di riferimento del canale uditivo, univocamente determinata dalla frequenza centrale di ogni notch. Mentre i modelli di riflessione precedenti tipicamente assumevano coefficienti di riflessione positivi, gli autori hanno verificato che l'utilizzo di coefficienti di riflessione negativi risulta un fattore chiave nella determinazione della frequenza centrale stessa. Assumendo dunque coefficienti negativi, la relazione tra frequenza del notch e distanza del punto di riflessione dal canale uditivo è completamente riassunta nella semplice equazione

$$d_i(\phi) = \frac{c}{2f_i(\phi)}, \qquad (1)$$

dove la costante c rappresenta la velocità del suono, ϕ è l'elevazione di cui si considera la PRTF, f_i è la frequenza centrale dell'i-esimo notch in tale PRTF e d_i è la distanza tra il corrispondente punto di riflessione e l'ingresso del canale uditivo. Dai punti così trovati e mappati su immagini di pinne di un *pool* di soggetti sperimentali, gli autori hanno notato l'ottima corrispondenza tra i punti di riflessione e i tre contorni principali della pinna, ovvero bordo dell'elice, antielice/parete interna della conca, e bordo esterno della conca.

Partendo da tali risultati, gli autori in [9] hanno realizzato la procedura inversa: a partire dall'immagine della pinna sono stati tracciati i tre sopracitati contorni e, attraverso semplici calcoli trigonometrici, trasformati in coppie di coordinate polari $(d_i(\phi), \phi)$ rispetto al canale uditivo. Dall'Equazione 1 si ricavano quindi le frequenze centrali dei notch ad ogni elevazione ϕ desiderata e per ognuno dei tre contorni. L'unico parametro indipendente usato dal modello è infatti l'elevazione della sorgente sonora virtuale, alla quale sono associate tre funzioni polinomiali che interpolano le frequenze centrali ricavate dai tre contorni per un certo passo di campionamento di ϕ (vedi Figura 1).

Per ciò che riguarda i parametri di larghezza di banda e profondità dei notch stessi, non è ancora stata trovata una relazione con la forma della pinna. Gli autori hanno precedentemente utilizzato per questi due parametri valori medi di una popolazione di soggetti sperimentali [9], così come per i parametri caratteristici delle risonanze. Si rende quindi necessario uno studio approfondito dell'impatto psicoacustico di tali parametri nella percezione dell'elevazione di una sorgente sonora.

3. CONDIZIONI SPERIMENTALI

Poon e Brugge [10] descrivono i risultati di esperimenti su gatti nei quali vengono registrate le risposte di singole fibre del nervo uditivo a rumori a banda larga. All'interno di tali rumori viene introdotto un notch sintetico di frequenza centrale variabile tra 2.5 e 40 kHz. La tesi degli autori è che l'informazione contenuta nella frequenza e nell'intensità di rilevanti notch nelle HRTF venga trasmessa al sistema nervoso centrale attraverso il tasso di scarica delle fibre uditive. Dai risultati si ricava che le fibre del nervo uditivo sono molto sensibili alla presenza di notch con frequenza centrale prossima alla frequenza caratteristica delle fibre. In particolare, si nota un'ampia riduzione nel tasso di scarica delle fibre attorno alla frequenza caratteristica delle stesse quando la frequenza del notch è vicina, tasso che scende occasionalmente anche al di sotto dell'emissione spontanea che si avrebbe in assenza di stimolo acustico. Gli autori hanno altresì stimato il livello dello stimolo, sopra la soglia della fibra, al quale il notch produce la maggiore riduzione del tasso di scarica, ricavandone un valore compreso tra 15 e 35 dB. Infine, i due autori hanno trovato un'ottima corrispondenza tra un notch di 35 dB di profondità e la curva di sintonizzazione della fibra uditiva di frequenza caratteristica più vicina.

In [11], Alves-Pinto e Lopez-Poveda mostrano come l'abilità di rilevare notch ad alta frequenza dipenda dalla larghezza di banda del notch così come dal livello dello stimolo. In generale, i risultati mostrano come la soglia di profondità del notch affinché risulti rappresentabile nella risposta del nervo uditivo (in termini di variazione nel tasso di scarica) incrementi al diminuire della larghezza di banda del notch stesso. Da questo lavoro si evince infatti che mentre per notch con larghezza di banda tra 2 e 4 kHz la soglia resta al di sotto dei 10 dB, il rilevamento di notch con larghezza di banda minore (1 kHz) risulta problematico anche per notch molto profondi.

Risultati simili erano precedentemente stati trovati da Moore [12]. Nel citato studio l'autore analizza l'abilità di un gruppo di soggetti di rilevare e discriminare picchi e notch con frequenze centrali di 1 e 8 kHz inseriti in stimoli di rumore a banda larga. Le larghezze di banda prese in esame per i notch sono di $0.125f_c$, $0.25f_c$ e $0.5f_c$, tutte

relative alla frequenza centrale f_c del notch. Considerando la soglia per la frequenza di 8 kHz - tipica di uno dei maggiori indicatori per il rilevamento dell'elevazione nel piano mediano nelle HRTF, ovverosia un notch di larghezza di banda di $0.7f_c$ [4] - osserviamo che i notch con larghezza di banda pari a $0.125f_c$ risultano di difficile discriminazione, mentre tutti i soggetti esaminati riescono a rilevare notch con larghezze di banda di $0.25f_c$ e $0.5f_c$. Le soglie di profondità variano tra 10 e 20 dB per larghezze di banda di $0.25f_c$ e diminuiscono sensibilmente per larghezze di banda di $0.5f_c$. In base a tali risultati, il sopracitato indicatore di elevazione dovrebbe essere sempre rilevato.

In [13], Iida *et al*. hanno effettuato test di localizzazione nel piano mediano utilizzando un modello parametrico di HRTF. Tale modello è costituito solo da alcuni dei picchi e notch estratti dalle HRTF originali. Gli autori hanno trovato che per la HRTF parametrica (*p-HRTF*) composta da tutti i picchi e notch rilevanti, l'elevazione percepita è accurata tanto quanto quella percepita utilizzando le HRTF misurate sul singolo individuo. Al contrario, si hanno scarsi risultati considerando *p-HRTF* costituite da un solo notch o da un solo picco. Per le *p-HRTF* costituite solo dai primi due notch rilevanti (*N1* e *N2*), estratti dalle HRTF a partire dalla frequenza di 4 kHz, i risultati mostrano un'accuratezza nella percezione dell'elevazione paragonabile a quella che si avrebbe con le HRTF originali. Questo implica che *N1* e *N2* assumono un ruolo importantissimo nella percezione dell'elevazione di un suono nel piano mediano. Dai risultati è evidente anche il ruolo del secondo picco nella percezione di sorgenti sopra la testa. Per quel che riguarda il primo picco ($P1$), pur non essendo legato alla percezione dell'elevazione, può essere visto come un importante riferimento per il sistema uditivo umano nell'analisi dei primi due notch.

Dalle osservazioni tratte dai lavori sopracitati abbiamo ricavato utili indicazioni per la scelta dei nostri parametri sperimentali. Consideriamo i 3 parametri associati ad un singolo notch:

- f_c, frequenza centrale del notch;
- G, guadagno (profondità) del notch;
- BW: larghezza di banda del notch calcolata a -3 dB.

Come detto in precedenza, l'unico parametro associato alle misurazioni antropometriche del soggetto è f_c. Per la larghezza di banda consideriamo invece tre possibili valori: $BW = 2$ kHz, $BW = 0.15f_c$, $BW = 0.25f_c$. La scelta di larghezze di banda di $0.25f_c$ e fissa a 2 kHz è dovuta alla constatazione che tali valori corrispondono alle minime larghezze di banda, relative e fisse, alle quali la maggior parte dei soggetti riesce a rilevare un notch alle frequenze di nostro interesse, come mostrato rispettivamente in [12] e in [11]. Anche se la larghezza di $0.15f_c$ risulta di difficile percezione in base alla letteratura [12], la decisione di testare tale larghezza di banda nasce dalla considerazione che a tale valore si ha meno sovrapposizione tra notch sintetici affiancati, ottenendo quindi una PRTF più verosimile.

Per quanto concerne la profondità del notch, abbiamo deciso di utilizzare 2 diversi valori:

Condizione	Guadagno	Banda	Nr. notch
C_1	-10 dB	$0.15f_c$	3
C_2	-10 dB	$0.15f_c$	2
C_3	-10 dB	$0.25f_c$	3
C_4	-10 dB	$0.25f_c$	2
C_5	-10 dB	2 kHz	3
C_6	-10 dB	2 kHz	2
C_7	-30 dB	$0.15f_c$	3
C_8	-30 dB	$0.15f_c$	2
C_9	-30 dB	$0.25f_c$	3
C_{10}	-30 dB	$0.25f_c$	2
C_{11}	-30 dB	2 kHz	3
C_{12}	-30 dB	2 kHz	2

Tabella 1. Condizioni sperimentali.

1. $G = -10$ dB: la minima soglia alla quale, con larghezza di banda di $0.25f_c$, gran parte dei soggetti riesce a rilevare il notch, come mostrato in [12], e, allo stesso tempo, la soglia sotto la quale, con larghezze di banda fisse a 2 e 4 kHz, si distribuiscono le soglie psicoacustiche di profondità dei notch, come mostrato in [11];

2. $G = -30$ dB: deriva dalla corrispondenza tra un notch con profondità di 35 dB, di poco superiore al valore da noi scelto, e le curve di sintonizzazione delle fibre uditive, come mostrato in [10].

Infine, l'ultimo parametro variabile delle nostre PRTF sintetiche è il numero di notch presenti. Abbiamo testato PRTF costituite da 2 picchi e 3 notch e PRTF composte da 2 picchi e 2 notch, eliminando quello centrale. Tale decisione è in accordo con i risultati ottenuti in [13], dove, come esposto poc'anzi, la presenza di due soli notch è risultata sufficiente per una localizzazione accurata nel piano mediano. In Tabella 1 sono riepilogate le 12 condizioni sperimentali testate, date da tutte le combinazioni di larghezza di banda, profondità e presenza/assenza del secondo notch.

4. SETUP SPERIMENTALE

Al fine di valutare le prestazioni soggettive di ognuna delle 12 condizioni sperimentali applicata al modello strutturale di PRTF, abbiamo effettuato un esperimento di localizzazione su 12 soggetti (9M/3F, età 23-41). Soltanto $1/3$ dei soggetti aveva precedentemente avuto esperienze con test di localizzazione, e nessuno è stato sottoposto a sessioni di training. Una procedura automatizzata per la rilevazione della soglia uditiva [14] ha certificato che nessun soggetto presentasse perdite rilevanti fino agli 8 kHz binauralmente.

4.1 Stimoli

Lo stimolo è costituito da un treno di 3 impulsi di rumore bianco uniformemente distribuito, ognuno della durata di 300 ms. In assenza di filtraggio con PRTF, esso è presentato a un livello di 60 dBA (misurati sulle cuffie, al livello della pinna). Gli impulsi sono separati da pause di 250 ms e sagomati con finestre di Hann in modo da ottenere

fade-in e fade-out. Lo stimolo originale viene fatto passare attraverso un filtro per la compensazione delle cuffie utilizzate per l'esperimento [15] e quindi convoluto con una delle PRTF sintetiche modellata a seconda della posizione spaziale della sorgente virtuale e della condizione sperimentale. La componente risonante del modello strutturale è fissata a priori per tutti i soggetti e riceve in ingresso parametri mediati su una popolazione di HRTF, i quali variano con l'elevazione [9].

È importante a questo punto chiarire due aspetti fondamentali. Innanzitutto, ciò che viene presentato all'ascoltatore è uno stimolo filtrato con una PRTF e non con una HRTF completa. Tuttavia, l'unica differenza sostanziale tra PRTF e HRTF nel piano mediano è rappresentata dalla presenza del contributo di busto e spalle nella HRTF. Dal momento che tale contributo non rappresenta che un debole indicatore di elevazione alle basse frequenze, possiamo assumere $PRTF(\phi) \approx HRTF(\phi)$ nel piano mediano. In secondo luogo, mentre i parametri delle risonanze e le frequenze centrali dei notch variano con l'elevazione, i parametri banda e profondità dei notch restano fissi lungo tutto il range a parità di condizione sperimentale. In realtà, secondo uno studio precedentemente svolto su un database di HRTF, all'aumentare dell'angolo di elevazione la profondità dei notch tende mediamente a decrescere e la banda ad aumentare [9]. Ciononostante, tale scelta sperimentale è stata dettata dalla difficoltà nel rilevare un andamento univoco dei due parametri all'interno dei set di HRTF analizzati.

4.2 Protocollo

Come primo step acquisiamo un'immagine di profilo del soggetto, in modo tale da avere una rappresentazione della sua pinna sinistra. In una seconda fase, l'immagine viene prima ruotata per allineare la punta del naso del soggetto con il trago dell'orecchio; quindi, il punto di massima protuberanza del trago viene scelto come punto d'ingresso del canale uditivo. I tre principali contorni dell'orecchio vengono tracciati manualmente e usati per ricavare le frequenze centrali di ogni notch come esposto in Sezione 2. Il soggetto entra quindi in una cabina silente Sound Station Pro 45 e indossa un paio di cuffie Sennheiser HDA 200 cablate a una scheda Roland Edirol AudioCapture UA-101 impostata a una frequenza di campionamento di 44.1 kHz.

Le elevazioni testate sono $\phi \in [-45°, -30°, -15°, 0°, 15°, 30°, 45°]$, tutte sul piano frontale e mediano, dove un'elevazione di $0°$ corrisponde esattamente alla direzione frontale. Elevazioni maggiori di $45°$ sono state scartate a causa della tipica mancanza di notch nelle corrispondenti HRTF [16], mentre per elevazioni minori di $-45°$ interverrebbero questioni di postura del soggetto a complicare la percezione delle elevazioni stesse.

Nel corso del test di localizzazione ogni stimolo viene presentato, in modalità diotica, 4 diverse volte per ogni elevazione e condizione. Tale scelta permette di evitare che i risultati finali siano influenzati da possibili errori di attenzione e di vagliare un possibile effetto di *learning*. Considerando dunque le 7 elevazioni, le 12 condizioni e le 4 ripetizioni, otteniamo un totale di $7 \times 12 \times 4 = 336$ stimoli, presentati al soggetto in 4 blocchi da 84 stimoli, ciascuno associato ad una singola ripetizione. L'ordine degli stimoli all'interno di ogni blocco è una permutazione pseudocasuale degli stessi.



Figura 2. La GUI utilizzata per l'esperimento di localizzazione.

Uno schermo posto di fronte al soggetto mostra l'interfaccia grafica riportata in Figura 2. Ad ogni presentazione dello stimolo sonoro il soggetto, utilizzando un comune mouse, indica sull'interfaccia un punto all'interno della corona verde (raffigurante il piano mediano) nella direzione percepita del suono. Pochi decimi di secondo dopo il *click*, un nuovo stimolo viene inviato in cuffia. Alla fine di ogni blocco di 84 stimoli compare una finestra di *countdown* e al soggetto viene concessa una pausa di 3 minuti prima di proseguire con i blocchi successivi. L'intero test di localizzazione dura, considerando le 3 pause inserite, circa 45 minuti in media.

5. RISULTATI

Per ognuno dei 12 soggetti e per ognuna delle 12 condizioni sperimentali abbiamo innanzitutto calcolato l'errore medio di localizzazione, inteso come differenza assoluta tra elevazione *target* ed elevazione percepita (calcolata a partire dal punto inserito nella corona circolare della GUI) a seguito di un'eventuale correzione di errori *front*/*back*,[1] e la percentuale stessa di errori *front*/*back* sul totale degli stimoli. Poiché non è stato trovato alcun trend significativo riguardo la percentuale di errori *front*/*back* al variare della condizione sperimentale, non riportiamo i dati associati a questa misura. I valori di errore di localizzazione sono invece riportati in Tabella 2.

Appare innanzitutto netta la differenza in termini di errore di localizzazione assoluto tra i 4 soggetti con esperienza pregressa in test psicoacustici (S_3, S_6, S_8, S_{12}) e gli altri. Al fine di non inficiare i risultati dei soggetti migliori nel prosieguo della nostra analisi, abbiamo convenuto di non considerare i soggetti la cui performance fosse paragonabile a una performance completamente casuale in oltre il 60% delle condizioni (8 su 12 o più). Abbiamo perciò calcolato l'errore di localizzazione associato a valori inseriti da un generatore di numeri pseudocasuali distribuiti

[1] La correzione consiste nel mappare, con simmetria assiale, tutti i valori posti nell'emisfero posteriore nell'emisfero frontale. Tale pratica è ormai comune negli esperimenti di localizzazione [17] ed è giustificata dall'elevato tasso di inversioni cui sono soggetti stimoli virtuali.

Sogg. \ Cond.	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}
S_1	31.4°	30.7°	32.1°	30.5°	33.6°	26.1°	37.2°	28.7°	37.3°	39.6°	40.8°	29.5°
S_2	32.9°	41.3°	40.9°	39.8°	46.2°	41.2°	38.0°	45.0°	50.0°	42.4°	48.3°	35.2°
S_3	20.0°	19.6°	12.3°	17.9°	16.6°	20.1°	13.3°	16.4°	14.4°	16.7°	11.5°	17.8°
S_4	40.6°	47.5°	44.5°	44.0°	39.1°	43.0°	47.5°	36.3°	47.3°	33.4°	52.6°	45.2°
S_5	24.1°	24.8°	24.3°	29.1°	22.7°	28.1°	24.7°	25.5°	25.6°	26.3°	23.1°	24.7°
S_6	26.9°	20.3°	22.8°	24.8°	27.0°	17.9°	20.5°	16.8°	26.6°	17.0°	19.4°	23.6°
S_7	35.2°	42.4°	47.6°	38.4°	47.2°	40.4°	36.3°	40.7°	43.4°	45.7°	45.5°	46.6°
S_8	19.9°	21.2°	20.4°	18.9°	17.5°	23.4°	15.9°	19.3°	19.3°	16.7°	16.6°	17.9°
S_9	30.4°	41.0°	36.3°	26.6°	28.6°	25.4°	37.8°	26.1°	35.1°	36.2°	32.4°	39.4°
S_{10}	39.6°	41.3°	46.6°	42.1°	32.1°	40.8°	45.5°	38.4°	36.3°	48.6°	45.8°	53.2°
S_{11}	35.3°	47.8°	44.0°	49.0°	42.4°	46.1°	39.9°	43.6°	26.6°	46.4°	28.6°	39.4°
S_{12}	33.6°	38.7°	31.4°	36.5°	29.0°	45.4°	35.2°	33.7°	34.3°	42.6°	28.7°	41.5°

Tabella 2. Errore di localizzazione medio diviso per soggetto e condizione sperimentale. In rosso sono evidenziate le condizioni le cui performance risultano paragonabili a una performance casuale. In verde sono evidenziati i soggetti considerati per la successiva analisi e le migliori condizioni sperimentali associate a ciascuno di essi.

Condizione	Errore	Inversioni *U/D*	*slope*+r^2
C_1	27.7°	17.0%	0.52
C_2	30.5°	16.5%	0.47
C_3	27.9°	17.0%	0.65
C_4	29.1°	16.1%	0.51
C_5	27.2°	16.1%	0.65
C_6	29.1°	16.1%	0.48
C_7	28.1°	19.6%	0.59
C_8	26.3°	12.5%	0.69
C_9	27.4°	12.5%	0.70
C_{10}	30.2°	17.9%	0.48
C_{11}	25.2°	12.9%	0.82
C_{12}	29.2°	16.5%	0.42

Tabella 3. Risultati mediati sugli 8 soggetti considerati buoni localizzatori. In rosso sono evidenziate le 3 condizioni migliori per risultato.

uniformemente nell'intervallo $[-90°, 90°]$. In questo caso l'errore medio di elevazione converge a un valore di 49° e il quinto percentile a un valore intorno ai 40°. Quest'ultimo valore è stato considerato come soglia dell'errore medio di localizzazione oltre la quale la performance risulta non attendibile, almeno per quella determinata condizione sperimentale. Di conseguenza, sono stati scartati dalle successive analisi i soggetti S_2, S_4, S_7 e S_{10}.

Nella successiva Tabella 3 sono quindi riportati i risultati medi per condizione sugli 8 soggetti restanti. In particolare, la tabella riporta

- il valore medio dell'errore di localizzazione, considerando anche i valori oltre soglia (errore $> 40°$);
- la percentuale di inversioni *up/down*, calcolata con una tolleranza di 15° attorno al piano orizzontale per tutte le elevazioni *target* tranne $\phi = 0°$;
- la somma dei parametri *slope* (pendenza) e r^2 (coefficiente di determinazione) della retta di regressione calcolata tra elevazioni *target* e percepite.

Confrontando i dati delle 3 colonne si evince che la condizione C_{11} risulta essere la migliore. A supporto di tale considerazione si noti in Tabella 2 come C_{11} dia, per 5 degli 8 soggetti selezionati, il migliore o il secondo migliore errore di localizzazione, e che in essa soltanto il Soggetto S_1 fallisce nel localizzare efficacemente la sorgente virtuale. Notiamo inoltre che le condizioni dispari (3 notch) danno risultati migliori rispetto alle condizioni pari (2 notch) associate, ad eccezione di C_7 e C_8; e che le condizioni globalmente migliori (C_8, C_9, C_{11}) hanno in comune una profondità di notch $G = -30$ dB.

A prima vista, la larghezza di banda dei notch sembra invece non avere un impatto significativo. Tuttavia, considerando i risultati dell'errore di localizzazione associati a ognuno degli 8 soggetti selezionati, notiamo che per per i soggetti che prediligono una condizione compresa tra C_1 e C_6 (S_1, S_5, S_9) l'elemento comune tra tali condizioni è la larghezza di banda, fissa a $BW = 2$ kHz. Al contrario, tutte e 3 le larghezze di banda appaiono tra le condizioni predilette comprese tra C_7 e C_{12} (soggetti $S_3, S_6, S_8, S_{11}, S_{12}$), tutte con $G = -30$ dB. Un'ispezione delle risposte in ampiezza dei filtri associati alle condizioni C_7, C_9, C_{11} (non riportate in questo articolo) rivela come queste siano tra loro molto simili a prescindere dalla larghezza di banda specificata.

Dal nostro esperimento risultano quindi avere una notevole importanza per una buona percezione dell'elevazione:

- una profondità di notch superiore a 10 dB, risultato in accordo con Moore [12] e Poon e Brugge [10];
- la presenza del secondo notch in una banda di frequenza tra 7 e 10 kHz, in accordo con Iida *et al*. [13];
- una banda sufficientemente larga affinché il notch venga rilevato, ad esempio fissa a 2 kHz, in accordo con Alves-Pinto e Lopez-Poveda [11].

6. CONCLUSIONI E SVILUPPI

In questo lavoro è stata analizzata, attraverso un esperimento psicoacustico all'uopo approntato, l'influenza della profondità, della larghezza di banda e del numero dei

notch presenti nelle PRTF nella percezione dell'elevazione di sorgenti sonore virtuali poste sul piano mediano. I risultati di tale esperimento hanno sostanzialmente confermato le indicazioni mutuate dalla letteratura precedente riguardanti tali parametri. Inoltre, questi ci hanno offerto incoraggianti riscontri sull'efficacia del modello strutturale nel rendere l'elevazione di una sorgente sonora virtuale, in particolare considerando i risultati dei soggetti cosiddetti "esperti". Una sperimentazione più approfondita sull'impatto psicoacustico del modello individualizzato sulla percezione dell'elevazione, paragonato a condizioni con HRTF individuali e non, rappresenta naturalmente un passo fondamentale verso la validazione dello stesso.

Un ulteriore aspetto da considerare per studi futuri riguardo i parametri caratteristici delle PRTF è la possibile relazione tra la componente risonante della PRTF e la forma delle cavità della pinna. Ricordiamo infatti che nel presente studio si è utilizzato il medesimo blocco risonante per tutti i soggetti. Infine, l'utilizzo di larghezze di banda e di profondità variabili per i notch potrà evidenziare o meno la rilevanza di tali variazioni nel determinare l'elevazione della sorgente sonora virtuale.

Nel corso del lavoro sono emerse alcune criticità dalla cui risoluzione tali successive sperimentazioni non potranno prescindere. In primo luogo, le differenze notate tra soggetti con e senza esperienza pregressa rende necessaria una sessione di *training* preliminare al test psicoacustico. In secondo luogo, le difficoltà riscontrate in alcuni soggetti nell'interpretare il metodo di inserimento dei dati da noi utilizzato pone la necessità di investigare metodi di puntamento alternativi, possibilmente di natura maggiormente ecologica [18].

7. RINGRAZIAMENTI

Questo lavoro è stato finanziato dal progetto di ricerca PADVA (Personal Auditory Displays for Virtual Acoustics), n. CPDA135702 dell'Università di Padova.

8. BIBLIOGRAFIA

[1] S. K. Roffler and R. A. Butler, "Factors that influence the localization of sound in the vertical plane," *J. Acoust. Soc. Am.*, vol. 43, pp. 1255–1259, June 1968.

[2] C. P. Brown and R. O. Duda, "A structural model for binaural sound synthesis," *IEEE Trans. Speech Audio Process.*, vol. 6, pp. 476–488, September 1998.

[3] A. Abaza, A. Ross, C. Hebert, M. A. F. Harrison, and M. S. Nixon, "A survey on ear biometrics," *ACM Trans. Embedded Computing Systems*, vol. 9, pp. 39:1–39:33, March 2010.

[4] J. Hebrank and D. Wright, "Spectral cues used in the localization of sound sources on the median plane," *J. Acoust. Soc. Am.*, vol. 56, pp. 1829–1834, December 1974.

[5] A. J. Watkins, "Psychoacoustical aspects of synthesized vertical locale cues," *J. Acoust. Soc. Am.*, vol. 63, pp. 1152–1165, April 1978.

[6] P. Satarzadeh, R. V. Algazi, and R. O. Duda, "Physical and filter pinna models based on anthropometry," in *Proc. 122nd Conv. Audio Eng. Soc.*, (Vienna, Austria), pp. 718–737, May 2007.

[7] K. J. Faller II, A. Barreto, and M. Adjouadi, "Augmented Hankel total least-squares decomposition of head-related transfer functions," *J. Audio Eng. Soc.*, vol. 58, pp. 3–21, January/February 2010.

[8] S. Spagnol, M. Geronazzo, and F. Avanzini, "On the relation between pinna reflection patterns and head-related transfer function features," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, pp. 508–519, March 2013.

[9] M. Geronazzo, S. Spagnol, and F. Avanzini, "A head-related transfer function model for real-time customized 3-D sound rendering," in *Proc. INTERPRET Work., SITIS 2011 Conf.*, (Dijon, France), pp. 174–179, November-December 2011.

[10] P. W. F. Poon and J. F. Brugge, "Sensitivity of auditory nerve fibers to spectral notches," *J. Neurophysiol.*, vol. 70, pp. 655–666, August 1993.

[11] A. Alves-Pinto and E. A. Lopez-Poveda, "Detection of high-frequency spectral notches as a function of level," *J. Acoust. Soc. Am.*, vol. 118, pp. 2458–2469, October 2005.

[12] B. C. J. Moore, S. R. Oldfield, and G. J. Dooley, "Detection and discrimination of spectral peaks and notches at 1 and 8 kHz," *J. Acoust. Soc. Am.*, vol. 85, pp. 820–836, February 1989.

[13] K. Iida, M. Itoh, A. Itagaki, and M. Morimoto, "Median plane localization using a parametric model of the head-related transfer function based on spectral cues," *Appl. Acoust.*, vol. 68, pp. 835–850, August 2007.

[14] D. M. Green, "A maximum-likelihood method for estimating thresholds in a yes-no task," *J. Acoust. Soc. Am.*, vol. 93, pp. 2096–2105, April 1993.

[15] A. Lindau and F. Brinkmann, "Perceptual evaluation of headphone compensation in binaural synthesis based on non-individual recordings," *J. Audio Eng. Soc.*, vol. 60, pp. 54–62, January 2012.

[16] S. Spagnol, M. Hiipakka, and V. Pulkki, "A single-azimuth pinna-related transfer function database," in *Proc. 14th Int. Conf. Digital Audio Effects (DAFx-11)*, (Paris, France), pp. 209–212, September 2011.

[17] F. L. Wightman and D. J. Kistler, "Headphone simulation of free-field listening.II: Psychophysical validation," *J. Acoust. Soc. Am.*, vol. 85, pp. 868–878, February 1989.

[18] L. Haber, R. N. Haber, S. Penningroth, K. Novak, and H. Radgowski, "Comparison of nine methods of indicating the direction to objects: Data from blind adults," *Perception*, vol. 22, no. 1, pp. 35–47, 1993.

"DONATONI: VERSO LA RIVALUTAZIONE DI QUARTETTO III"

Massimo Avantaggiato
Conservatorio Giuseppe Verdi di Milano
mavantag@yahoo.it

SOMMARIO

Donatoni scrisse ben 7 quartetti, l'ultimo dei quali è Luci III, del 1997: è, però, il terzo quartetto che emerge nella produzione del compositore per la peculiarità di essere concepito esclusivamente con mezzi elettronici.
Il terzo quartetto fa infatti parte di quell'insieme di brani quadrifonici realizzati presso lo Studio di Fonologia della Rai nel corso degli anni '60.

Persona timida e gentile, Donatoni dichiarerà che "non è difficile scorgere nel Quartetto III tutti i sintomi dell'inesperienza". Questa composizione, ad analisi più attenta, appare invece ben congegnata nel rapporto tra ministruttura e macroforma e nel progressivo processo di aggregazione, che, dalla "scala delle frequenze", conduce agli "elementi semplici" e "modulati", e quindi ai "gruppi", alle "colonne sonore di elementi formali". Nonostante l'opera sia stata, forse ingiustamente, svalutata da parte della critica, l'articolo cerca di rivalutarne l'importanza, non solo nell'ambito della produzione del compositore, ma anche nel ricco panorama di musiche elettroacustiche. Per raggiungere questo obiettivo sono utilizzate fonti archivistiche disponibili presso gli Archivi dello Studio di Fonologia; documenti, anche inediti, a cura del compositore, che ci sono stati gentilmente concessi da M. De Carli; inoltre, sono state impiegate varie tecniche di analisi proposte da compositori e musicologi per creare 8 partiture d'ascolto, poi opportunamente coordinate; si è fatto poi ricorso alla composizione assistita per rappresentare il piano pre-formativo dell'opera.
Se, da un lato, si procede ad una parziale analisi genetica, dall'altro, si definiscono gli oggetti sonori utilizzati, le tipologie di moto e si analizzano le
caratteristiche strutturali con riferimento ai gesti impiegati.

1. INTRODUZIONE

Nella produzione di Donatoni Quartetto III si presenta come un'opera di passaggio in quanto si colloca tra il periodo segnato, seppur con alcuni opportuni distinguo, dall'adesione allo Strutturalismo e le opere successive, come Quartetto IV del 1963 e Puppenspiel del 1966, che invece seguono la via segnata da Cage.
E' un'opera che, pur aderendo anch'essa allo Strutturalismo, anticipa alcune tendenze dell'arte di Donatoni, quale una certa "autonegazione": talvolta le attività del comporre sembrano ridotte a processi automatici di trasformazione del materiale che, parole di Restagno, *"nascevano da una maestria che il tempo e il lungo esercizio avevano trasformato in una manualità perfetta";* o meglio, come diremo più avanti, questi processi automatici nascevano dalla capacità del compositore di selezionare il materiale di partenza, posto a fondamento della composizione. Il Terzo Quartetto si colloca temporalmente dunque tra il Secondo Quartetto del 1958 (che, indicato talvolta come brano strutturalista, presenta una scrittura che alterna suoni tenuti e interventi puntillisti, con una elaborazione di frammenti musicali scelti facendo oscillare un pendolino)[1] e il Quarto Quartetto del 1963, che segna l'approfondimento delle tecniche dell'indeterminazione. Ma al di là delle tecniche impiegate e delle scelte estetiche effettuate, ciò che veramente colpisce è il vasto repertorio di brani scritti dal compositore per quartetto d'archi: l'influenza della personalità di Béla Bartók[2] e del suo Quarto Quartetto, sembra aver segnato anche quantitativamente la produzione del compositore veronese.
E' singolare, inoltre, che ci siano dei caratteri di continuità e molte similitudini tra Quartetto III e la restante produzione di Donatoni. Ne elenchiamo alcune:
- *l'ossessione numerologica:* si segnala la ricorrenza del numero 13, 11 o del numero 7 nei parametri musicali (quest'ultimo ricorre nel set di 7 frequenze che compongono gli "*elementi"*; nelle dinamiche; nella divisione in pannelli, nell'uso delle tecniche combinatorie, nell'impiego di metronomi etc.);
- *la passione per le simmetrie*: in Quartetto III una pausa centrale sancisce la divisione in due del brano. Que-

Copyright: © 2014 ***Massimo Avantaggiato.*** *This is an open-access article distributed under the terms of the* Creative Commons Attribution License 3.0 Unported*, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.*

[1] E. RESTAGNO, "Un'autobiografia di Donatoni raccontata da Enzo Restagno", E.d.T., Torino, 1990, p. 20 [16].

[2] L'influenza bartókiana è chiaramente leggibile nel primo Quartetto per archi (1950, premio Liegi), il Concerto per orchestra (1951), e il Concertino per archi, ottoni e timpano solista (1952, premio Concorso di Radio Lussemburgo). *"Senza dubbio, brani come il Quarto Quartetto per archi di Bartók, in cui si dischiude una libertà ritmica dai netti impulsi, ma totalmente svincolata dal metro della battuta, sono un presupposto ineludibile dell'esperienza di Donatoni".* G. MONTECCHI, Donatoni 1950-1972: da Bartók all'antimusica, in Donatoni:un'autobiografia dell'autore raccontata da Enzo Restagno, a c. di E. Restagno, E.d.T.,Torino 1990, pp. 77-109 [13].

sta caratteristica lo accomuna a molti altri brani come il quartetto The Heart's Eye del 1979, nel quale il materiale compositivo di partenza si trova nelle battuta centrale, la numero 131. Questa battuta crea due sezioni complementari e speculari[3];
- *la suddivisione formale in pannelli:* ricorrente anche in brani come "La souris sans sourire", dedicato alla formazione quartettistica dell'Ensemble InterContemporain. Anche in Quartetto III i pannelli sono accostati seguendo un principio di contrasto e manifestano coerenza interna di articolazione e sviluppo [3];
- *i metronomi impiegati:* simili a quelli adoperati in altri quartetti, come il già citato "La souris sans sourire", sono spesso numeri divisibili per 7 o per 11;
Una ulteriore curiosità si rileva, inoltre, col riferimento alla forma del sonogramma: notiamo una palese somiglianza con la forma a clessidra rovesciata indicata da Mattietti [11] per brani come "Gretchen am Spinnrade" nella Rapsodia per soprano, contralto e orchestra del 1994.

2. DONATONI: TRA STRUTTURALISMO E ATTENTATO REAZIONARIO

La Germania e l'Italia degli anni Cinquanta, anche attraverso una rinnovata politica culturale e la creazione e diffusione di nuova musica, cercavano di superare il senso di smarrimento creato dalle devastazioni della seconda guerra mondiale.
Dopo il 1945, la musica non poteva impiegare un linguaggio "massificato"[4]: la lacerazione e la distanza tra intellettuale e masse era definitivamente consumata.
Di fronte ai laceranti accadimenti della seconda guerra mondiale, lo Strutturalismo appariva a Donatoni come l'ultima modalità espressiva:
"(...) Lo strutturalismo mimava il gesto del linguaggio. Erano i gesti sonori di un ammutolito per assenza di sordità.. Coloro che avevano ascoltato il mondo erano stati ricompensati con l'ammutolimento, poiché solo ai sordi sarebbe stato concesso di continuare a parlare il linguaggio del mondo. Lo strutturalismo è l'ultima tensione linguistica, quella estrema, pare [9]*".*
Alla visione più serena e sicura di Stockhausen si contrappone quella più sofferta di *Donatoni: "Dopo i linguaggi della musica c'è un deserto che offre false piste e oasi inesistenti: unica legge è il moto, per chi sa di non arrivare [..]*. Nonostante questo desiderio di percorrere nuove strade, Donatoni riferirà di avere anche avuto, in quegli anni, ripensamenti e vaghi pensieri reazionari: *"Oscuramente avvertivo che la musica correva il massimo dei pericoli, che veniva messo in dubbio la sua sopravvivenza come arte: lo avvertivo oscuramente e incoscientemente. Riconosco nel pensiero turbato di quel tempo i sintomi che spingono all'attentato reazionario* [9]*".*

3. LA LIEVITAZIONE DELLA MATERIA

Oltre alle dittature politiche, vi era un'altra forma di dispotismo contro il quale alcuni compositori, tra cui lo Stockhausen di Studio I e II e il Donatoni di Quartetto III, si opporranno: la "dittatura" del materiale sonoro [1].
Donatoni parla in proposito di *"bisogno di costruire il proprio materiale acustico secondo gli artifici che le tecniche storiche consentivano"* perché ciò *"obbediva forse alla (mia) congenita passionalità per la riflessione sul suono, al mio istinto di mediare speculativamente ogni emozione primaria* [9]*..."*
Donatoni è consapevole che *"materia della musica può essere anche un suono pre-formato, come il canto di un uccello che "può prescindere dalla notazione e costituire materiale per elaborazioni elettroacustiche, quindi anche la materia scarta [...] diviene direttamente materiale, personalizzato mediante l'assunzione opzionale da parte del compositore* [6]"; ma il suo desiderio era quello di estendere i processi compositivi ai parametri del suono, conducendo una seria riflessione sulle sue proprietà [7]: Donatoni appare, pertanto, affascinato più dalle possibilità della *Elektronische Klangerzeungung* che dalle opportunità offerte dalla *Musique Concrète*.
Il rapporto tra materia e tecnica compositiva è meglio precisato dallo stesso compositore:
"*La tecnica compositiva reintegrata nella sua funzione meccanica come momento intermedio tra la materia sonora e manufatto musicale, implica (dunque) la presenza di materiali qualificati e contraddistinti dalla separazione avvenuta tra detti materiali e la virtualità ad essi inerente di cui sono, in un certo modo, il residuo primario. Ma proprio perché la qualificazione della materia viene accolta come il risultato di un processo discensionale che esercita la sua coercitività stessa, il problema della posizione di quest'ultima è del tutto secondario.*
Si può tuttavia precisare che l'indifferente sottomissione dell'esercizio redazionale sia alquanto più esatta e che il rovesciamento dei rapporti sia soltanto apparente. Infatti l'attività formante delle tecniche tende ad una lievitazione della materia sonora ottenuta mediante un'induzione lievitatrice proveniente dall'alto verso il basso [9]*".*
Ma quali sono i materiali "qualificati e contraddistinti" di cui parla Donatoni con riferimento a questa composizione? Un assunto, tipicamente strutturalista, diffuso all'epoca, consisteva nel creare forme di temperamento che abbandonassero la divisione dell'ottava in 12 intervalli uguali. Questo modo di procedere, che accomunava Donatoni e Stockhausen ad altri compositori, come Vlad e Togni, indicava un indirizzo, anche estetico, con-

[3] Altre strutture simmetriche si riscontrano: 1) in Duo pour Bruno, brano caratterizzato da una ripetizione ciclica di una "struttura che dal centro si dirama verso gli estremi" (Franco Donatoni, Un'autobiografia dell'autore raccontata da Enzo Restagno, cit., p. 36). 2) In Le ruisseau sur l'escalier (1980), ove la simmetria è basata su 130+1+130 battute. Ivanka STOÏANOVA, "Ruisseau e l'alchimia dei suoni. Frammenti di un discorso", in Donatoni, a c. di E. Restagno, E.d.T., Torino 1990, pp. 159-179: p.171 [16].

[4] "Ciò che è massificato [das massenhafte] non ha bisogno di tempo per crescere, solo di spazio […]. Là non c'è alcun limite, perché non c'è forma […] senza forma niente si differenzia, nessuna vita si impersona. La "Gleichschaltung" – il livellamento, la distruzione criminosa della diversità – era già là". M. PICARD, "Hitler in noi stessi", trad. it. Rizzoli , Milano, 1947, p. 40 [15].

diviso da molti compositori che si trovarono a lavorare negli anni '60 presso lo Studio di Fonologia.[5]

Franco Donatoni in particolare divide lo spazio sonoro compreso in un'ottava in 14 frequenze secondo un intervallo costante, avente rapporto di frequenza pari a 1,05[6].

Un altro aspetto importante di questo lavoro è che Donatoni basa la sua composizione non più su una serie, ma su una scala di 97 frequenze iniziali (suddivise in "elementi" di 7 frequenze): questo materiale iniziale, in quanto selezionato dal compositore, è suscettibile di ulteriori sviluppi e trasformazioni.

Dalla scala delle frequenze Donatoni ricava 4 gruppi, modificando la distanza intervallare intercorrente tra una frequenza e l'altra all'interno della scala, secondo una serie aritmetica (0,1,2,4).

Il primo gruppo sarà composto dalle frequenze in posizione multiplo di 1; il secondo gruppo dalle frequenze multiple di due; il terzo gruppo dalle frequenze multiple di 3; l'ultimo, dalle frequenze in una posizione multiple di 4.

In ciascuno di questi 4 gruppi ogni frequenza diventa la capostipite di altre 7 frequenze[7]: viene così pianificata la possibilità di rileggere, facendoli ruotare, gli "elementi" di 7 frequenze. E' importante rilevare che ciascuna formazione di sette suoni viene percepita come una struttura armonico timbrica che dà luogo ad una diversa "fusione di colore".

Il materiale di partenza aggregato in "gruppi" presenta infatti delle pregnanti proprietà musicali, che possiamo così riassumere:

- frequenza pivot fissa di 89 e ampliamento del registro (primo gruppo: 89-1.230 hertz; secondo gruppo: 89-4.390 hertz; terzo e quarto gruppo: 89-9.590 hertz);
- ampliamento dei rapporti intervallari tra le frequenze di ciascun gruppo (v. figura 1 e 2).

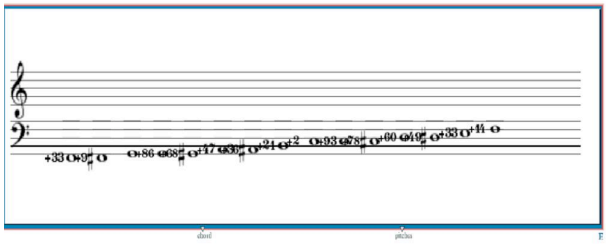


FIGURA 1: le prime 14 frequenze all'interno del primo gruppo.

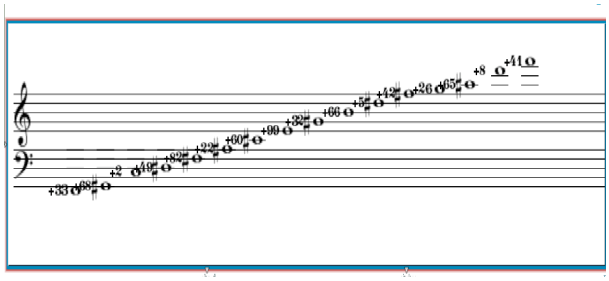


FIGURA 2: le prime 14 frequenze nel quarto gruppo: si noti, rispetto a figura 1, l'allargamento degli intervalli.

Il processo di aggregazione procede fino al montaggio delle 4 colonne, impiegando diversi processi compositivi:

1 - Moltiplicazione della materia iniziale tramite trasposizione e dislocazione nel tempo: la trasposizione intervallare di un evento a diverse altezze è una possibilità indicata talvolta dallo stesso materiale di partenza;

2 - Diminuzione dei valori delle note: la parte centrale del brano, sezione c) è costituita sovrapponendo gli *elementi formali* seguenti: i primi e gli ultimi delle colonne 1A; 2A;3A; 4A . Tali elementi vengono ravvicinati mediante una riduzione del 50% delle pause intermedie.

3 - Aumento graduale per accumulazione e condensazione: Il termine accumulazione, impiegato ampiamente per gli altri brani acustici, curiosamente, non viene utilizzato da Donatoni nella descrizione del suo brano elettronico.

Eppure, tali processi sono abbastanza evidenti e confermati dalle analisi condotte (cfr. Tabella 1):

Tecnica di analisi	
Vande Gorne (Figures d'Espace)	Accumulation; Envahissement
Smalley (Spettromorfologia)	Moto unidirezionale (fino 2'23"); multidirezionale nelle sezioni finali (da 2'23" a fine)
M.I.M. (T.S.U.)	Chaotic seguito da In suspension/ Suspending/ Questioning
Roy (Analisi Funzionale per processo)	Intensification/Accumulation seguita da Attenuation/Dispertion
Bayle	Le murmur/le cri

TABELLA 1: i processi di accumulazione indicati secondo le varie tecniche analitiche.

[5] Quartetto III presenta delle analogie con i brani degli autori citati:

1) "Recitativo per nastro magnetico" della durata di 4 minuti realizzato sempre nel 1961.

Togni ha individuato le caratteristiche delle 3 sezioni che compongono il suo brano:

- Sezione I: Entro lo spazio determinato dal rapporto armonico di 1: 4; un totale cromatico di 21 frequenze (all'intervallo minimo costante di 4/7 di tono); tre possibilità di trasporto, su registro diverso, del totale cromatico fondamentale;

- Sezione II: Spazio stabilito dal rapporto armonico 1:16; un totale cromatico di 35 frazioni di tono (l'intervallo minimo è quello di 24/35 di tono); una sola possibilità di trasporto;

- Sezione III: Campo determinato dal rapporto armonico di 1 e 64; 49 frequenze (secondo l'intervallo costante di 6/8 di tono); nessuna possibilità di trasporto su registro diverso.

2) "Ricercare elettronico" (1961) - 6'20" di Roman Vlad:

"Oltre ai suoni sinusoidali il materiale sonoro del "Ricercare elettronico" risulta dall'elaborazione in diverse guise di dodici mescolanze di sette toni sinusoidali ciascuno, di altrettante mescolanze di tredici suoni sinusoidali non temperati e dell'ordine dei quarti di tono, di dodici diverse strisce di rumore bianco filtrato, cioè di dodici diversi rumori colorati". Pamphlet "Per i 35 anni del Campus, 41° festival pontino di musica", 2005.

[6] Nel temperamento equabile il rapporto di frequenza è l'esatta radice dodicesima di 2, pari a 1.059463.

[7] Per es.: il primo gruppo viene formato impiegando 26 frequenze x 7 per un totale di 182 frequenze. Le 26 frequenze sono divise, a loro volta, in due sottogruppi, cui è assegnata una diversa intensità sonora in Db. Ogni frequenza diventa la capostipite di altre 7 frequenze.

Processi di accumulazione si riscontrano in 1'45"; quindi da 3'53" fino a 3'57" e, ancora, nella codetta finale da 4'50" a 4'53"; in alcuni casi, questi processi sono evidenziati mediante l'impiego dell'eco: 3'26"-3'31".
Nella versione quadrifonica, il processo di accumulazione e di rotazione delle altezze viene altresì enfatizzato attraverso la proiezione sui 4 canali.

4 - Diminuzione, o riduzione, graduale e progressiva dispersione e rarefazione: segue spesso ai processi di accumulazione.
Tale processo è evidente nella coda finale, dove il materiale è composto, dalle sovrapposizioni variate delle colonne formali B:

Colonna 1:	1a +3a +5a
Colonna 2:	2a +4 a+ 6a
Colonna 3:	7a+ 9a +11a
Colonna 4:	8a +10a+ 12a

TABELLA 2: Sintesi della struttura della coda [8].

5 - Sovrapposizione: nella colonna C vengono aggiunte delle "*tensioni"* in crescendo e diminuendo, per ottenere le quali Donatoni ha prodotto delle *"sovrapposizioni quadruplici a velocità variata* [8]*"*.

6 - Rovesciamento integrale di frammenti (grave-acuto e viceversa): Donatoni parla, a titolo di esempio, di "*elementi modulanti"* formati da frequenze assai gravi ottenute accoppiando, secondo un movimento divergente, i 7 elementi di ciascuna sezione del 4° gruppo e abbassando la risultante sonora di due ottave più una settima maggiore (es: 4.31 circa su colonna 1).

L'utilizzo di queste tecniche compositive e l'attenzione posta nel realizzare il progressivo processo di strutturazione, sono indice di una consapevolezza compositiva tutt'altro che "effimera"[8].

4. COORDINAMENTO DI VARIE TECNICHE DI ANALISI

Di Quartetto III esistono due versioni, peraltro coeve: una prima versione, stereofonica, registrata sulla bobina E018 ed una seconda versione, quadrifonica, incisa con altri brani su bobina Q002: entrambi i nastri sono stati registrati nell'aprile-maggio 1961 e sono custoditi presso gli archivi dello Studio di Fonologia.

Le differenze tra le due versioni sono evidenti all'ascolto: la versione quadrifonica permette di apprezzare i movimenti spaziali pensati dal compositore.

Disporre delle 4 colonne consente di individuare con minore difficoltà le frequenze impiegate; dall'altro, comprendere meglio il lavoro artigianale svolto nell'impiego dei nastri, nella gestione dei sincroni tra le diverse piste, etc.

[8] *"..altri musicisti si accostano in questi anni per la prima volta al mezzo elettronico, come Angelo Paccagnini cui tocca nel 1968 la direzione del Centro di Fonologia di Milano e , in modo più effimero, Franco Donatoni e Aldo Clementi (Collage 2, una manipolazione di canzoni dei Beatles del 1967)".* A. LANZA, "Storia della musica: il secondo Novecento", E.d.T. Torino, 1991, p.135 [10].

Ai numerosi tentativi di lettura e segmentazione della superficie acustica, è seguita l'organizzazione in immagini coerenti [12]: per raggiungere questo obiettivo, individuati gli oggetti sonori identifi- cati col vocabolario di Donatoni o con le tecniche di F. Bayle [2] si è proceduto all'identificazione di relazioni strutturali e di senso (*Functional Analisys* [17]; *Temporal Semiotic Units*).
Attraverso le metodologie analitiche sviluppate da Smalley [18] sono state inoltre individuate le tipologie di moto presenti: è stata rilevata la prevalenza del moto bi-direzionale/multidirezionale, diffratto e divergente, dopo 2'23''; si è verificato che il moto unidirezionale - ascendente o piano - prevale invece nella prima sezione (dall'inizio del brano fino alla pausa centrale che sancisce la divisione del brano in due parti di egual durata).
In base alle analisi effettuate, viene individuata una divisione in 7 pannelli o sezioni (Figura 6): questa divisione viene confermata dalla distribuzione e ricorrenza degli oggetti sonori nel brano, condotta secondo le tecniche analitiche di Bayle e dalla segmentazione realizzata mediante le tecniche delle T.S.U. (Figure 3 e 4).

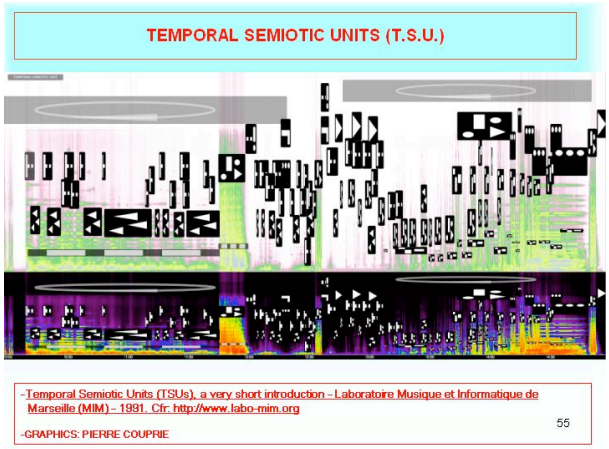


FIGURA 3: Temporal Semiotic Units. Grafiche © Pierre Couprie.

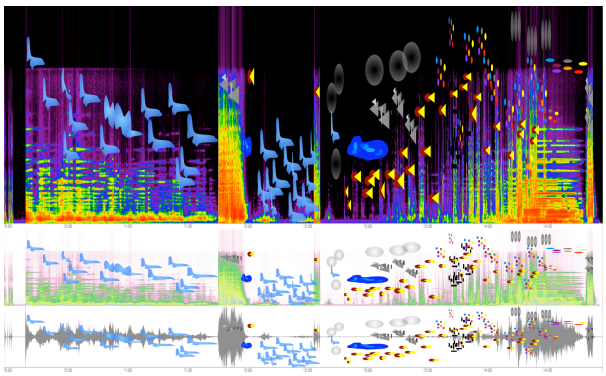


FIGURA 4: Individuazione degli oggetti sonori: Francois Bayle. Grafiche © Pierre Couprie.

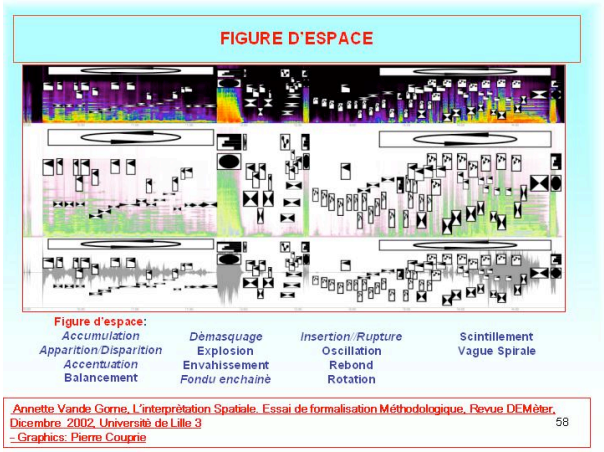


FIGURA 5: Individuazione delle figure spaziali (A. Vande Gorne). Grafiche © Pierre Couprie.

Alle analisi menzionate, si è aggiunta quella delle figure spaziali di Annette Vande Gorne [19], che ha permesso di comprendere quanto Donatoni fosse attento alla dimensione spaziale e all'organizzazione entro questa. In base a quest'analisi sono stati rintracciati alcuni tipi di figure indicati da A.Vande Gorne (Tavola III): Quartetto III sembra chiaramente preludere alla poetica di autori elettroacustici posteriori, che hanno valorizzato lo spazio e analizzato la possibilità di segmentarlo semanticamente sperimentando al suo interno figure spaziali e gesti variamente strutturati.
Tutte le analisi effettuate, opportunamente coordinate, evidenziano l'utilizzo di quelle tecniche di ripetizione o meglio di rilettura care a Bruno Maderna ed ereditate dal compositore veronese; "*Mi sono accorto",* diceva Donatoni, *"parlando con i miei allievi, analizzando con loro i miei lavori, che il mio modo di scrivere per riletture continue mi viene proprio da Bruno* [5]".

Accumulation:	Addizione successiva di piani; oppure messa in evidenza di un materia sonora corpuscolare o granulosa o ingrossamento progressivo di una fascia sonora
Apparition/Disparition:	Irruzione/chiusura non preparata verso uno stato spaziale differente, sovrapposto o successivo; sorpresa, magia, risveglio dell'ascolto.
Accentuation:	Messa in evidenza di una localizzazone/collocazione precisa presso gli altoparlanti.
Dèmasquage:	Cambiamento dei piani di profondità;
Envahissement:	Tragitto accumulativo orientato verso il pubblico con effetto generale di aggressione soggettiva;
Fondu enchainè:	Traiettorie tra sfondi collegati successivi, specie se si evoca un oggetto mobile;
Insertion/Rupture:	In uno spazio preciso si registra un passaggio brutale in sovrapposizione;
Rebond:	Salto rapido da un punto dello spazio ad un altro
Rotation:	Traiettoria circolare

TABELLA III: figure spaziali (Annette Vande Gorne).

Proprio la ripetizione e le variazioni negli oggetti sonori impiegati, nei gesti, nelle figure spaziali rinvenibili, consentono di individuare 7 sezioni e non tre[9], come indicato dal compositore. Tale risultato, apparentemente incoerente, è in realtà giustificato dall'ampio utilizzo delle tecniche combinatorie all'interno delle 3 (macro)sezioni indicate dal compositore veronese (figura 7).

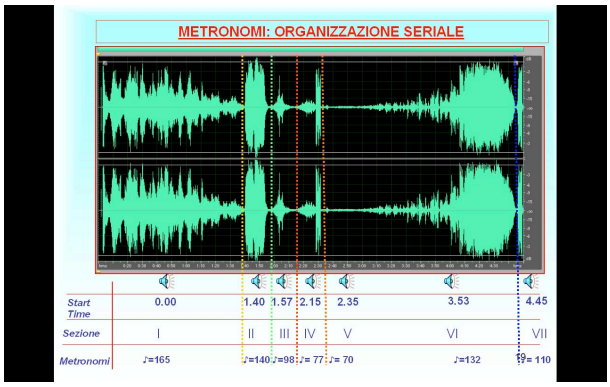


FIGURA 6: La macrostruttura percepita, derivata dal processo di ascolto e analisi.

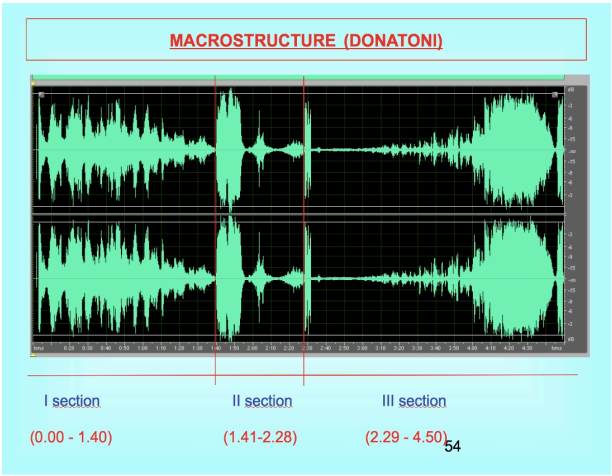


FIGURA 7: Le tre macrosezioni, individuate secondo le indicazioni del compositore [8].

5. CONCLUSIONI

L'immagine generale, che si trae da questo lavoro, è che si tratta di un brano che, pur dovendo molto allo Strutturalismo, ha delle connotazioni ampiamente personali: nell'uso delle tecniche combinatorie, nella scelta di materiali di partenza ben "qualificati e contraddistinti".

Nonostante la supposta imperizia di cui parla lo stesso Autore, rileviamo un sapiente uso strutturale dei parametri musicali, una migliore articolazione gestuale e regia drammaturgica rispetto ad altri brani strutturalisti dell'epoca.

Le analisi effettuate mettono in evidenza la capacità del compositore di tener sotto controllo, oltre ai parametri musicali tradizionali, anche quelli più squisitamente attinenti alla generazione del suono, con una

[9] Questo risultato non deve sorprendere: secondo la Psicologia della forma, una corrente psicologica fondata agli inizi del '900 in opposizione allo strutturalismo nello studio dei processi cognitivi, il "tutto è differente dalla somma delle singole parti". A. NEGRI, "Il Novecento filosofico e scientifico", Marzorati, Milano, 1993, vol. III, pp. 327-331 [14].

forte attenzione alla relazione tra oggetti sonori, ai significati espliciti e reconditi che essi portano con sé, singolarmente e in relazione agli altri.

Donatoni dimostra di essere attento non solo ai singoli processi compositivi e al loro coordinamento, ma anche alla dimensione più squisitamente percettiva del lavoro.

6. BIBLIOGRAFIA

[1] AA.VV., "La musica elettronica", a cura di H. Pousseur, Feltrinelli, 1976, p. 47.

[2] F. Bayle, "La langue inconnue. INA/GRM, La musique électroacoustique, édition hyptique.net", coll. Musiques Tangibles, 2000.

[3] G. Bataloni, **"**Sguardo complessivo sulla forma e analisi di alcune tecniche contrappuntistiche applicate all'arte combinatoria nello stile di Franco Donatoni", inedito.

[4] G. Coral, "Franco Donatoni: un ricordo del grande compositore veronese", inedito.

[5] R. Dalmonte, "Colloquio con F. Donatoni", in Musica/Realtà, anno IX, n°25, aprile 1988, pp. 65-78.

[6] F. Donatoni, "La musica e l'io: un'ipotesi sulle possibilità di autodidassi mediante adozione delle funzioni introproiettive quale metodo di comporre", inedito.

[7] F. Donatoni, "La musica elettronica", in "Discoteca", n°14, Milano, 1961, p.4.

[8] F. Donatoni, Quartetto III, "Note operative", Suvini Zerboni, 1961.

[9] F. Donatoni, "Questo", Adelphi, 1970, pp.12-13, p.15, p.17.

[10] A. Lanza, "Storia della musica: il secondo Novecento", E.d.T. Torino, 1991.

[11] G. Mattietti, "Geometrie di musica", LIM, Lucca 1996, pp. 172-181.

[12] S. McAdams, A. Bregman, "Hearing Musical Streams". *Computer Music Journal*, 1979.

[13] G. Montecchi, "Donatoni 1950-1972: da Bartók all'antimusica", in Donatoni, a cura di Enzo Restagno, E.d.T..

[14] A. Negri, "Il Novecento filosofico e scientifico", Marzorati, 1993;

[15] M. Picard, "Hitler in noi stessi", trad. it Rizzoli, Milano, 1947.

[16] E. Restagno, "Un'autobiografia di Donatoni raccontata da Enzo Restagno", E.d.T., Torino, 1990.

[17] S. Roy, "L'analyse des musiques électroacoustiques: Modèles et propositions", Paris, L'Harmattan, Univers Musical, 2003.

[18] D. Smalley, D. "Spectromorphology and structuring processes", The language of electroacoustic music (S. Emmerson ed.), MacMillan Press, 1986, pp. 61-93.

[19] A. Vande Gorne, **"**L'interprètation Spatiale. Essai de formalisation Méthodologique", Revue DEMèter, Dicembre 2002, Universitè de Lille 3.

RICOSTRUIRE NONO: DA *DIE ERMITTLUNG* A *RICORDA COSA TI HANNO FATTO IN AUSCHWITZ*

Carlo Barbagallo
Conservatorio "G. Verdi" di Torino
CoMET (Collettivo Musica Elettroacustica Torino)
info.barbagallo@gmail.com

SOMMARIO

Nel presente articolo viene descritto un lavoro analitico di approfondimento e ricerca intorno a due brani per nastro magnetico realizzati da Luigi Nono tra il 1965 e il 1966 allo Studio di Fonologia della RAI di Milano: *Musiche di scena per Die Ermittlung* e *Ricorda cosa ti hanno fatto in Auschwitz*. Obiettivo della ricerca è stata la ricostruzione su multitraccia di *Ricorda cosa ti hanno fatto in Auschwitz* a partire dalla versione definitiva di *Die Ermittlung*, a seguito di uno studio sulle relazioni tra le due opere e la *Composizione per Orchestra N.2: Diario Polacco '58* (1959) e attraverso l'analisi dei materiali intermedi per la produzione dei nastri finali dei due brani conservati presso la Fondazione Archivio Luigi Nono di Venezia. A seguito della ricostruzione è stato realizzato un ambiente esecutivo informatico per la spazializzazione multicanale in tempo reale del brano ricostruito, con l'obiettivo di rinnovare la pratica noniana di diffusione acustica per musica su supporto.

1. INTRODUZIONE

Il percorso compositivo che portò Luigi Nono alla composizione di *Ricorda cosa ti hanno fatto in Auschwitz* (1966) iniziò nel 1958 quando, invitato al II Festival Internazionale di Musica Contemporanea di Varsavia, ebbe la possibilità di visitare i campi di concentramento di Auschwitz e Birkenau. Questa esperienza lo ispirò alla scrittura di *Composizione per orchestra n.2: Diario Polacco '58*, opera che segnò l'inizio di una profonda speculazione da parte del compositore su alcuni materiali sonori nell'ambito dell'elaborazione elettronica su nastro magnetico, in una sorta di dinamico gioco della memoria; memoria della tragedia dei campi di concentramento, che nella pratica compositiva di Nono, si concretizza nel materiale musicale stesso che viene ripensato, rielaborato e riutilizzato, nel biennio 1965-1966, per ben tre opere sul medesimo tema: *Musiche di scena per Die Ermittlung*, una seconda versione di *Diario Polacco '58* e infine *Ricorda cosa ti hanno fatto in Auschwitz*, composizione per nastro magnetico solo autonoma dalle esigenze orchestrali e teatrali.

Copyright: ©2014 Carlo Barbagallo. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Per tentare di conoscere nel dettaglio questo percorso compositivo si è cercato in primo luogo di far luce sui legami tra le tre opere attraverso le fonti bibliografiche (sezione 2); in seguito si è proceduto nell'analizzare, tramite ascolto, ipotesi cronologiche, comparazione con le versioni definitive e dei cataloghi online dell'Archivio Luigi Nono di Venezia [1] e dello Studio di Fonologia della RAI di Milano [2], il contenuto audio delle bobine, conservate presso l'archivio veneziano e contenenti i materiali intermedi per la produzione dei lavori considerati (sezione 3); infine, dopo aver effettuato una dettagliata analisi formale di *Ricorda cosa ti hanno fatto in Auschwitz* sono stati realizzati una ricostruzione su multitraccia del brano a partire dalle scene costituenti *Musiche di scena per Die Ermittlung* (sezione 4) e un ambiente informatico per l'esecuzione su un sistema di spazializzazione multicanale, la cui interpretazione è stata supportata e ispirata dal lavoro di ricostruzione svolto (sezione 5).

2. DALL'ORCHESTRA ALLA SCENA

Il progetto iniziale di *Diario Polacco '58* prevedeva l'utilizzo del nastro magnetico per specchiare tre gruppi orchestrali in altrettanti doppi ma una considerazione sulle difficoltà di accesso allo Studio di Fonologia e su una conveniente economia dei mezzi portarono Nono ad approfondire una disposizione spaziale ad arco con specchi a simmetria bilaterale (quattro gruppi orchestrali composti da quattro ottoni, quattro legni, dieci archi, quattro percussioni) con l'obiettivo di aggiungere profondità al movimento delle sonorità dell'orchestra pur nella tradizionale disposizione frontale.

Quando, nel 1965, Nono ricevette la proposta da Erwin Piscator di lavorare alle musiche di scena per la realizzazione teatrale di *Die Ermittlung* [3] di Peter Weiss a Berlino, non esitò, dato il tema, nel riprendere i nastri della registrazione RAI della prima esecuzione italiana di *Diario Polacco '58*, diretta da Bruno Maderna con l'Orchestra Sinfonica della Rai di Roma (Venezia, festival di musica contemporanea della Biennale 1959) e ripensarli come materiali di partenza per la manipolazione e la composizione elettronica,

[1] http://www.luiginono.it - Data ultimo accesso: 24 Settembre 2014

[2] http://fonologia.lim.di.unimi.it - Data ultimo accesso: 24 Settembre 2014

[3] *Die Ermittlung, Oratorium in 11 Gesangen* è un testo basato sulle note prese da Peter Weiss durante le sedute del processo contro un gruppo di SS e di funzionari del lager di Auschwitz che si tenne a Francoforte tra il 1963 e il 1965. Si tratta di undici brevi canti in versi liberi, documentari del processo.

cogliendone inoltre l'occasione per ideare una nuova versione di *Diario Polacco '58* [1] integrando il nastro magnetico. Nono utilizzò estratti dalla già citata registrazione RAI, trattati con tecniche di *phasing* (sovrapponendo più volte ogni singolo frammento a se stesso, variandone leggermente la velocità di riproduzione), anticipando o riproponendo le battute in partitura, simultaneamente o in ritardo rispetto all'orchestra dal vivo

Nel cimentarsi con l'opera di Weiss, il compositore decise di ampliare la varietà dei materiali [4] , in particolare cori [5] , (il coro di bambini del Piccolo Teatro di Milano e probabilmente il coro della RAI di Milano), registrazioni di suoni e fonemi della solista vocale polacca Stefania Woytowitz [6] e materiale elettronico prodotto appositamente allo Studio di Fonologia.

Le versioni definitive di *Die Ermittlung* sono due: un nastro costituito da 35 scene suddivise *da test-signals* [7] e un nastro preparato per l'esecuzione in cui le scene sono 34 [8] . Come già detto, l'anno successivo Nono ricavò da queste la composizione per nastro magnetico solo *Ricorda cosa ti hanno fatto in Auschwitz.*

3. ANALISI DEI MATERIALI D'ARCHIVIO

Nel tentativo di cercare di conoscere a fondo le opere considerate e ricostruire il lavoro svolto da Luigi Nono nel rimontare in un'opera unitaria le musiche di scena prodotte precedentemente, si è proceduto all'analisi tramite ascolto del contenuto audio delle bobine [9] (riversate su CD) dell'ALN di Venezia contenenti i materiali intermedi per la produzione delle versioni definitive (vedi Tabella 1 e Tabella 2).

L'analisi, la comparazione e la formulazione di ipotesi cronologiche, hanno presentato alcune difficoltà, in particolare per l'assenza di informazioni dettagliate sia sulle elaborazioni elettroniche effettuate sui materiali scelti che sulla produzione del materiale esclusivamente elettronico. Il metodo di lavoro di Luigi Nono presso lo studio di Fonologia, era infatti coscientemente incentrato sulla sperimentazione empirica vera e propria:

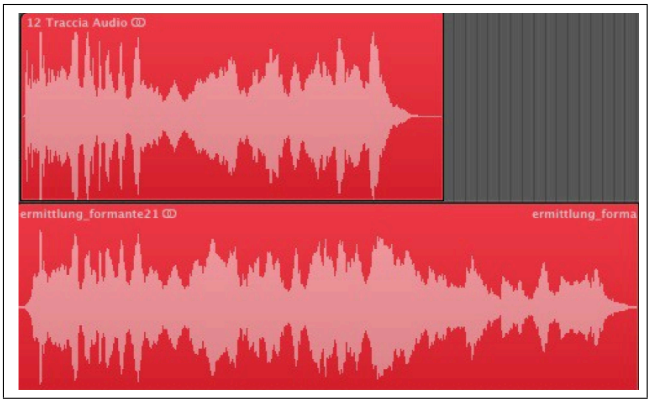


Figura 1: In alto il dodicesimo inserto di DP'58 versione 1965; in basso la scena 21 di Die Ermittlung.

> "...momento necessario di ricerca, di studio e di esperimento per poter finalmente iniziare la composizione vera e propria... questi tre stadi non sempre si sviluppano nella successione, ma spesso si sovrappongono ..."[2]

Nonostante spesso sia stato arduo capire la natura degli agglomerati sonori realizzati da Nono con la collaborazione del tecnico Marino Zuccheri, è stato comunque possibile dedurre mediante l'ascolto che probabilmente utilizzarono ampiamente tutti gli strumenti a disposizione dello Studio di Fonologia in quegli anni. [10]

Questa fase analitica ha inoltre permesso di individuare nella scena 21 del *Die Ermittlung*, ad eccezione di un nuovo attacco orchestrale e la dissolvenza a 1 minuto e 4 secondi (vedi Figura 1), il culmine espressivo del nastro magnetico di *Diario Polacco '58 v. '65* ovvero l'episodio più lungo (1 minuto e 14 secondi, dodicesimo inserto elettronico della versione).

La Tabella 1 e la Tabella 2 descrivono il contenuto dei CD numerati 22 [11] , 23 [12] , 24 [13] , 25 [14] . Le tabelle escludono le annotazioni dell'autore e le note fornite dall'Archivio Nono (vedi note 11-14), per le quali si rimanda al sito internet dell'archivio veneziano.

[4] L'utilizzo di materiali variegati venne giustificato da Nono così nel testo *Musica per Die Ermittlung di Peter Weiss* (1965): "Il materiale solo elettronico o solo strumentale o solo concreto resta molto limitato o serve solo all'ambizione dei puristi"[2]

[5] "Attraverso la musica solo ciò che la parola e la scena non possono rappresentare: i sei milioni assassinati nei campi di concentramento ... cori, quindi, in una concezione musicale autonoma, che si alternassero ai canti del testo, con un proprio tempo di sviluppo rispetto a quello scenico, altra formante dell'arco costruttivo della messinscena di *Die Ermittlung*."[2]

[6] E' interessante annotare come l'interesse del compositore si concentrò nella ricerca del raggiungimento di una carica espressiva, in qualche modo, pre-discorsiva, pre-verbale; elemento della sua poetica che ritornerà in maniera preponderante in due sue opere successive *A floresta È jovem e cheja de vida* (1966) e *Contrappunto Dialettivo della Mente* (1970): "...studiai come, componendo con semplici fonemi e suoni della voce umana, privi dell'elemento semantico di un testo letterario, si potesse raggiungere una carica espressiva significante e precisa."[2] Rinunciando così al supporto semantico della parola, Nono cercò di garantire una comunicazione emotiva ed esplicita attraverso i solo suoni, lavorati con rigore quasi scientifico durante la sperimentazione empirica, fondando in ciò il proprio impegno politico attraverso il mezzo musica.

[7] http://fonologia.lim.di.unimi.it/bobine_scheda.php?id=E.20029. Pubblicato in [3] http://www.luiginono.it/it/node/17024 - Data ultimo accesso: 24 Settembre 2014

[8] http://fonologia.lim.di.unimi.it/bobine_scheda.php?id=ZRic.20026 - Data ultimo accesso: 24 Settembre 2014

[9] La digitalizzazione dei documenti sonori dell'Archivio Nono è stata curata dal Laboratorio MIRAGE del DAMS-Musica dell?Università degli Studi di Udine.http://audiolab.uniud.it/lab_mirage.php - Data ultimo accesso: 24 Settembre 2014

[10] Nove oscillatori di cui era possibile controllare in maniera fine frequenza e ampiezza, un generatore di rumore bianco, un generatore d'impulsi, svariati registratori a nastro da uno a quattro canali, modulatori ad anello, modulatori d'ampiezza, una camera di riverberazione, un eco a nastro, due registratori a nastro a quattro testine rotanti con velocità variabile tramite oscillatori, svariati filtri e un banco di filtri d'ottava che permetteva di isolare la banda audio fino a due hertz.

[11] http://www.luiginono.it/it/node/18594 - Data ultimo accesso: 24 Settembre 2014

[12] http://www.luiginono.it/it/node/18595 - Data ultimo accesso: 24 Settembre 2014

[13] http://www.luiginono.it/it/node/18598 - Data ultimo accesso: 24 Settembre 2014

[14] http://www.luiginono.it/it/node/18600 - Data ultimo accesso: 24 Settembre 2014

CD track	Min. Sec.	Descrizione
22-3	0.00	Scena 1: coro e soprano filtrati, riverberati, a più velocità, quasi definitivo
	1.40	Scena 2: coro e soprano filtrati, riverberati, a più velocità, elettronica (sinusoidi alte frequenze, rumore bianco); nella versione definitiva i primi 10 secondi vengono tagliati, il frammento presenta un diverso inviluppo
	2.58	Scena 3: coro filtrato, riverberato, orchestra (percussioni e fiati)
	3,40	Scena 4: cori, orchestra
	4,02	Scena 5: cori, soprano, orchestra
	4,28	Scena 6: rumore bianco, cori, sinusoidi alte frequenze o rumore bianco filtrato
	5,37	Scena 7: percussioni, rumore, saturazione, cori
	6,12	Scena 8: come sopra
22-4	0,00	Scena 9: voce versi, cori, percussioni, fiati, sinusoidi, rumore bianco; nella versione definitiva i primi 8 secondi sono ripetuti e con chiusura differente
	0,40	Scena 10: sinusoidi o cori filtrati, suoni metallici, rumore bianco, tutto molto saturo
	1,05	Scena 11: percussioni, rumore bianco, cori, sinusoidi; nella versione definitiva la coda di "timpani rallentati" è tagliata, forse frammento utilizzato per *Auschwitz*
	1,43	Scena 12: voce sibilante, cori filtrati e rallentati
	2,30	Scena 13: soprano riverberato
	3,15	Scena 14: soprano riverberato e moltiplicato
	3,48	Scena 15: come sopra
	4,10	Scena 16: soprano moltiplicato, coro, elettronica, finale troncato, nella versione di *Auschwitz*
	4,40	Scena 17: coro, soprano, sinusoidi o feedback acustico in camera d'eco
22-5	5,06 0,00	Scena 8: timpani, soprano Scena 19: con lunga coda di sinusoidi e finale con scariche impulsive di percussioni rallentate; sfumata nella versione definitiva, uguale alla scena 20 per i primi 20 secondi con in aggiunta coro e crescendo di rumore bianco nel finale
	1,22	Versione alternativa della scena precedente con sinusoidi e scariche percussive isolate nel finale
	2,17	Come sopra, in aggiunta sinusoidi e percussioni rallentate
	3,34	Scena 20 con introduzione differente di scariche percussive; sinusoidi, cori e crescendo rumore bianco
	4,08	Scena 21: cori, orchestra, sinusoidi, versione alternativa forse utilizzata in *Auschwitz*, finale troncato. Dodicesimo estratto del nastro magnetico di *Diario Polacco '58 versione '65*.
22-6	0,00	Scena 22: timpani rallentati, sinusoidi, percussioni
	0,30	Scena 23: cori, percussioni, orchestra, soprano
	0,58	Scena 24 seconda parte: cori
	1,30	Scena 25: cori, nella versione definitiva crescendo del riverbero più in evidenza
	1,56	Esperimenti su cori
	2,37	Esperimenti su cori, probabilmente utilizzati con altro materiale nelle scene 26 e 27
	3,20	Come sopra
22-7	0,00	Scena 28: cori, sinusoidi
	0,38	Scena 29: rumore bianco, sinusoidi, cori
	1,20	Scena 30: cori modificati
	1,50	Scena 31: inizio impulsivo, cori
	2,10	Scena 32: cori, nella versione definitiva diverso profilo dinamico e taglio da 2,30 a 2,36
	3,06	Scena 33
	3,20	Scena 34
	3,35	Scena 35: probabilmente missaggio definitivo degli esperimenti su cori della traccia precedente
23		Si ipotizza che la bobina contenga versioni probabilmente precedenti (o in alternativa manipolazioni successive, poi abbandonate) a quelle riversate nel CD 22, tutte le scene in versioni abbastanza diverse da quelle definitive; qualità più scadente
24-1	0,00	Voci bianche e elaborazioni con filtri e rumore
	1,20	Percussioni, fortissimo
	1,35	Voci bianche con rumore bianco in crescendo fino al fortissimo, poi diminuendo, tratti impulsivi
	2,50	Montaggio: lastre o piatti riverberati, cori, bordone in bassa frequenza
	3,50	Montaggio: cori, percussioni, ottoni
	4,20	Come sopra
	4,30	Montaggio: percussioni su bordone in alta frequenza o voci bianche molo filtrate o rumore bianco filtrato
	4,58	Lento orchestra con filtri oppure sovra-incisione di clarinetti filtrati
	7,00	Come sopra
24-2	0,00	Voce: sussurri, sospiri, gemiti, grida soffocate, suoni gutturali, da 1,20 dice "kinder", nei montaggi descritti precedentemente e nelle versioni definitive sono così modificati da essere difficilmente riconoscibili
	2,15	Montaggi: clarinetto e voce
	3,40	Voce: come sopra; spesso è difficile capire se la modifica timbrica è effettuata dall'attrice o dalla lettura del nastro a velocità differenti
	7,35	Sovrapposizione delle take precedenti, a velocità differenti e forse filtraggio, a tratti riverberazione
24-3	0,00	Come sopra
	2,10	Montaggi: clarinetto e voce; variazioni dinamiche e di velocità
	7,05	Montaggi: gemiti, rumore bianco filtrato, frequenze medio alte, clarinetto e voce della sequenza precedente
	8,00	Montaggio tra sequenza precedente, fine del nastro riversato sulla traccia 24-1, suoni simili a campane e altri gemiti

Tabella 1: Tabella riepilogativa dell'analisi tramite ascolto dei nastri di lavoro per *Die Ermittlung* custoditi alla Fondazione Archivio Luigi Nono (CD 22-25). Prima Parte.

24-4	0,00	Coro di voci bianche: variazioni dinamiche e di velocità, riverberazione; probabilmente prove di montaggio, stop improvvisi, montaggi non precisi
	2,25	Poche voci bianche, vocali allungate, melodie casuali, a volte trattate con ritardi: da 4 minuti circa sembra che le melodie risultanti siano generate dalle variazioni di velocità di lettura del nastro
	5,00	Poche voci bianche in coro, variazioni graduali tra suoni consonantici bilabiali e suoni vocalici
	5,40	Come sopra
	6,00	Come sopra
	7,18	Suoni vocalici, passaggi graduali da una vocale all'altra su differenti altezze
	8,00	Passaggi più veloci o vocali tenute
	8,40	Passaggi più lenti
	8,50	Vocali separate e note tenute; un primo bambino inizia seguito subito dopo dagli altri
	11,00	Vocali separate, tutti insieme
	12,00	Versioni riverberate e duplicate delle take precedenti
	12,40	Singoli interventi, separati da click, mormorati a differenti e precise altezze del coro di voci bianche
24-5		Manipolazione elettronica delle registrazione del coro di voci bianche descritte precedentemente; inizialmente solo giustapposizioni, i tagli del montaggio sono evidenti
25-1/2		Nastro Ricordi per l'esecuzione composto da 34 scene: probabilmente copia del nastro originale *Zric. 026* [15] .

Tabella 2: Tabella riepilogativa dell'analisi tramite ascolto dei nastri di lavoro per *Die Ermittlung* custoditi alla Fondazione Archivio Luigi Nono (CD 22-25). Seconda Parte.

4. RICOSTRUIRE NONO

4.1 Analisi tramite ascolto

Luigi Nono suddivise la composizione in tre sezioni mirate alla condensazione formale degli undici canti del testo di Weiss e con il quale accostamento programmatico ha creato un triplice riferimento semantico alle sue riflessioni sulle realtà di Auschwitz: *Canto del lager*, *Canto per la morte di Lili Tofler*, *Canto della possibiliità di sopravvivere*. Sul piano musicale la suddivisione potrebbe rivelarsi nella presenza marcata, durante la parte centrale del brano, della voce solista e del coro di voci bianche, dall'uso prevalente di materiale corale nella parte iniziale e di quello orchestrale e percussivo in conclusione. Ad ogni modo, l'opera, nella sua compiutezza, vede l'intersecarsi continuo dei diversi elementi in agglomerati sonori spesso molto densi, anche qui in un gioco di rimandi alla memoria dell'ascolto (anticipazioni e posticipazioni, alle volte velate, altre evidenti) e in cui l'identificazione sia dei materiali che soprattutto il riconoscimento delle manipolazioni elettroniche applicate sfuma spesso in un continuum cangiante in cui cori ed elettronica agiscono da collante, non solo dal punto di vista tecnico-compositivo ma anche da quello poetico-simbolico.

4.1.1 Canto del Lager, prima parte

L'inizio del brano sembra essere una sorta di lungo crescendo formale diviso in tre blocchi distinti che iniziando in sordina nel primo raggiungerà il culmine nel terzo.

1A-x (0,00 – 0,55): il brano inizia con delle ondate di materiale vocale (soprano, cori, bambini). La struttura della sotto-sezione è speculare: con due interventi minimali a bassa dinamica posti all'inizio e in conclusione, entrambi collegati ad altre due formanti, di durata più lunga e caratterizzate da un profilo dinamico altalenante più variegato.

1A-y (0,55 – 2,00): qui il materiale vocale viene manipolato maggiormente rispetto a prima. Le due formanti di cui è costituito, collegate l'una all'altra, costituiscono la parte centrale del crescendo formale di cui si è detto. Le ondate del primo blocco sono caratterizzate da un montaggio più denso, mentre nel secondo appaiono lentamente, fino a diventare preponderanti, delle fasce di suono elettronico. Queste fasce sono probabilmente realizzate attraverso il pesante filtraggio di rumore bianco o altro materiale oppure prodotte attraverso oscillatori, sicuramente saturati dal nastro o trattati in altro modo. La sotto-sezione si conclude in dissolvenza sul solo materiale elettronico.

1A-z (2,00 – 2,50): blocco conclusivo del crescendo formale iniziale, nell'ottica di una graduale crescita della tensione emotiva, si apre in assolvenza su piatti orchestrali, seguiti dai cori per continuare con un fitto montaggio, anche qui in crescendo, di materiale orchestrale (prima ottoni, poi ottoni e percussioni) in un dialogo con le voci della soprano e dei cori, punteggiato da esplosioni percussive particolarmente modellate elettronicamente.

4.1.2 Canto del Lager, seconda parte

Questa sezione rappresenta, a mio avviso, un "ponte" tra il *Canto del Lager* e il *Canto per la morte di Lili Tofler*. Anch'essa è possibile suddividerla in tre blocchi compositi che nel loro insieme sembrano suggerire una forma piramidale per l'andamento dinamico nel complesso.

1B-x (2,50 – 3,22): parte sommessa aperta da leggeri timpani orchestrali, vede il ripresentarsi della fasce elettroniche di 1B accompagnate da leggere apparizioni di rumore bianco che si alternano al riaffiorare in sordina del coro e della voce della soprano. Quasi coda simbolica di 1C alterna il montaggio di incisi musicali cantati dagli ottoni e frasi nervose delle percussioni. Un taglio quasi improvviso a 3,18, il cui senso di stupore è aiutato dal breve crescendo di rumore bianco precedente, lascia prima gli ottoni e poi cori e fasce accompagnare l'inizio della sotto-sezione seguente.

1B-y (3,22 – 4,15): il coro dei bambini in solo si fonde, per scomparire e poi ritornare nuovamente, in un alternarsi di onde sonore: questa volta di rumore bianco più o meno filtrato e fasce elettroniche. Con due ultime onde di rumore bianco (4,00 – 4,15) seguito da lievi scie elettroniche la sotto-sezione si conclude.

1B-z (4,15 – 5,32): parte nuovamente sommessa in cui quasi solo le risonanze delle voci dei bambini in camera riverberante emergono, fondendosi gradualmente nelle basse frequenze risonanti di timpani, probabilmente molto rallentanti. Un colpo di rullante a 4,54 lascia riemergere del rumore elettronico che subito dopo svanisce lasciando una lunghissima coda dei timpani di cui prima.

4.1.3 Canto per la morte di Lili Tofler

Il *Canto per la morte di Lili Tofler*, sezione centrale e parte più onirica del brano, potrebbe essere suddiviso in due sottosezioni. Tutto il canto è caratterizzato dalla presenza quasi solista della voce soprano.

2A (5,32 – 7,04): la funzione di onda qui è svolta dalla voce della soprano e della sua ombra riverberante. Onde vocali successive si fanno sempre più lontane fino a 6,08 quando materiale elettronico da vita ad un moltiplicarsi delle voci. Da 6,30 a 7,02 il dialogo contrappuntistico tra i materiali (vocali ed elettronici), sebbene più diradato rispetto all'evento poco precedente, tende a confondere gli elementi tra loro; voci multiple filtrate si fondono con suoni sinusoidali in camera riverberante, accentuando le risonanze.

2B (7,04 – 8,39): sono code di suoni di timpani manipolati elettronicamente (in particolare nel profilo dinamico e nella velocità di riproduzione del nastro) che qui duettano con il canto della voce soprano. A 7,48 scariche di percussioni e ottoni rallentati e deformati dai filtri e ingigantiti dalla riverberazione, quasi come un ricordo di 1A-z e anticipazione del Canto successivo, lasciano emergere un canto di ragazzo sopra un coro di bambini moltiplicato. La chiusura è data da un netto suono di timpano, il cui *sustain* viene particolarmente allungato, anche qui come nel finale di 1B-z, sebbene più basso in frequenza.

4.1.4 Canto della possibilità di sopravvivere

Il canto finale di *Ricorda cosa ti hanno fatto in Auschwitz* è il più aggressivo dell'intero brano, basandosi interamente sullo sviluppo di quanto era già stato anticipato in 1A-z; il fitto montaggio di scariche percussive e frasi sconnesse degli ottoni fa da protagonista nell'alternarsi e il fondersi della presenza prepotente del coro e delle fasce elettroniche di cui si è più volte detto, della ripresa di alcuni incisi vocali (cori dei bambini e soprano) dalle sezioni precedenti. Un crescendo di rumore bianco decreta il finale in cui, anche qui, un timpano, questa volta ampiamente riverberato, chiude.

5. RICOSTRUIRE *AUSCHWITZ*

La realizzazione di *Ricorda cosa ti hanno fatto in Auschwitz* vede susseguirsi in diverso ordine le scene di *Die Ermittlung*, rimontate per l'occasione in un opera unitaria. Non vengono effettuate modifiche sostanziali ai frammenti, se non letture del nastro a velocità eggermente differenti e tagli all'interno di alcune sezioni; solo una manciata di frammenti sembrano effettivamente risultato di nuovi montaggi. Si rivelano sostanziali, invece, le modifiche al profilo dinamico.

I frammenti utilizzati, se si riferisce alla versione definitiva di *Die Ermitllung* comprendente le 35 scene[3], vengono presentati con svariate sovrapposizioni nell'ordine esemplificato dalla tabella Tabella 3. La tabella mostra anche un ipotesi concreta sulla possibile suddivisione in canti del brano, cercando di individuare, con una certa precisione, la suddivisione tripartita di cui si è accennato.

Contemporaneamente alla stesura della tabella si è ricostruito il brano su multitraccia e realizzata una versione a due canali unendo il nastro monofonico originale sul primo canale e la ricostruzione oggetto di questo studio sul secondo.

6. MISE EN ESPACE

Le analisi effettuate e descritte sono state di essenziale supporto alla progettazione di una *mise en espace* di *Ricorda cosa ti hanno fatto in Auschwitz* che ha inteso valorizzare le caratteristiche acustiche e drammaturgiche dell'opera, ispirandosi alle innovative intuizioni di Nono sull'integrazione di tecniche di diffusione acustica all'azione teatrale e all'esecuzione di musica su supporto. [16]

E' stato realizzato un ambiente esecutivo informatico per la spazializzazione multicanale del brano [17] utilizzando come "sorgente" un file audio a tre canali che include l'originale monofonico e le scene di *Die Ermittlung* montate come da ricostruzione e alternate sugli altri due canali. L'alternanza delle scene su tracce differenti e la loro eventuale conseguente sovrapposizione ha di per sé ispirato il lavoro interpretativo, permettendo di isolare e muovere in autonomia gli eterogenei frammenti nello spazio d'ascolto. Ad esempio, nel *Canto per la morte di Lili Tofler* (2A; vedi 4.1.3), la fitta sovrapposizione delle scene da 6,08 (vedi Tabella 3) ha permesso di poter rendere spazialmente il carattere a onde della sezione (mantenendo il monofonico originale in posizione centrale e ad un livello dinamico inferiore, si è cercato di localizzare ogni scena su un differente altoparlante per poi diffonderla verso il centro come ad immergere l'ascoltatore), oltre a rendere limpido il gioco contrappuntistico e di compenetrazione timbrica dei diversi materiali. Inoltre, le differenze, in termini dinamici e di fase, tra le forme d'onda dell'originale e dei frammenti da cui esso deriva, hanno reso possibile l'illusione di una profondità spaziale addizionale semplicemente disponendo le "sorgenti" su fronti opposti.

7. CONCLUSIONI

Il presente studio è un esempio di come l'approccio analitico ai documenti sonori relativi alla produzione di musica su

[16] Nel suo scritto *Musica e Teatro* [2] del 1966 Luigi Nono, discusse il rapporto tra queste due forme d'arte a fronte delle nuove possibilità tecniche offerte dalla manipolazione elettronica in studio e dalla possibilità di amplificazione dal vivo, nel tentativo di innovare i due linguaggi con stile contemporaneo e allo stesso tempo coerente con la tematica

[17] La realizzazione in ambiente Max della spazializzazione su n altoparlanti ha permesso con semplici algoritmi di gestire per ogni sorgente la sua velocità e direzione di rotazione, nonchÈ la sua distanza virtuale dal punto di ascolto previsto. Il controllo in tempo reale durante l'esecuzione è stato affidato a un interfaccia di controllo realizzata in ambiente Lemur (vedi Figura 2). Una versione quadrifonica è stata eseguita il 4 Novembre 2013 al Conservatotorio "G.Verdi" di Torino.

Canti	Min,Sec *Auschwitz*	Scene *Die Ermittlung* (sN)
Canto del Lager	0,00	s1
	1,05	s1 - s2
	1,59	s3
	2,25	s2 - s5
	2,32	s5 - s4
	2,42	s4 - s9; in silenzio fino a 2,50
	2,50	s9; dissolvenza in entrata dopo il picco percussivo, probabile LPF per tagliare i suoni elettronici oppure altra versione
	3,18	s9; taglio di circa un secondo e giunzione, eliminato crescendo e picco
	3,20	s9 - s6
	3,14	s6 - s12
	4,53	s12 - s11
	5,10 - 5,26	s11; coda allungata con eco o giunzioni di nastro oppure altra versione
Canto per la morte di Lili Tofler	5,31	s13
	6,06	s13 - s16
	6,14	s16; taglio di circa un secondo e giunzione
	6,15 - 6,26	s16; taglio e giunzione
	6,18	s16 - s14
	6,33	s14 - s19
	6,55	s19 - s15
	7,02	s15; veloce dissolvenza in uscita
	7,03	s18; in silenzio fino a 7.05, rivista dinamica attacco
	7,39	s24; in silenzio fino a 7.58
	7,44	s18 - s23
	7,58	s23 - s24
	8,03	s24 - s25
	8,14 - 8,16	s25; taglio e giunzione
	8,26	s25 - s22
	8,37	s22; dissolvenza
Canto della possibilità di sopravvivere	8,39	s21
	8,41 - 8,44	s21; taglio e giunzione
	8,49 - 8,52	s21; taglio e giunzione
	8,53 - 8,58	s21; taglio e giunzione
	9,56	s21 - s32
	10,29	s32 - s20
	10,52	s20 - s31

Tabella 3: Nella tabella le sovrapposizioni sono indicate con il trattino (-); le scene con sN.

supporto, può essere innestato nella prassi esecutiva, fornendo le basi per una rinnovata interpretazione del repertorio elettroacustico.

In considerazione dell'approccio metodologico adottato, che ha messo in chiaro alcuni aspetti dell'articolazione del percorso compositivo di Luigi Nono per questo gruppo di opere, è mia intenzione perfezionare l'analisi con l'estensione delle fonti a possibili documenti di rilievo non presi in esame dal presente studio.

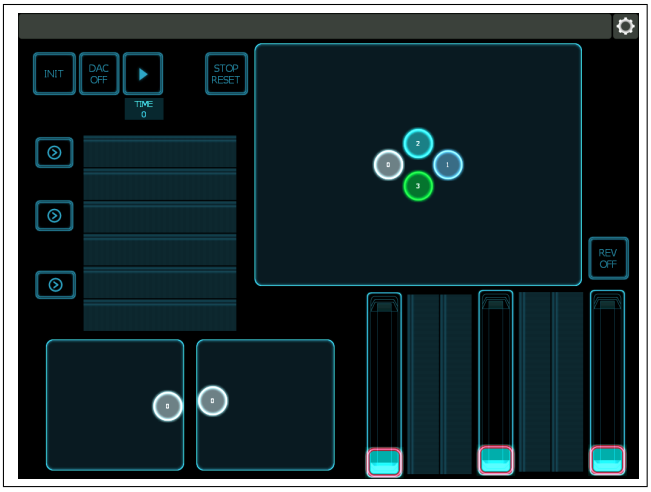


Figura 2: Interfaccia di controllo per la spazializzazione multicanale della versione ricostruita.

8. BIBLIOGRAFIA

[1] S. Bassanese, "Sulla versione 1965 delle composizione per orchestra n.2 (diario polacco '58)," in *La nuova ricerca sull'opera di Luigi Nono* (G. Borio, G. Morelli, and V. Rizzardi, eds.), Studi di Musica Veneta, Fondazione Cini, Archivio Luigi Nono, pp. 95–105, Olschki, 1999.

[2] V. Rizzardi and A. I. D. Benedictis, eds., *Luigi Nono. Scritti e Colloqui*. Ricordi, 2001.

[3] L. Nono, *Complete Works For Solo Tape*. Ricordi / Stradivarius, 2006.

[4] S. Canazza and M. C. T. Monti, *Ri-Mediazione dei Documenti Sonori*. Forum, 2006.

[5] L. Cossettini, "Le registrazioni audio dell'archivio luigi nono di venezia: linee per la conservazione e la critica dei documenti sonori," in *Musica/Tecnologia*, vol. 3, pp. 95–112, Firenze University Press, Fondazione Ezio Franceschini, 2009.

[6] M. Kontarsky, *Trauma Auschwitz. Zu Verarbeitungen des Nichtverarbeitbaren bei Peter Weiss, Luigi Nono und Paul Dessau.* PFAU Verlag, 2001.

[7] M. M. Novati, ed., *Lo Studio di Fonologia. Un diario musicale 1954-1983*. Ricordi-Universal, 2009.

[8] V. Rizzardi and A. I. D. Benedictis, eds., *Nuova musica alla radio. Esperienze allo studio di fonologia della RAI di Milano 1954-1959*. RAI-ERI, 2000.

[9] P. Violante, "Trauma auschwitz," in *Rivista di Storia delle Idee*, vol. 1, pp. 115–126, entro Studi Intrasformazione, Università degli studi di Palermo, 2012.

SOUNDSCAPEGENERATOR: SOUNDSCAPE MODELLING AND SIMULATION

Marinos Koutsomichalis
CIRMA/StudiUm - Università di Torino
marinos.koutsomichalis@unito.it

Andrea Valle
CIRMA/StudiUm - Università di Torino
andrea.valle@unito.it

ABSTRACT

This paper describes SoundScapeGenerator, a generic system for modelling and simulating soundscapes both in real and non-real time. SoundsScapeGenerator features algorithms for three-dimensional sound-localisation and is able to model complex spaces. The system relies on abstract rule descriptions (generators) to simulate deterministic, "cartoonified" (loosely modelled) sequencing of sound events. It further supports virtual P.O.H. (point of hearing) and is able to render the final output in various channel configurations. Based on generative algorithms, the SoundScapeGenerator implementation allows real-time user interaction with ever-changing soundscapes, and it can easily communicate with other applications and devices. Finally, we introduce SoundScapeComposer, a higher level module developed for the SoDA project (dedicated to soundscape automatic generation), that has been built on top of SoundScapeGenerator, hiding most of the latter's details to the user.

1. INTRODUCTION

Manual or automatic soundscape generation has been the focus of several projects hitherto. Yet, to our knowledge no system has been implemented that enables sophisticated and full-scale modelling/simulation of complex soundscapes. From our point of view such system should have the following features:

- modelling of complex soundscapes that may consist of an arbitrary number of zones with different geographical, acoustical and sonic characteristics;
- 3-dimensional sound-localisation that takes into account listener's and source's positions, source's size, sound-zone's acoustic features (reverberation, dampening, resonance, etc);
- both deterministic and non-deterministic sequencing and localisation of individual sounds;
- generation of sound events out of atomic sounds and samples in both deterministic and non-deterministic ways;
- modelling of sound events that may move in deterministic or non-deterministic ways in 3D space;
- modelling and simulation of POH (Point Of Hearing) "soundwalks" within the soundscape using virtual listeners;
- multi-purpose audio decoding to various speaker configurations –e.g. mono/stereo/quadraphonic/5.1/etc;
- Real-Time and Non-Real-Time operation.

Before examining the specifics of the proposed implementation, it is interesting to briefly discuss existing solutions, which in some cases successfully address some of the previously introduced issues and propose plausible soundscape generation paradigms. Regarding automatic soundscape generation, at least four relevant research projects need to be cited. The European project *Listen* [1] coordinated by the Fraunhofer Institut für MedienKommunikation is focused on the generation and control of interactive soundscapes, although it is specifically targeted at innovative, media-oriented experimentation on augmented reality. Its main goal is to create a new medium: the immersive audio-augmented environment (IAAE). Nevertheless, Listen is not targeted at explicitly modelling the soundscape. *Tapestrea* [2] is intended to create "environmental audio" in real-time: however, it does not define any explicit relationship between sound and space, and it does not provide user interaction. *Physis* [1] is an industrial research project led by IRCAM that deals with the modelling and the synthesis of virtual soundscapes. Physis is exclusively oriented towards the game industry and implementation details have not been published yet. *GeoGraphy* [3] is designed for the real-time simulation of existing soundscapes, starting from a database containing sound materials and other information. It can work interactively and in real-time, and includes the modelling of a virtual listener. Nonetheless, the organisation of audio materials is based on specific data structures (the so-called "graphs") which are potentially very complex and thus hard to handle. A generation technique of realistic soundscapes by means of a higher level methodology based upon GeoGraphy has been proposed too [4].

Copyright: ©2014 Marinos Koutsomichalis, Andrea Valle, . This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

[1] http://www.ircam.fr/305.html?&tx_ircamprojects_pi1%5BshowUid%5D=74&tx_ircamprojects_pi1%5BpType%5D=p&cHash=ed317fa8927e424c8700c020b1812a58&L=1 Retrieved June 14, 2014.

2. SOUNDSCAPEGENERATOR (SSG)

SoundScapeGenerator (SSG) is an autonomous soundscape synthesis engine capable of modelling soundscapes with a variable degree of realism. It has been designed trying to include all the features mentioned in Section 1, and has been prototyped using the SuperCollider programming environment. In SoundScapeGenerator the audio is generated by a Renderer that expects as arguments a SonicScape (the model of the soundscape, see Section 3), a Listener, which may be fixed or movable in the space, and a Decoder. The Decoder manages the desired output format. Internally, SSG relies on ambisonics spatialisation algorithms [5], so that, given the appropriate decoder, the same audio stream may be decoded (at least, theoretically) to any of the standard formats such as mono, stereo, 5.1, but also to a custom, arbitrarily speaker configuration in 2D or 3D space. The Renderer takes into account these three components to generate the final audio signal. It can operate both in real- and non-real time, thus making the system suitable for a wide range of applications. The SonicSpace is intended as a model of the desired space. A SonicSpace is built as an aggregation of an arbitrary number of individual "SoundZones" and with respect to their individual geographic, acoustic and sonic characteristics. SoundZone's geographical features refer to their spatial boundaries and their absolute positioning in 3D virtual space. Their acoustic features refer to modeled physical phenomena such as reverberation and resonance. Acoustic features are modelled independently and not with respect to the geometric properties of a modelled SoundZone—geographical informations are merely used to position/localise sounds. SSG also allows the user to model the acoustic properties of the boundaries (e.g. walls) that delimit the various SoundZones by means of various filtering. The sonic features of a SoundZone refer to the type of sound events that may occur inside a SoundZone and have to be specified using an arbitrary number of individual "SonicSources". SonicSources are conceived as containers for some sort of audio event which may or may not be repeated in time—their only differences lying in their spatial positioning and directionality. This means that a SonicSource consists of both the audio data to be reproduced and the information regarding timing and location for generation. In our model, audio data are intended as sound samples, as typically happens in real soundscape modelling. Timing is defined by means of a pattern-based mechanism (see next section); location depends also on the type of source. Figure 1 depicts the structure of a SonicSpace. SoundScapeGenerator features five different types of SonicSources:

- SonicAtmosphere: non-directional sonic ambience;
- FixedSound: directional and fixed in space;
- AmbulatorySound (2): governed by envelopes or by a callback function;
- SoundCloud: representing complex events, such as e.g. rain or crowds, that are characterised by multiple appearances of similar sonic sequences at ever-changing and random positions within a given cubic area and with respect to a density factor.

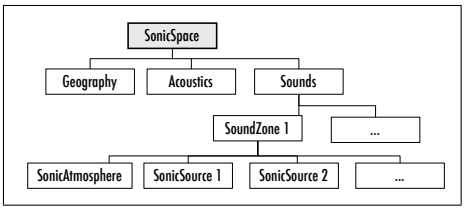


Figure 1. Structure of SonicSpace.

Each SoundZone has to be associated with at least one SonicAtmosphere, that functions as a background ambience and, therefore, provides a sort of canvas upon which individual SoundSources are positioned. The presence of such a background sound is useful also in order to mask unwanted artefacts in the sound samples, such as e.g. background noises in sound samples. The particular acoustic profile of a SoundZone will also add coherence to the scenery since all sound events within a given area will be processed in a similar way. The various directional SonicSources are localized using ambisonics algorithms thata have been further customized to allow for size-awareness (objects emitting sound from a larger area should have a broader spatial footprint than smaller ones) and listener-awareness (concomitant to the topological positioning and the listening radius of a virtual listener), as to be subsequently explained.The algorithm first encodes the audio source into an omnidirectional soundfiled (a four-channel b-format signal). Then the spherical coordinates of each SonicSource with respect to the virtual Listener's positioning in the SonicSpace are calculated as shown in equations 1, 2 and 3. In our implementation, the Listener's and Source's position coordinates are internally represented as three-channel control signals (to allow for ambulatory Listeners/Sources) and in Cartesian notation—channel 0 stands for the x, 1 for the y and 2 for the z dimensional axis. In the above equations, ρ stands for radius, ϕ for azimuth angle, θ for zenith, Δx, Δy and Δz for the difference between the Listener's and the SonicSource's coordinates for every dimensional axis. All coefficients are represented as functions of time (t) since they are audio signals. Note also that the equation to calculate the radius has been modified to account for the Source's size (R_s)—each SonicSource is assumed to be a spherical object that emits sound in all directions.

$$\rho(t) = \sqrt{\Delta x(t)^2 + \Delta y(t)^2 + \Delta z(t)^2} - \frac{R_s}{2} \quad (1)$$

$$\phi(t) = \arctan \frac{\Delta y(t)}{\Delta x(t)} \quad (2)$$

$$\theta(t) = \arccos \frac{\Delta z(t)}{\rho(t)} \quad (3)$$

Each SonicSource's corresponding audio signal is then localised according to equation 4 which demonstrates how a source soundfield is localised. Soundfields are notated as a standard b-format ambisonics signal (b-format signals comprise of 4 coefficients, namely w, x, y, z [5]). a_r

stands for relative amplitude and is a parameter associated with each SonicSource so that the algorithm takes into account the phenomenological amplitude characteristics of the Source (e.g. recordings of a space-rocket and a bird may be of the same amplitude yet the first is phenomenologically understood as much louder to the latter) which has to be taken into account for a dynamic and realistic SoundScape to be generated. P_r is a synthesis module used to simulate the proximity effect. P_u is also a synthesis module that 'pushes' the soundfield in a particular direction, taking into account a distortion angle (ω, calculated as in equation 5) and the azimuth and zenith angles as calculated in equations 2 and 3. A distortion angle of 0 stands for a non-directional soundfiled coming from all directions while one of $\frac{\pi}{2}$ stands for sound localised at a single spot. The proposed algorithm will result in 'pushing' all sounds that are outside a five meters range from the Listener's position to behave as nominal single-spot Sources while it will maintain a broader spatiality for those Sources located closer to it. Finally, the a_ρ represents a amplification factor which is calculated as an exponential mapping of the radius (ρ) (which represents the distance of a Source's to the Listener) to a range of $0, 1$ and with respect to the Listener's listening radius (ρ_l)—as shown in equation 6. The formula first produces a value between $1, 2$ and then subtracts 1, to compensate for the idiosyncrasies of the unit generators used internally in our implementation.

$$S_i \begin{bmatrix} w & x \\ y & z \end{bmatrix}(t) = a_r \times a_\rho(t) \times P_r(P_u(\begin{bmatrix} w & x \\ y & z \end{bmatrix}(t), \omega, \phi, \theta)) \tag{4}$$

$$\omega(t) = \begin{cases} \frac{\pi}{2} \times \frac{\rho(t)}{5} & \text{if } 0 \leq \rho(t) \leq 5 \\ \frac{\pi}{2} & \text{if } \rho(t) > 5 \end{cases} \tag{5}$$

$$a_\rho(t) = \begin{cases} ((\frac{1}{2})^{\frac{\rho}{\rho_l}} \times 2) - 1 & \text{if } \rho(t) \leq \rho_l(t) \\ 0 & \text{if } \rho(t) > \rho_l(t) \end{cases} \tag{6}$$

All SonicSources belonging to a SoundZone and the SonicAtmosphere associated with it are mixed and further processed with respect to the acoustic profile of the SoundZone, as shown in equation 8. Then, the audio output of all SoundZones (Z_{i-n}) is mixed and routed to the a Decoding module which will generate the SoundScape (S_f) in the desired format. Equation 7 demonstrates the decoding algorithm.

$$S_f = D(\sum_{i=1}^{n} Z_i(t)) \tag{7}$$

$$Z_i(t) = A_{coustics}(A_{tmo} + \sum_{i=1}^{n} S_i(t)) \tag{8}$$

Therefore, the proposed algorithm takes into account the Listener's relative three-dimensional positioning with respect to each SonicSource, their distance (also accounting for the the potential proximity effect), the size of the SonicSource and the acoustic profile of each SoundZone. Insofar as localisation of audio samples is concerned, it has to be kept in mind that sounds that already have spatial information registered within the recording should be treated differently when modelling a soundscape. Close-up of sound events, for example, should be positioned according to where the sound's source should be, e.g. in the sky in the case of a bird, while sound events recorded from a distance should be positioned somewhere next to the listener, since they already convey a listener's perspective of something occurring in a distance. When positioning such Sources, it should be also kept in mind that, as already explained the algorithm will progressively 'push' non-directional soundfields to directional ones within a five meters range from the Listener's positioning; therefore and for most cases, this means that such Sources should be positioned at a distance no less than 5 meters. Figure 2 demonstrates the flow of audio and control signals within our current implementation of SoundScapeGenerator. As already mentioned, the Listener's and the various Sources' positioning are represented as three-channel control-rate signals that feed into internal virtual buses. Audio is also streamed internally using four-channel virtual-buses.

2.1 Pattern-based Sequencing

One of the most difficult aspects in soundscape simulation and generation concerns the modelling of a source's behaviour in time. In real-life soundscapes, sound events may repeat themselves in irregular patterns. An interesting approach in relation to sound synthesis is cartoonification as devised by the Sounding Object project [6]; here sounds are described not in terms of the real mechanics of their production, but following a phenomenologically-compliant and physically-simplified approach. The idea at the base of cartoonification can be extended from sound synthesis to the organisation of sound sequences. In SoundScapeGenerator, time-organisation of sound events is thus cartoonified (loosely modelled) by means of specific data structures, namely generators [7]. A generator can be thought of as a rule for generating sequences. When executed, a generator will result in a stream of values of a certain length and with respect to some high-level rule. As the rule is specified rather than the actual data, the sequence can be of infinite length. In relation to SoundSources, a generator-based strategy allows for quick and efficient emulation of a variety of time behaviours: from one-shot to repetitive sound-events, from deterministic to stochastic sequences. Thus, in SoundScapeGenerator, all SoundSources have to be associated with a pattern that defines the exact time of their first appearance and their repetition scheme. Support for generators is native in the SuperCollider language by means of "Patterns" and "Streams" of data that result from their execution [8]. Hence on, we will use the term "pattern" and the relative SuperCollider notation. Built-in patterns include representations for linear sequences, random selections from lists, probability-based number generators, random walks, and other similar mathematical constructs that provide a conceptually straightforward way to model streams of values. Such patterns may be chained and/or nested recursively, allowing for a very compact notation of

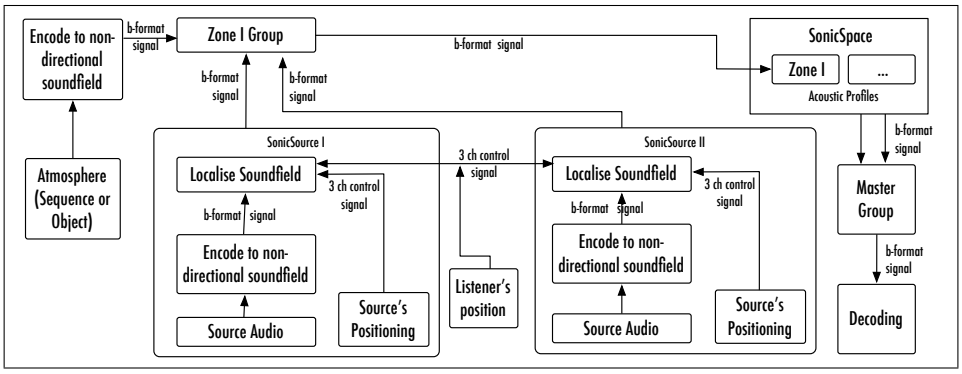


Figure 2. Audio flow within SSG.

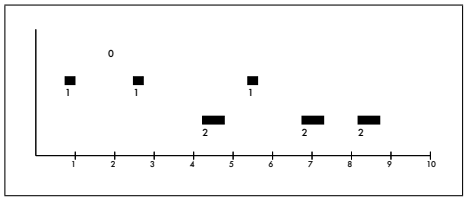


Figure 3. Barking Sequence.

very complex behaviour. Consider, for instance the following pattern:

$$Pseq([0,1],2)$$

This stands for a linear sequence of 0 and 1 repeated twice (2), representing time intervals. As a model of temporal behaviour, it will result in the corresponding SoundSource being reproduced 4 times: immediately once the Renderer is asked to play (0 seconds), 1 second after having finished playing back, immediately after this second appearance has finished (again, 0 seconds), and 1 second after the third appearance. As an example, it is easy to model a highly realistic, ever-permuting dog barking by means of simple patterns applied to just a few barks, like that:

$$Prand([a,b,s],inf)$$

$$Pwhite(0.5,2.0,rrand(3,10))$$

This two patterns will result in a virtual dog barking every now and then, as shown in Figure 3 in irregular patterns and having a varying duration. The first pattern defines the atomic sounds that will be used, with a and b representing two different bark atoms, s standing for silence and inf for infinity (it is upon the duration pattern to define the total number of atoms used). The second pattern defines their durations as random numbers between 0.5 and 2 seconds and will aggregate a random number of atoms between 3 and 10 (the $rrand$ function). Then, a Source which points to the aforementioned sequence object will cause it to generate a new audio sequence whenever needed and with respect to its particular repetition schemata. A graphical representation of a possible barking sequence (here made up of 7 sounds for ≈ 9 seconds) is shown in Figure 3, where each bark sound is given an index (1 and 2) and silence is represented by 0. To compensate for potential discontinuities when joining sound samples together and to seamlessly truncate audio files when needed, SSG uses parametrisable linear (cross)fades. The designer may select the appropriate cross-fade time with respect the idiosyncracies of the audio samples in use; e.g. joining individual footsteps together to form a larger sequence requires minimal or even not fade times between each sample, while joining city ambiences to construct a longer Atmosphere requires fade times of several seconds (even minutes). In any case, looping may be achieved using pattern sequences: the algorithm will simply create crossfades with a sound and a repetition of itself.

2.2 Composing soundscape as a hyper-narratives

As discussed, SoundScapeGenerator features different kinds of Listeners. Listeners are given a listening radius as well, outside of which no sound is audible. After having localised the sounds using the extended ambisonics techniques described before, the algorithm modulates their amplitude proportionally to their distance from the Listener's positioning to deliver a POH (Point of Hearing) interpretation of the SonicSpace. In that sense, SoundScapeGenerator follows a listener-oriented and phenomenological approach. The exported soundscape is analogous to a virtual listener's perspective of the SonicSpace and with respect to their spatial positioning, their listening radius and the particular acoustic features of the area in which they are positioned. Therefore, rendering with different Listeners may result in dramatically different versions of the very same SonicSpace.

Considered as a semiotics for the description of soundscape, SSG is able to output soundscapes as its utterances, depending on various factors. In contrast to a DAW-like purely "syntagmatic" (sequential) approach, SSG models a SoundScape as a "semiotics" where each Listener's particular trajectory through a SonicSpace can be seen as a realisation of virtual possibilities (its "paradigm"). In that sense, and from a user-perspective, SSG may be conceptualised as a tool to model a broader hyper-narrative (or maybe, a hyper-soundscape) that may yield very different outputs. Hyper-narratives are to be understood as the sum of the multiple trajectories through a paradigm [9] or, in this particular case, as the aggregate of all possible combinations of the elements in a SonicSpace. Each time SSG is asked to produce audio, a new narrative is generated, yet one which is always a subset of a broader hyper-narrative.

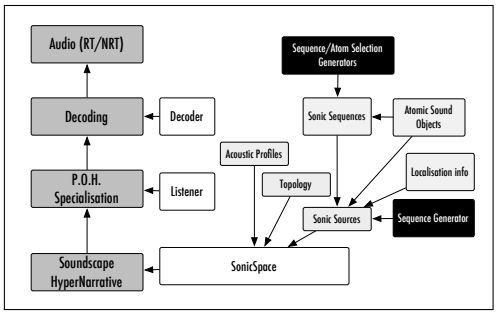


Figure 4. General structure go the SoundScapeGenerator Model.

From a practical point of view, using such an approach is highly advantageous in that coherency and consistency are easily achieved. A hyper-narrative remains identifiable as such and, if carefully designed, it outputs a unique, dynamic and unpredictable utterance–thus offering a highly personalised and dynamic experience that does not suffer from the problems normally associated with fixed soundscape recordings[10]. Figure 4 shows the complete architecture of SSG in four stages (dark grey): A "hyper-narrative" (designed by means of a SonicSpace) is to be "specialised" (with respect to a Listener object) and then "decoded" (with respect to an ambisonics decoder) so that audio is generated (in RT or NRT). Figure 4 further illustrates how a SonicSpace is designed by means of defining Acoustic, Topological and Sonic profiles (for each SoundZone). SonicSources are then shown to be depend on Generators (as already explained), localisation information and sound samples (be in an atomic sound object or in a sequence concatenated by generators).

3. THE SODA PROJECT: SOUNDSCAPECOMPOSER

SoundScapeGenerator has been designed as a generic system. While it is theoretically possible to directly embed it in third-party application, its current intended use is either as a stand-alone autonomous unit or as part of broader SuperCollider-based systems. Even in that form, however, it is quite straightforward to use SuperCollider's built-in communication channels (OSC, MIDI, etc) so that SSG can be controlled by other software or hardware applications. In this vein, SSG features no graphical user interface to avoid specialisation and keep the project generic [2] . Possible applications for SSG may include soundscape composition, acoustic ecology projects, virtual or augmented reality environments, interactive sound design. The system's NRT mode makes it a possible solution for audio-visual sound designers and composers looking for simulations of real or artificial environments to be used in other contexts. SSG's hyper-narrative paradigm in combination with its real-time capabilities makes it suitable for interactive or reactive navigation in virtual spaces. These include for example video games, animation, virtual reality or augmented reality applications, etc . We now describe

[2] In any case, SuperCollider has a sophisticated support for designing GUIs and it is a trivial task to create a GUI according to one's needs.

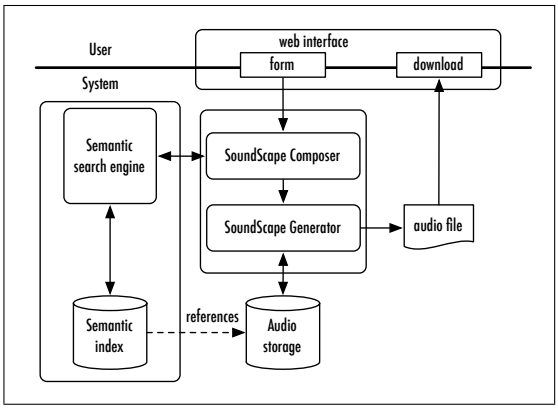


Figure 5. SoDA Architecture.

the SoDA project for which a specialised module has been devised as a higher level interface to SSG.

The SoDA (Sound Design Accelerator) project [3] aims at providing automatic soundscape composition by means of an ontologically annotated sound library [11]. SoDA requires a NRT sound engine which nevertheless should be automatically parametrisable according to the results of a semantical analysis engine. The idea is that the sound designer simply queries the system by inputting few keywords, and gets back an automatically generated soundscape. SoDA aims at providing –with respect to the creation of soundscapes– a twofold computational "acceleration" (hence its name): on the selection of relevant sound elements to be composited and on their organisation. Figure 5 shows SoDA's architecture and the various modules involved. The user is asked to provide a number of keywords through a web interface, then the semantical analysis engine is responsible for analysing the query so to return the sound files to be used in the generation step. Sound files have been previously annotated by a semantical analysis phase. Each of the audio files in the library is annotated so that technical as well as contextual information is associated, e.g. type of shot, relative amplitude, etc. SoDA then relies on an automated SoundScapeComposer. The latter is intended as a high level interface to SSG, as it provides SSG all the required data by taking into account the results of semantical analysis and by some ad hoc algorithms for compositing sound files. Once opportunely tuned, SSG is then responsible for the final audio rendering. Complying with SoDA's requirements for a fully automated soundscape composing paradigm, SoundScapeComposer (SSC) has being conceived as a bridge between SSG and the other components of SoDA. SSC has 4 tasks to address (6): parsing and interpreting the result of the semantical analysis engine; modelling a SonicSpace with the adequate features; populating it with SoundSources; placing a Listener within it. In the context of SoDA, a soundscape is intended as a background ambience with some possible moving sources: thus, a SonicSpace of a singleton Zone inhabited by a fixed Listener is enough. This is already an example of the specialisation achieved on SSG through SSC. SSC associates to the SonicSpace an ever-present Atmo-

[3] http://sodaproject.wix.com/soda

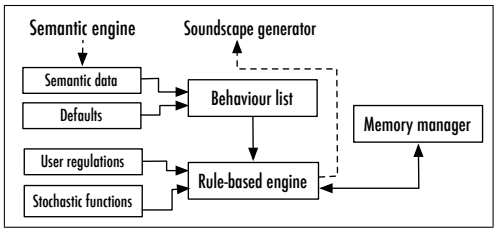


Figure 6. SoundScapeComposer Overview.

sphere, in order to provide a first background layer. One of the most difficult tasks of SSC is to figure out when and how individual atoms should be joined to form "meaningful" sequences. This is achieved by considering special meta-tags entries as well as the technical data retrieved from the files using a FEUtil –a special Feature Extraction Utility expressively designed for SoDA, which relies on machine listening algorithms to extract various spectral and psycho-acoustical information from audio files. To accomplish each task, SSC relies on three modules: an algorithmically generated Behaviour list; a Rule-based Engine, featuring general rules, stochastic and probability functions and user-defined regulations; a memory manager (Figure 6). The Behaviour list is intended as a description of the way a SonicSource behaves. It is built from the ontological data returned by the semantical engine, and provided with default states for everyday sounds: for example, animals or cars may occasionally move while architectural structures do not; speech is generally not to be repeated; cars move mostly on ground-level while elevators move vertically, etcetera. A behaviour list should be understood as numerical answers to certain properties, such as: the minimum/maximum allowed deviation in each spatial dimension for the localisation of a Sonic-Source, the minimum/maximum speed of their movement and their acceleration pattern, the maximum number of repetitions allowed for a SonicSource, and so on. The Rule-based Engine is, then, deputed to convert the data provided by the behaviour list into the required SonicSources and to provide them with their spatio-temporal features. SSC also features an intrinsic Memory module, initially empty, that is incrementally populated at runtime with the features of the generated SonicSources. Then, and unless the behaviour lists or the user-defined rules suggest otherwise, SSC attempts to accelerate variance by consulting the memory manager in order not to generate objects with almost identical features.

SSG is intended as a low-level, fine-tuneable engine. On the contrary, the purpose of SSC is to define an even higher level, by hiding the user the complexities involved in manually designing a soundscape with SSG. Interaction is intended to happen primarily through the various semantical filters s/he may apply to the query: the semantic engine becomes the main user interface.

4. CONCLUSIONS

The SoundScapeGenerator has been designed as a generic system suitable for generic soundscape simulation that involves sound samples. Its pattern-based logic and its hypernarrative compositional paradigm are intended to provide a flexible, interactive and generative low-level engine. Indeed, it still requires the user a certain amount of work to be set up. But its modular nature allows to define higher level interfaces that provide automatic parametrisation and specialisation, as in the case of SoDA's SoundscapeComposer. In fact, the latter relies on a minimum set of SSG's features. SoundScapeGenerator's current implementation is stable and fully functional. SSG has been initially tested in a wide range of soundscape-generation scenarios. A rigorous user evaluation has not been carried on yet, and it will be the next step of the project. Also, there is still room for improvements (e.g. concerning fidelity, efficiency, flexibility and ease of use), that will become more evident by taking into account users' feedback.

5. REFERENCES

[1] O. Warusfel and G. Eckel, "Listen-augmenting everyday environments through interactive soundscapes," *Virtual Reality for Public Consumption, IEEE Virtual Reality 2004 Workshop*, vol. 27, 2004.

[2] A. Misra, R. Cook, and G. Wang, "Musical tapestry: Re-composing natural sounds," in *Proceedings of ICMC*, 2006.

[3] A. Valle, V. Lombardo, and M. Schirosa, "Simulating the soundscape through an analysis/resynthesis methodology," *CMMR/ICAD*, pp. 330–357, 2009.

[4] M. Schirosa, J. Janer, S. Kersten, and G. Roma, "A system for soundscape generation, composition and streaming," in *Prossime distanze. Atti del XVIII CIM*, pp. 115–121, 2011.

[5] D. Malham and A. Myatt, "3-d sound spatialization using ambisonic techniques," *Computer Music Journal*, vol. 19, pp. 58–70, 1995.

[6] D. Rocchesso and F. E. Fontana, *The Sounding Object*. Edizioni di Mondo Estremo, 2003.

[7] T. Budd, *A Little Smalltalk*. Reading, Mass.: Addison-Wesley, 1987.

[8] R. Kuivila, "Events and patterns," in *The SuperCollider Book* (S. Wilson, D. Cottle, and N. Collins, eds.), pp. 179–205, The MIT Press, 2011.

[9] L. Manovich, *The Language of New Media*. Leonardo (Series) (Cambridge, Mass.), MIT Press, 2001.

[10] M. Koutsomichalis, "On soundscapes, phonography and environmental sound art," *Journal of Sonic Studies*, vol. 4, [Online], 2013.

[11] A. Valle, P. Armao, M. Casu, and M. Koustomichalis, "Soda: A sound design accelerator for the automatic generation of soundscapes from an ontologically annotated sound library," in *Proceedings ICMC—SMC—2014*, (Athens), pp. 1610–1617, 2014.

LADOP$_1$: DEFINIZIONE E COMPUTAZIONE DEGLI ADATTAMENTI IN OPERE D'ARTE INTERMEDIALI ADATTIVE

Alessio Gabriele
Centro Ricerche Musicali CRM, Via G. Peroni 452, Roma
Conservatorio di Musica "A. Casella", Via F. Savini, L'Aquila
alessio.gabriele@gmail.com

SOMMARIO

Ogni sistema utilizzato dall'uomo si evidenzia oggi come un agglomerato interagente di hardware, software e persone. Da qualche anno, dedicando la mia ricerca alla Musica e alla Scienza, ho lavorato su opere musicali e plastiche che inducono la percezione di un "presente continuo", con l'opera d'arte in rapporto dialogico con fruitori e ambiente. Si tratta di sistemi che adattano il proprio comportamento in risposta alle esigenze degli utenti, a possibili fallimenti, al cambiamento dell'ambiente in cui operano, alla variabilità delle risorse disponibili e al proprio output. Tali sistemi devono essere abbastanza flessibili da permettere l'introduzione incrementale di nuove funzionalità sia prima che dopo la messa in opera del sistema, assicurando requisiti di evoluzione *run-time*, disponibilità e qualità del servizio. LAdOp$_1$ propone un primo approccio alla definizione di un linguaggio utile alla computazione degli adattamenti in opere tempo-varianti adattive. L'opera musicale adattiva è definita nella sua forma completa solo al tempo di esecuzione e in un dato contesto, dipendentemente dalle risorse disponibili e dalle interazioni con i fruitori. LAdOp$_1$ permette di descrivere l'adattamento del sistema in termini di cambiamento di alcune parti del sistema stesso che, una volta applicate, conservano una proprietà di "stile" e soddisfano diversi livelli di granularità funzionale.

1. INTRODUZIONE

Il modello di computazione ubiqua immaginato da Mark Weiser [9] descrive persone che utilizzano diversi sistemi hardware e software durante azioni quotidiane, anche senza avere coscienza del lavoro che tali sistemi stanno compiendo. Questo paradigma di calcolo pervasivo è sempre più attuale e sta progressivamente sostituendo i sistemi tradizionali che risultano dall'attività umana cosciente, tant'è che la possibilità di pensare alla "elaborazione dell'informazione che si dissolve in comportamento" [10] ha promosso l'interesse anche per quei sistemi che riescono ad adattare il proprio comportamento in base alle sollecitazioni provenienti dall'ambiente circostante. La scienza è uno spunto e uno strumento indispensabile per la creazione artistica; l'arte spesso è opportunità anticipatrice e profetica nella comprensione del mondo che la scienza indaga. È così che l'esigenza di creazione artistica e di ricerca scientifica intorno ad un sistema capace di "sentire", "ricordare" e mutare il proprio comportamento conservando il proprio "carattere" ha dato nome e identità all'opera adattiva. Ad-Opera [8] è stato il primo frutto del lavoro di formalizzazione e schematizzazione di un framework per opere d'arte adattive. Un sistema costruito su Ad-Opera è aperto e dipendente dal contesto, può essere completamente definito solo al tempo di esecuzione ed è soggetto ad evoluzione per mezzo di continui adattamenti lungo tutto il suo ciclo di vita. Durante la fase di composizione non tutte le caratteristiche possono (o devono) essere definite. Ciò che "rimane" da specificare caratterizza tutte quelle parti dell'opera musicale che il compositore condiziona adeguatamente e affida agli adattamenti durante l'esecuzione. Ogni adattamento del sistema deve essere compiuto conservando lo *stile* dell'opera. Questo implica la necessità di un complesso processo di valutazione e verifica di invarianza dello stile al tempo di esecuzione, quando tutte le necessarie informazioni sono disponibili. LAdOp$_1$ è una prima proposta di linguaggio formalizzato in Ad-Opera per gli scopi di specifica del sistema, verifica dell'invariante stilistica e adattamento.

L'articolo è così organizzato: la sezione 2 introduce il concetto di adattività in musica e presenta alcune opere adattive già realizzate. La sezione 3 descrive alcune entità di alto livello previste nel paradigma computazionale di Ad-Opera. La sezione 4 descrive dominio teorico e formalizzazione del linguaggio LAdOp$_1$; nella quinta si illustra un esempio di adattamento basato su LAdOp$_1$. L'ultima sezione propone considerazioni sul lavoro svolto e suggerisce prossime attività di ricerca e sviluppo.

2. OPERA ADATTIVA

2.1 Accezioni di adattività

Il termine "adattivo" è utilizzato con accezioni diverse in ambito artistico e scientifico. Nelle scienze informatiche, con "adattività" si identifica una forma di personalizzazione che consiste nel costruire un'interfaccia in

Copyright: © 2014 Alessio Gabriele. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

grado di fornire l'interazione fra utente e computer in funzione delle caratteristiche dell'utente. I sistemi adattivi osservano i comportamenti degli utenti e modificano dinamicamente la loro interfaccia e/o il loro contenuto secondo preferenze e capacità individuali. Una delle conseguenze dell'adattività in senso sistemistico è la capacità del sistema di continuare ad assolvere ai propri compiti a fronte del cambiamento, o della deriva, di uno o più parametri di stato. Per quel che concerne l'arte e la produzione artistica, un'opera adattiva è quella capace di conservare il suo "stile" nel corso della sua esistenza, di preservare una sua integrità o identità formale ed espressiva anche sotto l'influenza delle interazioni con l'utente, del mondo circostante e degli effetti del suo stesso stato "vitale". Mentre in un'opera interattiva l'intero piano delle azioni e delle reazioni possibili è del tutto predeterminato dall'autore già in fase di composizione, l'adattività identifica un sistema che si auto-regola e che modifica il proprio stato in relazione a un'azione del fruitore e dell'ambiente circostante, restituendo al fruitore risposte che tengono conto non solo dell'azione attuale, ma anche della successione delle precedenti azioni e dell'intero contesto ambientale, conservando la propria identità stilistica [6]. Tipicamente, il fruitore ha la libertà di scegliere la sequenza delle azioni da effettuare ma gli esiti sono irreversibili, nel senso che a uno stimolo uguale sollevato in tempi diversi, non corrisponderà in generale un uguale effetto. Il sistema è in qualche misura capace di apprendere, di adattarsi agli stimoli e alle mutevoli condizioni intorno a sé. Gli aspetti legati all'adattività risultano in variazioni di funzioni ad alto livello, ma non toccano le regole organizzative del materiale sonoro soggiacente, perché quelle funzioni provvedono al comportamento per cui è possibile riconoscere un'identità, ergo uno stile. Quando il comportamento è adattivo, lo stile è un invariante dell'opera.

2.2 Opera adattiva: le esperienze

Il primo prototipo di opera adattiva è stato realizzato nel 2006 dal compositore Michelangelo Lupone per l'installazione sonora d'arte "Volumi Adattivi", insieme all'artista visuale Licia Galizia presso il Centro Ricerche Musicali CRM di Roma. I fruitori possono interagire con l'opera sfiorando e muovendo le sue parti mobili, le risposte sonore intervengono sia sul materiale musicale che sullo spazio d'ascolto. In relazione all'interazione e al suo stato interno, l'opera inizia un processo evolutivo legato all'ambiente più che alle interazioni con i fruitori. Numerose altre installazioni sonore d'arte adattive sono state poi realizzate presso il CRM da Lupone e Galizia, in un processo che ha permesso di definire in maniera sempre più raffinata le regole compositive e i principi che favoriscono una caratterizzazione "personale" e una evoluzione "naturale" dell'opera, evitando il rischio di degenerazione entropica dell'opera musicale. "OASI" (Lupone – Galizia, 2014) (Figura 1.A) è una installazione adattiva capace di sviluppare la forma musicale in relazione alle interazioni (tocco, pressione, presenza) con il fruitore senza barriere di accessibilità. "Spazio 1N" (Alessio Gabriele, 2008) (Figura 1.B) è una installazione adattiva in cui la modulazione e l'adattamento dello spazio sonoro è influenzato dalla presenza e dai movimenti del fruitore che viene osservato attraverso un sistema di rilevazione radio e video.

3. Ad-Opera

Il progetto Ad-Opera [8] nasce dalla necessità di ricerca artistica e scientifica sulla famiglia di opere d'arte intermediali definite adattive. La ricerca su Ad-Opera è volta alla definizione di un framework di supporto a progettazione e implementazione di opere adattive, mediante strumenti teorico-pratici, metodologici, informatici e ingegneristici. Si sono rese subito evidenti la notevole complessità dei temi oggetto di studio e la necessità di porre a fattor comune esperienze provenienti da molteplici ambiti di ricerca (ingegneria, fisica, informatica, intelligenza artificiale, sensoristica, networking, interfacce uomo-macchina, musica, arti visive e plastiche). Il lavoro di formalizzazione dei concetti astratti tipici dell'arte e quello di sistematizzazione della ricerca con metodo scientifico hanno fin'ora portato alla modellizzazione di alto livello del sistema "opera adattiva", all'esplicitazione dei concetti di base e dei principi formali che la caratterizzano, alla teorizzazione di metodologie di gestione del ciclo di vita dell'opera, alla realizzazione concreta di opere musicali e plastiche adattive. I risultati e le prospettive "visionarie" hanno inoltre ispirato nuovi approcci ingegneristici alla definizione e gestione di software che adatta il proprio comportamento in relazione all'ambiente circostante [1, 7].

3.1 Stile, regole compositive, partitura

La prima definizione di "stile" riferita a un'opera adattiva si trova sui documenti di progetto redatti nel 2002 da Michelangelo Lupone per l'installazione sonora d'arte permanente "Gioco delle risonanze" presso gli Scavi archeologici di Pompei: "lo stile è l'identità dell'opera, [...] è rappresentato dalle invarianze organizzative del materiale sonoro". Lupone stesso commenta che tali invarianze sono leggi compositive che stabiliscono: 1. la disposizione diacronica e sincronica degli accadimenti sonori; 2. la tempo-varianza del materiale o dell'accadimento sonoro (dinamica, spettro, durata, spazio); 3. la correlazione dei dati provenienti simultaneamente dai sensori (priorità); 4. la congruenza dei dati di controllo dai sensori (condizionamento); 5. le relazioni tra dati in ingresso e processi sonori attivati (modalità di attivazione/articolazione degli accadimenti sonori).
Quel che chiamiamo "stile" può essere visto come una libreria di "buone pratiche" che diventano familiari al compositore e che gli sono "gradite". Questa libreria si costruisce sulla base della conoscenza del compositore, della pratica e dell'invenzione. Nell'elaborazione e nella costruzione di uno stile assume rilevante importanza an-



A



B

Figura 1. Esempi di installazioni sonore d'arte adattive già realizzate che integrano musica e forma plastica (A. "OASI", M. Lupone – L. Galizia (2014); B. "Spazio 1N", A. Gabriele (2008)).

che l'ascolto: quel che viene prodotto è valutato dal compositore stesso e rientra nel consolidamento stilistico. Lo stile costituisce l'invariante del processo adattivo poiché le operazioni e i processi elaborativi svolti dal sistema durante la vita dell'opera devono sempre rispettare le regole stilistiche definite dal compositore.

Una "regola compositiva" (o comportamentale) definisce la sequenzializzazione degli eventi musicali e stabilisce in che modo il sistema reagisca al verificarsi di alcune ipotesi in un dato lasso di tempo, ovvero quale sia la reazione a una determinata configurazione o evento fino a che le ipotesi di sussistenza della regola siano valide. Una regola statica descrive l'esatta sequenzializzazione degli eventi musicali (semplici o aggregati), mentre una regola dinamica è espressa in maniera parametrica e il suo risultato può essere valutato solo a *run-time*.

La "partitura" è costituita dalla concatenazione di regole compositive e varia nel tempo come conseguenza del processo adattivo dell'opera. Durante l'esecuzione, la partitura deve essere sempre in grado di verificare l'invariante stilistica. A seguito dell'innesco di un processo adattivo, una regola compositiva deve essere selezionata a *run-time* e verificata: solo se si può garantire la validità dello stile la regola può essere inserita in un punto della partitura (punto di adattività) ed eseguita.

4. LAdOp$_I$

La capacità di reagire ai cambiamenti mantenendo valido un insieme di obiettivi di alto livello è il fattore decisivo che permette lo sviluppo di sistemi che possono soddisfare requisiti mutevoli senza soluzione di continuità. Tali obiettivi di alto livello, detti "invarianti", rappresentano le proprietà del sistema che devono essere verificate a prescindere dalle condizioni ambientali (lo "stile" è invariante nel sistema). Al contrario, le proprietà non critiche possono essere rilassate, aumentando così il grado di flessibilità del sistema durante gli adattamenti [3]. Il fine è supportare l'adattività *run-time* e l'evoluzione del sistema. Nella definizione del linguaggio LAdOp$_I$ è stato utilizzato un approccio di tipo assume-guarantee [2], capace di soddisfare tali requisiti.

4.1 Verifiche Assume-Guarantee

La tecnica di verifica [2] prevede di specificare, rispetto a ciascun modulo (componente software), ciò che può essere assunto rispetto all'ambiente e ciò che esso deve garantire (quando l'assunzione è verificata); si definisce inoltre un sistema logico per inferire su tali specifiche modulari. Ciascuna componente è intesa come interagente con il suo contesto. Questo è rappresentato come una proprietà che l'ambiente deve soddisfare per interagire correttamente con la componente. Questa proprietà è chiamata *assumption* e rappresenta l'assunzione che la componente compie sul contesto. Se l'assunzione è soddisfatta, la componente che opera in quel contesto soddisfa sicuramente un'altra proprietà denominata *guarantee*. Grazie ad una opportuna combinazione delle proprietà assume-guarantee, è possibile dimostrare la correttezza dell'intero sistema come conseguenza della correttezza di ciascun modulo. Questo processo è supportato da studi e metodologie che propongono possibili decomposizioni del sistema [4] e generazione automatica delle assunzioni [5].

Il paradigma assume-guarantee [2] permette di suddividere un sistema S in moduli che possono essere sostituiti o modificati senza violare la proprietà invariante. Consente inoltre di definire attributi verificabili in maniera efficiente a *run-time* in modo da garantire la correttezza dell'adattamento. Un sistema adattabile S risulta dalla composizione $V \circ C$, per mezzo di un operatore astratto "$\circ$", di una parte centrale C (core) e una parte adattabile V, tale che S verifica l'invariante I ($S \models I$). La semantica dell'operatore $\circ$ è stabilita dalla semantica operazionale del linguaggio S. Al fine di permettere operatività composizionale, l'operatore $\circ$ deve godere della proprietà associativa. Se l'operatore $\circ$ gode anche della proprietà commutativa, allora S può essere composto in differenti maniere. Per lo stesso sistema è quindi possibile avere diversi insiemi di annotazioni assume-guarantee. L'invariante globale I e le assunzioni sul contesto di S richieste per soddisfare I rimangono sempre le stesse, ciò che può cambiare sono le assunzioni locali (interne) e i guarantee delle varie parti del sistema.

4.2 Ipotesi restrittive

Al fine di agevolare la definizione iniziale del linguaggio e l'approfondimento della teoria di adattamento è stato assunto che (i) è possibile valutare e gestire una sola regola compositiva alla volta; (ii) un elemento compositivo è l'unità atomica del calcolo, quindi interventi di adattamento su una regola compositiva non possono interrompere l'esecuzione di un elemento compositivo.

4.3 Termini del linguaggio

L'insieme dei termini del linguaggio $LAdOp_1$ è costituito dagli elementi compositivi (es. algoritmo di sintesi, una traccia audio, un insieme organizzato di dati di controllo) impiegati nella costruzione dell'opera. Essi costituiscono l'entità di base su cui costruire le regole compositive per mezzo di opportune istanze ed offrono interessanti analogie con lo spazio delle variabili e delle costanti di un linguaggio di programmazione procedurale: sono dichiarati con diversi attributi che ne definiscono il comportamento e l'utilizzo consentito (es. id, nome, durata temporale, formato, suono/silenzio, durata fissa o variabile). Si possono dichiarare elementi compositivi "semplici" o "complessi" (definiti a partire da tipi semplici). Un elemento a durata fissa è identificato con x, uno a durata variabile con $x_v(T_v)$. La durata temporale di ogni evento è definita come $dur(x)= T_f$; $dur(x_v(T_v))=T_v$

4.4 Definizione formale

Si definiscono di seguito le componenti di una versione semplificata del linguaggio di programmazione $LAdOp_1$, in cui non vengono considerate questioni riguardanti le dichiarazioni dei tipi di dato, type checking, blocchi strutturati e scoping.

4.4.1 Definizione del linguaggio Cmat

La grammatica per *Cmat* è la seguente:

$$X ::= X_f \mid X_v \mid X \bullet X \qquad (2)$$
$$X_f ::= A \mid B \mid C \mid P1 \mid P2 \mid X_f \bullet X_f \qquad (3)$$
$$X_v ::= D_v(T) \mid P_v(T) \mid X_v \bullet X_v \qquad (4)$$

I termini generati da X (*Cmat*) sono chiamati *elementi compositivi:* X_f è l'insieme degli elementi a durata fissa, X_v è l'insieme degli elementi a durata variabile, $D_v(T)$ e $P_v(T)$ rappresentano rispettivamente elementi che implicano presenza e assenza di suono. La grammatica definisce anche l'insieme X che rappresenta tutte le possibili giustapposizioni temporali degli elementi compositivi a partire da elementi primitivi (A, B, C, $P1$, $P2$) fino ad elementi complessi. Per definire la semantica operazionale degli elementi compositivi bisogna considerare lo stato del sistema. Uno stato σ è una funzione che associa elementi di X a elementi della coppia (σ_w, σ_e) dove σ_w individua una finestra temporale che scorre nel tempo, delimitata da un tempo iniziale t_1 e un tempo finale t_2 ($t_1<t_2$); σ_e conta il numero di occorrenze di eventi sonori nella finestra temporale attuale. Si definisce l'operatore di incremento temporale: $\oplus$ sposta la finestra temporale individuata da σ_w incrementando di uno stesso valore gli estremi t_1 e t_2. Le regole semantiche di base per gli elementi compositivi primitivi sono:

$$\frac{\sigma'_w = (\sigma_w \oplus dur(A))\ \ \sigma'_e = (\sigma_e + 1)}{< A,\sigma > \xrightarrow[Cmat]{} \sigma'} \qquad (Cmat_A)$$

$$\frac{\sigma'_w = (\sigma_w \oplus dur(B))\ \ \sigma'_e = (\sigma_e + 1)}{< B,\sigma > \xrightarrow[Cmat]{} \sigma'} \qquad (Cmat_B)$$

$$\frac{\sigma'_w = (\sigma_w \oplus dur(C))\ \ \sigma'_e = (\sigma_e + 1)}{< C,\sigma > \xrightarrow[Cmat]{} \sigma'} \qquad (Cmat_C)$$

$$\frac{\sigma'_w = (\sigma_w \oplus dur(P1))\ \ \sigma'_e = \sigma_e}{< P1,\sigma > \xrightarrow[Cmat]{} \sigma'} \qquad (Cmat_{P1})$$

$$\frac{\sigma'_w = (\sigma_w \oplus dur(P2))\ \ \sigma'_e = \sigma_e}{< P2,\sigma > \xrightarrow[Cmat]{} \sigma'} \qquad (Cmat_{P2})$$

$$\frac{\sigma'_w = (\sigma_w \oplus dur(D_v(T)))\ \ \sigma'_e = (\sigma_e + 1)}{< D_v(T),\sigma > \xrightarrow[Cmat]{} \sigma'} \qquad (Cmat_{D_v(T)})$$

$$\frac{\sigma'_w = (\sigma_w \oplus dur(P_v(T)))\ \ \sigma'_e = \sigma_e}{< P_v(T),\sigma > \xrightarrow[Cmat]{} \sigma'} \qquad (Cmat_{P_v(T)})$$

$$\frac{< x,\sigma > \xrightarrow[Cmat]{} \sigma' \quad < y,\sigma' > \xrightarrow[Cmat]{} \sigma''}{< x \bullet y,\sigma > \xrightarrow[Cmat]{} \sigma''} \qquad composizione$$

4.4.2 Definizione del linguaggio Rule

La grammatica per *Rule* è così definita:

$$R ::= R_1(X) \mid R_2(X,X) \mid R_3(X_v) \qquad (5)$$

I termini generati da R (*Rule*) sono chiamati *regole compositive*. La semantica operazionale del linguaggio *Rule* è così definita:

$$\frac{< x_f,\sigma > \xrightarrow[Cmat]{} \sigma'}{< R_1(x_f),\sigma > \xrightarrow[Rule]{} \sigma'} \qquad (Rule_{R_1})$$

$$\frac{< x_f,\sigma > \xrightarrow[Cmat]{} \sigma' \quad < y_f,\sigma' > \xrightarrow[Cmat]{} \sigma''}{< R_2(x_f,y_f),\sigma > \xrightarrow[Rule]{} \sigma''} \qquad (Rule_{R_2})$$

$$\frac{< x_v,\sigma > \xrightarrow[Cmat]{} \sigma'}{< R_3(x_v,t),\sigma > \xrightarrow[Rule]{} \sigma'} \qquad (Rule_{R_3})$$

$$\frac{< R,\sigma > \xrightarrow[Rule]{} \sigma' \quad < R',\sigma' > \xrightarrow[Rule]{} \sigma''}{< R;R',\sigma > \xrightarrow[Rule]{} \sigma''} \qquad sequenzializzazione$$

$$dove\ x_f, y_f \in X_f, x_v \in X_v$$

4.4.3 Definizione del linguaggio Part

Una partitura del linguaggio $LAdOp_1$ è definita dalla seguente semplice grammatica:

$$P ::= R \qquad (6)$$

I termini generati da P (*Part*) sono chiamati *partiture*. L'operatore utilizzato per costruire una partitura è quello di sequenzializzazione ";". P è costituito da una sequenza di regole compositive che viene gestita dal sistema di derivazione $\rightarrow$*Part*. Se R_i è l'insieme delle regole eseguite durante una iterazione di P, si definisce la *durata* di un generico programma P

$$dur(P) = \sum_{i=1}^{n} dur(R_i) \qquad (7)$$

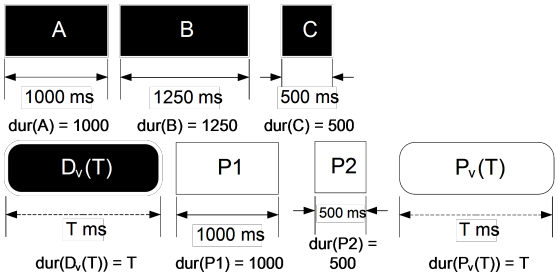
Semantica operazionale del linguaggio *Part*:

$$\frac{P = \{R\} \quad < R,\ \sigma > \xrightarrow[Rule]{} \sigma'}{< P, \sigma > \xrightarrow[Part]{} \sigma'}$$

5. UN ESEMPIO DI ADATTAMENTO

5.1 Definizione degli elementi compositivi

Si definiscono i seguenti elementi primitivi:



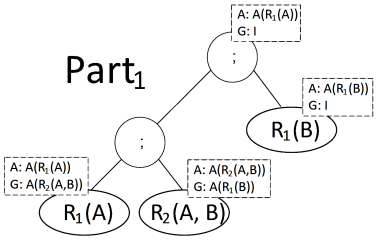
Per immediatezza espositiva si è scelto di dare una veste grafica agli elementi: i blocchi neri sono "contenitori" di suono, quelli bianchi di silenzio; angoli squadrati denotano una durata fissa, quelli arrotondati variabile.

5.2 Partitura con invariante non soddisfacibile

Una possibile partitura definita su $LAdOp_1$ è

$$Part_1=\{M_1:R_1(A);\ M2:R_2(A,B);\ M_3:R_1(B)\}$$

Il sistema *S* è costituito dall'insieme ordinato di moduli $S=\{M_1,M_2,M_3\}$. Si rappresenta $Part_1$ come albero sintattico astratto annotato con proprietà assume e guarantee:



Una volta decisa l'invariante *I*, si avvia un processo di calcolo mediante il quale si arriva alla esplicitazione delle proprietà assume-guarantee per ogni foglia dell'albero. Ad esempio, sia *I* l'invariante che garantisce la presenza di esattamente k eventi sonori in una finestra temporale di t' millisecondi. Per semplicità si assume *t'>dur(Part₁)*. Per t'=5000 e k=3 si esprime la proprietà $I:\{\sigma_w=(t\text{-}5000, t) \wedge \sigma_e=3\}$ dove t indica l'istante temporale raggiunto alla fine di una iterazione completa delle istruzioni di $Part_1$. Il calcolo delle proprietà A (assume) e G (guarantee), per ogni nodo dell'albero, deriva dall'analisi delle modifiche apportate alle variabili di stato dalla regola contenuta in quel nodo [2]. A partire da una invariante *I* desiderata per l'intero sistema *S*, l'obiettivo finale della valutazione, guidata dal processo di annotazione a ritroso, è trovare, se esiste, una possibile assunzione su *S* che soddisfi *I*. Se tale assunzione risulta non raggiungibile, è possibile concludere il processo con un responso negativo (*false*), oppure procedere da quel punto in avanti nel calcolo di un guarantee possibile G' per *S*, e quindi decidere se G' è ancora un invariante valido per *S*. Qui l'annotazione indica l'insoddisfacibilità dell'invariante.

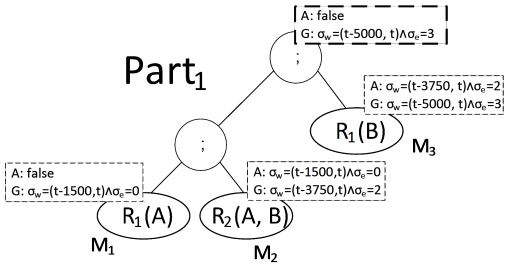


Figura 2. Albero sintattico astratto annotato per *S*.

Valutata la soddisfacibilità dell'invariante fino al modulo M_2 tramite opportuni ricalcoli, il sistema può proporre opportune sostituzioni dei moduli, o procedere in maniera automatica ad un adattamento del sistema che renda vera l'invariante. Supponiamo che il sistema decida di sostituire M_1 con un M_1' che rispetti l'assunzione di M_2, adottando un modulo costituito unicamente dalla regola $M_1':R_2(P_2,P_2)$. In questo caso l'adattamento produrrà una sostituzione del modulo M_1 con M_1' all'interno della partitura $Part_1$. Il sistema risultante è $S'=\{M_1',M_2,M_3\}$.

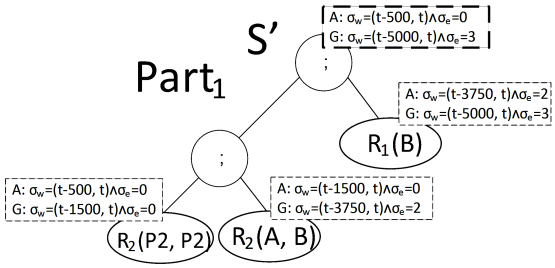


Figura 3. Albero sintattico astratto annotato per *S'*.

Dunque il sistema *S'* rispondente alla partitura $Part_1$ soddisfa l'invariante *I* se posto in un ambiente che assicura l'assunzione $A:\{\sigma_w=(t\text{-}500, t) \wedge \sigma_e=0\}$. Oltre alle sostituzioni, gli interventi di adattamento sul sistema possono determinare anche l'eliminazione di parte del sistema o, ad esempio, l'aggiunta di altri moduli.

5.3 Osservazioni

5.3.1 Granularità delle annotazioni

È possibile calcolare delle annotazioni e operare adattamenti di granularità maggiore rispetto agli eventi contenuti in una stessa regola compositiva. Supponiamo che due elementi B1 e B2 siano compresi rispettivamente nelle due regole $R_1(B1)$ e $R_1(B2)$, potrebbe allora accadere di dover compiere la sostituzione degli elementi B1 e B2 in maniera atomica. In questo caso un'annotazione complessiva dei due elementi (raggruppati come B') potrà essere utile alla esplicitazione delle condizioni di assume e guarantee necessarie all'adattamento nel rispetto dell'invariante. Con tale accorgimento, è possibile applicare su *S* degli adattamenti che possono riguardare anche insiemi di moduli.

5.3.2 Ricalcolo dell'invariante

Può succedere che durante il processo di annotazione a ritroso del sistema *S* ci si accorga che per un elemento generico x_i non è possibile trovare una assunzione Ax_i che soddisfi Gx_i e quindi il guarantee generale G. Una delle possibilità offerte dalla teoria consiste nel procedere proprio da Ax_i, stavolta non a ritroso ma in avanti, nel calcolo della catena di coppie assume-guarantee per tutti i moduli $M_i,\ldots,M_{n-1},M_n$ del sistema *S*. Si arriva alla definizione di un nuovo guarantee G' per *S*. Si avrà una invariante *I'* che potrà essere valutata e dunque accettata o rifiutata. Se *I'* viene accettata, allora un nuovo processo di annotazione (a ritroso) viene intrapreso.

5.3.3 Invariante degli elementi atomici

È possibile definire lo stile non solo per l'intero sistema *S*, ma anche per sue sottoparti, e garantire l'invariante stilistica anche su intervalli temporali ristretti, anche istante per istante. Si assume quindi che *I* possa predicare su intervalli più brevi rispetto alla durata del sistema, con discretizzazione minima data dalla durata di ciascun blocco. Il calcolo delle coppie assume-guarantee per ogni istante, avviene sempre a ritroso a partire dall'ultimo elemento su cui predicare [2].

5.3.4 Proprietà di frazionamento

L'individuazione di punti di adattabilità sottende il fatto che non è possibile lavorare sul continuo, cioè la valutazione di *I* o di sottinsiemi di *I* non può mai avvenire su una porzione di elemento compositivo.

5.4 Considerazioni per sviluppi futuri

La logica proposizionale appare inadeguata a esprimere predicati articolati che si fondano sul tempo, tant'è che non esiste un operatore adatto a scrivere formule temporali. Esiste invece un operatore adatto a questi scopi nella logica temporale, tale operatore è il "Globally" della LTL (Linear Temporal Logic). La logica temporale risulterebbe dunque indicata all'espressione di fatti musicali. Allo stato attuale la teoria sviluppata in Ad-Opera restringe il proprio interesse ad una singola traccia temporale; quando si passerà alla gestione di più tracce si passerà a logiche temporali di tipo Branching come CTL o CTL*.

6. CONCLUSIONI

Ad-Opera impone una ricerca multidisciplinare che è iniziata con la formulazione di un "vocabolario" condiviso e procede nello studio di tecniche, processi e strumenti comuni fra artisti, scienziati e fruitori dell'opera. La necessità oggi preminente riguarda l'approfondimento di aspetti teorici e sistemistici. I prossimi passi richiedono lo sviluppo di sistemi per la specifica e il controllo di leggi compositive sufficientemente capaci di supportare la non-degenerazione del materiale sonoro durante la stratificazione delle informazioni storiche del materiale stesso. Un sistema di auto-analisi è necessario per escludere processi degenerativi che possono danneggiare lo stile e per mantenere una coerenza in termini espressivi. Tali capacità di introspezione devono consentire analisi e valutazione dell'andamento dei parametri di elaborazione e sintesi più che il risultato sonoro prodotto dall'opera. Decisivi avanzamenti saranno possibili anche grazie al coinvolgimento di esperienze e saperi di altri ricercatori interessati ad indagare da più punti di vista il sistema uomo-opera-ambiente e le nuove possibilità espressive dell'opera d'arte, nell'ottica ormai non più visionaria di ambienti "liquidi" del vivere sociale, non tangibili, di computazione ubiqua.

7. BIBLIOGRAFIA

[1] A. Gabriele, M. Lupone, P. Inverardi, and P. Pelliccione: "Ad-Opera: Music-inspired Self-adaptive Systems" *Proc. FSE/SDP workshop on Future of software engineering research*, pp. 119-124, 2010.

[2] P. Inverardi, P. Pelliccione, M. Tivoli, "Towards an assume-guarantee theory for adaptable systems", *Proc. SEAMS 2009*, pp. 106-115. IEEE, 2009.

[3] B. H. C. Cheng, R. de Lemos, H. Giese, P. Inverardi, and J. Magee, editors. "Software Engineering for Self-Adaptive Systems", *volume 08031 of Dagstuhl Seminar Proceedings*. IBFI, Schloss Dagstuhl, Germany, 2008.

[4] Caporuscio M., Inverardi P., Pelliccione P., "Compositional verification of middleware-based software architecture descriptions" *Proceedings of the International Conference on Software Engineering* (ICSE 2004), ACM Press, 2004.

[5] Cobleigh J. M., Giannakopoulou D., Pasareanu C., "Learning Assumptions for Compositional Verification" *Ninth International Conference on Tools and Algorithm for the Construction and Analysis of Systems* (TACAS 2003), number 2619 in LNCS, Warsaw, Poland, April 2003.

[6] Lupone M., "Interattivo, Adattivo, Evolutivo" *Appunti per il Seminario all'Università di Belgrado – Facoltà di Architettura*, Centro Ricerche Musicali CRM, 12/12/2008.

[7] Inverardi P., Pelliccione P., Lupone M. and Gabriele A., "Ad-Opera: Music-inspired Self-adaptive Systems", *Computation for Humanity: Information Technology to Advance Society,* CRC Press, pp. 359-379, 2012.

[8] Gabriele A., "ADOPERA - definizione di un framework per la realizzazione di opere d'arte intermediali tempo-varianti", Tesi di Laurea, Dipartimento di Informatica, Università dell'Aquila, 2009.

[9] Weiser M., "The computer for the 21st century", *ACM SIGMOBILE Mobile Computing and Communications Review*, Volume 3 Issue 3, ACM New York, pp. 3-11, July 1999.

[10] Greenfield A., "Everyware: The Dawning Age of Ubiquitous Computing", New Riders Publishing, 2006.

LA MESSA ELETTROACUSTICA: SULL'UTILIZZO DELL'ORDINARIUM MISSAE IN ALCUNE COMPOSIZIONI ELETTROACUSTICHE.

Luca De Siena

lucadesiena@libero.it

SOMMARIO

Esiste un corpus di opere elettroacustiche che presenta nel titolo il termine *Messa* (o *Requiem*) e molte di esse sono inoltre costruite secondo la tradizionale successione di brani dell'*Ordinarium Missae*. Fino ad oggi, la musicologia non ha rivolto le sue attenzioni a queste composizioni che sono state invece il fulcro di interessanti sperimentazioni in campo internazionale. Un'indagine sistematica sull'esistenza di quel repertorio ha portato alla luce un *corpus* considerevole di musiche. Lo studio si è poi incentrato particolarmente su 4 *Messe* che utilizzano il testo e la suddivisione tradizionale dell'*Ordinarium* dato che, per le loro omogenee caratteristiche, si prestano maggiormente ad una analisi comparativa mirata a cogliere convergenze e divergenze fra quelle opere, nonché possibili riferimenti a scritture tradizionali della *Messa.* L'analisi ha utilizzato diversi parametri, tra i quali le tipologie d'articolazione vocale e i contenuti extramusicali utilizzati. È emerso che il trattamento elettroacustico dei principi della vocalità e della spazialità è risultato essere un tratto distintivo di tali lavori. Importanti supporti alla ricerca sono state le interviste ai compositori di messe elettroacustiche, che hanno permesso di definire numerosi aspetti: il perché della scelta compositiva; il rapporto del compositore con la scrittura dell'opera; i legami con la musica del passato; la committenza religiosa; l'eventuale - e non ovvio, come ci si aspetterebbe - utilizzo liturgico delle composizioni; gli elementi in comune tra questi nuovi lavori.

1. MESSE ELETTROACUSTICHE

1.1 Repertorio censito e catalogato

Più di un anno di ricerche e di contatti diretti con i compositori hanno portato alla raccolta e alla catalogazione di un repertorio di 22 opere**.** Di queste 22, si è riusciti ad entrare in possesso delle registrazioni di 11 *Messe* acusmatiche, di cui 6 *Requiem*.

È bene precisare da subito che esistono molte opere che nel titolo presentano la parola "Messa" o "Requiem" ma il ricondurle effettivamente alla forma tradizionale è un'operazione che richiede un enorme sforzo di astrazione dato che non sussistono più né il testo tradizionale né tanto meno la tipica suddivisione formale basata sull'*Ordinarium*.

la *Messa* elettroacustica è qui presa in analisi come presunto anello evolutivo di una tradizione musicale che parte da Machaut ed arriva a Beethoven passando per Palestrina. Pertanto, per ricongiungerci a questa tradizione o anche solo per lasciare trasparire un esile legame con questa produzione musicale, è necessario che, in assenza di riferimenti ad un sistema musicale comune, la suddivisione formale dell'*Ordinarium* (cioè la successione di *Kyrie, Gloria, Credo, Sanctus* e *Agnus Dei*) e la presenza del testo tradizionale siano state entrambe conservate. Tutte e quattro le opere analizzate in questo scritto, anche e soprattutto ai fini di una maggiore comparabilità, presentano queste due caratteristiche e sono tutte scritte per nastro solo (acusmatiche).

Copyright: © 2014 Luca De Siena. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

1.2 Repertorio analizzato

Come specificato, le 4 opere analizzate sono caratterizzate dalla suddivisione canonica dell'*Ordinarium Missae* e la presenza del testo tradizionale. Per l'analisi del repertorio si è cominciato con lo stilare una tabella di paragone che confronta caratteristiche quali: l'anno di composizione, la durata, i canali audio, la lingua del testo, il rapporto con l'*Ordinarium* e la suddivisione in numeri (coincidente con il numero delle tracce audio). Le quattro Messe elettroacustiche analizzate e comparate sono:

1. "Messe de Liverpool" di Pierre Henry (1967)
2. "Messe aux Oiseaux" di Jacques Lejeune (1987)
3. "Messe pour le mois d'octobre" di Bernard Fort (1995)
4. "Mass Utterance" di Simon Hall (1999/2000)

Di seguito sono riportati i dati scaturiti da questo primo livello di indagine:

	Messe de Liverpool	***Messe aux oiseaux***	***Messe pour le mois d'octobre***	***Mass Utterance***
Anno	1967	1987	1995	99/2000

Durata	47:11	60:05	41:36	23:56
Canali	2	2	2 e 8	2
Lingua del testo	Latino	Latino	Latino e Francese	Latino
Rapporto con l'*Ordinarium*	K,G,C,S, A e Communion	I parte: K II parte: G,C III parte: S,A	K,G,Le symbole des Apôtres, Sanctus/ Benedictus, A	I parte: K,G II Parte: C,S, A
Suddivisione dei numeri (tracce audio)	6	21	4	1

Tabella 1. Primo livello di comparazione fra le 4 Messe elettroacustiche

La *Messe de Liverpool* di Pierre Henry è quella che possiamo considerare la prima vera *Messa* elettroacustica (acusmatica) mai scritta. Oltre a questo primato l'opera di Henry vanta anche quello dell'unica *Messa* elettroacustica ad essere stata scritta per una committenza religiosa. La *Messe de Liverpool* infatti fu commissionata per l'inaugurazione della cattedrale del Cristo Re di Liverpool e venne eseguita durante la cerimonia inaugurale tenuta il 26 Maggio 1967. In questa occasione l'opera venne però presentata senza il *Credo.*

Scritta nel 1987*, la Messe aux Oiseaux* di Jacques Lejeune è l'opera di dimensioni maggiori fra tutte quelle analizzate. Dopo 20 anni dalla *Messe de Liverpool* di Pierre Henry siamo nuovamente in presenza di una commissione per una *Messa* elettroacustica; questa volta però la committenza non è religiosa ma si tratta dello Stato francese e dell'INA-GRM. Questa *M*essa, a differenza delle altre cinque, suddivide le sezioni dell'Ordinario in ulteriori numeri che prendono il nome dai versetti del testo latino per un totale di 21 tracce audio.

La *Messe pour le mois d'Octobre* di Bernard Fort, scritta tra il Luglio del 1992 ed il Dicembre del 1995, è un'opera scritta come se si trattasse di una sorta di preghiera personale. Il compositore nelle note personali dichiara di non aver pensato a come sfruttare l'opera che ritiene probabilmente inadatta tanto al concerto quanto all'uso liturgico. Eppure nella stessa sede certifica il fatto che si tratta di un'opera profondamente inserita nello spirito del Concilio Vaticano II pertanto fa uso (anche) della lingua nazionale, il francese, e dedica il suo lavoro a San Francesco d'Assisi.

Nelle note di sala, Simon Hall ci comunica che la sua *Mass Utterance* (1999/2000) consiste nelle esplorazioni di differenti tipi di masse sonore e di enunciazioni. Il testo liturgico è utilizzato come punto di partenza ed è presente in vari modi: cantato, recitato, armonizzato da un coro, sussurrato ed infine frammentato e trasformato. Queste enunciazioni del testo vengono contrapposte ad altre emissioni vocali di tipo collettivo ed individuale quali: cori da stadio, canto gregoriano, sillabe isolate senza senso, citazioni di personaggi famosi e politici nonché emissioni non vocali come quelle dei suoni della natura, della città, del computer e del sintetizzatore.

1.3 Risultati del confronto

Dopo aver analizzato le opere singolarmente si è passati alla fase di confronto dei dati ottenuti. Mettere a confronto le diverse tecniche elettroacustiche impiegate potrebbe non restituirci dati di grande interesse perché esiste un divario tecnologico troppo grande tra un lavoro composto nel 1967 ed uno composto 20 anni dopo.

Altre caratteristiche quali invece il tipo di articolazione vocale, la durata di ogni parte e la tipologia di suoni musicali ed extramusicali utilizzati, risultano sicuramente più confrontabili.

Per prima cosa si è confrontata la durata di ogni parte dell'*Ordinarium*. Questo confronto può restituirci delle informazioni circa l'importanza che il compositore ha voluto attribuire ad ognuna delle parti e l'atteggiamento comune fra le opere.

Facendo una media delle durate si è scoperto che è il *Credo* la parte a cui i compositori hanno dato maggiore spazio. Questo dato statistico sarebbe in linea con la lunghezza del testo della preghiera che è il più lungo di tutto l'*Ordinarium* e quindi lascerebbe pensare ad una attenzione particolare dei compositori verso il testo ed il suo impiego all'interno delle opere. D'altro canto dobbiamo dire che se questo è vero per le prime due *Messe*, per le seconde due è un dato assolutamente non veritiero poiché, al di là dell'incertezza sui confini tra Kyrie e Gloria nella *Messa* di Fort e della presenza di due preghiere all'interno di uno stesso numero (*Sanctus+Benedictus* in Fort e *Pater Noster+Agnus Dei* in Hall), Il *Credo* non risulta comunque essere la parte più lunga.

Per quanto riguarda i contenuti extramusicali, al di là della *Messe de Liverpool* che ne è priva, nelle altre tre si riscontra in ciascuna la presenza del canto degli uccelli che in un'opera si sostituisce addirittura al testo ed in un'altra dà il nome alla *Messa* stessa. Si tratta di una caratteristica che merita di essere analizzata con particolare attenzione.

Le tipologie di contenuti ed interventi musicali (tradizionali) comuni a tutte le *Messe* rientrano nell'ambito del canto, sia esso monodico o polifonico. In particolare, fatta eccezione per la *Messe aux oiseaux*, è il canto monodico in latino (gregoriano) a predominare. Questa scelta potrebbe essere dettata dalla potente forza evocativa del *sound* del gregoriano che, in opere come queste, dove vi si riscontra una grande varietà timbrica e di atmosfere, potrebbe avere il ruolo di ricordare all'ascoltatore che si tratta pur sempre di una *Messa*, cioè di qualche cosa che appartiene al mondo sacro o trae ispirazione da esso.

Riguardo le tipologie di articolazione vocale se ne rileva una grande diversità che va dal testo bisbigliato fino al testo gridato passando per la salmodia. Non si riscon-

tra una particolare accordanza nella scelta delle articolazioni vocali rispetto alla parte dell'Ordinarium.

Come già evidenziato sopra, la sostituzione della voce e del testo nel *Gloria* della *Messa* di Fort con il canto degli uccelli è un qualcosa che richiede di essere analizzato da un'ottica ermeneutica.

2. TESTO LITURGICO E VOCE NELLE *MESSE* ELETTROACUSTICHE

Al di là delle varie possibilità di articolazione vocale che sono state riscontrate nelle quattro *Messe* analizzate, occorreva un modello di analisi in grado di interpretare le strategie compositive messe in atto sia sul piano del contenuto testuale quanto su quello musicale. In questo ci viene in soccorso il modello bidimensionale elaborato da Rodrigo Segnini e Bruno Ruviaro, due ricercatori del Center for Computer Research in Music and Acoustics (CCRMA) dell'Università di Stanford, nell'articolo "Analisi di lavori elettroacustici con intersezioni fra linguaggio e musica" [4]. I due autori si sono concentrati su due caratteristiche fondamentali per quanto concerne la presenza vocale in musica e cioè il livello di intelligibilità e la dicotomia fra cantato e parlato. Solitamente, musica e linguaggio vengono considerati due domini separati. Nella musica elettroacustica questo confine tra i due domini viene spesso confuso proprio attraverso l'elaborazione elettroacustica.

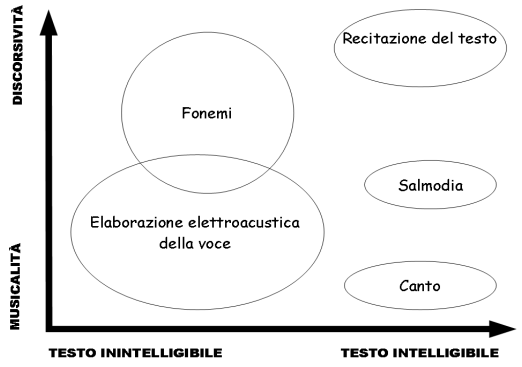


Figura 1. Il modello Ruviaro/Segnini applicato alle 4 Messe elettroacustiche prese in analisi

Il modello di analisi è stato qui presentato con le principali articolazioni della voce che sono state riscontrate all'interno delle quattro *Messe* prese in analisi.
Nonostante l'accuratezza di questo modello, esso rappresenta solo un primo livello di indagine. Denis Smalley riguardo la presenza della voce nella musica elettroacustica si esprime in questi termini:
"Nel momento in cui una voce viene percepita in un contesto sonoro l'orecchio di chi ascolta è attratto verso di essa e l'interpretazione si sposta a concentrarsi sull'invisibile presenza umana, cercando di decodificare il significato delle sue dichiarazioni e il rapporto con la persona e l'ambiente sonoro [...] *una presenza umana, dove in precedenza non vi era nessuno cambia tutto."* [5] (traduzione di chi scrive)
Nell'ambito della musica elettroacustica si può fare ricorso alla registrazione di paesaggi sonori in cui si muove una presenza umana individuabile per mezzo di una voce. Questa voce può esprimersi nel piano bidimensionale di Segnini/Ruviaro anche se la sua comprensione è ostacolata da elementi sonoro-ambientali presenti al momento della registrazione. La volontà del compositore di inserire un suono dove la comprensione del testo sia compromessa a priori, rappresenta una scelta compositiva degna di essere interpretata.
Come mostrato in *figura 1,* i contenuti vocali all'interno delle quattro *Messe* prese in considerazione si muovono dal testo recitato al canto nell'area dell'intelligibilità mentre passano dai fonemi slegati all'elaborazione elettroacustica dalla voce in un'area più vasta, ma sicuramente più sbilanciata verso il polo dell'inintelligibilità. Nell'ambito dei fonemi esiste una zona in cui l'organizzazione di questi si avvicina talvolta ad un costrutto di tipo musicale. Questa zona di confine è condivisa da un tipo di elaborazione elettroacustica della voce presente soprattutto all'interno di *Mass Utterance* di Simon Hall.

2.1 Afasia e sostituzione vocale nelle Messe elettroacustiche

In tre casi si è riscontrato un fenomeno definito come "afasia", termine mutuato dalla medicina per indicare l'incapacità di produzione del linguaggio, qui usato per indicare un'omissione della voce ove ci si aspetti la sua presenza.
Il primo caso è il *Gloria* della *Messe pour le mois d'octobre* di Bernard Fort. In questa sezione infatti troviamo solamente il canto degli uccelli a sopperire alla mancanza di qualsiasi altro contributo vocale nonché musicale.
Il compositore Bernard Fort nelle note di sala ha dichiarato che gli uccelli rappresentano per lui l'espressione della voce degli angeli. Pertanto qui non ci si trova semplicemente davanti ad un caso di afasia poiché la mancanza della voce umana nel *Gloria* è esplicitamente sostituita da quella animale degli uccelli. Non si tratta di una semplice sostituzione a fini coloristici, al contrario questa scelta attiva dei processi di significazione legati all'assenza della voce e alla presenza del canto degli uccelli in luogo di un testo con un suo significato preciso.
A proposito di questi processi di sostituzione Trevor Wishart ha scritto:
"Usando immagini sonore all'interno dello spazio virtuale degli altoparlanti, possiamo creare un mondo che stia in qualche modo a metà strada tra la concretezza della rappresentazione dell'opera e il mondo delle relazioni musicali. Non abbiamo bisogno di associare un contenuto musicale a, per esempio, un uccello e così ad un significato metaforico; potremmo usare il verso di un uccello direttamente. Così la concretezza della rappresentazione teatrale è rimpiazzata da un paesaggio sognante che fluttua tra l'articolazione musicale e gli eventi del "mondo reale" [6]

Anche nel primo numero del *Kyrie* della *Messa aux oiseaux* di Lejeune troviamo la presenza del canto degli uccelli ove ci si aspetterebbe la voce, anche se qui non si può parlare di vera e propria sostituzione vocale in quanto l'entrata in scena della voce viene solo ritardata.

A differenza di Fort però, nella *Messe aux oiseaux* il canto degli uccelli verrà ripreso ed integrato nello sviluppo formale proprio come accade in Messiaen. Il fatto che nella *Messa* di Fort il canto degli uccelli non sia stato rielaborato, non deve per forza farci giudicare negativamente questa scelta. Questa è infatti una possibilità offerta esclusivamente dalla musica elettroacustica, una caratteristica che permette una maggiore correlazione tra suono e processo interpretativo da parte dell'ascoltatore.

Per quanto riguarda invece il fenomeno dell'afasia, esso è riscontrabile nel *Pater Noster* della *Mass Utterance* di Simon Hall. In questa parte di cui non conosciamo gli effettivi confini con l'*Agnus Dei* (sono entrambe accorpate nell'ultimo numero) è solo la componente elettronica a dominare lo spazio frequenziale, mentre sullo sfondo si rileva la presenza di un *soundscape* di tipo urbano posto in lontananza in un range frequenziale medio-basso. È abbastanza singolare che il compositore abbia voluto inserire questa parte (che normalmente non rientra tra quelle che vengono musicate) e poi abbia scelto di lasciare la preghiera senza testo in un contrasto tra suoni elettronici molto taglienti (prevalentemente nel registro acuto) ed un paesaggio urbano alquanto tenue.

Un altro caso di afasia è riscontrabile nella sezione denominata *Communion* della *Messe de Liverpool* di Henry che è costituita solo da una tessitura di natura elettronica che cresce fino a metà brano per poi svuotarsi e decrescere fino al finale. Anche in questo caso ci troviamo in presenza di una parte aggiunta al classico *Ordinarium* ma lasciata priva del contributo vocale.

3. LO SPAZIO NELLE *MESSE* ELETTROACUSTICHE

Come abbiamo già visto, per Smalley la voce catalizza l'attenzione dell'ascoltatore verso se stessa nel momento in cui viene udita. La voce porta con sé delle informazioni riguardanti la persona che l'ha emessa ma anche importanti informazioni circa lo spazio che la contiene. La teoria di Smalley è supportata anche dalle parole di Michel Chion:

"La presenza della voce umana struttura lo spazio sonoro che la contiene […] *se una voce è parte di uno spazio sonoro, l'orecchio inevitabilmente è attratto da essa, la individua e struttura la percezione interamente attorno ad essa. L'orecchio prova ad analizzare il suono in moda da estrarne del senso e prova sempre a localizzarlo e, se possibile, ad identificare la voce"* [1]

Per questa ragione, essendo la *Messa* una composizione interamente costruita attorno alla voce (poiché costruita attorno ad un testo), è essenziale analizzare a fondo le componenti spaziali che questo elemento implica. Anche qui è stata necessaria la definizione di un modello analitico che permettesse lo studio dello spazio in queste quattro opere.

Lo "spazio" in un'opera elettroacustica è la risultante di tre componenti che combinandosi contribuiscono all'sperienza spaziale da parte dell'ascoltatore come mostrato nella F*igura 2:*

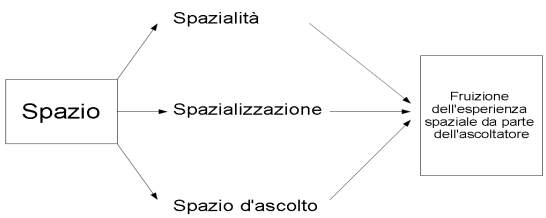


Figura 2. Componenti che concorrono alla fruizione dell'esperienza spaziale

Per "Spazialità" si intende la serie di interazione che intercorre fra i materiali audio usati (spazio frequenziale, lo spazio dato dal tipo di riverberazione etc.) prima che essi escano dagli altoparlanti per confrontarsi con l'ambiente dove la musica viene diffusa; per "Spazializzazione" tutti i movimenti dei materiali all'interno dello disposizione degli altoparlanti; infine con il termine "Spazio d'ascolto" ci si riferisce alla caratteristiche acustiche dell'ambiente in cui sono istallati gli altoparlanti. La combinazione di queste tre componenti influenza la percezione dello spazio.

La prima *Messa* acusmatica, la *Messe de Liverpool* di Pierre Henry è stata l'unica ad essere rappresentata in una chiesa per volere del clero inglese.

Sebbene sia disponibile solo una versione stereofonica di quest'opera è verosimile pensare che la sua rappresentazione sia avvenuta mediante una installazione multicanale durante la *premiere* tenutasi all'interno della cattedrale del Cristo Re di Liverpool. L'opera dimostra un'estrema parsimonia nell'uso del riverbero ed una certa tendenza a spezzare la continuità degli eventi sonori.

Le voci che enunciano il testo sono ridistribuite sul fronte sonoro ottenendo in questo mondo una maggiore intelligibilità. I suoni musicali, strumentali ed elettronici, compiono evoluzioni nello spazio esplorandone sia la profondità sia l'ampiezza. Questa povertà di elaborazione elettroacustica dei materiali è probabilmente dettata dalla necessità di riprodurre un'opera del genere all'interno di un ambiente altamente riverberante. Si tratta perciò di un tipo di scrittura elettroacustica fortemente delimitato dall'interazione spaziale che si intendeva mettere in atto al momento della sua riproduzione.

Non sono presenti nell'opera registrazioni di suoni extramusicali e quindi manca in questo senso l'evocazione di altri spazi e di conseguenza l'invito a trascendere l'ambiente d'ascolto attraverso l'associazione soggettiva di immagini sonoro-visive e simboli.

Di particolare interesse per questa trattazione è la rappresentazione della *Messe aux oiseaux* all'interno dello spazio di una chiesa (non durante una liturgia ma in forma di concerto). Lejeune ci informa che è stato necessario ridurre lo spazio d'ascolto nonché ridurre il volume generale per contenere gli effetti di riverberazione [3]. Per quanto concerne la *Messe pour le mois d'octobre* di Bernard Fort il discorso spaziale è alla base delle motivazioni che hanno spinto il compositore a scrivere la sua *Messa*. L'evocazione di altri spazi è rappresentata all'interno di questa *Messa* dalla registrazione del verso degli uccelli e di quella di una assemblea di fedeli che recita il credo all'interno di una chiesa. L'opera di Bernard Fort traccia delle traiettorie di senso proprio attraverso l'organizzazione dello spazio e degli spazi.

Si prenda in analisi il percorso circolare che l'avvicendarsi dei diversi tipi di spazio disegna lungo tratta l'arcata compositiva: si parte con uno spazio molto ridotto nel *Kyrie* in cui lo spazio ricreato è un qualcosa di ideale che nasce dal contrasto dell'intimità della voce registrata in prossimità del trasduttore con lo sfondo elettronico ad alta densità gestuale sulla quale essa si muove. Nel *Gloria* si esce verso uno spazio aperto dove c'è solo il canto degli uccelli a dominare lo spazio. Nel *Symbole des Apôtres* (*Credo*) si entra in un ambiente altamente riverberante come quello di una chiesa dove un assemblea di fedeli sta recitando (in francese) la preghiera che dà il titolo a questa sezione dell'*Ordinarium*. Tutti i crismi del suono altamente riflettente della cattedrale sono presenti. Con il *Sanctus* si torna nuovamente tra la natura (spazio aperto): un paesaggio sonoro tipicamente rurale sul quale una voce che canta una melodia gregoriana (o meglio, gallicana) si porta gradualmente in primo piano (disegnando una traiettoria) fino ad operare un brusco passaggio verso un ambiente nuovamente riverberante di una chiesa in cui vengono cantati gli ultimi versetti del *Sanctus*. L'ultima sezione, l'*Agnus,* è caratterizzata dallo stesso spazio ideale, ridotto ed intimo della prima, venendo a tracciare in questo modo un tipo di percorso circolare. Ma il discorso spaziale in quest'opera non si ferma alla sola spazialità. Il disegno analizzato poc'anzi è ricalcato ed estremizzato dalla spazializzazione, anch'essa tesa a tracciare una traiettoria compositiva circolare.Le due parti estreme (*Kyrie* e *Agnus*) sono state scritte in multicanale per un sistema ad 8 altoparlanti, quelle mediane (*Gloria* e *Sanctus*) costruendo uno spazio panoramico tramite l'unione di 4 registrazioni ambientali stereofoniche riprodotte su un sistema ad 8 vie. La parte centrale, il *Symbole des Apôtres* è invece scritta in stereofonia e riprodotta solo su due altoparlanti.

Possiamo perciò affermare che la *Messe pour le mois d'octobre* rappresenta un perfetto esempio di approccio simbolico all'organizzazione dello spazio all'interno di una composizione, un pensiero in linea con una tradizione antica. Nei secoli passati infatti non sono mancate le rappresentazioni di musiche sacre che occupavano lo spazio sacro in forma simbolica, una su tutte il *Vespro per lo Stellario della Beata Vergine* di Bonaventura Rubino (eseguito Basilica di San Francesco d'Assisi a Palermo nell'agosto del 1644) in cui dodici cori erano disposti a forma di stella per simboleggiare la corona della vergine che dà il titolo all'opera. Nella *Messa* di Bernard Fort L'idea di percorso circolare descritto dalla spazialità è, come abbiamo potuto osservare, inequivocabilmente confermato da quello organizzato attraverso la spazializzazione. Non è questa la sede per avventurarsi in questioni di semiotica, ma il fatto che un disegno circolare sia così palesemente tracciato in un'opera di ispirazione religiosa (si ricordi che il lavoro è dedicato a San Francesco d'Assisi), ci invita ad interrogarci più a fondo sulla presenza di questo simbolo.

Sull'ultima delle *Messe* analizzate, *Mass Utterance* di Simon Hall, non abbiamo notizie circa gli interventi di spazializzazione progettati o messi in atto durante la rappresentazione dell'opera. Ci limiteremo per tanto a prendere in esame il solo il parametro della spazialità.

Un simile discorso di circolarità lo ritroviamo anche nell'uso del soundscape che Simon Hall fa nel suo lavoro. *Mass Utterance* infatti si apre e si conclude con la registrazione di un paesaggio sonoro in cui risalta in particolar modo il canto degli uccelli. Il *Kyrie* presenta la voce cantata (gregoriano) sullo sfondo mentre un voce parlata è posta in prossimità dell'ascoltatore ad un volume appena percettibile.

Nel *Gloria* lo spazio diventa molto rarefatto a causa di una scrittura prettamente gestuale in contrasto con quella prettamente tessiturale della sezione precedente e di quelle successive. In questa parte però fa la sua comparsa anche la registrazione del coro da stadio che porta con sé tutte le informazioni di questa particolare immagine sonoro-visiva trasportando l'ascoltatore al di fuori della dimensione onirica creata fino ad allora, nuovamente verso uno spazio aperto**.** Il coro da stadio farà ancora delle breve incursioni durante l'arcata compositiva. Sul finale, nella transizione fra *Pater Noster* e *Agnus Dei* il compositore è ricorso all'uso di un altro *soundscape*, questa volta di matrice urbana che però è posto sullo sfondo e in gran parte coperto dall'uso dei suoni elettronici che si trovano più in primo piano.

3.1 "Lo spazio negato"

> Una domanda che nasce spontanea è cosa impedisca ad opere di questo genere, talune, come nel caso della *Messa* di Bernard Fort, così intrise di simbolismo di essere utilizzate durante la celebrazione del rito cristiano e ricongiungersi alla propria funzione originaria.

Delle quattro *Messe* analizzate sappiamo che solamente la *Messe de Liverpool* è stata commissionata da un ente religioso appositamente per essere eseguita dentro una chiesa. Ciò nonostante bisogna osservare che la *Messa* di Pierre Henry è stata eseguita nell'ambito di una cerimonia di inaugurazione di un edificio sacro, un'occasione che potremmo definire paraliturgica ma sicuramente non legata al rito inteso come celebrazione eucaristica.

Bernard Fort, infatti, in un intervento intitolato "Perché così poca musica religiosa o liturgica nella musica elettroacustico?" tenuto il 31 Gennaio 2010 nell'ambito delle «Journées de Lyon» organizzate dall'organizzazione «Universa Laus» di aria francofona, tra le possibili moti-

vazioni di questa carenza annovera la difficoltà di avere il permesso di installare gli altoparlanti nelle chiese. In Italia una delle poche esperienze di questo genere è stata ideata da Don Luigi Garbini (direttore del Laboratorio di musica contemporanea al servizio della Liturgia) che nel 2002 tenne una *Messa Elettronica* nella chiesa di San Marco a Milano in cui una postazione elettronica nascosta operava un'elaborazione in tempo reale sul suono prodotto dal coro e dall'assemblea durante la *Preghiera Eucaristica.* Nonostante ciò esperienze del genere restano isolate.

4. CONCLUSIONI

Come spiegato nell'introduzione, le domande che avevano motivato questa ricerca erano: 1. *Perché si sceglie di scrivere una Messa elettroacustica? 2. Quali e quanti sono i legami con la tradizione? 3. Esistono elementi in comune tra questi nuovi lavori? 4. È possibile pensare ad un loro utilizzo liturgico?*

1. Da quanto emerso da questa dissertazione, si scrive una *Messa elettroacustica* per 3 motivazioni principali: a. Su commissione: la commissione religiosa è rara se non inesistente. Nelle opere analizzate possiamo annoverare solo la *Messe de Liverpool* di Pierre Henry tra le opere commissionate dalla Chiesa. Anche la *Messe aux oiseaux* di Jacques Lejeune è stata composta dietro commissione. Non si tratta di una committenza religiosa, bensì laica, lo Stato francese. b. Come esercizio di stile: come nel caso di Jacques Lejeune. In molti dei brani censiti si è visto come il legame con la *Messa* sia rimasto solo nel titolo e/o nella suddivisione formale, ricorrendo ad altri testi poetici. Ma a che pro viene conservata una struttura prestabilita dove in alcuni casi i contenuti sono poi distanti in maniera siderale da quelli originali della *Messa*? Probabilmente questo ha a che fare con una crisi del "rito" all'interno della nostra società occidentale che ne è sempre più priva. Ma esiste una questione etica circa l'utilizzazione di un testo sacro in un'opera che ha finalità extra-liturgiche? Probabilmente no, o meglio, in questo caso non più, dopo che la tradizione musicale dell'occidente lo ha utilizzato così frequentemente per secoli anche in opere non destinate alla celebrazione del culto. c. Per motivazioni personali/religiose: è il caso della *Messe pour le mois d'Octobre* di Bernard Fort che lo stesso compositore non ha mai pubblicato ed ha eseguito poche volte in concerto. Nonostante ciò si tratta di un'opera intrisa di simbolismo e riferimenti alla tradizione che fanno della *Messe* di Fort il componimento più vicino al culto cristiano tra quelli qui analizzati. Un'opera come questa è sicuramente frutto della fede del compositore francese tanto più che è espressamente dedicata a San Francesco d'Assisi.
2. La questione, posta in termini funzionali, ci porta a considerare la *Messe de Liverpool* molto più vicina ad un modo di scrivere musica sacra che veniva adottato nel Medioevo/Rinascimento rispetto alle altre tre *Messe* analizzate. Nella Messe di Henry vi è, oltre ad un uso funzionale ben determinato a priori, l'assenza di un figuralismo di tipo musicale. La musica è totalmente in funzione del testo ed è al testo che è affidata al massimo la funzione di suggerire immagini extra-musicali. Le altre tre *Messe* sono in linea con un modo di scrivere musica sacra che prende piede dopo il Rinascimento. Si tratta di un approccio in cui la musica rafforza il significato del testo mediante delle metafore sonore.
3. Per quanto concerne gli elementi di comunanza tra i lavori, fatta eccezione per la *Messe de Liverpool*, si può parlare sicuramente di un'evoluzione del figuralismo in simbolismo. La presenza del canto degli uccelli in tutti e tre i lavori, in particolare, non va interpretata come un caso. È forse lecito chiedersi se al, pari dei madrigalismi, qui ci troviamo in presenza di un elemento tipico della scrittura di musica elettroacustica che prende spunto dal testo e fa uso di immagini extra-musicali e elaborazione spaziale per potenziarne ulteriormente il contenuto in termini emozionali.
4. Dalle parole di Don Luigi Garbini [2] si è appreso come una funzionalità liturgica di un'opera elettroacustica non può essere predeterminata ma va sperimentata sul campo. La creazione e la fruizione di *Messe* elettroacustiche in una funzione religiosa è un compito che solo la sinergia di Chiesa, musicisti e assemblea dei fedeli può realizzare, sperimentando sul campo, quello spesso acusticamente ostico delle navate delle cattedrali, soluzioni equilibrate tra nuovi linguaggi ed effettiva funzionalità. Le prime incursioni della *Messa* nel campo elettroacustico (o della musica elettroacustica nel campo della *Messa*) ci restituiscono delle preziose esperienze di cui il compositore dovrà assolutamente tenere conto nel caso voglia misurarsi con questo tipo di composizione.

5. BIBLIOGRAFIA

[1] M. Chion: *The Voice in Cinema,* Columbia University Press, New York, 1998.

[2] L. Garbini: *Breve storia della musica sacra*, Il Saggiatore, Milano, 2005.

[3] J. LeJeune: "La forme dans le paysage (II)"*, L'Espace du Son, Volume 2*, pp.79-83, 1991/2008.

[4] R. Segnini, B. Ruviaro: "Analysis of Electroacoustic Works with Music and Language Intersections"*, Proceedings of ICMC*, Barcelona, 2005.

[5] D. Smalley: "The listening imagination: Listening in the Electroacoustic Era", *Contemporary Musical Thought*, Volume 1, London, Routledge, pp. 541, 1992.

[6] T. Wishart: "Sound Symbols and Landscape"*, The Language of Electroacoustic Music*, Macmillan, pp. 41-60, 1987.

MUSICA (PRE) INFORMATICA NEGLI STATI UNITI

Alex Di Nunzio
Ricercatore indipendente
alex.dinunzio@gmail.com

SOMMARIO

Quest'articolo ripercorre le esperienze pionieristiche di computer music condotte negli Stati Uniti durante gli anni Cinquanta del XX secolo. L'attenzione è posta sugli esperimenti che appartengono alla categoria della composizione musicale assistita dal computer, tralasciando quelli sulla sintesi digitale dei suoni che pure si svolsero negli stessi anni. Nel corso di questa rassegna gli esperimenti presentati sono divisi in due categorie: quelli informatici, cioè che fecero uso del computer per la composizione delle musiche (Hiller, Klein e Bolitho, Brooks) e quelli pre informatici (Pierce, Pinkerton, Sowa), che non fecero uso di tecnologie digitali. Inoltre sono esposti i risultati musicali raggiunti, se ve ne furono, e le metodologie di lavoro adottate.

1. INTRODUZIONE

Il suggerimento offerto da questo ventesimo Colloquio, di rivolgere uno sguardo al passato, induce a riscoprire anche le pionieristiche attività condotte negli Stati Uniti durante il quinto decennio del XX secolo. Quando si legge del percorso che la musica ha svolto intrecciandosi con l'esperienza informatica, la questione storica è declinata in due differenti approcci, entrambi con origine negli Stati Uniti e nel medesimo periodo storico: quello della sintesi digitale dei suoni, che si espresse innanzitutto nelle sperimentazioni di Max Mathews, e quello della composizione musicale assistita dal computer. Questo secondo approccio ebbe un suo primo apice con la composizione della *Illiac Suite* di Lejaren Hiller. Tuttavia, pur rappresentando un importante traguardo nell'ambito della computer music delle origini, in questo articolo ho intenzione di soffermarmi maggiormente su quelle attività considerate marginali, vuoi per la qualità degli esiti musicali, dove ve ne furono, vuoi per la mancanza di continuità nel tempo di queste ricerche. Si tratta di esperimenti già noti nella letteratura di riferimento, sui quali il presente scritto svolge più che altro una rassegna, cercando di evidenziare i risultati ottenuti e la metodologia che in ciascuno di essi fu utilizzata per il raggiungimento degli scopi prefissati. La struttura dell'articolo prevede una prima sezione introduttiva in cui si accenna all'ambiente culturale nel quale nascono queste attività di ricerca. Seguono sei sezioni ognuna delle quali dedicata ad un diverso esperimento, presentati secondo un ordine che è cronologico, ma con un'eccezione per quanto riguarda la disposizione tra la ricerca di Richard Pinkerton e quella di John Pierce.

Copyright: © 2014 Alex Di Nunzio. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

2. UN AMBIENTE IDEALE

All'indomani della seconda guerra mondiale, gli Stati Uniti presentavano un contesto ideale per lo svolgimento di ricerche finalizzate a vagliare se, e come, il computer si potesse utilizzare in ambiti non scientifici, ai quali anche la musica appartiene. L'ambiente era ideale sia per lo sviluppo raggiunto dalla tecnologia digitale, sia grazie ad un ambiente culturale più favorevole di quello europeo. Questo secondo aspetto è quello di maggiore importanza per i successivi sviluppi musicali della tecnologia informatica. In effetti, un passato musicale meno condizionante di quello europeo, e l'elaborazione di alcune teorie fondamentali, contribuirono a generare interesse in figure professionali molto diverse tra loro ma accomunate dal voler verificare in che maniera due ambienti così distanti, quello creativo della musica e quello ingegneristico/matematico dell'informatica, si potessero legare insieme. Va considerato, come il titolo suggerisce, che alcune pionieristiche esperienze di computer music, pur essendosi sviluppate a ridosso di riferimenti teorici fondamentali per le future ricerche informatiche, non si avvalsero per niente della tecnologia digitale, e per questo sono da classificare come pre informatiche. È altrettanto vero che i punti di collegamento tra queste diverse esperienze sono tali da giustificare la loro trattazione in uno stesso articolo. Ad esempio vedremo che la ricerca pre informatica di Richard Pinkerton, offrì un importante modello di riferimento per la maggior parte delle successive ricerche che fecero uso del computer. Il filo principale che lega tutte queste esperienze, prescindendo dalla loro categoria di appartenenza, è quello della *Teoria dell'Informazione* così come formulata da Claude Shannon nel 1948 [12]. Questa s'interessa di trasmissione delle informazioni, nello specifico si occupa di valutare in che maniera un sistema di comunicazione è in grado di ricevere, elaborare, immagazzinare e trasmettere informazioni. In che modo era possibile trasferire in ambito musicale una teoria nata in un contesto così diverso da quello artistico?

3. UNA BANAL TUNE-MAKER

Una prima risposta al precedente quesito arriva da un ingegnere, Richard Pinkerton, il quale raccolse l'eredità

della teoria di Shannon e, circa dieci anni dopo, pubblica sulla popolare rivista *Scientific American*, un articolo dal titolo inequivocabile: *Information Theory and Melody*. Pinkerton, presupponendo la possibilità di applicare in musica i concetti sviluppati da Shannon, espone un metodo affatto complesso che avrebbe consentito a chiunque, anche privo di una specifica formazione musicale, di comporre delle originali melodie, semplicemente avvalendosi di una tecnica basata sulla selezione casuale di elementi musicali. Il punto di partenza della ricerca di Pinkerton fu di considerare la musica come una qualsiasi altra forma di comunicazione, riducendola così ad un problema affrontabile in termini matematici, coerentemente con il modello teorico elaborato da Claude Shannon. Il punto di arrivo, invece, doveva essere la composizione di canzoni per l'infanzia. Tra i due estremi si collocano le diverse fasi necessarie alla buona riuscita dell'esperimento. La prima fase era quella analitica, la quale, dato un certo repertorio di riferimento, avrebbe dovuto restituire un sistema di informazioni utili per la successiva fase di sintesi, o di composizione. Il repertorio scelto da Pinkerton, su cui svolgere l'analisi, era costituito da 39 canzoni, attinte dal volume *The Golden Song Book*, una raccolta di canzoncine per bambini pubblicata nel 1945 da Catherine Tyler Wessells.

Pinkerton uniformò armonicamente i 39 brani alla tonalità di Do, circoscrivendole nell'ambito di una sola ottava. Egli mirava ad estrapolare dal suo campione la frequenza con cui ciascuna nota si presentava all'interno di ogni melodia, così da poter calcolare quale fosse la probabilità che si potesse presentare una certa successione di note; ad esempio se fosse più probabile che dopo il Do ci fosse il Mi invece di un Fa, di un Re, e così via. Attraverso i dati ottenuti, annotati in apposite tabelle, Pinkerton ottenne un grafico che mostrava, data una nota iniziale, quali fossero le possibilità di movimento nel passaggio da una all'altra. Per ragioni d'esperimento, il ricercatore fu costretto ad un processo di semplificazione, tale che le scelte esprimibili per mezzo di una semplice moneta fossero sempre e soltanto tra due possibilità. Attraverso una reiterata applicazione di questo processo di selezione casuale, da svolgere per ogni nodo presente sul grafico in figura 1, Pinkerton fu in grado di comporre intere melodie originali, di cui in figura 2 è riportato un esempio. Pinkerton chiamò la propria invenzione Banal Tune-Maker, *generatore di semplici melodie*, di cui l'autore evidenzia un importante limite: le melodie ottenute erano troppo simili tra di loro; ma del resto si trattava di un problema difficilmente evitabile visto il numero assai limitato di scelte da poter compiere in prossimità di ogni nodo. Tuttavia è vero, al di là degli intenti e dei risultati, che l'importanza della Banal Tune-Maker era soprattutto nel modello che offriva a coloro interessati alle medesime applicazioni del computer; infatti, benché la sua ricerca non ebbe sviluppi successivi e le melodie ottenute non furono mai eseguite o tantomeno registrate, tutto questo non impedì all'articolo di Pinkerton di generare delle interessanti conseguenze. Non solo. Benché il suo esperimento appartenga alla categoria dei pre informatici, l'autore stesso suggerì che estendendo il processo di selezione anche ad altri elementi musicali, all'armonia ad esempio, e svolgendo il processo di selezione con un computer, al posto della moneta, ben altri risultati si sarebbero potuti raggiungere.

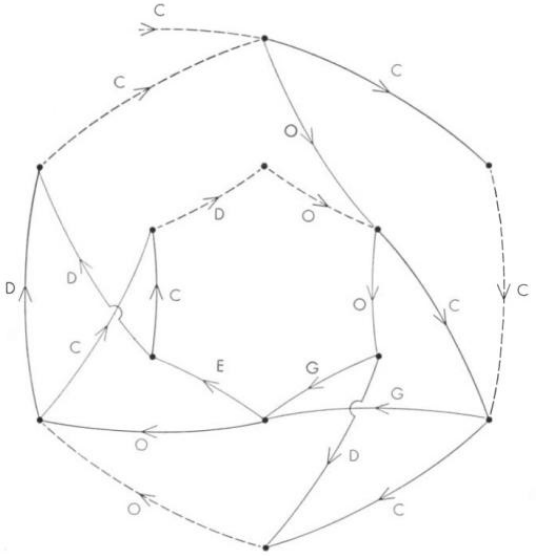


Figura 1. L'immagine, tratta dall'articolo di Pinkerton, mostra lo schema realizzato dall'autore per rappresentare graficamente la probabilità dei collegamenti tra le note. Ogni punto rappresenta uno snodo, dove la scelta rispetto all'una o all'altra possibilità era effettuata in maniera casuale attraverso il semplice lancio di una moneta.

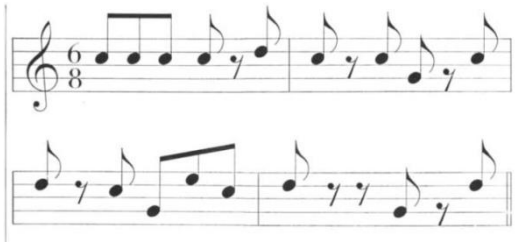


Figura 2. Una breve melodia ottenuta con la Banal Tune-Maker da Richard Pinkerton.

4. JOHN PIERCE, PRECURSORE?

Anche Lejaren Hiller, a sua volta un pioniere della computer music, ebbe modo di rilevare l'importanza dell'esperimento condotto da Pinkerton, evidenziando in particolare le numerose reazioni che si registrarono all'indomani della pubblicazione del suo articolo [8]. Hiller si sofferma in particolare su una lettera all'editore pubblicata nel numero di Aprile di *Scientific American*. Il mittente è John Pierce, che da lì a poco avrebbe contribuito alle sperimentazioni sulla sintesi digitale condotte da Max Mathews ai laboratori Bell dal 1957. La lettera di Pierce tratta ovviamente di un'altra questione. In effetti, vi menziona un esperimento di musica stocastica da lui condotto diversi anni prima e già presentato nel Novembre del 1950 con lo pseudonimo J. J. Coupling su *Astounding Science Fiction*, una rivista americana di fanta-

scienza.[1] Di questo esperimento, che condivide con quello di Pinkerton il riferimento alla Teoria dell'Informazione, Pierce ne racconta anche in una successiva pubblicazione, ribadendo di averlo realizzato nel 1949 insieme con Mary Elizabeth "Betty" Shannon, moglie di Claude Shannon [10]. Alla Teoria dell'Informazione di quest'ultimo, Pierce affianca un altro modello teorico, quello desunto dal *Musikalisches Würfelspiel* di Mozart. Ciò in cui differisce rispetto a quello di Pinkerton è nella metodologia. È assente, infatti, la fase analitica di un repertorio di riferimento, sostituita da un'arbitraria definizione di accordi costruibili sul primo grado della scala di Do. Realizzata questa prima tabella, tutte le altre - per i successivi gradi della scala - furono ottenute dal modello iniziale. Tutto ciò rappresentava l'insieme dei dati da cui, mediante l'uso di tre particolari dadi e di un'altra tabella di numeri, era possibile selezionare gli elementi utili alla composizione di musiche originali. La novità della ricerca di Pierce e Shannon, che ritroveremo anche nell'esperimento della *Illiac Suite*, era il sistema di regole da adottare per imporre dei limiti alla fase di composizione: (1) la lunghezza di ogni brano non doveva superare le 8 battute; (2) accordi formati da massimo 2 note; (3) almeno una delle due note doveva essere contenuta nell'accordo precedente; (4) La quinta e la sesta battuta dovevano essere una ripetizione della prima e della seconda. Grazie a questo metodo Pierce compose un certo numero di piccoli brani musicali con testo, quest'ultimo imposto senza l'adozione di procedure stocastiche; in figura 3 ne abbiamo un esempio. Tuttavia anche in questo caso il grosso problema era l'eccessiva ripetitività delle diverse melodie. Inoltre, come per Pinkerton, nessuna di queste fu mai eseguita e registrata, confinando questo esperimento dentro i limiti di un rendiconto teorico, allargando così l'insieme delle ricerche pre informatiche.[2]

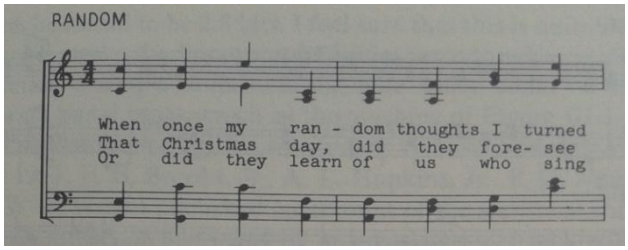


Figura 3. Uno dei tanti brani musicali ottenuti da John Pierce attraverso l'applicazione di alcune regole arbitrariamente impostate.

5. LE MELODIE DEL GENIUS ALMOST-AUTOMATIC COMPUTER

Una migliore comprensione del peso che l'articolo di Pinkerton ebbe sulla comunità americana, anche tra i semplici appassionati, e dell'attenzione che vi era nei confronti del progresso tecnologico, ci è possibile attraverso la storia di un altro esperimento, condotto questa volta non da un affermato ingegnere elettronico come John Pierce,[3] ma da un quasi anonimo sedicenne di Pittsburgh che, nel 1956, riuscì a concretizzare la Banal Tune-Maker di Pinkerton. John Sowa, questo il suo nome, progettò una macchina per comporre melodie musicali, avvalendosi del GENIAC, acronimo di *Genius Almost-Automatic Computer*; un dispositivo che con un computer condivideva soltanto una parte del nome. Il GENIAC, infatti, era una macchina analogica venduta per pochi dollari come un giochino in un kit con istruzioni di montaggio, che prevedeva: circuiti elettrici forati, rettangolari e circolari, sei switch elettrici, luci, cavi e batterie di alimentazione.[4] Cosa era possibile realizzare con un GENIAC? Non molto, piccoli esperimenti come l'accensione automatizzata delle luci oppure progetti fantasiosi quanto inutili come la macchina per le due mogli gelose [6].



Figura 4. Il kit di costruzione del GENIAC. Si nota sullo sfondo un grande circuito rettangolare e quattro circolari. L'uso di questi particolari circuiti consentiva di risparmiare sul numero di switch da utilizzare [6].

Nel 1956, il contest organizzato per promuovere la vendita del GENIAC fu vinto dal sedicenne di Pittsburgh, il quale presentò un progetto che concretizzava la Banal Tune-Maker teorizzata da Pinkerton. Avendo come riferimento il modello di quest'ultimo, Sowa impostò una metodologia del tutto simile che prevedeva, come punto di partenza, una fase di analisi. Questa fu svolta su un campione di melodie in 4/4 estrapolate da manuali per la didattica del pianoforte, private di alterazioni e livellate armonicamente sulla tonalità di Do. Per esigenze tecniche, Sowa limitò l'indagine alle prime cinque note della scala. Al termine di questa fase ottenne quattro tabelle che mostravano, similmente a Pinkerton, la probabilità delle transizioni da una nota all'altra, ottenendo così un grafico (figura 5) quasi identico a quello di Pinkerton mostrato in figura 1. Anche Sowa ridusse le scelte possibili a due, da poter selezionare mediante due metodi casuali:

[1] J. J. Coupling, "Science for Art's Sake," *Astounding Science Fiction*, Novembre 1950, pp. 83 – 92.

[2] Un esempio audio di questo lavoro, con il titolo *Music by Chance*, compare nell'album *Music From Mathematics*, prodotto dai laboratori Bell nel 1960, da non confondere con l'omonima raccolta pubblicata nel 1962 dalla DECCA. Lo si può ascoltare per intero al seguente indirizzo https://www.youtube.com/watch?v=6Ep6kXOha0U.

[3] John Robinson Pierce (1910 –2002) ha svolto importanti lavori in particolar nell'ambito delle comunicazioni satellitari con la costruzione del primo satellite commerciale, Telstar 1, contribuendo inoltre alla realizzazione del primo transistor, realizzato ai laboratori Bell nel 1948.

[4] Il progetto GENIAC era stato ideato da Edmund Berkeley, un informatico, che insieme a Oliver Garfield ne seguì l'evoluzione dal 1955 al 1958. Successivamente il due si separarono e Garfield portò avanti in solitario il progetto GENIAC per altri due anni, fino al 1960.

(1) attraverso il lancio di una moneta oppure (2) avvalendosi della conversione numerica da decimale a binario. In quest'ultimo caso, dato il numero 67, ad esempio, si otteneva il codice binario 1000011 che corrispondeva alla sequenza di una testa, quattro croci e altre due teste [13].
Al di là del metodo utilizzato, le sequenze di testa/croce o di 1/0, consentivano il funzionamento del GENIAC attraverso l'azione su sei switch, quindi attivando di volta in volta una differente linea di circuito, capace di far accendere soltanto una delle sei lampadine, una per ogni nota (Do, Re, Mi, Fa, Sol) con l'ultima che poteva indicare una pausa o un prolungamento di durata. Trascrivendo su un pentagramma le sequenze di luci, Sowa compose diverse melodie le quali soffrivano, come nei casi precedenti, di una forte ripetitività, un minimo attenuata da un lavoro di trasposizione su altre tonalità.

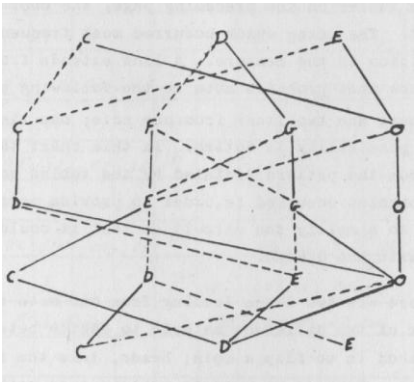


Figura 5. Il grafico delle transizioni ottenuto da John Sowa, elaborato anche questo sul modello di quello di Richard Pinkerton.

Anche le melodie di questo esperimento pre informatico non furono eseguite, né registrate; ma a differenza delle altre, per iniziativa dei costruttori del GENIAC, il lavoro di Sowa ricevette una buona visibilità, grazie alla pubblicazione di inserti pubblicitari su importati riviste dell'epoca, tra le quali anche *Scientific American*.

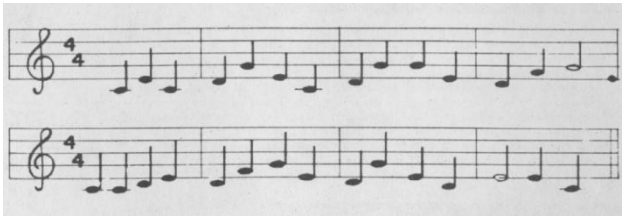


Figura 6. Una delle melodie composte da John Sowa con il suo particolare macchinario.

6. LA ILLIAC SUITE

A questo primo gruppo di esperimenti realizzati attraverso procedure non informatiche, si contrappone quello costituito da progetti di ricerca che avevano lo scopo di utilizzare il computer in funzione di assistente per la composizione di partiture originali, da eseguire con strumenti tradizionali. Il primo esperimento di questa categoria, per cronologia e importanza, è quello realizzato nel 1956 da Lejaren Hiller, in collaborazione con Leonard Isaacson, all'Università dell'Illinois; i cui risultati furono pubblicati tre anni dopo nel libro *Experimental Music*, che illustra dettagliatamente le diverse fasi attraverso cui fu composto il quartetto d'archi *Illiac Suite*. Quest'opera, di cui in appendice al testo di Hiller fu pubblicata l'intera partitura e di cui sono disponibili delle registrazioni, è considerata la prima opera musicale composta con un computer, frutto di una ricerca che lo spazio limitato di quest'articolo non consente di approfondire quanto sarebbe necessario.[5]
Ciononostante, vanno almeno sintetizzati alcuni importanti punti della metodologia adottata dai due ricercatori che, va detto, non era troppo dissimile da quella improntata da Pinkerton e Pierce. A differenza di quest'ultimo, Hiller scelse di svolgere l'analisi su un campione di opere già date, ma similmente a lui, e a differenza di Pinkerton, l'analisi non avrebbe dovuto restituire solo delle tabelle di probabilità, ma dei gruppi di regole che, opportunamente tradotte in programmazione, avrebbero fissato i limiti alla successiva fase di sintesi. La definizione delle regole rappresentava la parte più complessa e interessante di tutto l'esperimento. Innanzitutto i due ricercatori partirono con la premessa che era possibile assumere regole da diverse fonti: a) dalla tradizionale teoria musicale (ad esempio divieto delle quinte e ottave parallele); b) dettate dalla fantasia del programmatore, quindi prive di rapporto con la musica; c) ricavate da analisi statistiche di altre composizioni; d) regole auto-generate dal computer. Una volta definito, il sistema di regole era utilizzato per validare, o no, le selezioni del materiale musicale che il computer, in virtù di un particolare algoritmo, eseguiva rispetto ad un insieme potenzialmente illimitato di possibilità.[6]
Volendo banalizzare, il computer fu utilizzato come strumento in sostituzione dei dadi, delle monete e delle conversioni binarie adoperate nei precedenti esperimenti.
Questa metodologia, con importanti varianti, fu utilizzata in quattro esperimenti diversi, ciascuno corrispondente ad uno dei 4 movimenti della partitura finale che, viste le differenze apportate, si presentava ciascuno con delle caratteristiche marcatamente diverse. La prima presentazione della *Illiac Suite* vi fu il 9 agosto del 1956 all'Università dell'Illinois, in una versione in tre movimenti eseguita da un ensemble di studenti dell'Università.
Scrivendo del risultato finale della *Illiac Suite*, per altro afflitta da una certa complessità d'esecuzione,[7] Hiller sottolineò come ogni movimento si caratterizzava con uno stile particolare, anche molto vicino alla scrittura contemporanea, come nel caso del quarto. Nonostante le problematiche, la *Illiac Suite* rappresenta, sul piano estetico,

[5] La *Illiac Suite* fu edita in diversi album, spesso soltanto come estratto. Esecuzioni intere si possono trovare in *Computer Music From The University Of Illinois* (1967), LP della MGM Records e nel CD *Computer Music Retrospective* (1986) della Wergo.

[6] L'algoritmo utilizzato da Hiller è noto con il nome di Monte Carlo, il quale sfrutta la generazione di numeri casuali.

[7] Una curiosità: Hiller dovette attendere qualche anno prima di poter ascoltare un'esecuzione della sua *Illiac Suite* da parte di un ensemble di professionisti, e questo fu grazie a Max Mathews, un'altra figura di riferimento della computer music, più volte ricordato come il padre della musica informatica. Mathews e i laboratori Bell sponsorizzarono un'esecuzione della Illiac Suite da parte del WQXR String Quartet. Un estratto di questa esecuzione fu poi pubblicato nell'edizione 1960 di *Music From Mathematics*.

il punto più alto raggiunto tra tutte gli esperimenti (pre) informatici degli anni Cinquanta. Non meno importante, ai fini del peso storico della *Illiac Suite*, sarà anche la continuità che questa rara, ma non unica, figura di chimico-compositore diede alla propria ricerca nell'ambito della computer music, come dimostrano le diverse opere composte negli anni successivi, che lo avrebbero portato a sperimentare anche la sintesi digitale dei suoni. Del resto va rilevato che la *Illiac Suite* rappresentò anche per Hiller il momento di svolta della propria esistenza, poiché solo all'indomani di questa esperienza il compositore americano decise di abbandonare la professione di chimico per dedicarsi esclusivamente alla carriera musica.[8]

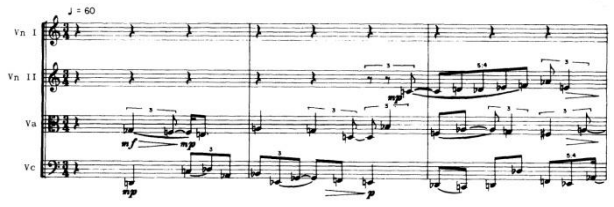


Figura 7. L'immagine mostra le prime tre battute del terzo movimento della *Illiac Suite*.

7. PUSH BUTTON BERTHA

Il concetto di continuità, invece, è assente nell'esperimento condotto da due matematici, Martin Klein e Douglas Bolitho, nello stesso anno della *Illiac Suite*. Benché ideato per scopi assai diversi, questo esperimento conobbe una certa fama, anche piuttosto duratura.[9] Perché questo successo? Semplicemente marketing. In effetti, questo esperimento nasce su richiesta della Burroughs Corporation, la quale nel 1956 aveva ufficializzato l'acquisizione della Data Electronic Corporation, azienda nota per la costruzione dei computer Datatron. Al fine di aumentare la propria visibilità, la Burroughs spinse per la realizzazione di un progetto che avrebbe dovuto stimolare la curiosità della stampa e quindi del grande pubblico. I due matematici, chiamati a dare una risposta alle esigenze dell'azienda, ritennero che un'idea vincente potesse essere quella di un computer capace di comporre musica nello stile di *Tin Pan Alley*. L'intuizione di affidare la scrittura del testo a uno dei cantautori più amati dell'epoca, Jack Owens, contribuì a rendere la forza comunicativa dell'esperimento più incisiva. Klein e Bolitho impostarono una metodologia che prevedeva innanzitutto una fase di analisi, da svolgere su un campione di musiche tratte dal repertorio classico e pop. Nel primo caso il riferimento furono alcune composizioni di Mozart, nel secondo si esaminarono i brani presenti nella Top Ten americana del 1956. Il risultato di questa fase furono 6 regole, 3 per ciascun repertorio, che avrebbero consentito la verifica e la validazione delle selezioni effettuate dal computer, secondo una procedura simile a quella adottata da Hiller e Isaacson. Il 3 luglio di quell'anno, Klein e Bolitho, e il computer Datatron, composero la partitura di Push Button Bertha.[10] Nelle settimane successive, il 9 agosto per essere precisi, la Burroughs fece eseguire questo brano all'interno del programma televisivo *Adventure Tomorrow*, compreso nel palinsesto della KABC-TV di Los Angeles. Poi furono realizzate delle registrazioni che innescarono un problema di difficile soluzione per la U. S. Library Congress, che si occupa di disciplinare il diritto d'autore negli Stati Uniti. La difficoltà per quest'ultima, assodata e riconosciuta la paternità del testo a Jack Owens, consisteva nello stabilire di chi fossero i diritti delle musiche: dei matematici Klein e Bolitho? Del computer Datatron oppure della Burroughs proprietaria del computer? Una questione rimasta irrisolta, come invendute restarono le prime cinque copie realizzate nella settimana successiva alla fine della sperimentazione.

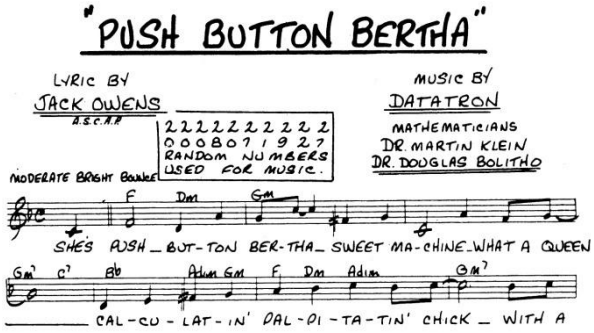


Figura 8. Le prime sei battute di Push Button Bertha, con il testo di Jack Owens. Si osservi l'attribuzione della musica al computer Datatron.

8. L'ESPERIMENTO DI HARVARD

Molto inferiore, invece, fu la visibilità pubblica che ottenne l'esperimento, non meno importante, condotto nel 1957 da un gruppo di lavoro composto da tre ricercatori del Computation Laboratories di Harvard e da un membro del settore sviluppo dell'International Business Machines Corporation. Anche questo esperimento prendeva spunto dalla Teoria dell'Informazione ed attingeva ad una metodologia molto simile a quella utilizzata da Lejaren Hiller, del quale recupera anche una certa complessità,

[8] Bisogna precisare che Hiller aveva una solida formazione musicale. Durante gli anni Quaranta era stato allievo di Roger Sessions, suo docente di composizione, poi di Milton Babbit, per la cattedra di contrappunto e composizione. Dopo aver compiuto questi primi studi, Hiller abbandonò la musica per poi riprenderne lo studio negli anni Cinquanta, insieme a Hubert Kessler, allievo di Heinrich Schenker, portando a termine il percorso con la laurea in musica nel 1958.

[9] L'attenzione suscitata dal lavoro di Klein e Bolitho ci è testimoniato dai tanti quotidiani che dedicarono spazio all'sperimento, e non faccio riferimento solo a testate come il New York Times, che scrisse del risultato ottenuto in un articolo del 3 luglio del 1956, ma anche a quotidiani molto meno blasonati come Toledo Blade o il The Milwaukee Sentinel, tanto per fare qualche esempio. Per quanto riguarda oggi, invece, è sufficiente una semplice ricerca su internet per visualizzare quanto è stato scritto sulla rete a proposito di Push Button Bertha. Io stesso ho scritto un articolo dedicato alla ricerca di Klein e Bolitho, a cui rimando per un maggiore approfondimento:
http://www.musicainformatica.it/argomenti/push-button-bertha.php. Su you tube, inoltre, ho caricato un esempio sonoro di Push Button Bertha ottenuto in maniera automatica dopo averne trascritto la partitura con il software notazionale MuseScore:
https://www.youtube.com/watch?v=V-XZKS4BItI.

[10] Il titolo fa riferimento ai bottoni del computer Datatron che, con una certa attenzione verso la comunicazione commerciale, fu chiamato amichevolmente Bertha.

ma se ne differisce per l'assenza di risultati musicali concreti e la mancanza di continuità nel tempo. Secondo i ricercatori, il problema principale nell'applicare i modelli della Teoria dell'Informazione alla musica era nella grande quantità di variabili di cui la musica era costituita (frequenze, intensità, durata, metro, ritmo, tonalità, dinamica, armonia e qualità del brano), questione che anche Hiller aveva già posto. Per semplificare il lavoro di composizione decisero che variabili come l'armonia e la qualità non si sarebbero presi in considerazione, mentre il tempo sarebbe stato arbitrariamente fisso. Come campioni furono selezionati 37 melodie religiose che utilizzavano un'unica struttura metrica. La scelta di rivolgersi al repertorio religioso era motivata dall'idea che in questo modo avrebbero potuto abbracciare epoche ed autori molto differenti [3]. Tutti i 37 inni furono selezionati a partire dall'ultimo quarto di una misura in 4/4, e nessuno di questi inni conteneva note più brevi di una croma. Il computer fu programmato per gestire le altezze comprese in un ambito di quattro ottave e attraverso delle complesse tabelle ottenute al termine della fase analitica, fu programmato per la composizione di brani originali. La fase di composizione, o di sintesi come definita nella ricerca, fu realizzata applicando dei processi stocastici basati su catene di Marcov e implementando un particolare algoritmo capace di non essere ripetitivo prima di $(10^8+1)/17$ numeri, così come specificato dagli autori [3]. Attraverso questo sistema, i ricercatori sintetizzarono circa 6000 melodie, delle quali i processi di validazione programmati per il computer ne selezionarono soltanto 600. Nessuna di queste fu mai eseguita con strumenti di alcun genere, e in seguito venne meno anche quella continuità cui ho accennato più volte [4].

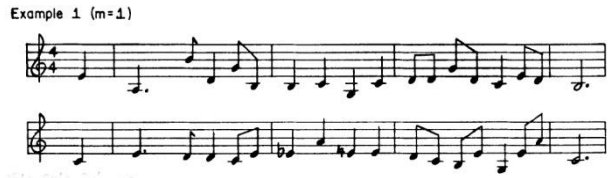


Figura 9. Una delle 600 melodie ottenute dai ricercatori di Harvard al termine del loro esperimento.

9. CONCLUSIONI

Dopo aver passato in rassegna queste pionieristiche esperienze, credo sia chiaro che il valore musicale dei risultati ottenuti da questi esperimenti, almeno sul piano estetico, ad eccezione della *Illiac Suite* per la quale sarebbe necessario un discorso a parte, è piuttosto irrilevante per la storia della musica. E non è un fatto nuovo, poiché già i primi esperimenti di musica concreta, o i primi lavori realizzati nello Studio di Colonia, ebbero la stessa sorte di queste esperienze di musica informatica. E per quanto riguarda questi ultimi, non poteva essere diversamente per delle attività in cui l'attenzione è spostata maggiormente verso l'aspetto tecnologico (Klein e Bolitho, Brooks) e quello metodologico (Pinkerton, Sowa, Pierce); al punto che per l'esperimento di *Push Button Bertha* e di Harvard, l'attenzione sul computer è tale che sarebbe più opportuno parlare di informatica sperimentale piuttosto che esperimenti di musica. Tuttavia questo non toglie nulla al valore storico di queste ricerche, che testimoniano un clima d'interesse assai diffuso per il nuovo strumento e confermano quel grande legame tra scienza e musica che è un tema importantissimo del Novecento musicale. Discorso diverso, dicevo, per la *Illiac Suite* di Hiller, che per quanto sia nata anch'essa da un orientamento sullo strumento, risente positivamente della maggiore sensibilità e consapevolezza musicale dell'autore. L'attenzione verso queste attività, quindi, nasce da un'esigenza di ripensare il passato a vantaggio di una migliore comprensione del presente e di quanto dovrà ancora avvenire in futuro, per questo ho ritenuto utile riordinare una storia di fatti che in ogni caso rappresentano il punto di origine di un cambiamento ancora in atto.

BIBLIOGRAFIA

[1] Anonimo, "Brain' Computes New Tune for TV," *New York Times*, Luglio 1956, p. 51.

[2] C. Ames, "Automated Composition in Retrospect: 1956 – 1986," *Leonardo*, Vol. 20 [2], 1987, pp. 169 – 185.

[3] F. P. Brooks, et al. "An Experiment in Musical Composition," *IRE Trans. On Electronic Computers*, EC-6, 1957, pp. 175 – 182.

[4] J. Cohen, "Information Theory and Music," *Behavioral Science*, Vol. 7 [2], 1962, pp. 137 – 163.

[5] A. Di Nunzio, *Genesi, sviluppo e diffusione del software Music N nella storia della composizione musicale*, Tesi di laurea, Università di Bologna: D.A.M.S. Musica, 2010.

[6] O. Garfield, *Geniacs: Simple Electric Brain Machines, and How to Make Them*, Manuale d'uso, 1955.

[7] L. Hiller, L. Isaacson, *Experimental Music: Composition with an Electronic Computer*, McGraw-Hill, 1959.

[8] L. Hiller, "Music Composed with Computers: a Historical Survey," *The Computer and Music*, a cura di Harry B. Lincoln, Cornell University Press, 1970, pp. 42 – 96.

[9] D. Levy, *Robots Unlimited: Life in a Virtual Age*, A. K. Peters, 2006, pp. 165 – 175.

[10] J. Pierce, *Symbols, Signals and Noise: the Nature and Processes of Communication*, Harper Torchbooks, New York 1961.

[11] R. Pinkerton, "Information Theory and Melody," *Scientific American*, Febbraio 1956, pp. 77 – 86.

[12] C. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, Vol. 27, 1948, pp. 379–423, 623–656.

[13] J. Sowa, *A Machine to Compose Music*, Oliver Garfield Co., Inc., New York, 1956.

LA MUSICA ELETTROACUSTICA E L'ANIMAZIONE ASTRATTA:
PROBLEMATICHE ESTETICHE, MODELLI ANALITICI E SVILUPPI FUTURI

Luca De Siena

lucadesiena@libero.it

SOMMARIO

Con la nascita della tecnologia digitale, immagini e suoni sono stati entrambi ridotti alla stregua di numeri e liberamente trasducibili da una all'altra forma. Questa facilità di trasduzione è stata sfruttata per creare dei lavori audiovisivi con una forte connessione fra materiali audio e video, aprendo ad esempio un'immagine in un software audio o un suono in un software grafico.

Ciò nonostante, sebbene l'interesse si sia spostato verso la generazione audiovisiva in tempo reale, tutt'oggi vengono prodotti lavori che giocano sulle asincronie (cioè un tipo di interazione audiovisiva non sempre puntuale) costruendo una sorta di contrappunto audiovisivo ma mantenendo una forte correlazione fra forma e materiali usati. I lavori a cui ci si è riferito in questo articolo sono lavori con immagini astratte e musica elettroacustica. I quesiti ai quali si è cercato di fornire una risposta sono stati: Cosa si perde abbandonando la narratività dell'immagine filmica? Cosa si guadagna utilizzando delle immagini astratte? Cosa si perde abbandonando il suono strumentale? Cosa si guadagna utilizzando il suono elettronico?

1. MODELLI ANALITICI

Occorre, per rispondere a questi quesiti, disporre di adeguati modelli analitici creati appositamente per lo studio di questa tipologia di lavori. I lavori a cui ci si riferisce in questo articolo hanno le seguenti caratteristiche:

- Narrazione non lineare
- Performance pre-composta (non in tempo reale)
- Materiali sintetici o concreti ma elaborati in digitale
- Esperienza di fruizione non corporea cioè come pubblico di una sala da concerto.

Copyright: © 2014 Luca De Siena This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Passiamo quindi in rassegna alcuni dei modelli analitici che sono disponibili attualmente per l'analisi di opere simili.

Un tentativo di ricerca di un'estetica comune ai lavori di "visual music" (termine con il quale come viene spesso etichettata questa tipologia di lavori) dal punto di vista del sistema percettivo umano è stata condotto da Tom DeWitt [1].

Tra i parametri presi in considerazione da DeWitt ci sono il colore, la dimensione temporale e la simmetria.

Nel colore DeWitt riscontra ancora una mancanza di opinioni comuni circa la percezione di questo parametro nonché una mancanza di univocità nelle corrispondenze colore-suono che nei secoli sono state teorizzate. Per questa ragione occorre fare un passo indietro alla soglia di passaggio fra il bianco e nero e il colore. DeWitt ipotizza che dal momento che i neuroni della retina sono più sensibili alla luce, il passaggio dal bianco al nero possa suscitare una sorta di sensazione di risveglio.

Per quanto riguarda la dimensione temporale, la percezione visiva e quella uditiva sono più vicini. È infatti all'incirca oltre le 20 oscillazioni per secondo (convenzionalmente il limite grave della banda udibile) che l'orecchio umano comincia a percepire le pulsazioni come suono continuo ed è oltre i 12 frame per secondo che le immagini ci danno un senso di continuità, di movimento.

Come sostiene DeWitt, "*Come la riduzione di una partitura in battute e misure, i film sono fatti di fotogrammi e scatti. Il mezzo cinematografico invita gli artisti a diventare musicisti visivi dividendo il tempo secondo un senso di cambiamento*. […] *Un movimento rapido può provocare eccitazione; un movimento lento può indurre un tipo di sentimento più languido.* ". [1]

Il problema del tempo ci conduce direttamente ad un'altra questione ad esso strettamente legata, quella del sincrono audiovideo cioè l'esito della sincronizzazioni di due flussi, quello visivo e quello uditivo, in un punto determinato del tempo al fine di creare senso.

Michel Chion considera il sincrono come uno degli eventi cardine nella storia dell'audiovisione: "*Per i primi spettatori del film sonoro, il sincronismo fu un fenomeno meraviglioso.* […] *L'unione sincrona di suono e immagine* […] *suscita un forte desiderio paragonabile al desiderio di un quartetto d'archi di raggiungere la perfetta sincronizzazione in una esecuzione*". [2]

Chion descrive la relazione fra suono e immagine come l'interazione "verticale" dell'*effetto Kuleshov*. Questo effetto fu sperimentato dal regista russo Lev Kuleshov in un video in cui la faccia senza espressione di un attore era mostrata alternandola ad altre immagini quali quelle di un piatto di zuppa, una ragazza, una piccola bara, etc. Il film fu mostrato ad un pubblico il quale ebbe l'illusione che l'attore cambiasse espressione in base a ciò che guardava, cioè la serie di immagini proposte, mostrando di volta in volta un'emozione diversa. Allo stesso modo Chion sostiene che suono e video proiettano i propri significati l'uno dentro l'altro ma non in maniera seriale come nel caso dell'effetto Kuleshov, bensì in maniera simultanea.
Se vi è la tendenza a dare particolare rilievo ai punti di sincronismo audiovideo è perché abbiamo imparato a considerare l'asincronia come casuale quindi come un disturbo o comunque qualcosa di irrilevante.
Sempre secondo Michel Chion i punti di sincrono audiovideo sono definibili come particolarmente salienti e significativi; questo meccanismo si applica soprattutto ai video in cui l'immagine è astratta poiché, come asserisce de Cardoso [3], in mancanza di riferimenti visuali esterni (che normalmente determinano l'organizzazione gerarchica di quello che guardiamo) i punti di sincrono diventano la maggiore fonte di interpretazione dell'esperienza di lavori audiovisivi aventi queste caratteristiche.

Un ulteriore punto di vista analitico dotato di grande rilevanza è quello dato da Anton Fuxjäger, professore dell'Università di Vienna. Fuxjäger classifica la produzione e la recezione della visual music all'interno di tre principali categorie: "Traduzione Musicale", "Strutture sintetiche" e "Disturbo reciproco".
Nella "Traduzione musicale" alcuni parametri musicali sono trasdotti in parametri visivi. Pertanto in questo caso è la musica a fornire la struttura sulla quale il video è costruito. Nelle "Strutture sintetiche", invece, gli eventi visuali e quelli sonori non sempre coincidono temporalmente, piuttosto formano una struttura audiovisiva unica che è il risultato di informazioni temporalmente differenti trasmesse nel canale visivo e in quello uditivo. Nel "Disturbo reciproco" non ci sono abbastanza coincidenze audiovisive e lo spettatore non è in grado di ricostruire una struttura audiovisiva. Lo stesso Fuxjäger specifica che le categorie di interazione audiovisiva da lui individuate possono alternarsi all'interno di uno stesso lavoro.
Diego Garro [5,6] parla di "visual music" rifendosi a lavori che non sono narrativi ma non per forza non-rappresentativi, citando Pliché, "Se il cinema è romanzo audiovisivo, la visual music è poesia audiovisiva". Pertanto, sempre secondo Diego Garro, il termine "visual music" è diventato un ombrello sotto il quale si racchiude un corpus di lavori nei quali immagini astratte o più comunemente "non-narrative" (generate a computer o catturate digitalmente e poi elaborate al computer) sono combinate con suoni elettroacustici. Sul perché la relazione fra immagini e astratte e suoni elettroacustici funzioni bene, Garro fa riferimento a Simon Emmerson sostenendo che possiamo parlare di esperienza monomediatica del suono solo nel caso si tratti di suoni puramente acusmatici ovvero che non portano traccia di riconoscibilità di suoni conosciuti quali il parlato, suoni strumentali e suoni d'ambiente/naturali. Questo perché, sul piano puramente uditivo, la presenza di suoni riconoscibili implica un tipo d'ascolto che porta l'ascoltatore a mettere continuamente in relazione il materiale astratto con quello concreto, alla ricerca di una traccia narrativa, risultando così essere quanto meno "multi-modale" se non "multimediale"; afferma Garro (traduzione dello scrivente): *"Il suono può portare con sé, e spesso lo fa, una moltitudine di categorie cognitive, culturali e artistiche."* Lo sviluppo di un discorso audiovisivo in assenza di un leitmotiv narrativo è inevitabilmente complesso. I suoni elettroacustici sono sospesi in uno spazio multidimensionale che non offre (quasi mai) griglie temporali e organizzazioni delle altezze, risultando così sicuramente affascinante ma anche molto difficile per l'ascoltatore, specie se inesperto, da decifrare. L'aggiunta di immagini e forme visuali aggiunge altre variabili ad una equazione di per sé già labirintica. Tuttavia il motivo recondito di questo incontro non sta nell'aggiungere ulteriore complessità ad un mezzo già privo i limiti, bensì nel combinare le ramificazioni di due domini artistici per arrivare ad una sintesi armoniosa delle due, un locus dove una variabile esplica l'altra, non risolvendo l'equazione ma rendendola se non altro più trasparente.
Diego Garro prova a condensare le modalità di interazione fra audio e video in questo modello:

Separazione>Complementarità intuitiva>Sincresi>Mappatura parametrica

Questo modello non è da intendersi come un commutatore a scatti bensì come un continuum di possibilità di interconnessione fra il gesto sonoro e quello visivo. Agli estremi di questo modello troviamo la "Separazione" e la "Mappatura parametrica". Per "Separazione" si intende l'assenza di interconnessioni palesi fra il flusso sonoro e quello visivo. In questo tipo di situazione è lo spettatore che è chiamato ad operare un'interpretazione delle ragioni per le quali quei due elementi siano stati messi in relazione audiovisiva (in una versione precedente di questo modello, risalente al 2005 [6], questo estremo era individuato proprio mediante il termine "Interpretazione"). All'estremo opposto del segmento troviamo indicata la situazione di "Mappatura parametrica" in cui alcuni parametri del suono sono in relazione isomorfica con taluni parametri delle immagini.
Un passo prima di questo tipo di relazione c'è il meccanismo della "Sincresi", termine che Garro mutua da Michel Chion, cioè la tendenza ad attribuire correlazioni particolari ai suoni e alle immagini basandosi solamente sulla loro simultaneità. Garro però si rifiuta di chiamare questo fenomeno "Sincronizzazione" perché sostiene che la sincronizzazione sia una *conditio sine qua non* per stabilire una connessione parametrica fra un evento visivo ed uno sonoro, ma non è una condizione suffi-

ciente a metterli in relazione profonda, perché due eventi in sincrono potrebbero essere spettromorfologicamente slegati.
L'altra posizione intermedia, indietreggiando verso un grado di forza associativa più bassa, è quella che Garro definisce come "complementarità intuitiva" cioè un tipo di situazione in cui i materiali audiovisivi sembrano sostenersi l'un l'altro nella creazione di una blanda gestualità multimediale, anche quando questi non si presentano simultaneamente.
Ciò nonostante, secondo Garro, esistono anche altre strategie associative che si pongono ad un livello gerarchico superiore rispetto a quelle gestuali appena esposte. Queste modalità di costruzione della sintassi audiovisiva vengono definite come "associazioni discorsive" e sono altrettanto importanti al fine di costruire una coesa identità idiomatica del lavoro. Questo tipo di strategie associative sono tipiche di collaborazioni fra artisti visivi e musicisti, in cui i due discorsi vengono costruiti separatamente ma tenendo presente alcune direttive comuni precedentemente stabilite.
Queste strategie vengono definite come "strategie allegoriche", "strategie sincretiche" e "strategie di saldatura". Le "strategie allegoriche" sono quelle in cui il suono e l'immagine sono similmente manipolati, ma non per forza in sincrono, pertanto l'uno è metafora dell'altro. Le "strategie sincretiche" estendono il concetto di sincresi, spiegato precedentemente, a tutta la forma e non solo agli interventi gestuali. Le "strategie di saldatura", invece, vengono definite come quelle che cercano di stabilire un discorso audiovisivo principalmente attraverso una corrispondenza morfologica, formale, caratteristica quest'ultima che diminuisce la sua forza progressivamente procedendo verso il discorso sincretico ed è pressoché assente in quello allegorico; quest'ultima affermazione prescinde dall'essere un giudizio di ordine estetico.
È inoltre fondamentale sottolineare che un lavoro audiovisivo non si limita soltanto all'utilizzo di una di queste strategie ma può articolarle tutte in alternanza.
Altra problematica che Garro affronta nella sua ricerca è quello del montaggio. Esiste infatti una difficile relazione fra il processo del montaggio nell'ambito della musica elettroacustica e il montaggio video. La difficoltà è principalmente tecnica: mentre per il montaggio audio si procede sia in senso verticale (spettrale/frequenziale) che in senso orizzontale (temporale), nel montaggio video, al di là delle tecniche del compositing e della trasparency, si opera sempre orizzontalmente.

2. CONCLUSIONI

Il sincrono, come sottolinea de Cardoso, è l'unico elemento fornito al pubblico per permettergli l'interpretazione di lavori di visual music, privi di riferimenti concreti, esterni.
Ma a che pro si sceglie di eliminare materiali sonori conosciuti rendendo ancora più remota la chiave di lettura? Cosa si pensa di guadagnare a livello audiovisivo con l'utilizzo della musica elettroacustica al posto di quella strumentale/tradizionale?

Alla base della nascita di questa particolare forma espressiva c'è sicuramente una convergenza tecnologica, cioè l'utilizzo della stessa tecnologia alla base della creazione del video e dell'audio: agli esordi della visual music era la pellicola, oggi sono i personal computer.
Secondo Diego Garro, la crescente potenza dei computer domestici permette ai compositori di musica elettroacustica di misurarsi con facilità anche con la dimensione visiva. Per alcuni compositori, tale rapporto rappresenta solo una conseguenza inevitabile del loro modo "visuale" di concepire il suono elettronico.
Ma non è solo questa convergenza tecnologica a rendere legittima l'accoppiata di immagini astratte e musica elettroacustica; come specificato da Garro "*il motivo recondito di questo incontro non sta nell'aggiungere ulteriore complessità ad un mezzo già privo di limiti bensì nel combinare le ramificazioni di due domini artistici per arrivare ad* una sintesi armoniosa delle due, un *locus* dove una variabile esplica l'altra, non risolvendo l'equazione ma rendendola se non altro più trasparente."
Sempre a tal proposito Garro afferma: *"i compositori si sono sempre sforzati di sviluppare un vocabolario sempre più ricco e diversificato di materiali sonori e di esplorare i modi in cui questi possono essere utilizzati per articolare il tempo e lo spazio. È naturale che l'introduzione della dimensione visiva nel linguaggio elettroacustico è considerata da alcuni membri della comunità come una sgradita intrusione nel mondo incontaminato delle entità sonore invisibili. Il video, se così si può dire, distrae l'attenzione dello spettatore dalla meraviglia del mondo sonoro che è possibile progettare nella colonna sonora: rompe la magia musicale.* [...] *L'aggiunta delle immagini in movimento non si limita a completare i suoni: essa comporta una ricchezza di nuove sfide compositive e sposta l'esperienza del compositore e quella dello spettatore in una posizione che è sicuramente affascinante, ma abbastanza distante da quella che sperimentano quando trattano con soli suoni."*[5]

A proposito del rapporto funzionale tra musica elettroacustica e immagini astratte, in uno scambio epistolare privato con il compositore Valerio Murat, questi suggeriva: "*Le immagini astratte, così come la musica composta da suoni con grado di surrogazione remota, ci interessano perché sono forme nuove; con "forma" intendo la fisionomia dell'immagine e la spettromorfologia dell'oggetto sonoro. Credo sia tutto lì. Nell'innata curiosità animale di sapere cos'è l'oggetto di fronte a noi. Se è velenoso, se è commestibile, se è un predatore o una preda. In un certo senso l'astrazione audiovisiva ha la capacità di rimetterci in contatto con la natura primordiale del nostro sentire. Bypassiamo l'interfaccia culturale e rivolgiamo la nostra attenzione alla risposta agli stimoli dei nostri sensi. Non ne possiamo fare a meno. Questo era il modus vivendi che ci permise di sopravvivere nella notte dei tempi e paradossalmente, oggi, riusciamo a ripristinare l'intimo rapporto con la*

natura attraverso uno dei prodotti più complessi e articolati del fare umano: la cultura".

Il concetto smalleyiano di "surrogazione remota" quindi, piuttosto che quello di "astrazione".
Questo tipo di immagini, interagendo con il sonoro, tende a creare un tipo di situazione in cui il canale visivo informa quello uditivo e viceversa, dandoci così un numero di informazioni tali da ridurre la quantità di surrogazione. L'unione di un suono e di una data immagine crea un rapporto di surrogazione non più con il materiale di partenza, bensì una surrogazione interna all'opera stessa e a quel particolare sintagma audiovisivo. Quanto detto da Murat conferma l'ipotesi di Garro circa l'unione dei due *canali attentivi* che tendono a creare una equazione più trasparente dove un canale cerca di completare il proprio senso nell'altro.

Tutte le ipotesi sembrano convergere sul fatto che immagini e suoni con "surrogazione remota" si associno al fine di dare maggiore fruibilità e maggiore consistenza all'esperienza audiovisiva allo spettatore.

Ma perché si preferiscono suoni elettroacustici a quelli strumentali? E ancora, perché si preferisce un tipo di organizzazione musicale basata sul timbro piuttosto che sull'organizzazione delle altezze?

Per citare nuovamente Garro, sul piano puramente uditivo, "*la presenza di suoni riconoscibili implica un tipo d'ascolto che porta l'ascoltatore a mettere continuamente in relazione il materiale astratto con quello concreto, alla ricerca di una traccia narrativa; il suono può portare con sé, e spesso lo fa, una moltitudine di categorie cognitive, culturali e artistiche"*.
A proposito di questi "detriti", che un suono riconoscibile si porta dietro e sul perché questo possa interferire con la fruizioni delle immagini, o meglio, impedire una esperienza audiovisiva pura, dionisiaca, e sgombra da condizionamenti pregressi, sembra opportuno citare il pensiero del regista italiano Michelangelo Antonioni. Nel saggio "Antonioni e la musica" di Roberto Calabretto si racconta il rapporto del regista ferrarese con la musica nel cinema, musica che secondo lui aveva spesso avuto *"una funzione di commento esterno, di commento inteso a creare un rapporto tra musica e spettatore, non fra musica e film"*; da qui la sua critica e la volontà, l'esigenza, di correggere questa anomalia di funzione: *"Questo mi sembra un rapporto sbagliato, un rapporto che non ha niente a che vedere con il cinematografo, proprio perché si tiene volutamente ai margini del fatto cinematografico, è un rapporto che si stabilisce tra musica e spettatore al di fuori dell'immagine"*. [7] Calabretto nel suo saggio mette in risalto come la ricerca musicale di Antonioni sia finalizzata a rivedere l'idea stessa di cinema che troppo spesso è inteso come un semplice accostamento di diverse espressioni artistiche (teatro, musica, fotografia...).

Quanto detto sembra confermare che a motivare la scelta di evitare il suono strumentale e i sistemi musicali tradizionali nei video astratti sia la volontà di evitare collegamenti verso l'esterno da parte dello spettatore; collegamenti che avvengono per mezzo dei sistemi musicali già codificati date le implicazioni psicologiche, culturali, etc. che un suono si porta inevitabilmente dietro.

Di conseguenza, la scelta ricade sui suoni elettronici, suoni liberi da condizionamenti pregressi che invece aiutano lo spettatore ad avere un'esperienza più immersiva e più "primordiale", in quanto ci si concentra solo sulle caratteristiche (spettromorfologiche) dei suoni.

L'assenza di una narrazione pone la questione della quantità di tempo che il connubio immagini e suoni, privi di legami con una sorgente riconoscibile, sia in grado di sostenere nell'attenzione dello spettatore rispetto ad un prodotto audiovisivo costruito con materiali aventi una più spiccata riconoscibilità.
In parole povere, quanto può durare un'opera come questa? In media questi lavori hanno una durata di circa 6 minuti.
La possibilità per queste opere di vivere in uno spazio con delle caratteristiche cromatiche e architettoniche che ne accentuano l'immersività[1], magari anche attraverso l'utilizzo di altoparlanti disposti attorno al pubblico (*surround sound*) per la spazializzazione del suono, permette dei tempi di sviluppo formale sicuramente più lunghi. Bisogna inoltre osservare che in una installazione (riprodotta in maniera ciclica, in *loop*) lo spettatore difficilmente fruirà dell'opera partendo esattamente dall'inizio. Questa evenienza potrebbe interferire con la volontà, da parte dell'autore, di mettere insieme uno sviluppo formale lineare che necessità di essere letto da un punto di inizio fino alla fine per rintracciare l'evoluzione del materiale, mediante l'utilizzo delle proprie capacità mnemoniche.
L'utilizzo di materiali riconoscibili aiuta il cervello a ritenere nella memoria più dati, permettendo lo sviluppo e la trasformazione degli stessi, perché più facilmente memorizzabili e quindi facilmente inseribili in una traiettoria "narrativa". L'uso di immagini sonore astratte costringe il compositore a plasmare il proprio oggetto sonoro in modo che questi presenti tratti salienti riconoscibili.
Un lavoro per la sala da concerto è invece relegato giocoforza a questa linearità di fruizione, al quale lo spettatore, incollato sulla poltrona, non può sottrarsi. Va inoltre considerato che spesso nei festival, opere del genere sono programmate assieme ad altro tipo di esibizioni (acusmatiche o per elettronica e strumenti) il che porta ad allestire i concerti in sale polivalenti che si adattano per l'occasione a questo tipo di audiovisione; si tratta senza dubbio di una situazione meno immersiva di quella installativa, che porta a dover ridurre la durata del lavoro proprio a causa della mancanza di questo coinvolgimento totale del proprio corpo nel mezzo dell'opera.

[1]Con "immersività" si intende il calarsi completamente nel mondo virtuale, a livello sensoriale

Il concetto di fruizione *embodied* e *disembodied* della visual music è stato trattato anche nella tesi di Leonardo de Cardoso. L'autore analizza la convivenza di immagini e suoni a remota surrogazione (astratti) in diverse situazioni che ne fanno impiego. Le situazioni descritte da de Cardoso vanno dal concerto di visual music alle performance in tempo reale di *vjing* (*visual jockey*, corrispettivo visivo del *disk jockey*) che accompagna alcune serate di musica dance nei club. In quest'ultimo caso, la presenza di immagini astratte viene tollerato tranquillamente e anzi diventa necessario al fine dell'immersione in una esperienza prettamente estatica.
D'altra parte, senza scomodare situazioni che implicano la presenza della musica ai fini della danza, le performance audiovisive dello stesso Carsten Nicolai (di solito con lo pseudonimo di Alva Noto) hanno durate superiore ai 6 minuti; è sicuramente corretto osservare come molte di queste mettano in primo piano l'aspetto ritmico (al pari di quello che succede in discoteca), ma si tratta di ritmi che non mirano ad essere ballati: è piuttosto una meccanicità che tende a riprodurre i comportamenti delle macchine, in linea con i dettami dell'estetica glitch. Non a caso questo genere musicale è stato definito *braindance*, *blip hop*, *Intelligent Dance Music* etc. tutti termini che, come viene spiegato bene nel saggio di David Byrne "What is Blip Hop" [8], giustificano la presenza del ritmo non come mezzo di stimolazione fisica bensì come processo totalmente cerebrale di avvicinamento alla logica comportamentale di un macchinario.
Ciò detto, la presenza del ritmo in una performance audiovisva di Carsten Nicolai non va interpretata alla stregua di un set di musica dance.
Dunque cos'è che consente durate superiori ai 6 minuti se escludiamo anche la componente ritmica?
Probabilmente è la corretta combinazione di spettacolarità, realizzata mediante schermi di grosse dimensioni, illuminazione e livello di intensità sonora, che riesce a tenere lo spettatore prigioniero dell'*hic et nunc* per tempi di molto superiori agli 8 minuti. La musica e il video sono solo due degli stimoli presenti in una performance del genere che, amplificati enormemente e uniti ad altri aspetti spettacolari, tendono a monopolizzare l'attenzione del pubblico non solo a livello mentale ma anche e sopratutto fisico.

Tutto questo sembra dimostrare che una mancanza di riferimenti precostituiti, nel canale visivo e in quello uditivo, non implica necessariamente una riduzione dei tempi di tolleranza di una performance audiovisiva, purché vengano implementate altre condizioni di fruizione.

Pertanto si può ipotizzare una diretta proporzionalità tra la durata temporale sostenibile dallo spettatore e il coinvolgimento fisico nella performance. Sarebbe a dire, al crescere della quantità di coinvolgimento fisico dello spettatore all'interno della performance, cresce anche la durata temporale della performance che il pubblico è disposto a sostenere, anche in mancanza di una narrazione.

2.1 Sviluppi futuri

Occorrerà in futuro massimizzare la qualità di questo tipo di lavori, sia sul piano visuale che su quello musicale. Le opere di Carsten Nicolai, ad esempio, puntano più sull'impatto sonoro che sulla qualità dello sviluppo del video. D'altro canto, i lavori di Candaş Şişman sembrano occuparsi con più attenzione dell'aspetto visuale che di quello musicale. La nuova generazione di compositori elettroacustici dovrà quindi colmare le lacune nell'ambito della creazione e della gestione di un flusso video. La nuova generazione di artisti visivi dovrà, invece, imparare a gestire una forma musicale e a padroneggiare l'elaborazione dei suoni.
Alternativamente, si potrà pensare alla collaborazione tra specialisti del settore della composizione musicale elettroacustica e quelli dell'arte visiva, al fine di creare un'opera corale che implementi lo stato dell'arte su entrambi i canali sensoriali.

Quanto detto a proposito dell'immersività ci invita invece a considerare nuove strategie di interazione di immagine astratta e suono elettronico, al fine di coinvolgere maggiormente il pubblico nella performance.

Diego Garro sembra convenire a tal proposito quando dice: "*Le sfide tecnologiche future non devono concentrarsi esclusivamente sui mezzi per la progettazione, la sintesi e la manipolazione di materiale, ma si spera anche sui modi di proiettare i suoni in veri e propri spazi architettonici in modo che il rituale concerto, ancora un pilastro della cultura elettroacustica, possa essere estesa alle opere audiovisive. La diffusione multicanale del suono di musica acusmatica può fare meraviglie articolando lo spazio sonoro intorno agli ascoltatori e creando sculture invisibili di entità sonore che sfidano la percezione dello spazio fisico che il pubblico occupa. I tecnici e gli artisti coinvolti nei nuovi media digitali basati sul tempo dovrebbero forse immaginare dei modi in cui anche le immagini possono occupare uno spazio tridimensionale e non una semplice sezione bidimensionale di esso sullo schermo.*" [6]

Per conquistare un'esperienza audiovisiva pura dobbiamo necessariamente astrarci dall'utilizzo di materiali sonori, che si portano dietro un sostrato di informazioni già sedimentate nel nostro pensiero musicale. Per rendere questa esperienza veramente dionisiaca abbiamo bisogno di essere totalmente immersi nella performance. Il suono, allo stato attuale, registra una supremazia sull'immagine per quanto riguarda la tridimensionalità. È forse superfluo ricordare che chiudendo gli occhi possiamo smettere di vedere un'immagine ma non possiamo affermare con la stessa facilità che, tappandoci le orecchie, possiamo smettere di ascoltare un suono, e questo proprio a causa della natura spaziale del fenomeno sonoro che lo porta ad impattare sul nostro corpo.
Le nuove tecnologie, quali il video 3D e la progettazione di ambienti in grado di ospitare una installazione

multipla di schermi, permetteranno una conquista spaziale anche da parte dell'immagine animata.

Nella conquista della dimensione spaziale da parte del video c'è l'opportunità di realizzare concretamente quanto sognato dai pionieri della visual music: *creare una musica visiva paragonabile alla musica acustica.* (W.-Moritz) [9]

3. BIBLIOGRAFIA

[1] T. DeWitt: "Visual Music: Searching for an Aesthetic", *Leonardo*, n.20, pp.115-122, 1987.

[2] M. Chion: *Film, a Sound Art*, Columbia University Press; Film and Culture Series edition, pp.37, 2009.

[3] L. De Cardoso: *Visual Music: an Ethnography of an Experimental Art in Los Angeles,* Presented to the Faculty of the Graduate School of Music, University of Texas, 2010.

[4] A. Fuxjäger: "Translation, Emphasis, Synthesis, Disturbance: On the function of music in visual music*, Organised Sound*, Volume 17, Issue 02, pp. 120-127, 2012.

[5] D.Garro: "A glow on Pythagoras' curtain. A composer's perspective on Electroacoustic music with video"*, Proceedings of the Electroacoustic Music Studies Conference - Sound in multimedia* , Montréal, 2005, Versione 2, 2011.

[6] D.Garro: "From Sonic Art to Visual Music: Divergences, convergences, intersections", *Organised Sound, Organised Sound*, Volume 17, Issue 02, pp 103-113, 2012.

[7] R. Calabretto: *Antonioni e la musica,* Marsilio, 2012.

[8] D. Byrne: "What is Blip Hop", *http://www.luaka-bop.com/comparativesound/whatis/whatis.htm* (ultimo accesso Febbraio 2013).

[9] W.Moritz: "*The Dream of Color Music and the Machines hat Made it possible"*, *Animation World Magazine*, Issue 2.1, 1997.

CONOSCERE L'ARMONIA TONALE NELL'AMBIENTE INTERATTIVO "HARMONIC WALK"

Marcella Mandanici, Leonardo Amico, Antonio Rodà, Sergio Canazza
Università di Padova
{mandanici, amico, roda, canazza}@dei.unipd.it

SOMMARIO

Harmonic Walk è un ambiente interattivo progettato per l'apprendimento e la pratica dell'armonia. L'interfaccia dell'applicazione consiste in una telecamera posta sul soffitto in grado di tracciare la presenza di un utente che cammina su una superficie piana come un pavimento o un tappeto. Utilizzando l'analogia esistente fra spazio bidimensionale e spazio armonico, tradizionalmente rappresentato come un reticolo di accordi, *Harmonic Walk* offre all'utente la possibilità di camminare fisicamente all'interno della struttura armonica, ottenendo un risultato sonoro relativo alla zona occupata. In questo articolo sono analizzate le potenzialità educative dell'applicazione nel caso dello studio dell'armonia tonale e della pratica compositiva da parte di bambini della scuola primaria. I risultati ottenuti hanno mostrato che, impiegando marcatori visivi o un opportuno *feedback* sonoro, la maggior parte degli utenti è in grado di orientarsi e di utilizzare correttamente l'applicazione. Inoltre, fra i partecipanti al test, un numero ridotto di bambini è stato in grado di armonizzare spontaneamente e senza aiuto esterno una melodia tonale.

1. INTRODUZIONE

L'armonia tonale, il cui massimo sviluppo coincide approssimativamente col culmine del periodo Romantico, è tutt'oggi alla base della maggior parte della musica che comunemente ascoltiamo. Questo fatto, unitamente all'alto livello di definizione tecnica e di complessità espressiva, ne fanno il terreno privilegiato per la didattica sia nei Conservatori e Licei Musicali, sia nella scuola dell'obbligo. Mentre nella formazione professionale dello strumentista si fatica ancora a raggiungere un adeguato livello di consapevolezza e di pratica dell'armonia, nella scuola dell'obbligo essa viene utilizzata per l'accompagnamento di canti avvalendosi principalmente di basi preregistrate o, nella migliore delle ipotesi, dell'accompagnamento eseguito da un docente alla chitarra o alla tastiera. Ciò introduce nella pratica musicale un elemento certamente gratificante, ma di scarso valore formativo per i bambini, i quali non possono che limitarsi a seguire l'armonizzazione senza poter minimamente interagire con essa. *Harmonic Walk*, avvalendosi delle grandi potenzialità dell'utilizzo del movimento nello spazio e dell'apprendimento enattivo [1], vuole fornire a un utente inesperto la possibilità di entrare in contatto diretto col mondo dell'armonia tonale, offrendogli occasioni di esplorazione dello spazio armonico, di esecuzione di collegamenti accordali e di armonizzazione di semplici melodie. Nelle Sottosezioni seguenti viene presentato il contesto generale relativo ad *Harmonic Walk*, con una breve rassegna di applicazioni correlate. Nella Sez. 2 vengono definite le armonie utilizzate e le caratteristiche dei percorsi accordali da svolgere sulla superficie dell'applicazione. Nella Sez. 3 sono illustrate le caratteristiche del sistema e la disposizione della struttura armonica nello spazio interattivo. La breve Sezione finale contiene la relazione di alcuni test effettuati con bambini della scuola primaria.

1.1 Apprendimento e movimento

Nel campo della psicologia dello sviluppo e della pedagogia, da tempo esistono diverse teorie a suffragio di una maggiore integrazione della componente motoria nelle attività didattiche ([1], [2], [3]). Più recentemente, la diffusione capillare delle tecnologie digitali, ha necessariamente portato psicologi dello sviluppo e pedagogisti a considerare le potenzialità che strumenti innovativi possono avere nel campo dell'apprendimento. In particolare Moreno e Mayer [4] hanno cercato di definire metodologie e strumenti per la valutazione di tecniche e tecnologie di apprendimento interattive. Infine è rilevante notare come anche le istituzioni si siano interessate a promuovere la ricerca di innovativi metodi di apprendimento. Nel 2004 l'Unione Europea ha infatti finanziato il progetto ENACTIVE Network [2], allo scopo di coordinare la ricerca sull'apprendimento e sulle interfacce enattive.

1.2 Applicazioni Correlate

Negli scorsi anni sono stati sviluppati diversi strumenti tecnologici di supporto alle attività didattiche basate sul movimento. Di particolare interesse per la presente ricerca sono quelli che, analogamente a *Harmonic Walk*, implementano un'interazione *whole body*, ovvero reagiscono ai movimenti dell'intero corpo. Tra questi *Kids Room* [5], *Mediate* [6], *SMALLab* [7] e la *Stanza Logomotoria* [8],

Copyright: ©2014 Marcella Mandanici, Leonardo Amico, Antonio Rodà, Sergio Canazza. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

[1] L'apprendimento enattivo avviene attraverso una relazione dinamica fra il soggetto e l'ambiente che lo circonda. I dati sensomotori raccolti dal soggetto nel corso dell'interazione con l'ambiente costituiscono infatti la fonte primaria del significato e la base per la formazione della conoscenza.

[2] Il sito web del progetto ENACTIVE Network http://www.enactivenetwork.org/

in cui un utente utilizza la memoria spaziale per collocare correttamente *files* sonori all'interno di una narrazione.
Per quanto riguarda poi in particolare l'insegnamento dell'armonia, esistono diversi lavori che propongono strumenti tecnologici enattivi per facilitare l'approccio alla materia. *Harmonic Navigator* [9], è una piattaforma interattiva in cui attraverso il controllo gestuale un utente è in grado di esplorare lo spazio armonico di differenti stili musicali, estrapolati da un database caricato nel sistema. *Harmony Space* [10] del Music Computing Lab dell'Università di Stanford, nasce come applicazione desktop dotata di un'interfaccia grafica per semplificare lo studio dell'armonia, analizzare brani musicali e fornire un supporto alla composizione. In una più recente versione [11] l'applicazione è poi stata adattata per lo spazio fisico: il controllo avviene attraverso i movimenti dell'utente su superficie delimitata, in modo simile a *Harmonic Walk*. La sostanziale differenza tra le due applicazioni consiste nelle loro diverse finalità. *Harmony Space*, il cui utilizzo dipende da una preselezione di diverse opzioni, è concepita per un utilizzo esperto, mentre *Harmonic Walk*, rivolgendosi a un utenza non specializzata, offre una modalità semplice e intuitiva per un approccio immediato al mondo dell'armonia.

2. BASI TEORICHE

In questa sezione verranno delineati i più importanti elementi costitutivi del linguaggio tonale e sarà affrontata l'analisi dei suoi principali aspetti cognitivi, al fine di elaborare delle valide linee guida per la progettazione e per l'utilizzo didattico di *Harmonic Walk*.

2.1 Principali elementi costitutivi del linguaggio tonale

Il linguaggio tonale è basato su una struttura armonica fortemente gerarchizzata, al centro della quale si trova l'accordo di tonica, affiancato alle armonie di dominante (V grado, posto una quinta sopra la tonica) e sottodominante (IV grado, posto una quinta sotto la tonica). Queste tre armonie sono dette fondamentali perchè assommano in sè tutti i suoni della scala e costituiscono pertanto la struttura portante dell'armonia tonale. La composizione basata sull'armonia tonale richiede un utilizzo predominante dell'armonia di tonica e un utilizzo assai frequente, anche se non paritario, delle armonie di dominante e di sottodominante. Esistono poi molti modi per variare e arricchire le armonie fondamentali: uno di questi è l'utilizzo delle armonie parallele. Questi accordi sono costruiti su un basso che si trova una terza sotto rispetto al basso delle armonie fondamentali e differiscono da queste ultime per una sola nota, come si può vedere in Fig. 1. Tuttavia va detto che le armonie parallele costituiscono un arricchimento del colore armonico ma non rappresentano una necessità strutturale, dal momento che qualsiasi melodia tonale può essere accompagnata utilizzando le sole armonie fondamentali.
Se l'armonia può essere pensata come la struttura portante del sistema tonale, la melodia ne rappresenta invece lo svolgimento orizzontale. L'esempio di Fig. 2 riproduce un'elementare melodia popolare scritta in fa maggiore,

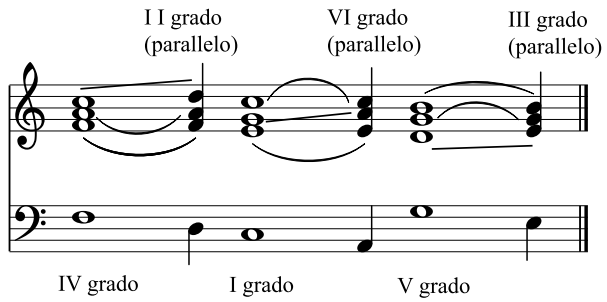


Figura 1. Quadro delle armonie fondamentali con le relative parallele. Le note comuni sono collegate da una legatura, mentre la nota che cambia è contrassegnata da una linea.

tratta da [12]. L'armonia di tonica è costituita dalle note fa-la-do mentre quella di dominante dalle note do-mi-sol. Nella Fig. 2 sono segnate in azzurro le aree di tonica e in rosso le aree di dominante, secondo uno dei possibili livelli di armonizzazione qui denominato livello *a*. Nel rigo sottostante la melodia sono riportati i valori ritmici relativi alla durata temporale delle varie aree, segnate rispettivamente con T per l'area di tonica e con D per l'area di dominante. Queste durate danno luogo al cosiddetto "ritmo armonico", che stabilisce i punti di cambio dell'armonia. I meccanismi che portano a questo risultato sono stati studiati e formalizzati da Lerdhal e Jackendoff [13], i quali hanno anche mostrato come la percezione musicale sia organizzata secondo vari livelli, dal più superficiale (raggruppamento) al più profondo (*prolongational reduction*). In Fig. 2 sono riportati tre successivi livelli di astrazione del canto popolare precedentemente citato. Il livello *b* mostra una prima riduzione in corrispondenza della prima e della quarta battuta. Il motivo della riduzione sta nel fatto che in queste due battute l'armonia di dominante cade sul tempo debole della battuta e quindi si configura come elemento secondario, tanto che al livello *c* entrambe le prime due battute sono assorbite in un'unica area di tonica. Questo procedimento evidenzia come la composizione tonale sia costituita da elementi ricorsivi (in questo caso il collegamento tonica-dominante) che si fondono man mano si procede ai livelli superiori in movimenti simili ma di più vasta entità. Quindi questi collegamenti armonici ricorsivi e le loro durate, scandite secondo precisi accenti forti e deboli, definiscono a livello schematico la composizione tonale.

2.2 Aspetti cognitivi della composizione musicale tonale.

Normalmente la tavolozza di possibilità offerta dalle tecniche compositive tonali è molto più elaborata rispetto all'esempio su esposto. Tuttavia queste tecniche necessitano di tempo e di una lunga fase di elaborazione e di perfezionamento, come fa notare J. Sloboda in [14]. Il riferimento a una struttura armonica con un percorso prestabilito costituisce pertanto un'utile rappresentazione schematica di basso livello della struttura compositiva tonale, laddove alla composizione si vogliano sostituire pratiche estemporanee come l'improvvisazione o l'interazione in tempo

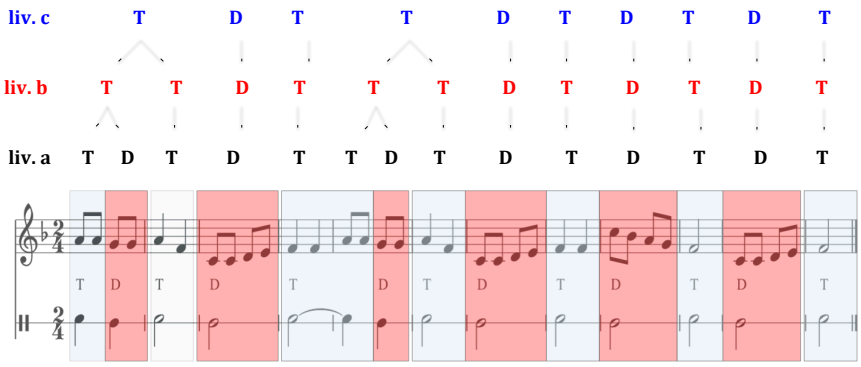


Figura 2. Tre livelli di astrazione della melodia popolare su riportata, ricavati secondo le regole di raggruppamento e di struttura metrica di Lerdhal e Jackendoff.

reale. In questi casi la struttura mentale del compositore-improvvisatore è principalmente di natura procedurale, laddove il processo compositivo consiste nel collegamento e nell'organizzazione temporale di elementi melodici e armonici che appaiono ricorsivamente. Questo tipo di strutturazione è descritta comunemente, anche in ambito professionale, con metafore di tipo spaziale. Espressioni come "percorso armonico", "il procedere della melodia", il "collegamento" di zone adiacenti, il "saltare" della melodia o il suo "procedere" per grado congiunto ("*step*", che in lingua inglese significa anche "passo"), il "camminare" del basso ("*walking bass*"), l'"attraversare" zone tonali, l'eseguire "giri armonici", eccetera, ne sono la più evidente testimonianza. La realizzazione della composizione-improvvisazione può quindi essere vista come lo svolgimento di un viaggio che deve necessariamente avere un suo punto di partenza e di arrivo e che per realizzarsi deve individuare anche delle precise tappe intermedie.
Un altro aspetto peculiare che caratterizza i percorsi musicali è la particolare scansione temporale richiesta per il raggiungimento delle loro varie tappe che è piuttosto rigida e dipendente da precisi accenti metrici. Gli aspetti della ricorsività delle tappe e della loro temporizzazione sono dunque gli elementi cognitivi peculiari della composizione tonale che costituiscono gli obiettivi dell'apprendimento da perseguire nell'uso didattico dell'applicazione.

3. CARATTERISTICHE DEL SISTEMA

In questa sezione viene descritta l'architettura generale del sistema *Harmonic Walk* sia per quanto riguarda la progettazione del software sia per la definizione della maschera di suddivisione dello spazio bidimensionale dell'applicazione.

3.1 Architettura del Sistema

Il sistema è composto da una telecamera posizionata al centro di un soffitto a circa tre metri di altezza. La telecamera riprende un'area libera del pavimento di circa sedici metri quadrati ed è connessa ad un computer dove risiedono i moduli software. Questi sono divisi in due parti. Una prima applicazione, chiamata *Zone Tracker* (Fig. 3), si occupa di tracciare la posizione degli utenti, e, in base ad una serie di maschere che permettono di suddividere in zone lo spazio interattivo, rileva la porzione occupata. Una seconda parte, responsabile della produzione musicale, è implementata attraverso una patch *Pure Data*.

L'algoritmo di *Zone Tracker* [15] si compone di tre stadi. In primo luogo, attraverso l'utilizzo di un modello statistico dello sfondo costruito per ogni canale dell'immagine, viene ricavata un'immagine in scala di grigi, in cui il valore di ogni pixel rappresenta la probabilità che esso sia diverso dallo sfondo, secondo il metodo proposto in [16]. Dopo aver elaborato l'immagine con l'ausilio di filtri convolutivi e morfologici, un algoritmo di soglia crea un'immagine binaria discriminando i pixel incerti in base alla loro vicinanza con pixel la cui probabilità di appartenere allo sfondo è nulla o certa. L'immagine in bianco e nero risultante identifica in bianco il *blob* dell'utente ripreso dall'alto, dalla cui silhouette si possono ricavare le coordinate cartesiane del baricentro. Le informazioni raccolte da *Zone Tracker* vengono poi inviate via *OSC* [3] a *Pure Data* [4], dove, in base alla zona occupata, vengono sintetizzati i suoni corrispondenti agli accordi della gamma armonica utilizzata da *Harmonic Walk*.

3.2 Strutture Armoniche e loro Rappresentazione Spaziale

Come si è visto nella Sez. 2, l'intera gamma armonica di una data tonalità si esprime in un totale di sei diversi accordi. Nella Sottosez. 2.2 è stato inoltre evidenziato come la struttura compositiva di un brano tonale presenti delle qualità spaziali che metaforicamente la accomunano ad un itinerario che colleghi diverse tappe. Tuttavia, l'elevata frequenza di elementi ricorsivi non favorisce la realizzazione di un percorso armonico disposto come una semplice successione lineare di varie postazioni accordali. La struttura del percorso tonale infatti richiede il frequente ritorno sulle tappe più significative e impone inoltre che nel corso del viaggio si possa ritornare ad un determinato punto da una qualunque posizione. La disposizione accordale del *tonne-*

[3] http://opensoundcontrol.org/introduction-osc
[4] http://puredata.info/

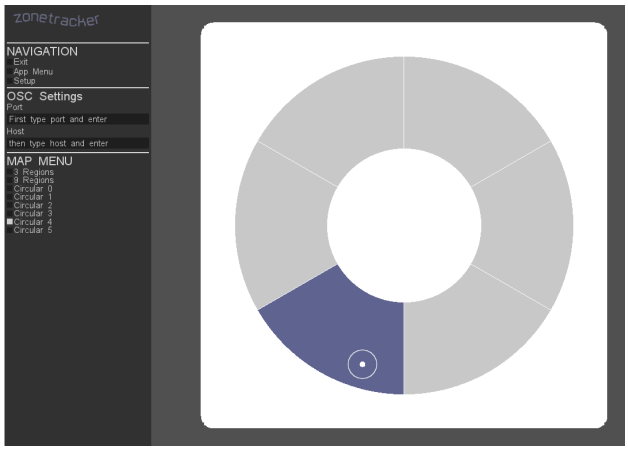


Figura 3. Interfaccia grafica dell'applicazione *Zone tracker*. Nella parte destra dell'interfaccia è visualizzata la maschera che suddivide in zone la superficie interattiva. Nel sezione sinistra sono invece presenti i comandi per scegliere tra diverse maschere e per inserire la porta e l'indirizzo del client *OSC*.

tz [17] non offre questa possibilità, dal momento che qui gli accordi sono concatenati per grado di prossimità.

Se si osserva infatti la parte superiore dello schema di Fig. 4 si vede come dal II grado si possa accedere direttamente solo al IV, mentre per toccare il VI si debba necessariamente transitare per il IV. Per questa ragione abbiamo pensato di duplicare i vertici che rappresentano il I, il III e il V grado della scala (C, E e G), dilatando la maglia del *tonnetz* originale per ricavare una zona interna che possa permettere il passaggio ad una qualunque delle restanti cinque zone senza dovere necessariamente toccare le altre. Normalizzando la figura ne risulta una forma a corona circolare, in cui la parte superiore contiene le tre armonie fondamentali, mentre l'altra parte ospita le armonie parallele. Questa disposizione è riprodotta dalla maschera di Fig. 3.

4. I TEST SPERIMENTALI

L'applicazione *Harmonic Walk* per l'apprendimento della musica tonale è stata sottoposta a una serie di test effettuati presso la Scuola Elementare di Paderno Franciacorta (Brescia) nel mese di Maggio 2014. Hanno partecipato ai test un totale di 56 bambini così suddivisi: 8 di I elementare e 16 di II (ex primo ciclo); 16 di IV elementare e 16 di V (ex secondo ciclo). Lo scopo dei test era raccogliere una prima serie di informazioni utili per i successivi sviluppi del sistema osservando le reazioni dei bambini rispetto a diverse disposizioni spaziali e a diversi *feedback* sonori. Nessuno dei bambini che hanno partecipato al test aveva particolari conoscenze musicali oltre a quelle fornite dal normale curriculum scolastico. Nonostante non sia sempre stato possibile condurre i test in maniera formale, tuttavia essi hanno rivelato aspetti interessanti circa la percezione dell'applicazione da parte dei bambini. Consideriamo quindi il materiale raccolto come una serie di spunti utili a progettare in futuro dei veri e propri test di usabilità dell'applicazione.

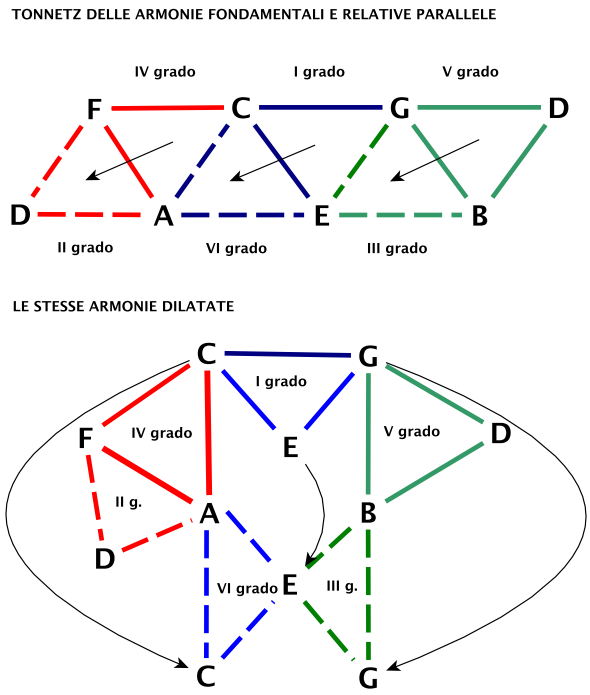


Figura 4. La rappresentazione spaziale delle armonie fondamentali con le loro relative parallele (indicate con una freccia nella figura superiore). La figura in basso mostra la dilatazione rispetto alla disposizione originale. Le frecce tonde indicano la proiezione dei vertici dei triangoli duplicati.

4.1 Tipologie e materiali

La maggior parte dei test sono stati svolti in parallelo con due gruppi di bambini, uno dell'ex primo ciclo e l'altro dell'ex secondo ciclo. Sono state effettuate due diverse tipologie di esperimenti. Una prima tipologia in cui ai bambini è stato chiesto di esplorare liberamente la superficie interattiva utilizzando tre diverse soluzioni di rappresentazione spaziale degli accordi, e una seconda in cui si è proposto di accompagnare la melodia di una canzone popolare a loro nota, muovendosi nello spazio dell'applicazione. Va precisato che si è proceduto alla seconda tipologia (armonizzazione della melodia) solo nel caso che la prima tipologia (esplorazione) era stata sperimentata con successo, cioè una volta che il bambino aveva dimostrato di sapersi muovere agevolmente nell'ambiente interattivo, eseguendo correttamente i collegamenti accordali. Entrambe le tipologie di esperimento sono state poi effettuate utilizzando due diversi tipi di *feedback* sonoro.

4.1.1 La disposizione degli accordi nello spazio

Sono state sperimentate le seguenti tre disposizioni spaziali:

A - Corona circolare divisa in sei sezioni, con la posizione delle zone segnata sul tappeto.

B - Corona circolare divisa in sei sezioni, senza indicazione delle zone sul tappeto.

C - Metà corona circolare divisa in tre sezioni, contenenti le sole armonie fondamentali, senza indicazione delle zone sul tappeto.

Le disposizioni A e B sono state testate con l'obiettivo di osservare quanto e come la presenza di un marcatore spaziale influenzasse il comportamento dei bambini, mentre la disposizione C è stata testata per verificare se l'utilizzo dei soli accordi fondamentali non potesse facilitare l'utilizzo dell'applicazione.

4.1.2 Il feedback sonoro

Per gli esperimenti sono stati utilizzati due diversi tipi di *feedback* sonoro.

1 - Accordi generati con *sustain* costante, ottenuti tramite un processo di sintesi di diverse forme d'onda con semplici inviluppi ADSR e senza alcuna componente ritmica.

2 - Accordi generati utilizzando un sintetizzatore MIDI con una pulsazione ritmica regolare.

Il primo tipo di *feedback* produce accordi dal timbro non riconducibile a strumenti acustici. Il *sustain* viene mantenuto per tutto il tempo di occupazione della zona e decresce fino a scomparire non appena la zona viene abbandonata. Il secondo tipo di *feedback* utilizza timbri strumentali più riconoscibili e, soprattutto, l'accordo è ripetuto secondo una pulsazione costante, riproducendo una situazione sonora assimilabile a semplici *pattern* di accompagnamento della melodia.

4.2 L'esplorazione dello spazio armonico

Per i test effettuati nella fase di esplorazione sono stati scelti prevalentemente bambini che utilizzavano il sistema per la prima volta e che non avevano assistito ad altri esperimenti nell'ambiente, con lo scopo di ottenere alcuni dati indicativi sull'interazione diretta tra i bambini e lo spazio armonico il più possibile priva di filtri e di condizionamenti. La richiesta del conduttore del test o dell'educatore era di esplorare liberamente lo spazio interattivo, senza aggiungere altri commenti o osservazioni per un tempo di circa tre minuti. L'osservazione dei test, effettuati nelle tre disposizioni A, B e C utilizzando il *feedback* sonoro di primo tipo rivelano che la disposizione A, con le zone segnate, si dimostra abbastanza robusta ed esecutivamente affidabile. I bambini individuano facilmente le zone da occupare, ottenendo il risultato sonoro con un buon margine di correttezza. Rilevata tuttavia una certa attitudine a seguire passivamente i segni sul tappeto senza soffermarsi ad ascoltare, si è provato a sperimentare se, utilizzando la stessa disposizione accordale senza segnare le zone, non si potesse indurre i bambini a un comportamento più consapevole. Su sedici bambini del II ciclo solo cinque hanno individuato la corona circolare lungo la quale erano disposti gli accordi, e sono stati quindi in grado di attivare un'efficiente modalità esecutiva, senza peraltro andare oltre. Il test effettuato secondo la disposizione C, teso a facilitare il discernimento degli accordi riducendone il numero e privilegiando le sole armonie fondamentali, ha in realtà quasi annullato la percezione dell'applicazione, rendendo insignificante la stessa esplorazione. Molti bambini hanno messo in atto schemi motori appresi in altri contesti (come correre, ballare o strisciare sul tappeto) dimostrando un'attenzione quasi nulla al *feedback* sonoro, percepito come casuale e privo di significato. Un ulteriore tentativo di indurre comportamenti più consapevoli è stato fatto modificando il *feedback* sonoro, cambiando dal primo tipo al secondo e mantenendo la disposizione B. L'obiettivo di questo esperimento era verificare se la maggior definizione del *feedback* sonoro potesse compensare la mancanza di segni identificativi delle varie zone. L'esperimento, anche se condotto parzialmente, si è rivelato assai significativo. Innanzitutto si sono molto allungati i tempi di permanenza nell'applicazione da parte dei bambini, che hanno mostrato maggior gradimento, maggior interesse nell'esplorazione e maggior attenzione ai cambi di zona. In particolare i cambi sono diventati naturalmente più regolari, come conseguenza della natura ripetitiva della pulsazione dell'*input* accordale. Anche la qualità dell'esecuzione si è fatta più precisa, nonostante l'assenza di marcatori visivi. Dati gli esiti positivi dell'esplorazione, si è proceduto nella stessa sessione sperimentale all'armonizzazione della melodia.

4.3 L'accompagnamento della melodia

Questa seconda sperimentazione è stata attivata solo nei casi in cui la localizzazione degli accordi e la loro esecuzione fossero stati positivamente sperimentati dal soggetto nella fase di esplorazione. Ciò è avvenuto in un caso su cinque nel test con la disposizione accordale A e *feedback* sonoro n. 1 e in cinque casi su dodici nel test con disposizione accordale B (senza segni sul tappeto) e *feedback* sonoro n. 2. In questa sperimentazione l'attività è stata diretta dall'educatore, che poteva intervenire con suggerimenti in dipendenza del comportamento del bambino. Da osservare che, rispetto alla fase di esplorazione, qui vengono coinvolte le abilità cognitive tipiche della composizione tonale analizzate nella Sez. 2 e cioè:

- l'individuazione delle tappe del percorso armonico e della loro struttura ripetitiva (ricorsività)
- la percezione temporale dei cambi di armonia secondo il metro della melodia (temporizzazione).

Dopo qualche tentativo le zone sono ugualmente individuate, anche se con qualche errore. Piuttosto, la pulsazione regolare del *feedback* sonoro n. 2 richiede tempi precisi nel cambio di zona armonica e non tutti i bambini sono in grado di individuare in corrispondenza di quale sillaba del testo della melodia questi si collochino. Fra i cinque bambini che sono riusciti ad armonizzare la melodia, solo uno di undici anni è stato in grado di raggiungere l'obiettivo senza essere guidato dallo sperimentatore.

5. CONCLUSIONI E SVILUPPI FUTURI

L'insegnamento dell'armonia è un obiettivo ambizioso ma cruciale dell'educazione musicale. Già in questa fase di sperimentazione preliminare si è potuto confermare come

Harmonic Walk possa costituire uno strumento utile a tale fine. Dai risultati ottenuti nei test si possono ricavare diverse indicazioni sui futuri sviluppi dell'applicazione. In primo luogo la scelta del *feedback* sonoro, che si è rivelato un mezzo assai potente per influenzare il comportamento dei bambini all'interno dell'applicazione. A questo riguardo si dovrà individuare una metodologia per la scelta di *feedback* sonori che risultino appaganti per i bambini, in modo da indurli ad un ascolto attento e ad un comportamento attivo nell'utilizzo di *Harmonic Walk*.

Infine la questione più critica ma fondamentale per lo sviluppo di *Harmonic Walk*: favorire un utilizzo creativo dell'armonia tonale, la cui conoscenza può portare all'esecuzione in maniera autonoma dell'accompagnamento di una melodia. Ciò riguarda sia la formazione musicale dei bambini, sia le modalità di interazione con l'applicazione. La didattica, favorendo pratiche di ascolto e discernimento, può fornire ai bambini gli strumenti basilari per avvicinarsi al mondo dell'armonia, mentre le attività da svolgere possono essere agevolate attivando meccanismi premiali che incoraggino i bambini a conseguire il completamento dei loro obiettivi all'interno dell'applicazione.

5.1 Ringraziamenti

Si ringraziano il Dirigente della Scuola Primaria di Paderno Franciacorta (Bs), tutto il personale della scuola e in particolare le docenti Marilena Abrami e Mariangela Agazzi per la loro partecipazione ai test sperimentali di *Harmonic Walk*, effettuati nel mese di Maggio 2014.

6. BIBLIOGRAFIA

[1] J. Bruner, *Toward a theory of instruction*. Belknap Press of Harvard University Press, 1966.

[2] S. Papert, *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc., 1980.

[3] H. Gardner, *Frames of Mind: The Theory of Multiple Intelligences*. Basic, 1983.

[4] R. Moreno and R. Mayer, "Interactive multimodal learning environments," *Educational Psychology Review*, vol. 19, no. 3, pp. 309–326, 2007.

[5] A. F. Bobick, S. S. Intille, J. W. Davis, F. Baird, C. S. Pinhanez, L. W. Campbell, Y. A. Ivanov, A. Schütte, and A. Wilson, "The kidsroom: A perceptually-based interactive and immersive story environment," *Presence: Teleoperators and Virtual Environments*, vol. 8, no. 4, pp. 369–393, 1999.

[6] S. Gumtau, P. Newland, C. Creed, and S. Kunath, "Mediate: A responsive environment designed for children with autism," in *Proceedings of the 2005 International Conference on Accessible Design in the Digital World*, Accessible Design'05, (Swinton, UK, UK), pp. 14–14, British Computer Society, 2005.

[7] M. C. Johnson-Glenberg, D. Birchfield, P. Savvides, and C. Megowan-Romanowicz, "Semi-virtual embodied learning-real world stem assessment," in *Serious Educational Game Assessment*, pp. 241–257, Springer, 2011.

[8] S. Zanolla, S. Canazza, A. Rodà, A. Camurri, and G. Volpe, "Entertaining listening by means of the Stanza Logo-Motoria: an Interactive Multimodal Environment," *Entertainment Computing*, vol. 2013, no. 4, pp. 213–220, 2013.

[9] B. Manaris, D. Johnson, and Y. Vassilandonakis, "Harmonic navigator: A gesture-driven, corpus-based approach to music analysis, composition, and performance," in *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2013.

[10] S. Holland, *Learning about harmony with Harmony Space: an overview*. Springer, 1994.

[11] S. Holland, P. Marshall, J. Bird, S. Dalton, R. Morris, N. Pantidi, Y. Rogers, and A. Clark, "Running up blueberry hill: prototyping whole body interaction in harmony space," in *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, pp. 93–98, ACM, 2009.

[12] A. Rossi, *200 canti*. Carish, 1981.

[13] F. Lerdahl and R. Jackendoff, *A Generative Theory of Tonal Music*. Mit Press series on Cognitive theory and mental representation, MIT Press, 1985.

[14] J. Sloboda, *La mente musicale*. Il Mulino, 2002.

[15] L. Amico, "La stanza logo-motoria. un ambiente multimodale interattivo per l'insegnamento a bambini in situazione di multi-disabilità," Master's thesis, Dipartimento di Ingeneria dell'Informazione, Università di Padova, 2012.

[16] S. Jabri, Z. Duric, H. Wechsler, and A. Rosenfeld, "Detection and location of people in video images using adaptive fusion of color and edge information," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, vol. 4, pp. 627–630, IEEE, 2000.

[17] M. Mandanici, A. Rodà, and S. Canazza, "The harmonic walk: an interactive educational environment to discover musical chords," *In via di pubblicazione in Proceedings of ICMC-SMC Conference 2014, Atene*, 2014.

CRAFTING SOUNDS WITH THE POTTER

DIBRIS-University of Genova
Viale Causa 13, 16145 Genova, Italy

Giacomo Lepri
giacomo.lepri@unige.com

Corrado Canepa
corrado@infomus.org

Gualtiero Volpe
gualtiero@infomus.org

Simone Ghisio
ghisio@infomus.org

Antonio Camurri
toni@infomus.org

ABSTRACT

This paper introduces *The Potter* a cross-modal reflexive interactive application for learning-by-exploring sound and its qualities. It is conceived mainly for children. The child is guided to perceive different basic qualities of sound, such as timber, pitch, loudness, and density, i.e., sound morphology. The adopted interaction paradigm is the manipulation of a sound object. Interaction with sound is simultaneous, i.e., sounds are produced as the child moves. The metaphor it builds upon can be summarized as follows: As the potter takes the clay and moulds it yielding a pot, in the same way the sound artisan takes a sound and moulds it yielding a crafted sound.

1. INTRODUCTION

If on the one hand, sound and music are nowadays more and more ubiquitous and accessible even to non-professionals, on the other hand such a trend risks to put music in the background, as something that more or less constantly accompanies human activities, but which is not really the focus of our attention. To give music its cultural and social role back, it is important to start music education since early childhood. The first fundamental step in this process is to stimulate attention to listening to sound and its qualities.

This paper introduces *The Potter*, an application having a twofold objective: (i) stimulating the attention of a child with respect to sound, (ii) learning-by-exploring sound morphology and its major qualities. It is conceived mainly for children 5-9 years old. The child is guided to perceive and scout different basic qualities of sound, such as timber, pitch, loudness, and spectral density, i.e., sound morphology.

The interaction paradigm The Potter adopts requires that the user images to manipulate with the hands a virtual object. Since the pioneering concept of Theremin [5], the design of interfaces for real-time moulding of sound and music by means of gesture, with a particular focus on hand gesture, is a very active research topic in the Sound and Music Computing research community. This includes, for example, design of novel digital musical instruments (see [9] for a review), control of digital audio (e.g., [13]) and gestural control of sound synthesis (see [14] for a review).

Applications exploit either computer vision techniques (e.g., [3] and [4]) or dedicated hardware (e.g., [10] and [7]). However, most of such work is intended to support music players in live performances, whereas considerably little attention is devoted to its possible application to music education. Among the examples in this area, for the science center Città dei Bambini, Camurri and colleagues, developed several interactive hands-on applications to manipulate and experience sound by means of hand and body gesture [2]; Bevilacqua and colleagues developed a complete gestural interface built to support music pedagogy [1]; *SoundBlocks* is a tangible environment where young users connect blocks for digital sound manipulation [6]; Machover carried out *The Toy Symphony* project, providing an integrated series of activities as an alternative entry for children into music. In particular Music Toys is an instruments that do not require any special skill but which do reward curiosity, imagination, and expression [8].

The Potter grounds on the concept of reflexive interaction [11]. This is based on the idea of letting users manipulate virtual copies of themselves. Reflexive interactive systems are designed as intelligent mirrors of the user's behavior. The Potter addresses reflexive interaction since the crafted sound, which is produced and stored, is the direct result of the manipulation of the child, i.e., it is a kind of cross-modal mirror reflecting her movement.

Section 2 discusses the metaphor The Potter is grounded on, Section 3 describes the set-up of the application, Section 4 presents how the application works, i.e., how sounds are crafted, Section 5 provides a technical description.

2. METAPHOR

The metaphor The Potter builds upon can be summarized as follows: As the potter takes the clay and moulds

Copyright: © 2014 Giacomo Lepri, Corrado Canepa, Gualtiero Volpe, Simone Ghisio, Antonio Camurri. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

it yielding a pot, in the same way the sound artisan takes a sound and manipulates it yielding a crafted sound object. Indeed, the work in a traditional potter workshop implies an intuitive discipline of gestures and basic conventions.
Moreover, the metaphor creates a context enabling affordances to gestures to manipulate sound: continuous and detailed iterative manipulation of sound is central to the process. By gestural interaction the user has the ability to listen, shape, and "play" a particular sound. Interaction with sound is simultaneous, i.e., sounds are produced as the child moves. Rules apply to movement only, whereas sound manipulation is open to the free and creative interaction of the child. For each session of crafting the focus is on a specific sound object. All the different sounds can be easily related to a well-known physical object/element (e.g., bells, human voice, animals, natural elements, and so on). Each sound is designed to be constantly manipulated through this association. The focal point is the concept of sound object exploration inspired by the theories introduced in Musique Concrète by Pierre Schaeffer (see [12] and [13]). The adopted pedagogical paradigm hence draws on researches on electroacoustic music experiences. The user manipulates the sound which, as a concrete sound object, is characterized by an imaginary physical autonomy. As a piece of clay detains an original shape that can be manipulated, also the sound object has an original morphology, which can be modified through the manual user intervention. It is therefore required from the user a process of objectification that leads to identify the sound in something external and independent. A key issue is also the role of memory: actions and gestures made previously will guide the future experience of the user.
From a more general point of view, the metaphor of the Potter is extremely useful to give children the possibility to experience, through the game, a simple and effective working methodology – such as the artisan's work discipline. In addition, this working process, involves important preliminary abilities related to the learning of a musical instrument. Amongst the others: (i) rational usage of space (proper use of the working areas), (ii) capability of following methodologies and tasks over time (execution of simple procedural steps aimed to carry out the sound manipulation), (iii) development of cross-modal creativity (association gesture/sound, internalization of the relations that occurs between the two media and experience a free creative control over both).

3. SET-UP

In her experience with The Potter, the child operates in front of a piece of furniture that, according to the adopted metaphor, recalls what happens in a potter's workshop. It consists of the following areas, corresponding to the major areas a potter uses during her work (see Figure 1).

3.1 The draft area

The draft area is a repository where the sound artisan keeps the sound objects (the left table in Figure 1). Sound objects are contained in sound pots. Sound objects are sounds of limited duration with specific features. The draft area corresponds to the basket holding the clay in the potter workshop.

3.2 The final area

The final area is a repository where the sound artisan puts the crafted sounds (the right table in Figure 1). The final area corresponds to the table or basket that will hold the manipulated crafted pots.

3.3 The working area

The working area, i.e., the area where the sound artisan carries out her work. This area consists of the personal space of the sound artisan herself. Moreover, the sound artisan requires an area where she can listen to a sound object. This is a table (the small table in the middle of Figure 1) in between draft and final areas.



Figure 1. The Potter's set-up.

4. SOUND CRAFTING

Sound crafting is carried out with the supervision of a teacher. Through a graphical user interface (the teacher interface, see Section 5), the teacher chooses a set of sound objects. Then, she metaphorically fills the sound pots (physical objects) with the selected sound objects. Finally, she chooses the movement mapping, i.e., which morphological qualities of the sound objects are subject to manipulation. This may range from one single quality to a collection of qualities, depending also on the pedagogical objectives of the teacher. For example, the teacher may decide to focus on a single quality at a first stage in order to make the child aware of the variation of such a quality, and move at a second stage to more complex mappings. This allows for different pedagogical paradigms. For instance a linear approach: the different sound qualities are introduced one by one so that

the child can train in perceiving them gradually. A reticular approach can also be adopted: all or most of the qualities are introduced at the same time and the child can experience the interaction between them simultaneously. The child takes a sound pot from the draft area. She puts the pot in the listening area, to listen to the sound object the pot contains. The whole sound object is reproduced once. Then, she grasps the pot and takes in her hands the contained sound. The transition between the state in which the user can listen to the original sound and the state in which she manipulates it, is related to the action of holding the sound objects with the hands. This transition is carried out by the user through the symbolic gesture of inserting one of the two hands inside the pot in order to pick the sound. After "wearing" the pot (see Figure 2) she can start manipulating the carried sound object with her movement. The sound object is a virtual object, so manipulation is performed by moving the hands in the air, rather than acting on a physical object (e.g., the pot). The processed sound is reproduced in loop as long as the child keeps moving. Finally, if the child freezes for a few seconds, the resulting sound is stored. The child can then put again the sound in its pot and place it in the final area. This will be the crafted sound. At a first stage, the interaction of the child with The Potter can even be limited to grasping a sound pot, putting it in the listening area, listening to the sound object, and putting again the pot in the draft area. Especially for young children, this is already an important step, since it stimulates the attention to sound and makes the child appreciate the difference between different sounds. Figure 2 shows a child crafting a sound. The child can explore and manipulate the following sound qualities.

4.1 Pitch

Once the child has taken the sound object in her hands, she can move it in whatever direction. The pitch of the sound object changes with the height of the hands with respect to the floor. The child can even throw the object toward the ceiling or the floor: the object bounces and pitch changes accordingly.

4.2 Lateralization

Keeping the sound object in her hand, the child can move it to her left or her right, listening to the sound moving coherently in that direction. She can also throw the sound object from her left and collect it on her right, listening to the sound moving.

4.3 Dynamic (volume accent)

The child can make the sound object bounce in the working area (e.g., on the floor or on the table between the draft area and the final area). This is associated to a percussive attack. The attack is proportional to the impulse given to the sound object, a percussive sound being associated to a sharp gesture and a slow attack being associated to a smooth gesture.

4.4 Density (variation of the spectral energy)

Compression and stretch of the sound object with both hands determines a variation in the density of the sound timbre. For example, the sound of water can change from the sound of a single drop, when hands are very close to each other, to the sound of rain when hands are in an intermediate position, to the sound of a river or of a waterfall when hands are far from each other. In the same way, multiple voices (e.g., from two to a choir) can be added to a sound (e.g., a bell, a human voice).

4.5 Distortion

The strength applied in the above-defined compression/stretch is associated to a temporary distortion of sound. If the gesture is smooth and effortless no distortion is applied. To experiment a strong distortion without varying density the child can press the sound object with both hands on the table in the working area. The amount of distortion changes with the strength applied to the pressing gesture, making the child perceive a dirtier and noisier sound when the sound object is pressed onto the table. Since the child does not wear any sensor on her hands, the distance from the table provides a first approximation of such strength. A quick sequence of compressions and expansions makes the child clearly perceive the difference between original and distorted sound.



Figure 2. A child crafting a sound with The Potter.

5. TECHNICAL DESCRIPTION

The Potter runs on a personal computer equipped with a Microsoft Kinect sensor. Figure 3 shows its architecture. Thick arrows represent data flows and thin arrows represent specific settings the teacher can configure by means of the teacher interface. The kernel of The Potter is the crafting module. It includes a movement detection and feature computation module operating on the video and Motion Capture (MoCap) data Kinect captures, and an audio processing module, operating on the audio

stream an audio reader provides. Kinect is placed in front of the piece of furniture described in Section 3. Indeed, besides conceptual and interaction design issues, such a set-up also enables to accommodate technical needs. For example, by putting the Kinect sensor in front of the tables, the child is naturally induced to move in front of the sensor, yielding a more robust and reliable analysis of her movement. Kinect was chosen, instead of common video cameras, in order to simplify the set-up so that The Potter can be prospectively installed at schools or kindergartens. Indeed, Kinect can perform background subtraction and motion tracking in almost non-controlled environments, thus removing constraints such as the need of constant background and lights. Moreover, Kinect does not need any further hardware (e.g., frame grabber cards), it can track multiple users, and it provides 3D MoCap data reliable enough for the aims of the application. The movement detection and feature computation module is responsible for (i) tracking the movement of the sound pots, and (ii) extracting features from the movement of the child's hands. In order to track sound pots, these are endowed with a band of tape reflecting infrared light. Each pot has a band of different height. The infrared light emitted by Kinect is thus reflected and captured by the video camera. The different areas are used to detect which pot is currently moving and to track it. This approach has also the advantage that it does not need any special set-up (e.g., specific lighting for colour tracking), being the Kinect sensor enough to exploit it. Kinect Mo-Cap data are processed in order to obtain (i) 3D position of the right and left hand, used to control pitch and lateralisation, (ii) impulse in the hands movement, used to control dynamic accent, (iii) distance between hands, used to control density, (iv) distance between hands and table, used to control distortion. The audio processing module applies real-time signal processing techniques to the audio stream in order to get a strict connection between the digital sound processes and the child's gestures. In order to modify pitch without changing the time dimension (reading speed of the sample), a pitch-shifting algorithm based on the use of variable audio delays is applied. As for lateralisation, sound is spatialised on the right and left channels using a square root pan algorithm to avoid reduction of volume when the sound is in a central position. Dynamic accent is obtained with an impulsive exponential increment of the volume. Variation of density depends on two different audio processes: (i) when the distance between hands is short - usage of filtering through a chain of band-pass filters, and (ii) when the distance between hands is large - activation of multiple delays with different delay times and/or utilization of harmonization algorithms with different simultaneous tuned voices. Distortion is achieved by applying granulation and wave-shaping to the sound material. Motion features and crafted sounds are stored in an archive containing what a child produced along a session. Through the teacher interface, the teacher can choose the draft sound to be used (selected sound set) and which sound qualities the child can explore (selected mapping). A particular possible setting consists of enabling the child to work with her own voice. In this case, one sound pot remains empty. When it is put in the listening area, the child can fill it with her own voice by means of a microphone. Once the pot is filled, the new sound object, i.e., the voice of the child, can be manipulated as usual.

The teacher interface is displayed in Figure 4. The radio buttons on the left enable the teacher to select the sound qualities to be used in the experience. The teacher can enable just one of them, or a subset, or all of them. The box in the middle allows the selection of the sound-set. Several sound sets are currently available (e.g., bells, animals, voices, and so on). The panel on the right enables settings the parameters for recording a new sound to be included in the sound set. In this way, the teacher can for example allow a child to record her own voice and play with it. Finally, the widgets in the bottom allow controlling the volume of the audio output and let the teacher know when the application is ready for recording the sound the child is actually making. Indeed, both for pedagogical and for technical reasons (time needed for filling audio buffers), the child has to play with a sound object for some time (in the order of 10s) before being enabled to record it and to move it to the final area. Movement detection and feature extraction are developed in EyesWeb XMI[1]. Audio processing is implemented in EyesWeb XMI and in Pure Data[2]. These are connected through Open Sound Control[3] (OSC). The different phases of The Potter and the selection of different sounds and mappings are implemented with the support of the MetaEyesWeb platform. This platform enables developing Finite State Machines (FSMs) and to control EyesWeb XMI depending on the current state of an FSM. Python scripts are used to specify FSMs and to control the MetaEyesWeb. The teacher interface is developed with EyesWeb Mobile.

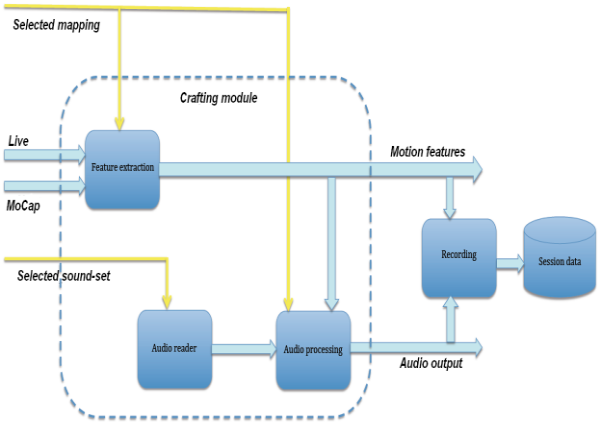


Figure 3. The Potter's architecture. Blue thick arrows represent flows of data and yellow thin arrows represent

[1] http://www.infomus.org/eyesweb_ita.php
[2] http://puredata.info
[3] http://opensoundcontrol.org/introduction-osc

specific settings the teacher can select through the teacher interface.

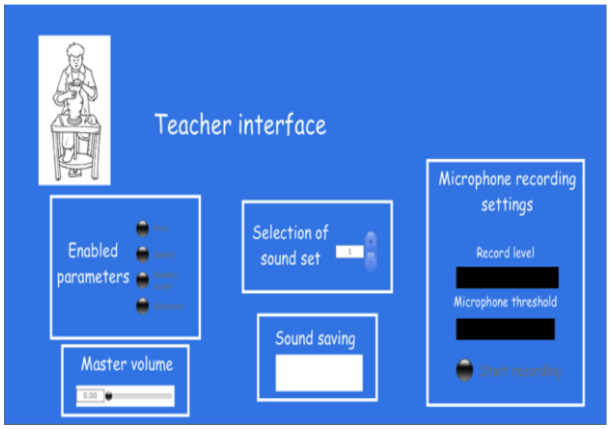


Figure 4. The Potter's teacher interface.

6. CONCLUSION

This paper presented The Potter a cross-modal reflexive interactive application for learning-by-exploring sound and its qualities. Since The Potter is an interactive musical system that requires a simple set-up (a laptop, two speakers and a Microsoft Kinect device) it can be easily installed at schools or kindergartens. The application was tested with children in several occasions. An initial version was presented at Festival della Scienza 2011, a science festival yearly hold in Genova, Italy. Further, a preliminary qualitative evaluation was carried out by psycho-pedagogical partners in specifically organized sessions at University of Gothenburg, Sweden, in April and October 2012. From a musical-pedagogical point of view, the application offers to the users the opportunity to experience an important process of internalization of different sound qualities. Consequently such process allows the acquisition of fundamentals knowledge useful for any kind of musical practice (e.g., playing a musical instrument or listening to and understanding a piece of music). From the interviews with children, it emerged that children show different degrees of awareness of the different sound qualities. The most evident quality was the difference in pitch. Children generally enjoyed interacting with the sound and the technology.

7. ACKNOWLEDGMENTS

This work was partially supported by the EU-ICT Project MIROR (Musical Interaction Relying On Reflexion, www.mirorproject.eu). The authors thank the teams of University of Gothenburg (Åsa Bergman, Bengt Olsson, Pernilla Lagerlöf, Sarah Mercieca) and of University of Bologna (Anna Rita Addessi, Marina Maffioli) for the useful feedback. Special thanks to Giovanna Varni, Paolo Coletta, Michele Pizzi, and Luca Serra for their precious contributions to this work..For a demo of The Potter application visit *https://www.youtube.com/watch?v=_PB29Z_Nq9I&feature=youtube&app=desktop.*

8. REFERENCES

[1] F. Bevilacqua, F. Guedy, N. Schnell, E. Flety, and N. Leroy: "Wireless sensor interface and gesture-follower for music pedagogy", *Proceedings of the 7th international conference on New interfaces for musical expression (NIME '07)*, pp. 124-129, 2007.

[2] A. Camurri and P. Ferrentino: "Interactive environments for music and multimedia". *ACM Multimedia Systems*, 7(1) pp. 32-47, 1999.

[3] J. M. Comajuncosas, A. Barrachina, J. O'Connell, and E. Guaus. Nuvolet: "3D Gesture-driven Collaborative Audio Mosaicing", *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 252- 255, 2011.

[4] S. Şentürk, S. W. Lee, A. Sastry, A. Daruwalla, and G. Weinberg: "Crossole: A gestural interface for composition, improvisation and performance using kinect.", *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME'12), pp.* 21-23, 2012.

[5] A. Glinsky: "Theremin: Ether Music and Espionage", *University of Illinois Press*, Illinois, 2000.

[6] J. Harrison: "Soundblocks and soundscratch: Tangible and virtual digital sound programming and manipulation for children", *Massachusetts Institute of Technology, School of Architecture and Planning, Program in Media Arts and Sciences*, Boston 2005.

[7] J. Homan: "Q&A: The sound catcher", *Nature International weekly journal of science*, pp. 486:472, 2012.

[8] T. Machover: "Shaping minds musically" *BT Technology Journal*, No. 22(4) pp. 171-179, 2004.

[9] E. R. Miranda and M. Wanderley: "New Digital Musical Instruments: Control And Interaction Beyond the Keyboard", *Computer Music and Digital Audio Series* A-R Editions, Inc., Madison, WI, USA, 2006.

[10] A. Mulder, S. Fels, and K. Mase: "Mapping virtual object manipulation to sound variation", *IPSJ Sig Notes*, No. 97(122) pp. 63-68, 1997.

[11] F. Pachet: "Enhancing individual creativity with interactive musical reflective systems", *Musical Creativity: Multidisciplinary Research in Theory And Practice*, I. Deliège and G. Wiggins, editors, Psychology Press, 2006.

[12] P. Schaeffer: "À la recherche d'une musique concrète", *édition du Seuil,* Paris. 1952.

[13] P. Schaeffer: “Traité des objets musicaux”, *édition du Seuil,* Paris 1966.

[14] V. Verfaille, M. M. Wanderley, and P. Depalle: “Mapping strategies for gestural and adaptive control of digital audio effects” *Journal of New Music Research*, No. 35(1) pp. 71-93, 2006.

[15] M. Wanderley and P. Depalle: “Gestural control of sound synthesis” *Proceedings of the IEEE*, No. 92(4) pp. 632-644, April 2004.

AUDIO 3D E ANCORAGGIO SONORO PER L'ESPLORAZIONE MULTIMODALE DI AMBIENTI VIRTUALI

Michele Geronazzo
Università di Padova
Dip. di Ingegneria dell'Informazione
geronazz@dei.unipd.it

Luca Brayda
Fondazione Istituto Italiano di Tecnologia
Dip. Robotics, Brain and Cognitive Sciences
luca.brayda@iit.it

Alberto Bedin e Federico Avanzini
Università di Padova
Dip. di Ingegneria dell'Informazione
avanzini@dei.unipd.it

SOMMARIO

Questo lavoro presenta un sistema interattivo audio-aptico di ausilio all'orientamento e alla mobilità per soggetti non vedenti, e un esperimento soggettivo volto a studiare i meccanismi cognitivi nella rappresentazione spaziale in assenza di informazione visuale. Si presenta in particolare un esperimento di riconoscimento di oggetti, che investiga il ruolo dell'informazione auditiva spaziale dinamica integrata con feedback aptico, in un semplice ambiente virtuale. Tale informazione è strutturata come una "ancora sonora", erogata in cuffia tramite tecniche di rendering 3D binaurale del suono (in particolare tramite *Head-Related Transfer Functions*, o HRTF, opportunamente personalizzate). I risultati sperimentali relativi al tempo di riconoscimento da parte dei soggetti mostrano una relazione tra la posizione dell'ancora sonora e la forma dell'oggetto riconosciuto. Inoltre, un'analisi qualitativa delle traiettorie di esplorazione suggerisce l'insorgere di modifiche comportamentali tra le condizioni monomodale e multimodale.

1. INTRODUZIONE

La nostra interazione con il mondo è intrinsecamente multimodale. In particolare, le interazioni che coinvolgono la modalità uditiva sono legate all'esplorazione attiva dell'ambiente. La natura multimodale dell'interazione rende difficile comprendere e quantificare la rilevanza dell'informazione uditiva nello svolgimento di attività motorie complesse, poiché essa dipende dall'accuratezza e l'accessibilità dell'informazione associata alle diverse modalità sensoriali [1]. Inoltre diverse modalità si combinano e si integrano [2], anche attraverso corrispondenze crossmodali [3]. La comprensione di questi meccanismi è essenziale per progettare interfacce uomo-macchina in cui vari flussi di informazione unimodale si fondano efficacemente in un singolo display interattivo multimodale.

Molti sistemi interattivi in diversi ambiti applicativi fanno un uso crescente dell'informazione spaziale legata al suono, come dimostrato anche dall'introduzione del termine "auralizzazione" già all'inizio degli anni '90 del secolo scorso [4]. Una panoramica [5] sugli approcci usati in letteratura nella sonificazione di quantità fisiche, categorizzati per strategie di mapping e la loro relazione con il mondo fisico, mostra che la dimensione spaziale uditiva è usata tipicamente per sonificare quantità cinematiche. Ciò motiva un'indagine sulla rilevanza di tale dimensione in un contesto di interazione continua con un ambiente virtuale.

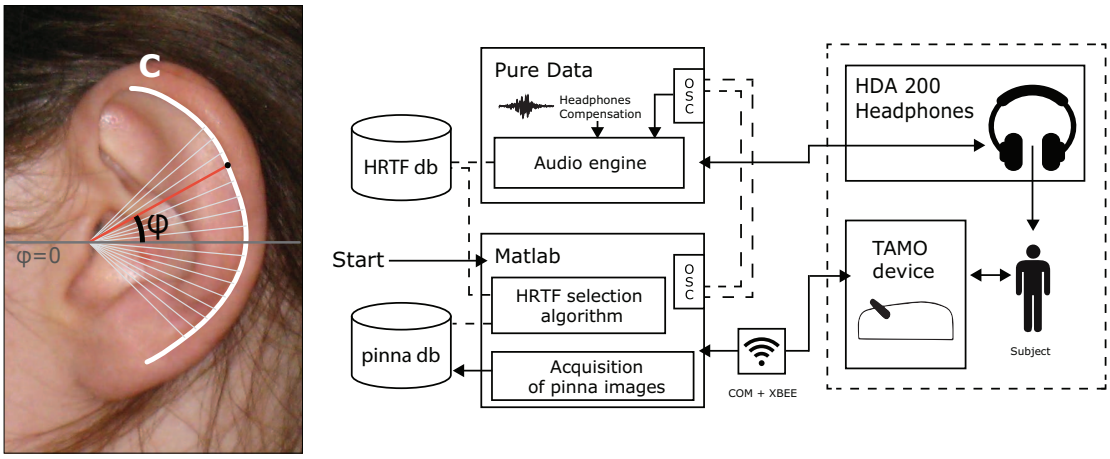
Il feedback uditivo assieme a quello aptico è stato usato in letteratura per approcci di sostituzione sensoriale, allo scopo di verificare se l'acquisizione di informazione spaziale attraverso modalità non visuali rispecchia rappresentazioni interne costruite attraverso la vista [6]. Un esempio notevole di questo tipo di sistemi di sostituzione sensoriale è costituto da dispositivi per l'orientamento e la mobilità (*Orientation & Mobility*, O&M) per soggetti non-vedenti, i quali includono riconoscimento di oggetti/ostacoli, definizione di punti di riferimento, assistenza nella pianificazione di tragitti, e così via.

Un altro aspetto rilevante e non completamente compreso in letterature è la relazione tra l'informazione sensoriale (sia quella "idiotetica", associata a comandi motori per il movimento e alla propriocezione, sia quella "allotetica", associata alla percezione dell'ambiente esterno), e i livelli di attenzione e memoria coinvolti nei processi di apprendimento spaziale [7]. Un dispositivo multimodale per applicazioni in ambito O&M dovrebbe minimizzare i conflitti intermodali, fornire informazioni ecologicamente consistenti con quelle disponibile nel mondo reale, e quindi non appesantire eccessivamente il carico cognitivo richiesto all'utente.

2. COGNIZIONE SPAZIALE

Nella nostra società, la visione è la modalità sensoriale principale nell'espletamento di innumerevoli attività quotidiane, così come nella comunicazione. In particolare, una semplice attività di esplorazione e riconoscimento della forma di un oggetto si complica notevolmente in assenza di informazione visiva. Lederman e Klatzky [8] hanno studiato il legame tra movimenti della mano ed esplorazione aptica di oggetti, con soggetti non-vedenti. I loro risultati mostrano che l'informazione aptica è utile nel riconoscimento di un singolo oggetto, ma non sufficiente in un task più complesso di riconoscimento di layout e strutture spaziali (ad esempio un ambiente 2D con dati di altezza [9]). In maniera complementare, Yu et al. [10] hanno mostrato che la percezione spaziale e la propriocezione sono invece elementi necessari all'esplorazione non-visuale.

Copyright: ©2014 Michele Geronazzo, Luca Brayda, Alberto Bedin e Federico Avanzini. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



(a) (b)

Figura 1. (a) Immagine di lato e contorni dell'orecchio esterno di un soggetto. (b) Uno schema semplificato della configurazione sperimentale.

Nello specifico ambito degli ausili per O&M, la letteratura presenta molti sistemi aptici di sostituzione sensoriale. Le principali limitazioni riguardano gli ingombri, come nel caso del OPtical to TActile CONverter (Optacon) [11], o conflitti intersensoriali, specialmente a danno della modalità uditiva quando presente [12]. Tra i dispositivi proposti, il sistema *HOMERE* [13] usa la metafora del "bastone bianco" per l'esplorazione lungo tragitti predefiniti, e richiede un ingombrante sistema di audio multicanale. Lahav et al. [14] hanno svolto studi qualitativi sull'esplorazione di mappe da parte di soggetti non-vedenti, mostrando una riduzione dei tempi di esplorazione di un ambiente reale, nel caso di soggetti sottoposti a training in ambiente virtuale.

Un ambiente desktop di realtà virtuale più completo e complesso è *BlindAid* [15], il quale usa un Phantom® [1] per il rendering aptico. Azioni e comandi sono tipici di molti contesti di interazione uomo-macchina (come zooming, scrolling, undoing, ecc.). Lo scopo è accelerare il processo di apprendimento spaziale. Il suono è reso spazialmente attraverso *Head-Related Transfer Functions* (HRTF), ossia quei filtri peculiari di ogni ascoltatore che catturano gli effetti acustici della testa umana e delle orecchie [16]. Tuttavia questo sistema fa uso di HRTF generiche misurate su un manichino KEMAR [2], invece che HRTF individuali, pregiudicando l'accuratezza dell'informazione uditiva.

3. ESPLORAZIONE AUDIO-TATTILE

La soluzione tecnologica proposta in questo articolo segue l'idea del "tablet virtuale audio-tattile" per l'esplorazione attiva non visuale di una mappa. L'informazione "locale" viene resa tramite feedback aptico (ad es. urti con oggetti incontrati durante l'esplorazione), mentre un'informazione "globale" viene fornita tramite feedback uditivo (in particolare un punto di ancoraggio sonoro, fisso nell'ambiente virtuale).

I punti di ancoraggio sono collegati ad informazioni tattili locali in relazione biunivoca con ogni punto disponibile nell'area di lavoro; é stato utilizzato un rendering audio basato su HRTF personalizzate, che permettono una simulazione fedele del segnale audio che arriva all'ingresso del canale uditivo in funzione della posizione della sorgente sonora rispetto all'ascoltatore. Le tecnologie binaurali sono infatti fortemente legate alle caratteristiche antropometriche e percettive individuali e hanno un ruolo fondamentale nella caratterizzazione e modellazione di HRTF [17].

3.1 Spazializzazione con audio binaurale

Il nostro sistema utilizza un'innovativa procedura di selezione di HRTF generiche. Per ogni soggetto si seleziona un set di HRTF da un database utilizzando la procedura descritta in [18], con l'obiettivo di ottimizzare automaticamente l'impressione spaziale individuale. L'acquisizione di un'immagine dell'orecchio esterno permette di calcolare una distanza *ad hoc* definita tra contorni dell'orecchio esterno e le caratteristiche a livello di segnale delle HRTF presenti nel database.

Le frequenze centrali dei principali notch di una determinata HRTF nel piano mediano possono essere estratte con ragionevole precisione a partire da alcune caratteristiche peculiari della forma del padiglione auricolare (pinna); viene calcolata la distanza tra un punto all'ingresso del canale uditivo (il punto di *focus*) e ogni punto che si trova sul bordo dell'elice, responsabile della principale riflessione dell'orecchio esterno, Fig. 1(a). Ad elevazione ϕ, la distanza tra il punto di riflessione e il *focus* é di $d(\phi) = ct(\phi)$ dove $t(\phi)$ é il ritardo temporale tra raggi diretti e riflessi, e c é la velocità del suono. Supponendo che ogni coefficiente di riflessione sia negativo e responsabile per un solo notch, la frequenza centrale corrispondente, $f_0(\phi)$, è

$$f_0(\phi) = \frac{c}{2d_c(\phi)}. \quad (1)$$

[1] Un dispositivo aptico con feedback di forza controllato tramite uno stilo: http://www.sensable.com

[2] http://kemar.us/

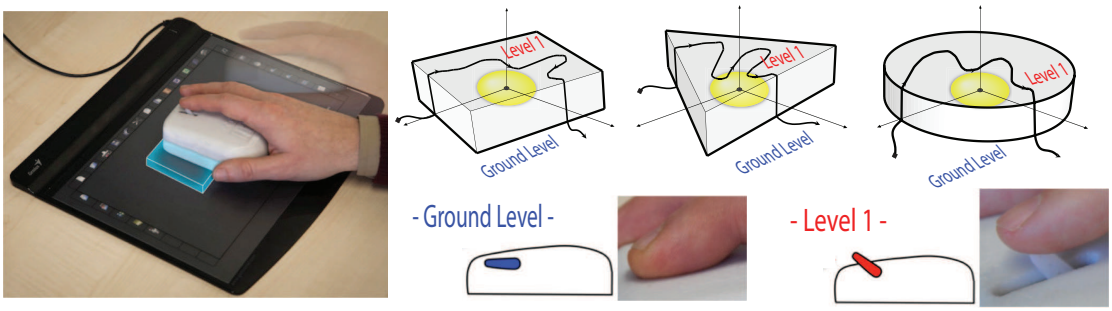


Figura 2. Il dispositivo TAMO in un singolo frame di esplorazione di un oggetto virtuale, raffigurato in azzurro (a sinistra). Le traiettorie di esplorazione (linee nere continue) di tre oggetti virtuali con un'ancora sonora abbozzata in giallo (in alto). Due stati della leva del TAMO (in basso): quando il puntatore é sopra l'oggetto virtuale (Livel 1) la leva sale; quando il puntatore non tocca l'oggetto virtuale (Ground Level), la leva si abbassa.

Dato un soggetto per cui le HRTF individuali non siano disponibili, la precedente relazione permette di selezionare da un database una HRTF generica che abbia il minimo *mismatch* m tra le frequenze estratte dai contorni personali dell'orecchio esterno, f_0, e le frequenze centrali dei notch, F_0, del piano mediano per la HRTF disponibile:

$$m = \frac{1}{|\phi|} \sum_{\phi} \frac{|f_0(\phi) - F_0(\phi)|}{F_0(\phi)}, \qquad (2)$$

dove l'elevazione ϕ spazia tutti gli angoli frontali disponibili tra $-45°$ e $45°$, guidando il confronto tra le frequenze centrali dei notch delle HRTF prese in considerazione con i corrispondenti punti di riflessione ai bordi dell'elice.

3.2 Mouse tattile

Abbiamo scelto un feedback tattile minimalista, il dispositivo TActile MOuse (*TAMO*) [19] che fornisce uno stimolo tattile minimale con una metafora di interazione tipo mouse (v. Fig. 2). Una leva mobile posta sopra il dispositivo al posto della comune rotellina crea un contatto tattile con il polpastrello del soggetto: quando il puntatore raggiunge un oggetto virtuale sulla mappa, la leva segnala un bordo virtuale di una certa altezza. Il feedback tattile corrispondente allo spostamento minimo in altezza viene chiamato *taxel*: le informazioni relative ad un *taxel* vengono memorizzate e rese in analogia con il pixel in *computer graphics* [20]. Il TAMO genera un singolo *taxel* per ogni corrispondente pixel dell'area di lavoro (210×297-mm), come in una rappresentazione a bassorilievo. La leva si sposta da una posizione orizzontale corrispondente ad altezza terra, $\varphi_0 \approx 0°$, ad una quasi verticale, corrispondente ad una rotazione di $\varphi_{max} \approx 80°$.

L'utilizzo di tale dispositivo é motivato dalle seguenti considerazioni:

- la combinazione di tatto e di esplorazione attiva é adeguata per il riconoscimento e la ricostruzione di geometrie semplici;
- la risposta minimalista del dispositivo garantisce una curva di apprendimento rapido con importanti implicazioni pratiche per gli utenti finali;
- la maggior parte dei dispositivi tattili, es. il Phantom, sono difficilmente accessibili in termini di costi e per questo motivo non é mai stato riscontrato un loro uso diffuso; contrariamente il TAMO fornisce una soluzione a basso costo.

4. L'ESPERIMENTO

Alcune indagini recenti [21, 19] hanno studiato l'influenza nella qualità della costruzione di mappe tattili della combinazione di fattori quali carico cognitivo, tasso di acquisizione delle informazioni e strategia casuale/identificabile. Seguendo un approccio simile, l'esperimento qui proposto vuole indagare le strategie di esplorazione in un'attività di riconoscimento di oggetti utilizzando degli stimoli audio-tattili. Ai soggetti coinvolti é stato chiesto di riconoscere il più rapidamente possibile semplici oggetti virtuali con forme geometriche di base. Tali oggetti sono stati collocati al centro della zona di lavoro virtuale, ossia lo spazio limitato dai bordi di un tablet (Fig. 2). A seconda delle condizioni di feedback, un suono spazializzato di ancoraggio viene sintetizzato al centro del tablet come indicatore di orientamento globale.

Viene adottata una visuale egocentrica della mappa virtuale in cui il puntatore corrisponde alla testa dell'ascoltatore; spostando il TAMO dal centro verso il lato destro della tavoletta, il suono di ancoraggio viene spazializzato verso sinistra, come se la testa dell'ascoltatore fosse posizionata sul puntatore; tale metafora, chiamata *ears in hand*, é stata proposta da Magnusson et al. [22].

4.1 Partecipanti e apparato

Hanno preso parte all'esperimento nove soggetti (6 maschi e 3 femmine) di età compresa tra i 21 a 40 anni (media 29, DS 5, 7). Tutti normo-udenti e con diversi livelli di esperienza relativi alla partecipazione ad esperimenti psicofisici.

L'esperimento é stato eseguito in una cabina silente. Le cuffie Sennheiser HDA 200 [3] sono state collegate a una scheda audio esterna Roland Edirol AudioCapture UA-101 ad una frequenza di campionamento di 44, 1 kHz. La Figura 1(a) propone una rappresentazione schematica del nuovo prototipo di interfaccia per il rendering aptico che utilizza il TAMO.

4.2 Stimoli multimodali

Una sorgente sonora virtuale viene posta al centro della mappa (Fig. 2.). Il suono spazializzato viene riprodotto in

[3] Queste cuffie dinamiche circumaurali chiuse offrono un'attenuazione passiva che é efficace contro il rumore ambientale, permettendo di mascherare i suoni meccanici del TAMO.

cuffia in accordo con la posizione relativa tra il puntatore TAMO e l'ancora sonora. Lo stimolo uditivo presentato é un treno continuo di impulsi di rumore gaussiano della durata di 40-ms intervallato da un silenzio di durata pari a 30 ms [23]. L'ampiezza massima all'ingresso del canale uditivo viene fissata a 60 dB(A) dando la possibilità ai soggetti di regolarla a piacimento per ottenere un livello confortevole di ascolto.

Viene acquisita l'immagine dell'orecchio esterno di ogni soggetto, ed avviata la procedura di selezione dell'HRTF descritta nella Sez. 3.1. Il rendering della distanza segue la legge dell'inverso del quadrato sull'attenuazione del livello sonoro ed ha il suo massimo quando il puntatore ricopre la posizione della sorgente sonora, e il suo minimo sulla posizione più lontana che sia raggiungibile nell'area di lavoro del tablet. Un cerchio di raggio 25 px ($\approx 12\ mm$) attorno al suono di ancoraggio definisce l'area in cui il feedback uditivo rimane costante.

Gli stimoli uditivi vengono filtrati attraverso il set di HRTF selezionato, e successivamente con un filtro di compensazione per le cuffie ottenuto con l'algoritmo presentato da [24]. Tale compensazione, misurata su un manichino KEMAR senza orecchie, garantisce un'effettiva equalizzazione fino a $8 - 10$ kHz, e replica uno scenario applicativo reale dove non é possibile ottenere dei filtri di compensazione personali.

L'ambiente multimodale da noi proposto é in grado di simulare apticamente la presenza di oggetti virtuali sulla tavoletta come descritto nella Sez. 3.2. Tutte le mappe sono circondate da pareti virtuali sintetizzate con φ_{max} e quando il dispositivo viene spostato al di fuori della zona di lavoro, la leva si muove in maniera alternata da φ_{max} a $\varphi_{max} - 26°$ a frequenza di *refresh* per segnalare il superamento dei confini delimitati dalle pareti virtuali.

4.2.1 Integrazione multimodale

Il sistema proposto sfrutta diversi componenti software / hardware (ad esempio porte COM, adattatori X-Bee®, mouse, interpolazione di HRTF, ecc.). Per questo motivo é fondamentale verificare che i vincoli di tempo reale siano soddisfatti, dando particolare attenzione alla sincronizzazione dello stimolo multimodale all'interno di una finestra temporale di integrazione capace di mantenerne la coerenza percettiva. Dato che i segnali provenienti da diverse modalità sensoriali posseggono diversi tempi di arrivo ed elaborazione all'interno del cervello, una finestra temporale di circa 200 ms garantisce l'integrazione multisensoriale [25].

Per scegliere la frequenza di aggiornamento del rendering multimodale é stata misurata la latenza degli stimoli audio e tattili posizionando due microfoni a condensatore collegati ad un Tascam 680 con frequenza di campionamento di 192 kHz: uno vicino alle cuffie e l'altro in prossimità della leva del TAMO. L'intero sistema introduce un ritardo medio di 68 ms (< 200 ms), guidando la nostra scelta del periodo di *refresh* del rendering multimodale di 80 ms.

4.3 Procedura

Una breve sessione di esercitazione agevola un'introduzione all'esperimento. I soggetti vengono verbalmente informati del dover identificare da bendati un solido geometrico sconosciuto in una mappa virtuale attraverso l'uso di udito (cuffie) e tatto (mouse tattile), utilizzando una precisa metafora di esplorazione; vengono inoltre istruiti nel tenere la testa con la stessa posa per ottenere coerente il rendering spaziale del suono. Lo sperimentatore guida i soggetti bendati verso la cabina silente e successivamente li conduce ad ogni *trial* verso una posizione di partenza rappresentata dal dispositivo TAMO al centro del bordo inferiore dell'area di lavoro.

Ogni *trial* si conclude quando i soggetti hanno identificato verbalmente l'oggetto o dopo un tempo massimo (impostato a 150 s); non vi é limite alle ipotesi che un soggetto può effettuare all'interno del tempo a disposizione, ma non viene fornita alcuna indicazione in merito alle ipotesi non corrette. Se i soggetti non sono in grado di identificare l'oggetto, il *trial* si conclude con esito negativo.

Gli oggetti virtuali di base, sconosciuti ai soggetti, sono: un prisma regolare a base triangolare, un prisma regolare a base quadrata, e un cilindro (vedi Fig. 2). Le dimensioni degli oggetti vengono fissate con riferimento a uno studio precedente [21] ed hanno tutti uguali altezza virtuale ($\varphi_0 + 18°$), ne risultano cosí movimenti del TAMO che spaziano da ≈ 100 mm a 150 mm sopra agli oggetti virtuali. Con questi vincoli, gli oggetti posseggono un'area maggiore del 50% dello spazio di lavoro disponibile. Ogni oggetto viene cosí presentato in due condizioni: (i) TAMO, condizione aptica unimodale, e (ii) TAMO + audio 2D, condizione bimodale. Ne risulta un totale di sei stimoli. Le sequenze di presentazione vengono ordinatamente disposte a quadrato latino rispetto alla forma dell'oggetto. Le condizioni di feedback sensoriale, unimodale e bimodale, vengono presentate in maniera alternata per minimizzare gli effetti di apprendimento.

5. RISULTATI E DISCUSSIONE

Un'analisi preliminare ha rivelato un tempo medio di riconoscimento degli oggetti pari a $72,25$ s nella condizione TAMO e di $62,87$ s nella condizione bimodale, con una deviazione standard di $29,8$ s e $35,9$ s, rispettivamente. Inoltre, i fallimenti nel riconoscimento sono stati sporadici (10 *trial*).

In generale, i prismi regolari sono stati gli oggetti più facili da riconoscere, mentre il cilindro é stato quello più impegnativo (Fig. 3.(A)). Il suono spazializzato di ancoraggio ha avuto inoltre un effetto limitato sul tempo medio di riconoscimento, tranne nel caso di esplorazione di un prisma regolare a base quadrata dando origine ad un aumento del 30% sulle prestazioni tra TAMO e TAMO + Audio 2D con deviazioni standard di $46,62$ s e 23 s, rispettivamente.

In aggiunta all'analisi quantitativa dei tempi di riconoscimento, un'analisi qualitativa delle strategie di esplorazione dei soggetti può essere in grado di rivelare le differenze tra le condizioni unimodali e bimodali. La discussione che segue é incentrata sull'esplorazione di prismi re-

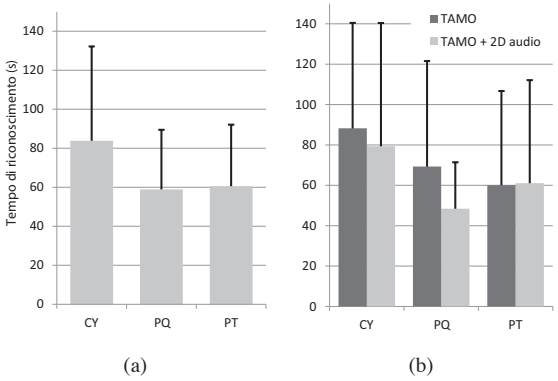


Figura 3. Media e deviazione standard del tempo di riconoscimento per 9 soggetti (a) per ogni oggetto virtuale e (b) raffinato sulle diverse condizione di stimolo. Gli oggetti virtuali sono identificati come: CY - Cilindro, PQ - prisma regolare a base quadrata, e PT - prisma regolare a base triangolare.

golari con base quadrata per i quali sono state riscontrate differenze marcate. Sono state identificate le stesse due strategie principali riportate in [21]:

1. *Grid-scan*: esplorazione della mappa su una griglia con movimenti lungo linee verticali e orizzontali (Fig. 4(c));

2. *Z-scan*: inseguimento dei profili dell'oggetto tentando di attraversare i bordi con movimenti ortogonali o diagonali (Fig. 4(e)).

Possiamo sorprendentemente notare che emerge una terza strategia in presenza dell'ancora acustica: i soggetti seguono i bordi dell'oggetto controllando la distanza dal centro dell'oggetto stesso, corrispondente alla posizione del suono di ancoraggio. Questo comportamento é illustrato nella Figura 4 (seconda colonna) per tre soggetti esemplificativi. Questa nuova strategia promuove la riduzione della quantità di tempo trascorso al di fuori dell'oggetto, guidando i soggetti attraverso un'esplorazione ottimizzata. Un'analisi dei micro-movimenti (piccoli movimenti precisi) e dei macro-movimenti (rapidi cambiamenti macroscopici sulla mappa) potrebbe fornire, in una successiva valutazione sperimentale, un'ulteriore comprensione circa l'uso dell'ancoraggio sonoro in task di esplorazione che combinano attività in campo lontano e vicino. Si può ipotizzare che l'esplorazione di ambienti di grandi dimensioni possa essere supportato in modo più efficace utilizzando un suono di faro fisso. Nel caso in cui l'area di lavoro abbia dimensioni paragonabile a quelle del corpo umano, gli utenti possono modificare la posizione dell'ancora sonora utilizzando i riferimenti spaziali uditivi che man mano vanno a crearsi per confermare le diverse viste dell'oggetto.

6. CONCLUSIONI

Il sistema multimodale proposto in questo articolo fornisce un banco di prova per l'utilizzo di audio spazializzato al servizio della navigazione in spazi virtuali. Un task di

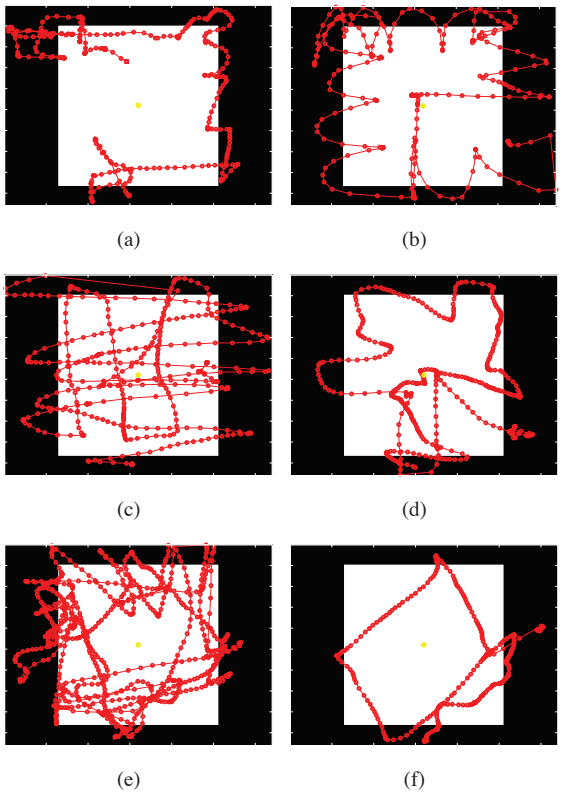


Figura 4. Esplorazione di un prisma regolare a base quadrata da parte di tre soggetti (righe), in condizione unimodale e bimodale (colonne). Le traiettorie sono mostrate in rosso.

riconoscimento di oggetti con soggetti bendati ha fornito dei primi incoraggianti risultati in merito a come i suoni di ancoraggio possano migliorare le prestazioni e modificare le strategie di esplorazione.

Una futura analisi dovrà correlare le caratteristiche dei movimenti esplorativi ai cambiamenti di feedback e strategie, nonché rivelare alcune criticità riguardanti la rappresentazione delle HRTF, la difficoltà di riconoscimento di oggetti complessi, l'efficacia di diversi posizionamenti per i punti di ancoraggio sonoro. Le modifiche temporali e spettrali dell'ancora sonora influenzeranno il carico cognitivo, e di conseguenza le prestazioni nell'attività di navigazione.

7. RINGRAZIAMENTI

Questo lavoro é stato finanziato dal progetto di ricerca PADVA (Personal Auditory Displays for Virtual Acoustics), n. CPDA135702 dell'Università di Padova.

8. BIBLIOGRAFIA

[1] E. J. Gibson and A. D. Pick, *An Ecological Approach to Perceptual Learning and Development*. Oxford University Press, Aug. 2000.

[2] M. O. Ernst and H. H. Bülthoff, "Merging the senses into a robust percept," *Trends in Cognitive Sciences*, vol. 8, pp. 162–169, Apr. 2004.

[3] C. Spence, "Crossmodal correspondences: A tutorial review," *Atten. Percept. Psychophys.*, vol. 73, pp. 971–995, Jan. 2011.

[4] M. Vorländer, *Auralization: Fundamentals of Acoustics, Modelling, Simulation, Algorithms and Acoustic Virtual Reality*. Springer Publishing Company, Incorporated, 1st ed., 2007.

[5] G. Dubus and R. Bresin, "A systematic review of mapping strategies for the sonification of physical quantities," *PLoS ONE*, vol. 8, p. e82491, Dec. 2013.

[6] Z. Cattaneo, T. Vecchi, C. Cornoldi, I. Mammarella, D. Bonino, E. Ricciardi, and P. Pietrini, "Imagery and spatial processes in blindness and visual impairment," *Neuroscience & Biobehavioral Reviews*, vol. 32, pp. 1346–1360, Oct. 2008.

[7] A. Afonso, A. Blum, B. Katz, P. Tarroux, G. Borst, and M. Denis, "Structural properties of spatial representations in blind people: Scanning images constructed from haptic exploration or from locomotion in a 3-d audio virtual environment," *Memory & Cognition*, vol. 38, no. 5, pp. 591–604, 2010.

[8] S. J. Lederman and R. L. Klatzky, "Hand movements: A window into haptic object recognition," *Cognitive Psychology*, vol. 19, no. 3, pp. 342 – 368, 1987.

[9] L. E. Magee and J. M. Kennedy, "Exploring pictures tactually," *Nature*, vol. 283, pp. 287–288, Jan. 1980.

[10] W. Yu and S. Brewster, "Comparing two haptic interfaces for multimodal graph rendering," in *Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2002. HAPTICS 2002. Proceedings. 10th Symposium on*, p. 3–9, 2002.

[11] W. Schiff and E. Foulke, *Tactual Perception: A Sourcebook*. Cambridge University Press, Mar. 1982.

[12] R. Farcy, R. Leroux, A. Jucha, R. Damaschini, C. Grégoire, and A. Zogaghi, "Electronic travel aids and electronic orientation aids for blind people: technical, rehabilitation and everyday life points of view," in *Conference & Workshop on Assistive Technologies for People with Vision & Hearing Impairments Technology for Inclusion*, p. 12, 2006.

[13] A. Lecuyer, P. Mobuchon, C. Megard, J. Perret, C. Andriot, and J.-P. Colinot, "HOMERE: a multimodal system for visually impaired people to explore virtual environments," in *Virtual Reality, 2003. Proceedings. IEEE*, pp. 251 – 258, Mar. 2003.

[14] O. Lahav and D. Mioduser, "Exploration of unknown spaces by people who are blind using a multi-sensory virtual environment," *Journal of Special Education Technology*, vol. 19, no. 3, p. 15–23, 2004.

[15] D. Schloerb, O. Lahav, J. Desloge, and M. Srinivasan, "BlindAid: virtual environment system for self-reliant trip planning and orientation and mobility training," in *Haptics Symposium, 2010 IEEE*, pp. 363 –370, Mar. 2010.

[16] C. P. Brown and R. O. Duda, "A structural model for binaural sound synthesis," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 5, p. 476–488, 1998.

[17] S. Spagnol, M. Geronazzo, and F. Avanzini, "On the relation between pinna reflection patterns and head-related transfer function features," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, p. 508–519, Mar. 2013.

[18] M. Geronazzo, S. Spagnol, A. Bedin, and F. Avanzini, "Enhancing vertical localization with image-guided selection of non-individual head-related transfer functions," in *In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014)*, (Florence, Italy), pp. 4496–4500, May 2014.

[19] L. Brayda, C. Campus, and M. Gori, "Predicting successful tactile mapping of virtual objects," *IEEE Transactions on Haptics*, vol. 6, pp. 473–483, Oct. 2013.

[20] M. Siegel, "Tactile display development: the driving-force for tactile sensor development," in *Haptic Virtual Environments and Their Applications, IEEE International Workshop 2002 HAVE*, pp. 115–118, 2002.

[21] L. Brayda, C. Campus, R. Chellali, G. Rodriguez, and C. Martinoli, "An investigation of search behaviour in a tactile exploration task for sighted and non-sighted adults.," in *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*, CHI EA '11, (Vancouver, BC, Canada), p. 2317–2322, ACM, 2011.

[22] C. Magnusson, H. Danielsson, and K. Rassmus-Gröhn, "Non visual haptic audio tools for virtual environments," in *Haptic and Audio Interaction Design* (D. McGookin and S. Brewster, eds.), no. 4129 in Lecture Notes in Computer Science, pp. 111–120, Springer Berlin Heidelberg, Jan. 2006.

[23] B. F. G. Katz, S. Kammoun, G. Parseihian, O. Gutierrez, A. Brilhault, M. Auvray, P. Truillet, M. Denis, S. Thorpe, and C. Jouffrais, "NAVIG: augmented reality guidance system for the visually impaired: Combining object localization, GNSS, and spatial audio," *Virtual Reality*, vol. 16, pp. 253–269, June 2012.

[24] A. Lindau and F. Brinkmann, "Perceptual evaluation of headphone compensation in binaural synthesis based on non-individual recordings," *J. Audio Eng. Soc*, vol. 60, no. 1/2, p. 54–62, 2012.

[25] N. P. Holmes and C. Spence, "Multisensory integration: Space, time, & superadditivity," *Curr Biol*, vol. 15, pp. R762–R764, Sept. 2005.

CONFRONTO TRA SISTEMI DI RILEVAMENTO DEL GESTO BASATI SU SENSORI A INFRAROSSI O ULTRASUONI PER APPLICAZIONI DI PIANOFORTE VIRTUALE

Yuri De Pra
Università di Verona
Dipartimento di Informatica
Strada le Grazie 15
Verona 37134, Italia
yuridepra@libero.it

Federico Fontana
Università di Udine
Dipartimento di Matematica e Informatica
via delle Scienze 206
Udine 33100, Italia
federico.fontana@uniud.it

Fausto Spoto
Università di Verona
Dipartimento di Informatica
Strada le Grazie 15
Verona 37134, Italia
fausto.spoto@univr.it

SOMMARIO

Presentiamo una tastiera virtuale che mira a sostituire quella del pianoforte con dei tasti in realtà aumentata. Lo sforzo maggiore è stato compiuto nel rilevamento accurato nonché rapido dei movimenti delle mani. Per raggiungere questo obiettivo sono stati confrontati un dispositivo di rilevamento a infrarossi ed uno a ultrasuoni, valutandone pro e contro nell'impiego specifico. Abbiamo poi realizzato un sistema di feedback multimodale riprendendo le azioni delle mani dell'esecutore mediante la fotocamera posteriore di un comune dispositivo mobile, e presentandole all'utente unite alla visualizzazione di una tastiera di pianoforte animata. Connettendo il dispositivo Leap Motion, responsabile del rilevamento a infrarossi, e altri dispositivi hardware per la generazione di stimoli audio e vibrotattili, il prototipo promette potenziali applicazioni come interfaccia musicale mobile a basso costo, trasformando un comune tavolo in uno scenario di realtà aumentata dove possono essere eseguiti semplici esercizi con buona accuratezza e una risposta realistica in assenza di una tastiera reale.

1. INTRODUZIONE

Le interfacce degli strumenti musicali virtuali di nuova concezione sono progettate per assicurare un'esperienza utente completa. Esse emulano, mediante il feedback multimodale, le sensazioni percepite dall'esecutore durante una performance con lo strumento reale. Nello sviluppo di questo tipo di interfacce, non è banale garantire la sincronizzazione nella generazione dei diversi stimoli sensoriali in modo che siano percepiti come un evento unitario. In particolare, la finestra temporale in cui gli stimoli visivi, uditivi e somatosensoriali sono associabili allo stesso evento da parte di un utente è molto ristretta. I diversi esperimenti percettivi che sono stati condotti hanno con buona approssimazione determinato la dimensione massima di tale finestra in 100 ms, nel caso ottimale in cui l'ordine temporale delle modalità associate agli stimoli sia tattile, uditivo e visuale [1, 2, 3, 4].

I sensori dei dispositivi mobili reperibili oggi sul mercato spesso non possiedono l'accuratezza e la reattività richieste per effettuare il riconoscimento e la gestione di eventi nella finestra temporale precedentemente descritta. Inoltre la gestione delle risorse del sistema operativo tende ad assegnare bassa priorità ai processi audio e video. Nel caso specifico del piano portatile, il riconoscimento della pressione dei tasti e della relativa forza applicata in tempo reale è possibile solo mediante periferiche dedicate collegabili al dispositivo mobile secondo le specifiche MIDI-USB. A tutt'oggi non esiste un'interfaccia immateriale che copra il divario esistente nella qualità dell'interazione tra i software di emulazione su dispositivi mobili e il pianoforte fisico.

Malgrado le difficoltà tecniche che non permettono la realizzazione di interfacce su dispositivi mobili, sono stati realizzati diversi progetti software [5] che permettono il riconoscimento di gesti mediante una videocamera a infrarossi a frequenza di cattura elevata. Inoltre esistono progetti software di realtà aumentata per pianoforte basati su soluzioni desktop [6, 7], e studi sulla loro applicabilità in contesti educativi [8, 9].

Possiamo concepire un contesto in cui l'interazione con il pianoforte avvenga su un comune tavolo di legno su cui viene proiettata o disegnata una tastiera. Un'altra possibile implementazione potrebbe prevedere di visualizzare porzioni di tastiera sullo schermo di un tablet portando l'interazione sul touch-screen del medesimo dispositivo. Esistono diverse applicazioni (oltre 250) che sfruttano questo paradigma di interazione; tra queste citiamo Piano 3D (mobileagency.com.au), Real Piano 3D (imudra tech.), Play Piano (android technologies), Learn Piano, My Piano e Grand Piano.

Gli schermi touch-screen consentono un riconoscimento preciso del tocco a livello spaziale e prevedono la possibilità di generare vibrazioni come risposta agli eventi (anche se non modulabili). D'altra parte essi sono affetti da una considerevole latenza nella trasmissione dell'evento e non prevedono funzioni di stima della forza esercitata dalle dita sulla superficie; tale informazione risulta indispensabile per un'applicazione di pianoforte.

Date queste premesse, sono state ideate e testate diverse

Copyright: ©2014 Yuri De Pra, Federico Fontana, Fausto Spoto. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

soluzioni hardware e software atte a costruire un sistema a bassa latenza che consentisse di stimare posizione e velocità relativa delle dita della mano. Nel sistema proposto la realtà aumentata viene realizzata sullo schermo del tablet dove viene mostrata la mano dell'esecutore ripresa dall'alto (mediante la videocamera del dispositivo mobile) a cui viene sovrapposto un modello 3D di tastiera del pianoforte in semi-trasparenza. Attualmente il sistema si avvale di un dispositivo mobile Nexus 10 di Samsung, su cui opera Android 4.4.

L'obiettivo di questa ricerca non è riprodurre fedelmente le sensazioni percepite suonando un pianoforte reale, quanto piuttosto ricreare un'esperienza utente accettabile mediante il supporto di un sistema di feedback multimodale. L'assenza della tastiera fisica non compromette totalmente l'esperienza dell'utente, ancorché la sensazione tattile di contatto con una superficie (per esempio un tavolo) debba essere supportata da una pronta, consistente ed accurata risposta audio/video.

La sintesi in tempo reale del suono di pianoforte a oggi ha raggiunto ottimi risultati; è quindi possibile generare suoni realistici mediante campioni di pianoforte o sintesi a modelli fisici. La riproduzione degli stessi può avvenire tramite trasduttori vibro-tattili posizionati sulla superficie di interazione che simulino la radiazione della tavola armonica del pianoforte.

Per quanto riguarda la visualizzazione, un piccolo ritardo è tollerabile qualora sia rispettato l'ordine degli stimoli: tattile, audio, visuale. Nello specifico, sul sistema presentato dovranno essere effettuati esperimenti per quantificare il grado di coesione della risposta multimodale [4].

In questo lavoro riportiamo i risultati della ricerca compiuta per ottenere un'interazione naturale attraverso l'utilizzo del feedback multimodale. In particolare, l'articolo verte principalmente sugli aspetti del riconoscimento delle dita e sulla stima della velocità dei loro movimenti.

2. RICONOSCIMENTO DELLA MANO

I dispositivi mobili di recente fabbricazione sono dotati di fotocamere ad alta risoluzione ma soffrono di problemi di latenza, soprattutto durante le operazioni di autofocus e compensazione che non possono essere disattivate. Come detto precedentemente, latenze video ridotte possono essere accettate dal pianista nell'interazione; d'altra parte i sistemi di riconoscimento basati sulle immagini risultano inutilizzabili qualora affetti da latenze di quest'ordine di grandezza [10]. Per questo motivo abbiamo investigato soluzioni differenti.

2.1 Sistema basato su sensori a infrarossi

I sensori a infrarossi in commercio come Microsoft Kinect™ e Leap Motion™ hanno ottenuto un buon grado di successo come strumenti per il riconoscimento di parti del corpo e di oggetti, consentendo di determinarne la posizione nello spazio e stimarne la velocità dei movimenti. Per quanto riguarda il caso specifico dell'inseguimento della mano il dispositivo più adatto risulta essere Leap Motion, un dispositivo progettato per il tracking della mano e delle dita a bassa latenza che prevede l'integrazione con i più diffusi linguaggi di programmazione (C++, Java, Pyhon, etc.). Secondo le specifiche il prodotto consente un grado di precisione sub-millimetrico e un'elevata frequenza di cattura (circa 300 fps), consentendo il riconoscimento di movimenti anche rapidi. In realtà, diversi test hanno evidenziato prestazioni al di sotto delle aspettative per quanto riguarda la robustezza dell'inseguimento nel caso di movimenti rapidi o di condizioni di luce non ottimali [11]. Nel nostro studio sono emersi gli stessi limiti, riportati a conclusione dell'articolo.

Nel caso particolare abbiamo sviluppato un modulo software su un laptop PC Intel Core i3, il quale riceve i dati dal driver del dispositivo di rilevamento e invia le coordinate spaziali ad un'applicazione Android tramite protocollo UDP. L'applicazione è progettata per virtualizzare sullo schermo del dispositivo mobile lo scheletro della mano (Figura 1).

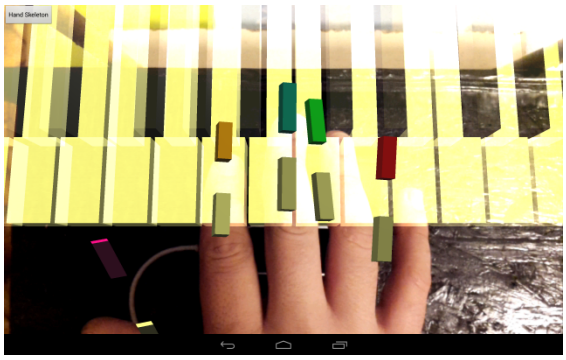


Figura 1. Riconoscimento basato su Leap Motion con virtualizzazione della mano.

Le caratteristiche del sistema di riconoscimento impongono che Leap Motion venga posizionata al di sotto del palmo della mano a una distanza non inferiore a 7 cm; per questo motivo l'utente interagisce con una superficie trasparente agli infrarossi posizionata al di sopra del dispositivo. Questo vincolo limita l'accuratezza della stima della velocità delle dita lungo l'asse di principale importanza per gli scopi dell'applicazione.

Una prima implementazione dell'algoritmo di inseguimento è stata realizzata utilizzando la gesture "key pressed" fornita nativamente dalla API del dispositivo. Tale metodo restituisce la posizione e stima della velocità delle dita sugli eventi riconosciuti. Al seguito di alcuni test preliminari è stato abbandonato l'approccio a causa di eccessivi falsi negativi e positivi nonché di una considerevole latenza [1] .

Una successiva implementazione è stata realizzata sulla base delle coordinate spaziali rilevate dal dispositivo e fornite dalla API. Sono state create due soglie virtuali poste ad altezza predeterminata dalla superficie in vetro (2 cm e 0.5 cm): è stato generato un evento di pressione al raggiungimento della seconda soglia ed è stato calcolato il tempo

[1] Al momento della scrittura di questo manoscritto il produttore di Leap Motion ha rilasciato una nuova versione della API, le cui prestazioni a una valutazione preliminare degli autori non appaiono ancora soddisfacenti.

di percorrenza del dito nella finestra per determinarne la velocità. Le prestazioni di questa implementazione sono risultate migliori sia in termini di latenza che di precisione nella cattura degli eventi.

2.2 Sistema basato su sensori a ultrasuoni

Esistono tecniche alternative di riconoscimento, basate per esempio sull'utilizzo di ultrasuoni [12] e dispositivi laser [13] dotati sia di emettitore che di ricevitore. Questi trasduttori sono programmati per emettere a intervalli regolari una data sequenza di impulsi, le cui riflessioni su eventuali ostacoli sono rilevate dal ricevitore. Con questa tecnica si stimano posizione e distanza degli ostacoli. Non abbiamo trovato in letteratura applicazioni di pianoforte basate sul riconoscimento laser o a ultrasuoni.

Abbiamo dunque realizzato un sistema costituito da una matrice di otto sensori a ultrasuoni HC-SR04 (Figura 2) in grado di ottenere un'elevata frequenza di cattura (circa 200 fps) se configurato per riconoscere oggetti posti ad una distanza tra 20 e 30 millimetri; l'elevata frequenza è possibile grazie alla breve distanza percorsa dagli impulsi prima di incontrare l'ostacolo ed esserne riflessi.

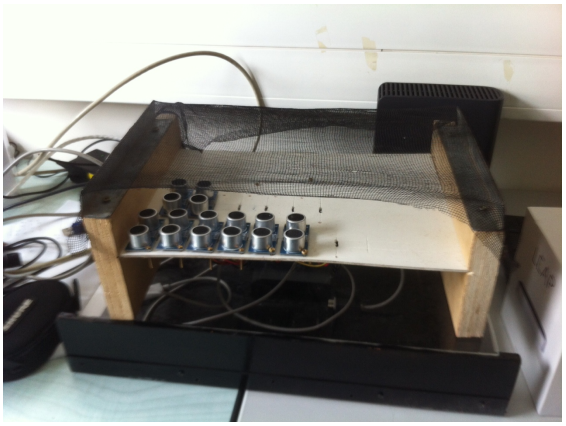


Figura 2. Matrice di sensori a ultrasuoni per 8 tasti.

In una prima implementazione la matrice di sensori è stata pilotata da un microcontrollore Arduino 2009, collegato a sua volta tramite porta seriale a un laptop PC che generava segnali MIDI inviati a un'applicazione Android utilizzando il protocollo UDP (Figura 3). Successivamente è stato utilizzato il controllore Arduino ADK, collegandolo direttamente al dispositivo mobile mediante l'apposita interfaccia seriale mini USB. La seconda implementazione è stata realizzata al fine di ridurre la latenza di comunicazione e aumentare a 14 il numero di sensori pilotabili; per ogni sensore, infatti, è necessario utilizzare un'uscita digitale per inviare il segnale di trigger e un'uscita di tipo PWM per determinare il tempo di riflessione e la corrispondente distanza dell'oggetto.

Il sistema di riconoscimento è basato su un algoritmo operante su tre soglie di distanza, che determina il sensore coinvolto e stima la velocità del dito durante la discesa. Per determinare la velocità viene trasmesso il tempo del microprocessore in ogni frame generato nel superamento di una delle soglie stabilite per un determinato sensore.

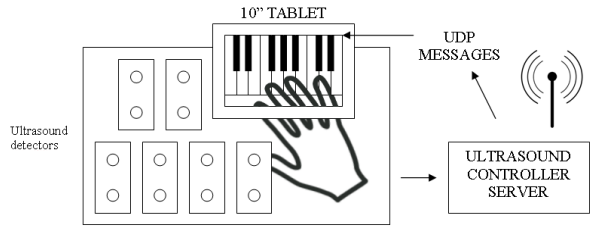


Figura 3. Sistema di realtà aumenta con rilevamento a ultrasuoni.

Nel messaggio inviato all'applicazione è stato inserito il numero del sensore, il valore di soglia e il tempo di generazione in modo da evitare distorsioni nell'informazione dovuti al canale di comunicazione. Il calcolo della velocità e la creazione dell'immagine dello stato del sistema sono stati demandati all'applicazione Android. Le tre soglie servono a determinare rispettivamente: il punto di attivazione del sensore, l'inizio del calcolo della velocità e il punto finale dell'azione. Una corretta configurazione delle soglie permette di sviluppare strategie predittive che possono compensare i ritardi di altri componenti del sistema nell'elaborazione dell'informazione

3. FEEDBACK

Come detto in precedenza la mano viene visualizzata sullo schermo del tablet sovrapposta alla tastiera virtuale. Per fare ciò è stata usata la fotocamera del sistema mobile, e quindi la visualizzazione risulta affetta da una latenza a oggi irriducibile. Il risultato è comunque piacevole data la risposta coerente del sistema alle azioni compiute dall'utente: risultano infatti sincronizzate, sebbene affette da latenza, la visualizzazione della mano e dei movimenti dei tasti controllati dal sistema. La visualizzazione sullo schermo del tablet può essere sostituita da altri riferimenti visivi (es. tastiera disegnata o proiettata su un tavolo); ogni possibile opzione dovrà essere valutata da rigorosi esperimenti.

La parte relativa all'audio non è stata considerata prioritaria in questo stadio della ricerca: al momento viene utilizzato un set di campioni di pianoforte PCM di buona qualità. La parte riguardante il feedback aptico è stata meno esplorata, ma dovrà essere effettuato ulteriore lavoro di ricerca al fine di simulare la resistenza del tasto del pianoforte quando viene premuto. Riportiamo sotto i risultati riguardanti questa problematica.

3.1 Feedback Tattile

Riprodurre, anche parzialmente, la consistenza di un tasto di pianoforte in modo virtuale non è banale. In letteratura sono state studiate tecnologie basate su ultrasuoni [14, 15], generazione di vortici d'aria [16] e repulsione magnetica [17] al fine di riprodurre la materialità di oggetti virtuali interattivi, ottenendo promettenti risultati. È stato fatto un tentativo di applicare tali tecnologie nel contesto della riproduzione del tasto di pianoforte considerando le esigenze di portabilità del sistema.

Nel caso dell'inseguimento visuale è stata utilizzata una superficie trasparente agli infrarossi per determinare il punto di contatto necessario a fornire fisicità all'interazione dell'utente con l'interfaccia; soluzioni simili sono state adottate in applicazioni come Cristal Piano [11].

L'interazione con la matrice di sensori a ultrasuoni ha sollevato il problema di selezionare materiali trasparenti agli stessi, in grado di ricreare il feedback tattile; per questo motivo si è scelto di utilizzare una rete elastica con maglie di circa 2 millimetri posta ad una distanza di 33 millimetri dalla matrice sensori (Figura 2). Questa soluzione ha creato un interessante effetto somatosensoriale in quanto la resistenza della rete elastica cresce proporzionalmente alla pressione del dito sulla stessa.

In alternativa è stato chiesto ai pianisti di indossare dei guanti costruiti ponendo dei piccoli magneti scorrevoli sulle punte delle dita (Figura 4); è stata predisposta una parte

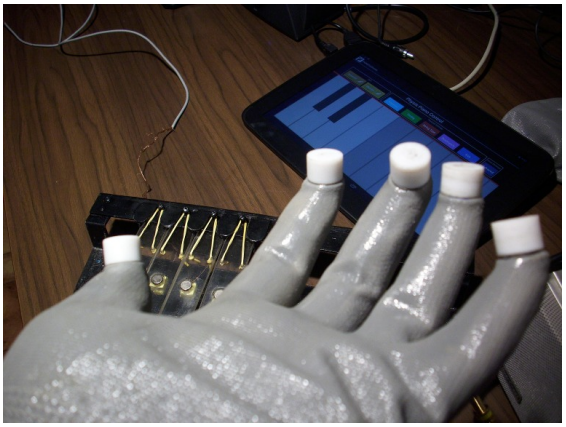


Figura 4. Guanti magnetici.

di tavolo con una serie di magneti polarizzati inversamente rispetto a quelli dei guanti nei punti corrispondenti ai tasti del pianoforte. La forza repulsiva risultante e la conseguente sensazione sono state valutate realistiche, ma il disagio manifestato dai pianisti nell'indossare dei guanti così costruiti ha portato a escludere questa soluzione applicativa.

Sono stati fatti dei tentativi di concentrare fasci di ultrasuoni utilizzando trasduttori ad alta potenza. I segnali sono stati sfasati temporalmente in modo da creare dei punti di fuoco posti a pochi centimetri dalla superficie del tavolo. Al momento non sono emersi risultati significativi, probabilmente a causa del ridotto numero di trasduttori impiegati.

4. RISULTATI

In Tabella 1 riportiamo una comparazione tra i sistemi di rilevamento, illustrandone le differenze in termini di prestazioni e costo computazionale.

La Leap Motion consente una rilevazione continua e molto precisa mentre la matrice di sensori a ultrasuoni prevede di utilizzare un sensore per ogni tasto. L'algoritmo attuale utilizzato dalla Leap Motion non permette un corretto impiego nel tracking verticale dei movimenti delle dita in quanto, molto frequentemente, la rilevazione viene

Proprietà	**Leap Motion**	**Ultrasuoni**
Punti rilevati	Rilevamento continuo	Un sensore per tasto
Requisiti hardware	Android e laptop	Android e Arduino ADK
fps	30∼300	200
Falsi positivi	eccessivi	limitati
Falsi negativi	limitati	nessuno
Latenza	accettabile	molto ridotta
Feedback tattile	superficie rigida	continuo e proporzionale
Tolleranza dei soggetti	bassa (guanti)	buona (rete elastica)
Integrazione diretta con Android	non possibile al momento	utilizzando Arduino ADK (USB)

Tabella 1. Comparazione tra Leap Motion e matrice di sensori a ultrasuoni

interrotta a causa di movimenti rapidi della mano. Distanze inferiori ai 7 centimetri non sono rilevabili impedendo la costruzione di prototipi facilmente trasportabili. Il driver della Leap Motion è inoltre esoso in termini di CPU, e una sua integrazione nei dispositivi mobili sembra non banale. Per funzionare al meglio, l'applicazione richiede il posizionamento frontale del sistema di tracking; al momento questo non è possibile a causa dell'algoritmo utilizzato dal dispositivo. Data l'assenza di un'API che fornisca direttamente la nuvola di punti, sono stati fatti dei tentativi di accedere direttamente alle informazioni di basso livello provenienti dal dispositivo. L'assenza di specifiche del protocollo di comunicazione non permette un'ottimale gestione e, di conseguenza, la frequenza massima ottenibile è bassa (circa 15 fps). E' stato implementato un algoritmo di riconoscimento che, partendo dalle immagini catturate (destra e sinistra), effettua il calcolo della matrice di disparità, la determinazione dei contorni della mano e dei difetti di convessità al fine di localizzare le estremità delle dita. I risultati di tale algoritmo consentono di costruire la scheletrizzazione della mano che risulta però affetta da un'instabilità dei punti dettata dalla bassa frequenza di cattura. Inoltre il costo computazionale di tale algoritmo al momento risulta comparabile a quello originale. Il produttore del dispositivo ha previsto a breve nuove implementazioni che consentano un rilevamento più stabile e una API estesa;

La rilevazione ad ultrasuoni è stata valutata come robusta e ripetibile. Infatti, i sensori pilotati dal microcontrollore consentono una rapida e accurata stima della distanza, successivamente convertita in eventi grazie ad un algoritmo dal costo computazionale limitato. L'uso della tecnologia a ultrasuoni permette di utilizzare materiali, come la rete elastica impiegata nel sistema, che risultano trasparenti al rilevamento ma permettono di generare una risposta somatosensoriale al tocco. Di contro non è possibile porre superfici estese in prossimità dei sensori per evitare riflessioni accidentali; per questo motivo è neces-

sario ruotare il dispositivo mobile sullo stand in modo che eventuali riflessioni non interessino l'area dei sensori.

Un test informale del sistema a ultrasuoni ha mostrato come, a seguito di un evento di pressione sulla rete elastica, la generazione in sequenza degli stimoli tattile, audio e video restituisca la sensazione di unità dell'azione anche se la visualizzazione risulta affetta da latenza.

5. CONCLUSIONI

Abbiamo realizzato un ambiente di test in realtà aumentata per dispositivi mobili nel quale sono state confrontate due differenti tecniche di rilevamento dei movimenti delle dita per la realizzazione di interfacce di pianoforte. Indagini preliminari hanno mostrato come sistemi basati su tastiere immateriali possano essere realizzati; rimangono però irrisolti molti problemi, sopratutto per quanto riguarda la latenza complessiva e la portabilità dei sistemi. La sfida maggiore sembra essere la generazione di un feedback tattile sufficientemente realistico. Allo scopo, verranno compiute ulteriori ricerche riguardanti i sistemi rilevamento nonché la ricerca di materiali e tecniche per migliorare le soluzioni di generazione del feedback tattile.

6. RINGRAZIAMENTI

Gli autori segnalano il supporto del progetto PiaNo - *Piano from Nothing*, finanziato da Intel Corporation – Santa Clara (CA) e coordinato da Tsinghua University – Pechino (Rep. Pop. Cinese) nella persona del Prof. Linmi Tao.

7. BIBLIOGRAFIA

[1] D. J. Levitin, K. Maclean, M. Mathews, L. Chu, and E. Jensen, "The perception of cross-modal simultaneity (or "the Greenwich Observatory Problem" revisited)," vol. 517, pp. 323–329, AIP, 2000.

[2] N. Miner and T. Caudell, "Computational requirements and synchronization issues for virtual acoustic displays," *Presence: Teleoper. Virtual Environ.*, vol. 7, pp. 396–409, Aug. 1998.

[3] T. Mki-patola and P. Hmlinen, "Latency tolerance for gesture controlled continuous sound instrument without tactile feedback," in *Proc. International Computer Music Conference (ICMC)*, pp. 1–5, 2004.

[4] J. M. Silva, M. Orozco, J. Cha, A. E. Saddik, and E. M. Petriu, "Human perception of haptic-to-video and haptic-to-audio skew in multimedia applications," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 9, pp. 9:1–9:16, May 2013.

[5] "Leap motion air piano."

[6] J. L. Crowley, F. Brard, and J. Coutaz, "Finger tracking as an input device for augmented reality," in *Workshop on Automatic Face and Gesture Recognition,Zurich, Switzerland*, pp. 195–200, 1995.

[7] J. Letessier and F. Berard, "Visual tracking of bare fingers for interactive surfaces," in *Proceedings of the 17th annual ACM symposium on User interface software and technology (UIST 04)*, (Santa Fe, NM, USA), pp. 119–122, ACM Press, 2004.

[8] A. Goodwin and R. D. Green, "Key detection for a virtual piano teacher," in *IVCNZ*, pp. 282–287, 2013.

[9] F. Huang, Y. Zhou, Y. Yu, Z. Wang, and S. Du, "Piano ar: A markerless augmented reality based piano teaching system," in *2011 Third International Conference on Intelligent Human-Machine Systems and Cybernetics* (U. IEEE Computer Society Washington, DC, ed.).

[10] T. Kaaresoja and S. A. Brewster, "Feedback is... late: measuring multimodal delays in mobile device touchscreen interaction," in *ICMI-MLMI*, p. 2, 2010.

[11] E. S. Silva, J. A. O. de Abreu, J. H. P. de Almeida, V. Teichrieb, and G. L. Ramalho, "A preliminary evaluation of the leap motion sensor as controller of new digital musical instruments," in *14th SBCM - Brazilian Symposium on Computer Music*, 2013.

[12] M. Reunanen, K. T. Palovuori, T. Ilmonen, and W. Mkel, "Npr - affordable fingertip tracking with ultrasound," in *IPT/EGVE* (E. Kjems and R. Blach, eds.), pp. 51–58, Eurographics Association, 2005.

[13] S. Perrin, A. Cassinelli, and M. Ishikawa, "Laser-based finger tracking system suitable for moems integration," in *Proc. IVCNZ03, New Zealand*, pp. 131–136, 2003.

[14] T. Hoshi, M. Takahashi, T. Iwamoto, and H. Shinoda, "Noncontact tactile display based on radiation pressure of airborne ultrasound," *EEE Trans. Haptics*, vol. 3, pp. 155–165, July 2010.

[15] T. Carter, S. A. Seah, B. Long, B. Drinkwater, and S. Subramanian, "Ultrahaptics: multi-point mid-air haptic feedback for touch surfaces.," in *UIST* (S. Izadi, A. J. Quigley, I. Poupyrev, and T. Igarashi, eds.), pp. 505–514, ACM, 2013.

[16] S. Gupta, D. Morris, S. N. Patel, and D. Tan, "Airwave: Non-contact haptic feedback using air vortex rings," in *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, (New York, NY, USA), pp. 419–428, ACM, 2013.

[17] K. Karunanayaka, S. Siriwardana, C. Edirisinghe, R. Nakatsu, and P. Gopalakrishnakone, "Haptic mouse, enabling near surface haptics in pointing interfaces," in *Proc. ACHI 2013 - Conference on Advances in Computer-Human Interactions*, 2013.

THE GATE MODULATOR. AN EXPERIMENT IN DIGITALLY-CONTROLLED ANALOG SYNTHESIS

Andrea Valle
CIRMA/StudiUm, Università di Torino
andrea.valle@unito.it

Simone Pappalardo
Cons. N. Piccinni (Bari) - Dip. Nuove Tecn. e Linguaggi
Cons. O. Respighi (Latina) - Informatica Musicale
simpapp@yahoo.it

ABSTRACT

This paper describes the Gate Modulator, a device for computer-controlled analog audio processing using the Arduino board. The Gate Modulator is based on the 4066 quad bilateral switch, an integrated circuit typically used for control applications, that is here used for audio processing at different time ranges, providing a variety of possible effects and uses. In the paper, first we consider the increasing role of the Arduino platform in audio applications, then we introduce the design and implementation of the Gate Modulator and the benefits of using Arduino as a controller, finally we describe the *Otosimbionte* project, an installation that exploits the capabilities of the Gate Modulator.

1. ARDUINO AND AUDIO PROCESSING

Arduino is renown as a prototyping board for physical computing [1]. But it is gaining an ever increasing popularity in various forms of sound processing. Its main benefits are low cost, small size, prototyping easiness. If compared to computer based DSP, it has lowest computational resources, but symmetrically it has lowest requirements (e.g. operating system). Moreover, it can be easily embedded into small physical objects, thus becoming "transparent", i.e. it can be hidden to the user, who is not forced to take into account a specific interface layer. Various experiences involving the Arduino microcontroller platform and audio are available:

- some projects directly use the Arduino ADC input (8 bits) to sample the audio signal [1]. Those include granular synthesis by means of a 8-bit weighted resistor output (R2R) as a 8-bit DAC [2]; a "bitcrusher" that introduces a weighted pin technique [3]; reimplementations of the most famous sound effects [4], as typically used in guitar pedals [5];
- real-time sound synthesis project have been conceived [6] [7] [8];
- dedicated audio shields are available. Some shields embed only passive components (connectors, resistors, capacitors, potentiometers) [9] and are specifically targeted at guitar pedals [10]: DSP is realised by software. Other shield includes various DAC/ADC modules [11] [12];
- Arduino has also been used as an analog audio effect controller, e.g. as a digital LFO [13] or as a driver for controlled resistors that modify some analog distortion effect parameters [14].

2. ANALOG GATE MODULATION

Our approach does not involve Arduino as a DSP board, rather it focuses on Arduino as a computational controller for analog audio processing. In this sense, it is a form of hybrid system for synthesis [2]. Before introducing Arduino, we discuss some aspects of the use of the 4066 IC. The featured analog processing is very crude per se, but it still allows to produce interesting results and does not belong (if not very loosely) to previously existent techniques. Its design directly springs from technological exploration with integrated circuits. In particular, it uses the well-known 4066 quad bilateral switch integrated circuit (see Figure 1) as controllable pass/not pass gate [3]. The 4066 IC acts as a logical gate for the input signal, its internal state being determined by a control signal. The audio signal is fed into the input, and the gate is then open/closed by a control signal. When the control signal has an audio frequency, indeed the technique results in a form of synthesis/processing that belongs to the modulation family (hence the name "gate modulation"). From a theoretical perspective (and in a possible digital implementation) this kind of processing would result in a windowing function uncoupled from the input signal, with very sharp, noisy edges introduced by the abrupt change in signal amplitude.

[1] http://www.uchobby.com/index.php/2007/11/11/arduino-sound-part-1/
[2] http://www.amandaghassaei.com/arduinodsp.html
[3] http://www.instructables.com/id/Lo-fi-Arduino-Guitar-Pedal/
[4] http://interface.khm.de/index.php/lab/experiments/arduino-realtime-audio-processing/
[5] http://wiki.leaflabs.com/Guitar_Audio_Effects

Copyright: ©2014 Andrea Valle, Simone Pappalardo. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

[6] http://playground.arduino.cc/Main/ArduinoSynth
[7] http://playground.arduino.cc/Main/PbSynthCode
[8] http://www.mycontraption.com/sound-effects-with-and-arduino/
[9] https://www.kickstarter.com/projects/2006771911/the-arduino-dsp-shield
[10] http://blog.arduino.cc/2014/01/28/program-sound-effects-for-your-guitar-with-pedalshield/
[11] https://learn.adafruit.com/adafruit-wave-shield-audio-shield-for-arduino
[12] http://174.136.57.214/html/rugged_audio_shield.html
[13] http://www.beavisaudio.com/Projects/DSWF/
[14] http://www.youtube.com/watch?v=_X0bL6WS-VY

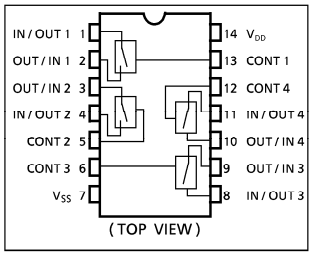


Figure 1. The 4066 quad bilateral switch.

The technique can be understood as a form of amplitude modulation, where the carrier is the input signal and the modulator (the signal resulting from the gating behaviour) can be thought as a square wave with varying frequency and duty cycle. While such a technique could prove interesting per se, its analog implementation by means of the 4066 IC introduces a relevant –analog-style so to say– smoothing due to nonlinearity in the onset phase. The IC requires a certain amount of time (which is variable from 15 to 70 nanoseconds, depending on the supply voltage and, in part, on the ratio from supply voltage and instantaneous amplitude of the audio signal) to change its state from open to close or vice versa, which results in a specific envelope applied to the input signal. A comparison between a digital implementation and the real 4066 IC signal is shown in Figure 2: in the second case, the same digital signal (a 1 Hz sinusoid) is fed into the IC and re-recorded, with the same gating parameters of the digital implementation.

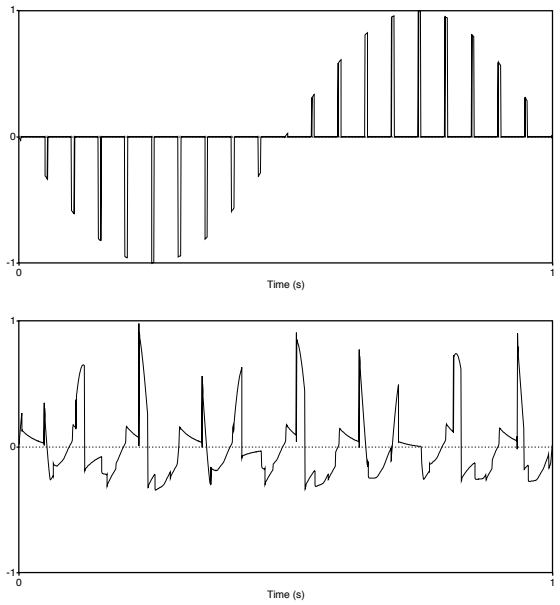


Figure 2. 1 Hz sinusoidal signal modulated by a gate signal: digital implementation (top) and 4066 analog implementation (bottom).

Even from these very first, empirical observations, nonlinearities are apparent [15] . When IC is on play, the modulator is no more a square wave, rather a sort of smoothed sawtooth. Complex spectral effects are thus obtained. The 4066 IC behaviour depends on the control signal, which can be thought as an unipolar square wave. Thus, if considering the wave resulting from an open/close cycle, two parameters are available: frequency and duty cycle. Frequency is related to the open/close period while duty cycle can be expressed as the open/close ratio in a period. Figure 3 shows a sonogram of a 1000 Hz sinusoidal signal fed into the 4066 IC with a fixed duty cycle (0.5) and variable frequency (frequency starts varying after some seconds). Modulation effects are apparent in the first 10 seconds, then, while frequency increases, time effects become relevant, i.e. a impulsive/granulating sound result. Harmonic frequencies related to the 1000 Hz fundamental gain progressively an increasing energy. Very low frequencies in

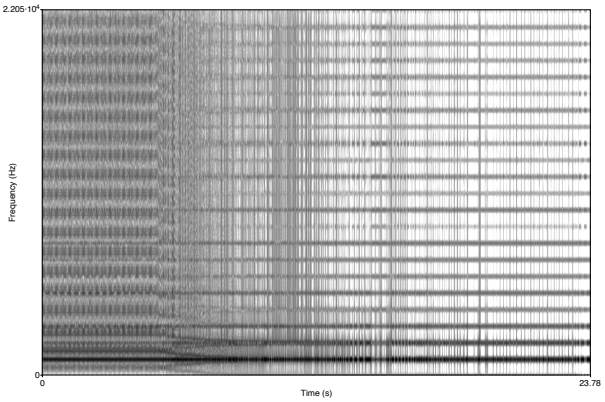


Figure 3. Sonogram of a 1000 Hz sine modulated with a variable frequency and fixed duty cycle.

the carrier (even in the sub-audio range) are still viable as the carrier acts like a "support" for various clicking effects introduces by modulation. In Figure 4, a 50 Hz sine is modulated with fixed frequency but decreasing duty cycle, that "dries" the signal introducing a stronger pulse/clicking effect.

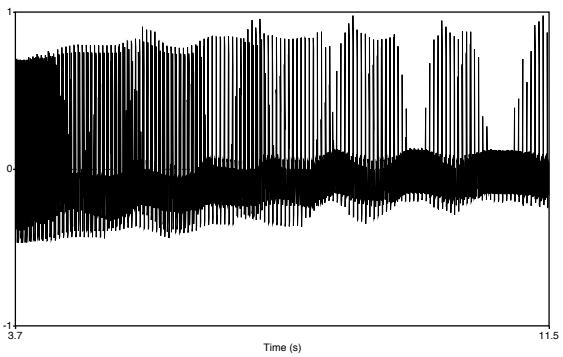


Figure 4. 50 Hz sine modulated with fixed frequency and decreasing duty cycle.

[15] Also, the gating process seems to introduce a certain amount of DC offset. Moreover, the Arduino's clock resolution may introduce some quantisation distortion.

3. USING ARDUINO AS A CONTROLLER

Gate modulation via 4066 IC can indeed be implemented in a complete analog fashion. In that case, a clock is needed to provide the control signals to the IC, that can be realised in various ways. Figure 5 shows a possible implementation by means of a 40106 Hex inverting Schmitt trigger [4], to which other diodes can be added in order to control the clock duty cycle. Another possible solution may include the notorious 555 timer IC. We will not deal into details of

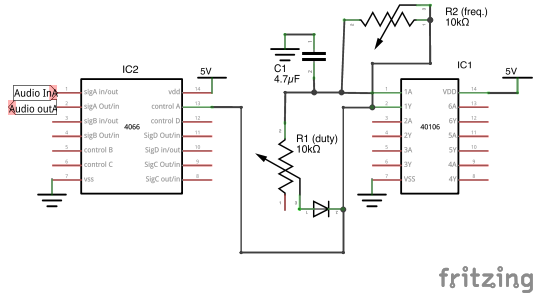


Figure 5. Analog implementation via 40106 Schmitt trigger.

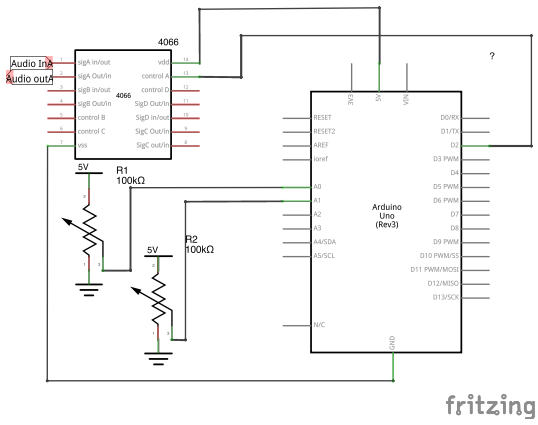


Figure 6. Gate Modulator circuit with Arduino.

analog implementation, as in our case we chose an hybrid approach, where the control signal is digitally generated and provided directly by the Arduino board. Our implementation is shown in Figure 6, where Arduino's digital outs are connected directly to the 4066 IC control ins. The use of a microcontroller results in various benefits with respect to a completely analog implementation:

- fast prototyping: in an experimental situation such as the one presented here, the relevance of the results lies only on the musical side. Thus, it must be assessed empirically by testing various control situations. While it is indeed possible to use well-known tools for electronic prototyping (e.g. by means of a breadboard), still the minimal setup Arduino/4066 is particularly light, as it shifts all the control logic on the Arduino language side. This allows the designer to work with a smooth trial-and-error methodology;
- multiple gate modulation: the 4066 IC contains four individually controllable switches, each one related to an input signal (audio, in our case). As it is trivial to connect four Arduino outs to the 4066 controls, four gate modulation processes are immediately available, that can operate on four different audio signal ("multiple ins" setup) or by applying four independent modulations to the same signal ("multiple modulator" setup). As this possibility is already provided by a standard Arduino (e.g. Uno), this expanded behaviour does not affect the hardware size.
- time independency: the scheduling of the control signals is handled by software. This means that it is not bound to particular upper time resolution. This allows to experiment with various modulating frequencies, with a time interval ranging from microseconds up to minutes. In this was, a continuum from audio synthesis to time scheduling is available. In our case a gate modulation device can easily become an audio signal distributor, a chopper, a tremolo, or a combination of these and other effects.
- hardware extension: the use of a microcontroller opens up other possibilities in hardware communication and control. The serial communication bus of the Arduino can be connected to a master software controller. In this way, the a gate modulation device can be controlled in real-time e.g. via USB port from a host computer, thus linking analog sound processing to algorithmic composition. Moreover, other features of Arduino can be exploited: as an example, by connecting the 4066 control ins with the Arduino PWM dedicated ports it is possible to generate control signals that can modulate the carrier with audio frequencies (typically, up to 980 Hz [16]). On the input side, it is easy to integrate the a gate modulation device in a physical computing framework, e.g. by connecting sensors to the Arduino and by associating them to a certain control logic for the modulation.

4. THE GATE MODULATOR

In this section we describe our implementation: the Gate Modulator. We chose a module-like design, that was inspired both by analog synth/processing modules and by guitar effect pedals. Also, it had to match, both functionally and aesthetically, the design of other devices that were meant to share the same hardware setup. Figure 7 shows three modules (see next section for an explanation), with the "Gate Modulator" on top (a). The design is indeed rooted into the DIY tradition in electronic music, that focuses on technological re-appropriation by means of low cost materials, hardware hacking, and knowledge sharing,

[16] http://arduino.cc/en/Reference/AnalogWrite

that is gaining a high momentum in recent years ([5], [6]). Following the "multiple ins" setup (see before), the Gate



Figure 7. The Gate Modulator (a) with other DIY modules (b and c).

Modulator features four parallel independent inputs and four related outputs for modulated signals [17] . As with the typical guitar pedal model, connectors are $\frac{1}{4}$ inch jack plugs. Modulation frequency for each input channel can be set by a dedicated knob. Knobs are connected to potentiometers read by Arduino's analog ins. In short, Arduino acts as the general controller for the whole modulation process, mediating between the user and the 4066 IC behaviour. Each knob has been programmed to exhibit a three-state behaviour. When a knob is set to minimum, its relative gate is always closed, so that the channel is "muted"; symmetrically, when it is set to maximum, the gate is always open, so that no modulation is applied; inside the range, gate modulation is active. This behaviour provides the user a mute/pass functionality without extra hardware interfaces. Two other knobs have been made available. The first (see the larger knob, right bottom, in Figure 7 a) allows to modify the duty cycle in the range $[0, 1]$. The second (smaller knob, Figure 7 a, right top) controls a scaling factor to be applied to all the modulations. In this, way it is possible to have a large range of frequency values while fine tuning the resulting values by means of each independent channel knob: as a result, time resolution varies from microseconds up to minutes, a range typically outside analog implementations. Figure 8 shows the internals of the Gate Modulator, with the 4066 IC in the foreground. An Arduino Uno is in use, that is placed in the box so that it shows externally the power inlet and the USB port. In this way it is possible to power the module externally, so that the Gate Modulator is in all respects an autonomous module. Moreover, by means of the USB port it is still possible to connect it to a computer and reprogram the Arduino so that the IC may be controlled from an external application bypassing the knobs.

[17] To independently manage the logical channels of 4066 the Arduino must be programmed for multitasking operations.

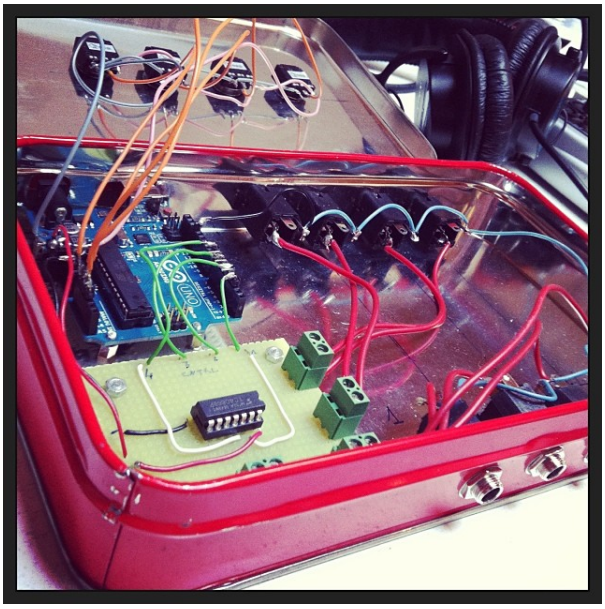


Figure 8. The Gate Modulator. Internals.

5. USAGE: THE OTOSIMBIONTE PROJECT

The *Otosimbionte* project started as a collaboration between the two authors in the occasion of the exhibition *Il sogno verde* curated by *Brecce per l'arte contemporanea* (March 22-23, 2014) at Villa Gregoriana, Tivoli. Otosimbionte is intended as a sound installation (see in general [7]) that, while focusing on sound production and symmetrically on aural perception, still retains a specific physical presence. The design results from the intervention of various elements:

- the DIY tradition in electronic music, that focuses on low-cost in order to assemble experimental sound devices, following the path of the American tradition ([5], [8]);
- the attention to reused, recycled, poor materials as a political way to challenge late Capitalism attitude towards the waste of resources, and as a way to recontextualize everyday objects (see also the Arte povera movement or the Nouveau Realism; for a discussion in the context of physical computing see [9]).
- the interest into feedback as a general, powerful expressive resource for generating content (see [10] for a discussion) that emerges directly from interaction between agents of variable nature (be them software or hardware), without prior definition.
- the collaborative attitude in artistic production that underlines the role of cooperation against competition and that blurs the distance between "pure" technical and artistic phases.
- the blurring of the installation/performance contexts, so that an installation can be performed, and vice versa a performance system can be turned into an installation.

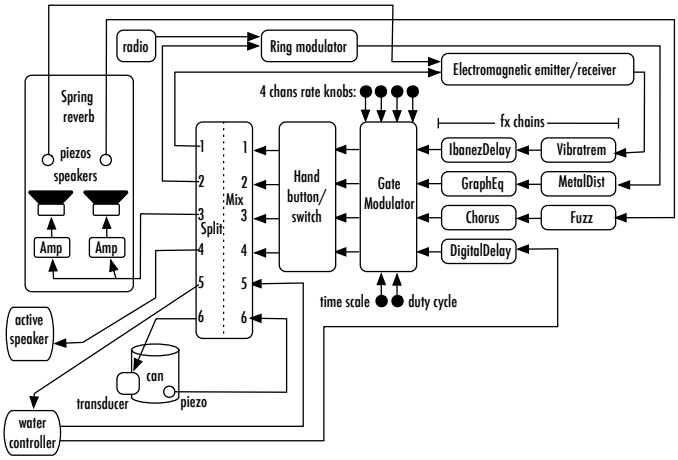


Figure 9. Diagram of Otosimbionte components.

Otosimbionte results from the hybridisation via feedback of each author' sound system. Hence the name, that refers to symbiosis and to listening. The whole setup is shown in Figure 10. The installation has been placed on a "raft" inside the ruins of an ancient Roman villa, so that it emerged from the context as an isolated organism. It has been conceived and designed at distance (the authors living in two different cities), and finally assembled and fine-tuned in loco, where it has taken its final shape. Components are detailed in Figure 9. We describe Otosimbionte as an example of the benefits of the Gate Modulator and as an extension to an artistic object of the principles at its basis. The whole system is based on a continuous circulation of audio material from its output to its input by means of six channels of audio streams. The effect chain block (right) is made by standard guitar pedals: they are connected to the Gate Modulator that operates by chopping them at variable time rates. The four outs are sent to a custom made hand button/switch module (Figure 7b, see later). The four audio channels are sent to a minimal DIY passive mixer/splitter (Figure 7c) that mixes all 6 the input into a single output and split this mono signal into 6 outputs. These outputs are sent to other custom devices. The spring reverb is assembled by connecting small loudspeakers to springs and by capturing their physical vibrations by means of piezo microphones. The ring modulator is a classical passive DIY effect made up of two small transformers and four diodes that allow to modulate the incoming signal from the splitter with a hacked radio. The electromagnetic emitter/transducer is a system of 4 modified coils. Three of these are used as electromagnetic emitters of signals coming respectively from the mixer/splitter and from one of the two channels of the spring reverb. The fourth coil captures the resulting signal.

Indeed, it is possible to derive from each point of the feedback chain an audio signal to be delivered to the audience. We chose two "extraction points". On one side, a large recycled metal can (visible in Figure 11, left) is used as a resonator by applying an amplified transducer to its surface, thus delivering the sound in the environment. A second sound output is provided by an active loudspeaker connected to one of the outputs of the mixer/splitter.

Even if it is mainly based on the constant, internal recirculation of audio signals, Otosimbionte gathers sound also from the external world: first, by integrating the radio signal in the ring modulator, secondly by providing four interaction points to the user.

- the hand button/switch module: by simply muting the desired streams, the whole sound output can be greatly affected;
- electromagnetic receiver/transmitter: its output varies in relation to the mobile coils, available to the user. By placing the receiving coil closer to one or the other transmitting coils it becomes possible to actively control the amplitude of the relative audio signal. The same signal may be modulated by inserting various electromagnetic noises, e.g. by introducing near the receiver small motors, fans, the same power supplies in use for Otosimbionte;
- spring reverb: the user can effectively intervene on the springs.
- water controller: an Arduino-based tangible system, it recognises various types of touch in the water (Figure 9, middle left). A copper wire is placed below the bowl –partly filled with water– that acts as a proximity sensor. The hardware and software configuration has been redesigned following an example from Disney laboratories [18] in order to make it compatible with Arduino [19]. While in the original design a computer was in use, in our case the whole control processing is entirely performed by the Arduino. The system basically distinguishes between two main types of touches: if the bowl is touched externally or the water is lightly perturbed, the signal is routed to the output A; rather, if the hand is immersed in the water, the signal is routed to the output B.

[18] http://www.disneyresearch.com/project/touche-touch-and-gesture-sensing-for-the-real-world/

[19] http://www.instructables.com/id/Touche-for-Arduino-Advanced-touch-sensing/
http://madlabdk.wordpress.com/2013/04/08/2-touche-meets-pd-and-maxmsp/



Figure 10. The *Otosimbionte* installation at Villa Gregoriana, Tivoli.

Both the knobs on the Gate Modulator and the hand button/switch module are not relevant to the installation per se, but allow to immediately turn the installation into a performance device, where the musicians operate on the various control points dramatically varying the sonic output. In fact, various improvisation performances on the "raft" have been realised by the authors as a part of the exhibition [20]. Figure 11 shows a live performance in a club where the raft has been "transported". In the context of Otosimbionte, the Gate Modulator has proven to be a crucial feature. A typical feature of analog feedback systems is to generate continuous, drone-like sounds ("sons homogènes", to speak with Pierre Schaeffer), making it difficult to obtain impulsive, granular, isolated sounds. The switching mechanism has been a key element in escaping this typical "feedback sound", rather providing a much more "event-oriented" sound perspective, from modulation to granulation, up to long sound events surrounded by silence. In relation to this aspect, by varying the duty cycle and the time interval, we were able to obtain a sort of "sound breathing" in the installation, that slowly evolved through a cycle of presence/absence of sound, further emphasised by the natural reverberation of the location.

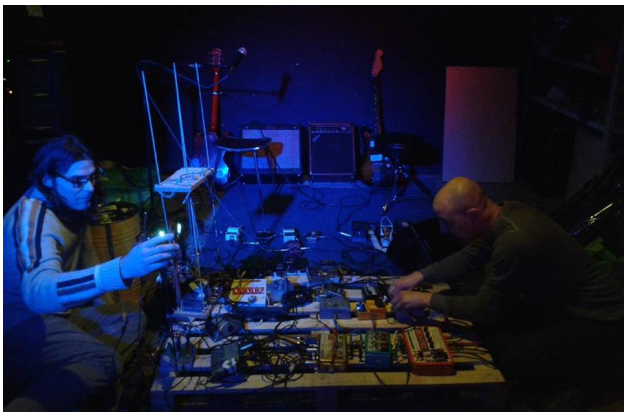


Figure 11. Improvising on the Otosimbionte's raft.

[20] A video can be seen here: https://vimeo.com/106018857

6. CONCLUSION

The Gate Modulator is intended both as an exploration of hybrid synthesis processes and as an experiment in a low-cost, unconventional hardware construction. Digitally-controlled gate modulation has proven to be an unusual yet interesting technique for sound generation and processing, in which the limited parameters (substantially, frequency and duty cycle) were functional to an immediate expressive usage. On the other side, the Arduino microcontroller allowed us to build a very versatile, portable hardware device (as seen in Figure 8, most of the case space is dedicate to connectors). Versatility is to be intended as a twofold feature: first, in relation to a large range of gating frequencies; second, in relation to hardware expansion (e.g. reprogrammability of the device, a feature that "softens" the hardware). Future work will focus on a more in-depth analysis of the acoustic distortion introduced by the analog gate, and on the expansion of the computer-controlled switching logic to a multiplexer for bridging synthesis processes with mixing.

7. REFERENCES

[1] M. Banzi, *Getting started with Arduino*. O'Reilly, 2009.

[2] C. Roads, *The Computer Music Tutorial*. Cambridge, MA, USA: MIT Press, 1996.

[3] Toshiba, "TC4066BP/BF/BFN/BFT." datasheet, 1997.

[4] NXP B.V., "HEF40106B Hex inverting Schmitt trigger." datasheet, Nov. 2011.

[5] N. Collins, *Handmade Electronic Music. The art of hardware hacking*. New York–London: Routledge, 2006.

[6] VV.AA., "Re-wiring electronic music," *Organised Sound*, vol. 18, 2013.

[7] B. LaBelle, *Background noise: perspectives on sound art*. New York–London: Continuum, 2006.

[8] VV.AA., "Composers inside electronics: Music after David Tudor," *Leonardo Music Journal*, vol. 14, 2004.

[9] A. Valle, "Making acoustic computer music: The Rumentarium project," *Organised Sound*, vol. 18, pp. 242–254, 2013.

[10] D. Sanfilippo and A. Valle, "Feedback systems: An analytical framework," *Computer Music Journal*, vol. 37, no. 2, pp. 12–27, 2013.

SELF-ORGANIZING THE SPACE OF VOCAL IMITATIONS

Davide Rocchesso, and Davide Andrea Mauro
Department of Architecture and Arts, Iuav University of Venice
roc, dmauro @iuav.it

ABSTRACT

The human voice is a powerful instrument for producing sound sketches. The sonic space that can be spanned with the voice is vast and complex and, therefore, it is difficult to organize and explore. In this contribution, we report on our attempts at extracting the principal components from a database of 152 short excerpts of vocal imitations. We describe each excerpt by a set of statistical audio features and by a measure of similarity of the envelope to a small number of prototype envelopes. We apply k-means clustering on a space whose dimensionality has been reduced by singular value decomposition, and discuss how meaningful the resulting clusters are. Eventually, a representative of each cluster, chosen to be close to its centroid, may serve as a landmark for exploring the sound space.

1. INTRODUCTION

The EU project SkAT-VG [1] (Sketching Audio Technologies using Vocalizations and Gestures, 2014-2016) is aimed at finding ways to exploit voice and gestures in sonic interaction design. Research in SkAT-VG proceeds along three directions: (i) improving our understanding on how sounds are communicated through vocalizations and gestures; (ii) looking for relations between vocal/gestural primitives and the physical characteristics of sound-producing phenomena; (iii) designing tools for converting vocal and gestural actions into parametrized sound models.

In this paper we describe a research exercise that may be ascribed to the first direction of the SkAT-VG project. We try to see how the space of vocal imitations could be rearranged and simplified to highlight clusters of sounds that are acoustically similar. We also check if the clusters, produced by algebraic and algorithmic manipulations, make sense to humans as well.

The paper is organized as follows: we first introduce the sonic space we want to be able to explore and understand. Then in sections 3 and 4 we tackle the problem of how to encode and efficiently manipulate the data in order to organize them. We present then perspectives on how to relate the automatic analysis with the subjective classifications obtained from human subject.

[1] http://www.skatvg.eu/

Copyright: ©2014 Davide Rocchesso. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

2. THE VOCAL SONIC SPACE

In a sense, the human voice has for acoustic communication a role similar to what the hand and pencil have for visual communication. Humans use their voice for verbal communication as well as for non-verbal acoustic expression, similarly to the hand which is used both for writing and for drawing. Just as the hand and pencil are extensively used for visual sketching, the voice has potential to be exploited for sketching sounds. Indeed, sketching comes before verbal – oral or written – expression in development of both the human species and the human individuals [1].

In order to devise tools that facilitate sound design by vocal sketching we must gain a better understanding of what the voice can do and how vocalizations are interpreted by listeners. The space of voice-produced sounds need to be described both in acoustic and in articulatory terms. We need to know the characteristics of a comprehensive repertory of vocal sounds and how these can be achieved by our voice organ. From a sound design perspective, it is particularly useful to organize the vocal sound space on a low-dimensional layout whose navigation can be facilitated by landmarks, or sounds that represent distinct neighborhoods. The purpose of this study is to construct such a layout automatically from a database that significantly spans the possible non-verbal uses of the human voice.

A database of 152 audio segments were manually extracted from the Fred Newman's repertory of vocal imitations described in his book [2] and included in the companion CD. The segments were all $500\ \mathrm{ms}$ long ($22050\ \mathrm{samples}$ at $44100\ \mathrm{samples/s}$) and were taken to represent a single sound event or process. There is a degree of arbitrariness in this operation, as some events may be the result of a concatenation of articulatory actions of a shorter length, but for the scope of this study each audio segment may be considered to include a single vocal utterance.

The idea of using landmarks to facilitate navigation in the sound design space was previously explored in the context of parametric sound synthesis [3, 4], and auditory representations were used both to give a visual snapshot to each sound and to compute distances that would allow locating new sounds in the map. In this work we aim at extracting landmarks as representatives of clusters.

3. REDUCING DIMENSIONALITY: A COMPACT DESCRIPTION OF SOUNDS

Reducing the dimensionality seems to be a reasonable approach to organize a sonic space. A classic way to do that is

by means of Principal Component Analysis (PCA), which is based on Singular Value Decomposition (SVD).

Taken our reference database of 152 audio segments, a PCA on the raw audio files would reveal that as many as 110 singular values need to be retained to account for 90% of the energy. The basis vectors, while being recognizable as vocal sounds, are not easily associated to different articulatory or acoustic characteristics. The situation does not change much if we apply PCA on the (Fourier, or Wavelet) transformed version of the audio segments, in the sense that the number of dimensions that would retain most of the signal energy is almost as large as the number of exemplars in the database. Things get better if we giveup invertibility and, for example, apply PCA on the magnitude Fourier spectrum. In this case 95 singular values are sufficient, yet still too many, to account for 90% of the energy. Giving up invertibility means that it would not be possible to select a point in the reduced sonic space and have the sound produced by inverse-transforming. Still, it would be possible to localize a given sound in the space and, for example, to interpolate between landmarks to synthesize a similar sample.

In the area of music information retrieval a lot of research has been devoted to extract audio descriptors (or features) that could concisely represent sound and music [5]. Several software libraries are available to easily extract brightness, spectral flux, and other descriptors from a given soundfile, and to collect statistical information from them. For this study, we have been using the MIR toolbox [6], and we applied some of its feature extractors to summarize each of our audio segments with statistical information. In particular, we used the median and interquartile range values (as recommended in [5]) of spectral flux, centroid, roughness, flatness, entropy, skewness, and RMS energy computed over 18 windows spanning the $500\mathrm{ms}$ of each audio segment.

In addition to the statistical audio features, we added some features that would account for the temporal morphology of each audio segment. The idea is that, for example, such features would mark a clear difference between a sustained noise and an impulsive click. However, there is the problem of where short temporal events actually occur in time, as it should be irrelevant if an impulsive click occurs at time $100\mathrm{ms}$ or $300\mathrm{ms}$ in the considered time span. In order to account for possible elastic deformations of time, Dynamic Time Warping (DTW) is used to compare distances between the extracted RMS profile and each of a set of prototype temporal envelopes, namely upward slope, downward slope, up-down profile, and impulses.

All collected features are non-negative real numbers, but their range and units are quite different from each other. For the subsequent step of PCA, we perform a normalization to the maximum value of each feature in our population of samples. Still, most of the distributions are heavily skewed toward zero. In order to obtain feature distributions that more closely resemble a gaussian we distort the distribution of values of each feature by its cumulative histogram, a cheap trick that is called histogram equalization in image processing. As an example, Figure 1 shows

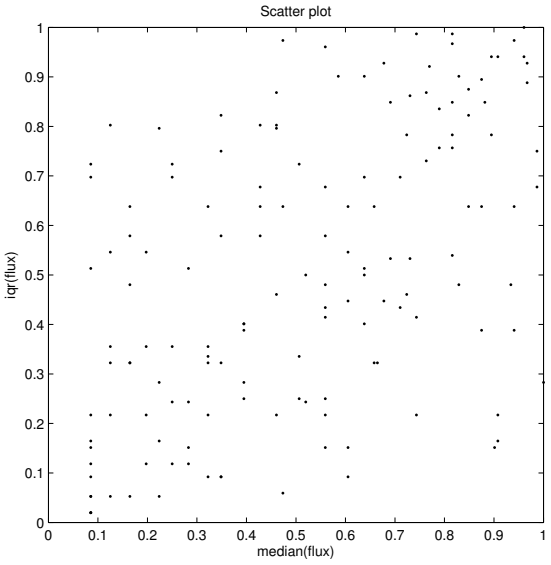


Figure 1. Distribution of median and interquartile range values of spectral flux after histogram equalization

the distribution of median and interquartile range values of spectral flux after histogram equalization.

Before the extraction of principal components, the mean is subtracted from the distribution of each feature, and the distribution is further normalized to range between -1 and 1. Then, the thin SVD is computed on the matrix $B \in \mathbb{R}^{m \times f}$, where $m = 152$ is the number of audio segments and f is the number of features:

$$B = USV'. \tag{1}$$

$S \in \mathbb{R}^{f \times f}$ is the diagonal matrix of singular values in descending order, $U \in \mathbb{R}^{m \times f}$ is the matrix of orthonormal basis vectors (principal components) that best represents the set of audio segments (described as features) in a L^2 sense. The $i-$th row of U expresses the $i-$th audio segment as a set of coefficients of a combination of principal directions, or "feature modes". These modes are expressed as columns of $SV' \in \mathbb{R}^{f \times f}$.

To reduce dimensionality, we retain only columns 1 to l of matrix U, corresponding the l largest singular values, or to the most prominent feature modes.

4. CLUSTERING

In general, clustering in the PCA-reduced subspace is more effective than doing it in the original space, because the subspace of $l + 1$ cluster centroids is spanned by the first l principal directions of data [7].

For our database of audio segments, each summarized by the 18 features described in Section 3, the singular values are plotted in Figure 2.

With $l = 2$ (two principal components) a run of k-means clustering with a variable number of clusters returns a squared sum of errors as represented in Figure 3. With such low value of l, the extraction of three clusters is particularly effective, and such clusters can be displayed in the 2-D space of principal components, as in Figure 4. In

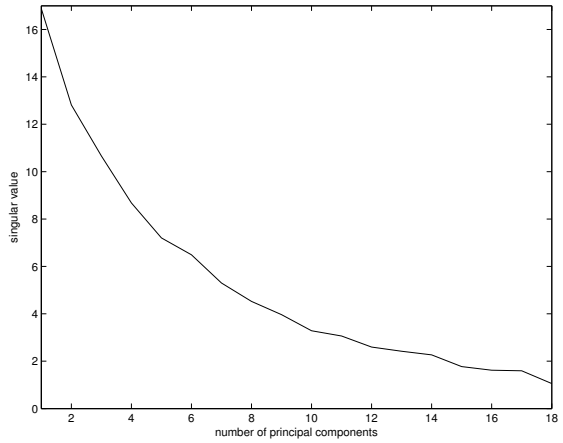


Figure 2. Singular values of matrix B of descriptions of audio segments

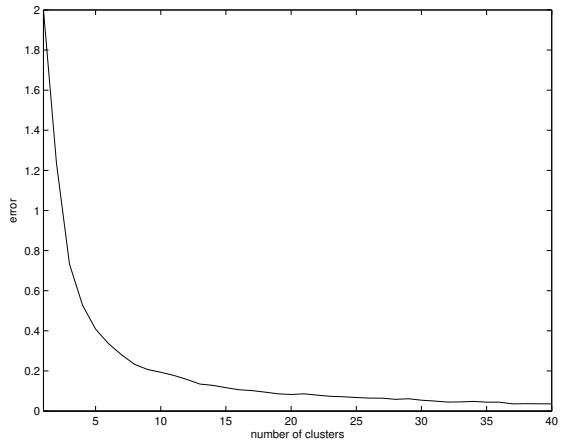


Figure 3. Squared sum of errors as a function of number of clusters ($l = 2$)

that figure, the larger circles correspond to the cluster centroids, which ideally should be selected as representatives of each cluster. In practice, since resynthesizing a vocalization that corresponds to such centroids is not possible, we can choose the closest member as a cluster representative. The spectrograms of such representatives are depicted in Figure 5. The first is described as imitation of a trumpet, the second is a prototype of "glottal fry", and the third is a "tongue flop" that could be used to imitate horse steps. Given such a relatively small number of clusters compared to the number of elements and the vague nature of the terms and categories that can be used to describe sounds it is not easy to interpret them. In the first cluster we have sounds that are (mostly) pitched or exhibit an intonation. The second cluster contains sounds that are continuous. Finally the third cluster encompasses sounds that are characterized by an impulsive behavior or a temporal evolution.

By visualizing the cells of the equalized, centered, and normalized matrix of features we can make sense of how clustering operates on audio descriptors projected along principal components. Such visualization is displayed on Figure 6, where a row-wise sorting has been performed to

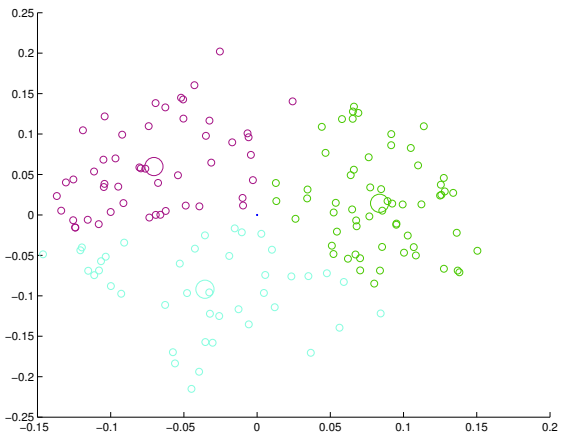


Figure 4. Three clusters in the space of the two principal components

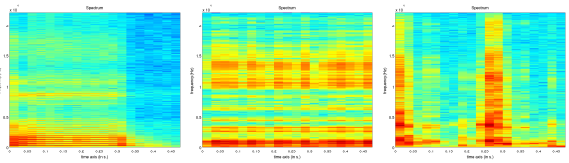


Figure 5. Spectrograms of representatives of clusters 1 (60 elements), 2 (42 elements) and 3 (50 elements)

group all audio segments belonging to the same cluster. In the cells, deep blue corresponds to -1 and deep red corresponds to +1 as normalized feature values.

Even more meaningful is connectivity analysis [7]. Figure 7 shows, discretized to a binary color, the matrix $U_2 U_2{}'$, where U_2 are the first two columns of U, with their rows sorted according to the extracted clusters. Apart from the three clusters, which are clearly visible, elements of "contamination" between and within clusters are also visible. This indicates the opportunity of refined clustering, either increasing l, or looking for more than $l + 1$ clusters, or running PCA and clustering hierarchically on each cluster. In figure 7, the degree of connectivity is $c = 65\%$, i.e.,

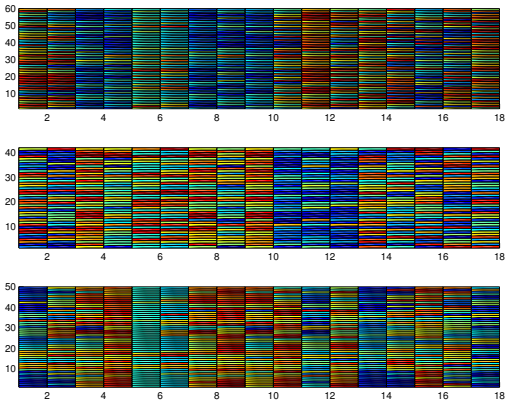


Figure 6. How clusters are derived from features (one per column)

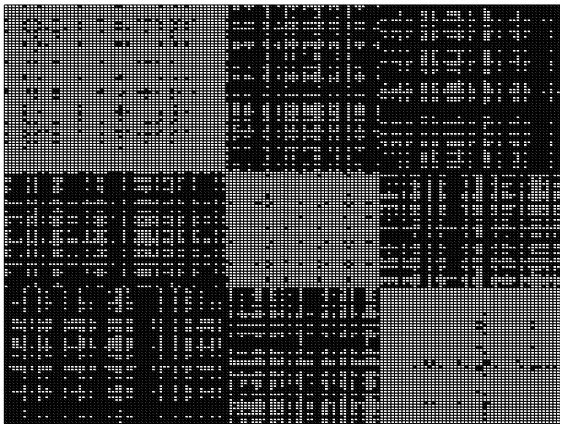


Figure 7. Connectivity of 152 audio segments in two principal components

65% of the white cells belong to the three squares on the diagonal, showing strong connection within clusters.

5. HOW WOULD A HUMAN DO?

Considering a small (i.e., 3) number of clusters, we asked three colleagues, not involved in this research exercise, to memorize the three cluster representatives and then to assign each of the remaining 149 sounds to one of the representatives. From these associations, we computed the confusion matrix and the clustering accuracy for each subject. The three subjects showed the values of accuracy: 0.48, 0.53, 0.65, where a random assignment would return a value 0.33 of accuracy. For example, for the subject that is the closest to automatic clustering, the confusion matrix is $C = \begin{bmatrix} 46 & 13 & 1 \\ 6 & 24 & 12 \\ 10 & 11 & 29 \end{bmatrix}$, where element $c_{i,j}$ represents the number of audio segments that have been assigned to cluster i by the machine and to cluster j by the human. It is an interesting coincidence that the accuracy expressed by this subject (65%) is the same as the degree of connectivity expressed in figure 7.

6. POSSIBLE EXTENSIONS

The degree of connectivity c depends on the number l of principal components and on the number k of clusters. The maximum value $c = 100\%$ is obtained for $l = 1$ and $k = l + 1 = 2$, where the clear separation into two clusters seems to be largely determined by spectral centroid and flatness. Increasing the number of clusters by just one ($l = 1, k = 3$) gives a much more confused picture ($c = 0.62$).

In general, subdivision into k clusters is best done on $l = k - 1$ principal components. With this constraint, and for $l = 1, 2, 3, 4, 5$ we get degrees of connectivity $c = 1, 0.65, 0.49, 0.39, 0.33$, respectively. In all cases, the prototype sounds (cluster representatives) found are perceptually distinct from each other, and they may well serve the purpose of automatically finding landmarks in the space of vocal imitations.

Further explorations of the space are currently being pursued, especially experimenting with hierarchical clustering, to see if more meaningful subdivisions will emerge. So far, relatively little attention has been payed to the features, which were chosen from a set of standard audio features used for musical signals extended with signatures of temporal envelope. The fact that the sounds are all of vocal origin should be exploited to include specific features that come from the literature of speech and voice analysis.

7. ACKNOWLEDGMENTS

The authors are pursuing this research as part of the project SkAT-VG and acknowledge the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 618067.

8. REFERENCES

[1] M. Changizi, *Harnessed: How Language and Music Mimicked Nature and Transformed Ape to Man*. Perseus Books Group, 2013.

[2] F. Newman, *MouthSounds: How to Whistle, Pop, Boing, and Honk... for All Occasions and Then Some*. Workman Publishing, 2004.

[3] C. Drioli, P. Polotti, D. Rocchesso, S. Delle Monache, K. Adiloglu, R. Annies, and K. Obermayer, "Auditory representations as landmarks in the sound design space," in *Proceedings of Sound and Music Computing Conference*, 2009.

[4] K. Adiloglu, C. Drioli, P. Polotti, D. Rocchesso, S. Delle Monache, *et al.*, "Physics-based spike-guided tools for sound design," in *Conference on Digital Audio Effects*, 2010.

[5] G. Peeters, B. L. Giordano, P. Susini, N. Misdariis, and S. McAdams, "The timbre toolbox: Extracting audio descriptors from musical signals," *The Journal of the Acoustical Society of America*, vol. 130, no. 5, pp. 2902–2916, 2011.

[6] O. Lartillot and P. Toiviainen, "A Matlab toolbox for musical feature extraction from audio," in *International Conference on Digital Audio Effects*, pp. 237–244, 2007.

[7] C. Ding and X. He, "K-means clustering via principal component analysis," in *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, (New York, NY, USA), pp. 29–, ACM, 2004.

RIFLESSIONI SU UN NUOVO MODELLO VOCALE

Concetta Cucchiarelli
Conservatorio "A. Casella" L'Aquila
info@concettacucchiarelli.com

SOMMARIO

Se esiste un elemento *liquido* per antonomasia, mutuando un'espressione di Zigmund Bauman [1], è proprio la voce. La liquidità è la caratteristica principale della voce, ancor più in un tipo di società anch'essa liquida. In effetti non si può non considerare come i mutamenti dei paradigmi estetici e culturali, i profondi mutamenti delle relazioni e gerarchie sociali, abbiano reso la voce sempre più polimorfa, tendente continuamente all'espressione di una o più identità. Da un punto di vista artistico, l'estetica della vocalità d'opera ha ingabbiato per lungo tempo la voce in uno stereotipo che rappresenta soltanto uno degli infiniti modi possibili della voce, in sostanza l'ha resa *solida*. Non è un caso che anche gli studi, incentrati su questo argomento, si siano fatti *solidi* trattando o il parlato o il canto, nell'unica forma espressiva della voce d'opera. Questo lavoro vuole dimostrare come sia necessario ormai un modello che sia in grado di descrivere una fattispecie vocale molto più ampia. Facendo, inoltre, una ricognizione trasversale dei progressi individuati frammentariamente nei diversi domini di ricerca, vuole dare qualche anticipazione degli elementi caratterizzanti e delle tappe previste per la realizzazione di un modello utile al compositore che decida di avvicinarsi a questo "strumento".

1. INTRODUZIONE

«Definire l'identità come un compito e lo scopo dell'impegno di tutta una vita era, se paragonato all'attribuzione automatica a un ceto dell'era premoderna, un atto di liberazione: una liberazione dall'inerzia delle strade tradizionali, delle autorità immutabili, delle routines preordinate e delle verità incontestabili» (Bauman, Intervista sull'identità [1]).

Quale elemento è più legato intimamente al concetto di identità se non la voce? In effetti la voce rappresenta nello stesso tempo l'aspetto più culturale e meno culturale in assoluto.

Copyright: © 2014 Concetta Cucchiarelli This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Se consideriamo il termine culturale come opposto di naturale, e quindi prodotto da fatti umani, la voce è al contempo la cosa meno culturale, perché la più legata alla dimensione istintiva e nello stesso tempo il massimo prodotto, sedimento o effetto collaterale di scelte umane conscie e inconscie. Nell' opera la voce è una fattispecie ben definita ed uniforme che mira a mantenere un elevato livello di omogeneità timbrica. Questo non vuol dire che il canto d'opera risulti monotono o senza sfumature. Ricordiamo una Cathy Berberian entusiasta che dimostra come sia diverso eseguire un brano vicino alla tradizione popolare evitando un modo di cantare tradizionale e concertistico [2]. Di fronte alla varietà più ampia di vocalità esistente, tuttavia, possiamo paragonare queste sfumature a dialetti appartenenti ad un'unica lingua in un contesto di pluralismo linguistico.

D'altra parte, oltre a mantenere una certa omogeneità timbrica, il canto d'opera è funzionale anche a far emergere la voce su un organico di tipo orchestrale senza l'ausilio di alcuna amplificazione. Nella musica popolare, ad esempio, questa esigenza viene meno, e la voce si può scegliere in funzione della necessità di farsi simbolo di una personalità ben definita, di un personaggio riconoscibile anche ad un breve ascolto, di essere imitata nel suo rimanere inimitabile. Nel canto etnico-tradizionale la voce è espressione di una provenienza, di un'appartenenza ad un gruppo o etnia, e spesso serve a creare un rapporto di tipo imitativo con la natura o di comunicazione con la collettività. Nel parlato la voce si fa espressione di sentimenti, di stati d'animo, ma è anche un modo con il quale affermare il proprio posto nella società, la propria identità sessuale, ecc.. Alcuni autori ritengono che la ragione primaria che determina il modo di cantare è, in realtà, di tipo ideologico molto prima che musicale e che la continua attenzione alla enunciazione di un testo, alla trasmissione di un significato abbia, prima di ogni altra cosa, condizionato e, nello stesso tempo, inibito la sperimentazione vocale [15]. Nell'ambito della musica contemporanea sperimentazioni di un utilizzo *esteso*[1] della voce iniziano a metà del novecento con Luciano Berio, John Cage fino ad arrivare a Trevor Wishart, solo per fare alcuni esempi, ma non trovano quella formalizzazione puntuale e condivisa che, invece, si è riusciti a raggiungere per le tecniche esecutive alternative degli strumenti musicali

[1]Per utilizzo esteso della voce si intende quell'insieme di tecniche utilizzate soprattutto a partire dagli inizi del ventesimo secolo in ambito teatrale e musicale.

[15]. Paradossalmente lo studio sulla voce è il primo a nascere e il primo ad arenarsi. Inoltre, la riuscita e complessità di certe sperimentazioni resta molto legata al ruolo dell'esecutore, tanto da poter azzardare il considerare quest'ultimo come una parte integrante nel processo di composizione. In realtà le potenzialità della voce sono tuttora sconosciute, soprattutto per un suo utilizzo artistico. Questo accade perché le varie discipline che si interessano di voce si sono mosse e continuano a farlo in modo frammentario e non coordinato, con obiettivi non coincidenti.

In quanto fenomeno multidimensionale, in effetti, la voce può essere analizzata da diversi punti di vista [14]. Quelli che prende in considerazione questo lavoro sono quello acustico, fisiologico e didattico. Il motivo per il quale gli studi non procedono quasi mai in maniera uniforme o coordinata è che in ogni campo esiste un obiettivo diverso, ad esempio nel campo medico lo studio è volto soprattutto alla valutazione della patologia al fine di riportare il fenomeno nell'ambito dell'eufonia. Crescenti sono i lavori sulla voce artistica, ma ancora relativamente pochi e concentrati sulla voce d'opera [16]. L'analisi artistica deve tener in considerazione tutti questi aspetti, facendo un lavoro di composizione, laddove per composizione si intende selezionare suggestioni utili e significative e metterle a sistema, quello di cui questo lavoro rappresenta il primo passo.

2. VOCE CANTATA E VOCE PARLATA

Gli studi sulla voce, tradizionalmente, sono suddivisi in base alla dicotomia parlato-cantato. Alcuni autori hanno individuato una serie di distinzioni [11]. La maggior parte dei suoni emessi nel canto, ad esempio, è per il 90% intonata (*voiced*), mentre nel parlato il valore corrisponde al 40%, anche se differisce per tipo di lingua (i valori riportati nell'esempio, in effetti, si riferiscono alla lingua inglese). Considerando la parte intonata di entrambi le emissioni, inoltre, risulterebbe evidente come il canto si muova su un'estensione molto più ampia rispetto a quella nella quale ci si muove in un discorso. Ci sarebbe, inoltre, anche un fattore di spontaneità delle due emissioni che le differenzia: la voce parlata sarebbe molto più spontanea di quella cantata che, tra l'altro avrebbe anche l'obbligo di emergere anche su un'orchestra al completo. Quello che si vuol fare non è confutare quanto appena esposto, che è vero se si considera il canto coincidente con quello d'opera ma, invece, fare una piccola rivoluzione copernicana, spostando al centro dello studio la voce nella sua manifestazione timbrica, anziché come un'attività specifica. Quando si analizza uno strumento musicale, se ne studia il timbro nella sua totalità, mentre per la voce è come se la *voce cantante* e la *voce parlante* fossero due strumenti diversi.

Per capire la parzialità delle affermazioni appena esposte dobbiamo necessariamente esulare dalla voce d'opera. Solo in tal modo si riesce a pensare ad un tipo di vocalità non caratterizzata in maniera predominante dalla parte intonata. Si riesce, così, a comprendere l'incompletezza della affermazione secondo la quale, ad esempio, il canto si muova su estensioni frequenziali maggiori del parlato: ci sono molti tipi di canto che si caratterizzano proprio per la loro continuità rispetto al parlato e che quindi rimangono in un range frequenziale molto ristretto, così come ci sono modalità di espressione parlata, come le grida, soprattutto in alcuni contesti, che possono raggiungere escursioni vocali non riscontrabili nel canto. Quanto alla spontaneità del parlato rispetto al canto, basti pensare a certi esperimenti di improvvisazione vocale, tanto cari anche alla tradizione contemporanea, per capire quanto una vocalità non spontanea sia rappresentativa solo di una parte del fenomeno.

Il nuovo paradigma vocale si muove su una definizione molto più ampia. A livello concettuale e teorico bisogna spostare l'attenzione sul timbro e superare le riduzioni fatte finora come, ad esempio, l'identificazione della voce parlata con l'articolazione vocalica e non, senza considerarne l'aspetto di variazione timbrica che rende unica l'espressione emotiva, non riducibile solo a variazioni di pitch nell'ambito della frase. Ad esempio un semplice *vocal fry* inserito in un inizio di frase come attacco riesce ad esprimere benissimo indecisione o sensualità. Allo stesso modo parlare in modalità *twang*[2] rende la voce presente, sicura di sé e vivace.

3. LE ORIGINI

L'inizio di questa contrapposizione culturale-metodologica, in parte, deriva molto anche da un fattore tecnologico. In effetti i primi studi sulla sintesi vocale si sono sviluppati negli ambienti delle telecomunicazioni ed hanno puntato ad ottenere una rappresentazione del suono più realistica possibile ma, soprattutto, che potesse essere trasmessa efficientemente a distanza di chilometri con una quantità ridotta di dati. Di conseguenza l'intelligibilità e la riconoscibilità hanno avuto la meglio sull'indagine del timbro ed hanno finito per radicare sempre più la distanza tra il mondo del canto e quello del parlato.

Le prime simulazioni vocali cominciano nei primi anni del novecento e sono il territorio preferenziale per i pionieri della sintesi per analisi. Da quei primi passi lo sviluppo è stato continuo e ha visto l'implementazione di molte metodologie. Nello specifico MUSSE (*Music and singing synthesis equipement* sviluppato presso il KTH), CHANT (sviluppato presso l'Ircam), SPASM (*Singing Phisical Articulatory Synthesis model* sviluppato presso il CCRMA) e LYRICOS Project (sviluppato presso la Texas Instruments) ed altre ancora, sono state sviluppate principalmente per la sintesi della voce cantata [11]. Ognuna di queste tecniche ha un diverso grado di efficacia ed efficienza, risultando inversamente proporzionali la qualità del risultato e il numero di parametri ne-

[2]Il *twang* è una tipologia vocale introdotta dal metodo didattico Voicecraft.

cessario a ottenere la resintesi. La ricchezza di questi metodi è indiscussa ma presenta dei limiti, come affermano alcuni degli autori da tempo impegnati in questi studi [5] . Ci sono elementi che certamente, richiedono un necessario approfondimento o implementazione. La cosa importante è che questi studi hanno dimostrato quanto per questa materia non si possa prescindere dalla conoscenza di molti elementi di diversi campi del sapere: linguistica, fisica, acustica, anatomia ed oltre.

4. IL MODELLO SORGENTE-FILTRO

Il modello alla base della maggior parte degli algoritmi di sintesi per analisi è il modello sorgente-filtro perché ritenuto il più rispondente al meccanismo fisico di produzione vocale [7]. Semplificando molto il modello, si pensa ad un segnale sorgente correlato alla qualità delle corde vocali, come lunghezza e spessore, e forma del tratto vocale, e un filtro, selezionato quasi totalmente dalla posizione di lingua, mandibola, oltre che alla lunghezza totale del tratto vocale [3]. La posizione di questi elementi creerebbe una mappa di spazi, rappresentati in un grafico chiamato *area funzionale*, responsabili della formazione di una serie di formanti (solitamente cinque) in grado di riprodurre la voce. La teoria delle aree funzionali [19] si rivela molto utile per la riproduzione di vocali, ma lascia inspiegate molte caratteristiche predominanti nella voce non d'opera.
Nella sintesi digitale della voce, gli effetti non rappresentabili nel modello sorgente-filtro, come le trubolenze ed il rumore, vengono inseriti in un residuo segnale di rumore aggiunto separatamente (vedi Figura 1) nell'ambito di una logica di linearità del sistema [8].

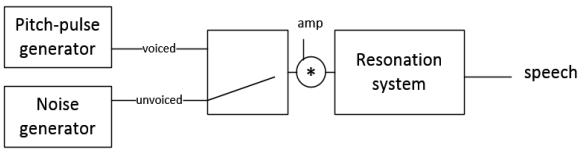


Figura 1. Algoritmo per la sintesi digitale del parlato.

Questo procedimento semplifica sicuramente il lavoro di implementazione ma non rende possibile considerare il rumore nella sua varietà (più o meno correlato al segnale, più o meno filtrato e via dicendo). Il problema è che questo sistema fa riferimento ad una descrizione dell'apparato fonatorio così come indagato da una strumentazione che oggi risulta essere obsoleta.
Questo porta a considerare un sistema di fonazione incompleto nel quale i protagonisti principali sono le corde vocali per il livello sorgente e il *vocal tract* soprattutto nella sua sezione orale per il livello risuonatori. Quello che, invece, si può leggere in alcuni studi al proposito [4] è che, innanzitutto, il panorama vocale è molto più vasto e difficile da descrivere con terminologie già esistenti. Il problema di avere una terminologia condivisa e chiara è molto importante per non ricadere nelle confusioni e fraintendimenti legati ad una terminologia sedimentata e confusa nella quale, ad esempio, lo stesso concetto di *registro* corrisponde ad almeno cinque diverse definizioni. La seconda cosa che risulta evidente è, in effetti, la difficoltà di descrivere il fenomeno vocale attraverso indicatori acustici conosciuti [12].
Se cambia la fattispecie devono cambiare anche i parametri descrittivi. In ambito clinico, ad esempio, ci si è resi conto che parametri basati sulla valutazione della periodicità del segnale (shimmer, jitter,..) non funzionano sulla valutazione di una vocalità non eufonica [9, 10, 13]. Si va alla ricerca, quindi, di nuovi indicatori in grado di aggirare il problema periodicità come, ad esempio, l'analisi cepstrale[3].

5. NUOVI ATTORI DEL SISTEMA FONATORIO

La maggior parte delle suggestioni determinanti ai fini della elaborazione di un nuovo modello vocale, tuttavia, provengono dalle ricerche effettuate in campo didattico. I metodi didattici con un fondamento scientifico (finora molto pochi ma significativi) presentano il vantaggio di stimolare la ricerca sul funzionamento del sistema fonatorio in modo direttamente collegato ad un utilizzo artistico della voce (non necessariamente cantata) ma con strumentazione professionali di altissima qualità, non essendo direttamente subordinate alla programmazione della ricerca medica [4, 17].
Solo per elencarle senza entrare nel dettaglio, dato che saranno esposte in un lavoro futuro, le novità rilevanti rispetto al sistema fonatorio tradizionalmente concepito sono quattro.

Innanzitutto non si possono considerare come sorgente solo le corde vocali. Osservazioni effettuate con strumentazioni particolari[4] che riescono ad ovviare alle limitazioni legate all'utilizzo dalla stroboscopia (inutilizzabile nella valutazione di suoni aperiodici), hanno reso possibile isolare il segnale sorgente che si è rivelato essere significativamente diverso per tipologia di emissione vocale sin dalla sorgente.
Questo tipo di indagine, combinata ad una visione in cinematografia ultrarapida, ha reso possibile accertare che l'eventuale rumore nella voce è contenuto direttamente nel segnale sorgente e viene generato da un'interazione tra corde vere e corde false che vibrano a frequenze diverse, a volte in rapporto di numero intero (ad esempio *overtone singing* o canto diplofonico) e a volte in rapporto irrazionale (ad esempio nei suoni distorti).
Facendo un controllo incrociato a livello medico si vede, in effetti che non solo ci sono due livelli di vibrazione, ma che tra un tipo di corde e l'altro,vere e false, si crea uno spazio di risonanza (c. d. *ventricoli di Morgagni*) che rappresenta il primo filtro acustico sovraglotttico agente sul segnale, potendo restringersi o allar-

[3]Lavoro sulla definizione di indicatori descrittivi della gravità di disfonia spasmodica svolto con L'Università La Sapienza, a cura del prof. G. Ruoppolo, in corso di pubblicazione.
[4]Al proposito si veda http://www.laryngograph.com/.

garsi in misura variabile. Le caratteristiche morfostrutturali determinano, assieme alla massa e lunghezza delle corde vocali vere, le variazioni individuali del timbro [18].

Questo anche se in anatomia era già accertato, non è stato considerato prima perché nella emissione vocale d'opera le corde vocali false non entrano in funzione. Oltre alle corde false va considerata la chiusura delle cartilagini aritenoidee grazie ad un movimento combinato e sinergico di muscoli considerati fino a poco tempo fa antagonisti [18].

La capacità di inclinazione dello sfintere ariepiglottico, inoltre, determina un restringimento dato dall'avvicinamento delle cartilagini aritenoidee alla parte più bassa dell'epiglottide e questo processo (tipicamente coincidente con la modalità vocale c. d. *twang*) rende possibile produrre un suono più "proiettato" e squillante[5]. [17, 20].

Tutto questo può essere considerato e racchiuso più che in un semplice livello sorgente, in un vero e proprio *sistema sorgente* che già forma gran parte del suono (il movimento dei *Ventricoli di Morgagni*, insieme alle tensioni del muscolo costrittore della faringe, unitamente ancora di più all'abbassamento o innalzamento dell'unità tiroidea solidale con la base della lingua, trasforma significativamente lo spazio immediatamente sovrastante del vibratore glottico).

Risalendo lungo il tratto vocale ci si imbatte in un nuovo soggetto che, come le corde vocali false, non viene preso in considerazione se non come un interruttore che chiude o apre la strada nasale: si tratta del *velo palatino* o *palato molle*. Tale passaggio si rivela di importanza fondamentale per la formazione del timbro [4]. Nei modelli classici il suo stato è o chiuso o aperto. Si è riusciti, attraverso indagini empiriche incrociate (video sul movimento degli articolatori, laringoscopia e fibroscopia endorale) a capire che il suono risulta fortemente modificato attraverso un movimento del palato molle (utilizzato nella vita quodidiana sostanzialmente per vomitare e sbadigliare), a posizione invariata di lingua, mandibola e labbra. Il palato molle mosso da diversi muscoli che agiscono in direzioni diverse, è in grado di innalzarsi, abbassarsi, allargarsi o restringersi, assumendo diverse forme che influiscono immediatamente sul timbro.

Questo comporta un immediato ridimensionamento di quello che è stato da sempre il ruolo degli articolatori orali, fondamentali per l'articolazione del linguaggio ma di rifinitura per la formazione del timbro[6].

[5]La modifica del tratto provoca una selezione di frequenze medio alte, in particolare, comprese tra i 1500-2000 Hz che si sono rivelate essere la componente media dello spettro vocale e, nello stesso tempo, quella componente che l'orecchio ascolta meglio. Questo compensa la contestuale perdita di ampiezza riscontrata nel segnale.

[6]E' significativo come, nell'ambito degli studi del Complete Vocal Technique, si facciano eseguire diversi tipi di emissione vocale ad un cantante ispezionato con fibroscopia rigida, durante la quale il dottore tiene ferma ancorata la lingua tirandola verso l'esterno, per introdurre ed inclinare del grado necessario la telecamera e, nonostante questa immobilizzazione della parte orale, il suono continui ad essere percettivamente molto diverso al variare dell'emissione.

6. VERSO UN NUOVO MODELLO

Detto questo, come modificare un modello di fonazione che recepisca tutte queste variazioni ed innovazioni? Innanzitutto bisogna immaginare un sistema composto da un *sistema sorgente* e un *sistema superiore*.

Il sistema sorgente deve ricomprendere non solo le corde vocali ma anche le false e tutte le modifiche dei filtri dell'area immediatamente sovraglottica. Questi elementi sul segnale proveniente dalla vibrazione delle corde vocali agiscono applicando una vera e propria modulazione di frequenza e un filtraggio per lo più di tipo passa alto. In più le stesse corde, a seconda di come vibrano, con più o meno tempo di adduzione, generano un interazione variabile con l'aria che si crea a livello sottoglottico producendo direttamente al livello sorgente anche voci soffiate (*breathy*) o particolarmente ricche d'aria.

Il sistema superiore sarà costituito sostanzialmente da faringe e palato molle che non funge solo da interruttore, ma funge da filtro selettore sia di quantità di suono che deve passare per il naso o che deve proseguire nella cavità orale, sia di frequeze enfatizzate. Basti pensare che anche il suono emesso a bocca chiusa e ferma può avere equalizzazioni diverse grazie al movimento del palato molle.

Rimane da verificare, tuttavia, l'eventuale dipendenza tra un elemento e l'altro come, per esempio, il palato molle e la base della lingua o i muscoli che agiscono a livello glottico.

La seconda caratteristica di un nuovo modello vocale è la sua rappresentazione su tre dimensioni. L'inserimento di modifiche di forma nei vari livelli di tratto vocale, anche di restringimenti delle pareti del "tubo" faringeo o di modifica del palato molle rendono necessaria una ricostruzione tridimensionale del modello. Già ci sono alcune implementazioni di modelli in *3D*, ma progettati per l'utilizzo dei parametri previsti dal vecchio modello, con il palato che fa da interruttore e un ruolo centrale degli articolatori [6].

Da ultimo, tutte le modifiche trattate finora devono essere inglobate in un modello fisico in modo da ovviare alle limitazioni della sintesi per analisi come la dipendenza dei risultati dalla voce analizzata e la difficoltà di modifica dei parametri, così importante per un utilizzo creativo del modello.

Quello che ne deriverà sarà un sistema in grado di superare ogni distinzione per tipo di vocalità, mirando direttamente allo studio timbrico in grado di rappresentare il continuum delle infinite possibilità vocali. Focalizzare l'attenzione sulla qualità vocale è un cambio di prospettiva necessario se non si vuole accettare la permanenza su modelli che funzionano molto bene ma che sono solo parzialmente rappresentativi della realtà.

7. SVILUPPI FUTURI

Per quanto riguarda gli sviluppi futuri quello di cui si ha bisogno è, innanzitutto, monitorare i movimenti dei singoli elementi durante la fonazione per ogni tipologia di

emissione vocale. Questo verrà fatto con la ricostruzione di un modello tridimensionale attraverso immagini da risonanza magnetica dinamica, con osservazione anche palatale. Questo tipo di visione arricchirà le indagini endoscopiche già effettuate e permetterà di confermare alcune intuizioni che da una visione dall'alto (quella della laringoscopia) risultano meno chiare. Da questo modello sarà possibile anche misurare puntualmente gli spazi e le distanze da utilizzare nel modello fisico costruito *ad hoc*. Ripensare la voce come un *contiuum* timbrico che non prevede distinzioni tra parlato e cantato potrebbe essere utile anche nelle procedure di riconoscimento vocale dove i diversi tipi di emissione possono essere considerati come espressione di diversi stati d'animo.

8. CONCLUSIONI

La voce cantata, da un punto di vista filogenetico, è la tappa più sofisticata dell'evoluzione vocale. Questo ci impone una riconsiderazione estesa della voce, che passa per la demolizione di un postulato culturale, l'unicità della voce d'opera, e la necessità di ripensare i modelli e i descrittori esistenti.

Alcuni autori avevano evidenziato già da tempo come alcune ipotesi fossero una forzatura necessaria alla modellizzazione ma che non tenevano conto della varietà della realtà che, invece, richiede un modello molto più complesso e sofisticato [5].

Per far questo è stato e sarà sempre più importante far convergere le conoscenze e puntare ad obiettivi comuni. Non si inventa nulla di nuovo, si cambiano solo le prospettive di osservazione e si creano dei legami, dei collegamenti che ci fanno riscoprire un fenomeno che sembrava già sufficientemente esplorato. Avere il punto di vista dell'*utilizzatore*, avere una conoscenza personale del proprio strumento, essere cantante e sperimentare nuove vocalità, cercando di non danneggiare il proprio strumento, permette di porsi delle domande, sentire a livello propriocettivo che accade qualcosa di diverso da quanto già conosciuto e permette di formulare delle domande che un osservatore esterno non avvertirebbe come necessarie.

Le riflessioni contenute in questa esposizione possono essere considerate quelle che Bauman definirebbe *riflessioni in progress*, ossia riflessioni che non si accontentano di definire o concettualizzare un evento, ma che puntano a stabilire connessioni ed echi con fenomeni sociali, fisici, psicologici, culturali che sembrano molto lontani dall'oggetto iniziale, seguendo una metodologia che punta ad individuare il filo che lega le tante e diverse osservazioni tra di loro e con l'oggetto indagato.

9. BIBLIOGRAFIA

[1] Z. Bauman: *Intervista sull'identità*, Editori Laterza, Bari, 2009.

[2] L. Berio: *C'è musica e musica*, Rai Eri, Milano, 1985.

[3] P. R. Cook, *Identification of Control Parameters in an Articulatory Vocal tract model, with applications to the synthesis of singing*, Dissertation at Stanford University, 1991.

[4] C. Cucchiarelli, *Per un nuovo modello di analisi e sintesi della voce: I colori vocali nel canto moderno,* Dissertation at Conservatorio "A. Casella", 2013.

[5] C. Dodge, T. A. Jerse: *Computer Music. Synthesis, composition and performance*, Schirmen Thomson Learnning, Usa, 1997.

[6] O. Engwall, *Vocal tract modeling in 3D*, Departement for Speech, Music and Hearing Quarterly Progress and Status Report, Vol. 40, No. 1-2, pp. 31-38, 1999.

[7] G. Fant, *Analysis and synthesis of speech process*, in *Manual of phonetics*, Malmberg Edition, Amsterdam, 1968.

[8] G. Fant, *Glottal source and excitation analysis*. Speech Transmission Laboratory Quarterly Progress and Status Report KTH, Stockolm, 1979.

[9] R. Garrett: *Cepstral and Spectral-Based Acoustic Measures of Normal Voices*. Theses and Dissertations. University of Wisconsin, 2013.

[10] J. Hillenbrand, R. A. Houde, Acoustic Correlates of Breathy Vocal Quality: Dysphonic Voices and Continuous Speech, *Journal of Speech and Hearing Research*, Vol. 39, pp. 311-321, 1996.

[11] Y. E. Kim: *Singing voice analysis/synthesis*, Dissertation at Massachusetts Institute of Technology, Massachusetts, 2003.

[12] E.Marchetto, *Automatic speaker recognition and characterization by means of robust vocal source features*, Dissertation at Università degli studi di Padova, A.A. 2010-2011.

[13] J.Muñoz E.Mendoza M.D.Fresneda G.Carballo P.Lopez: Acoustic and Perceptual Indicators of Normal and Pathological Voice, *Folia Phoniatrica e Logopaedica*, No. 55, pp. 102–114, 2003.

[14] L. Pigozzi: *A nuda voce, vocalità, inconscio e sessualità*. Antigone Edizioni, Torino, 2008.

[15] J. Potter: *Vocal Authority, singing style and ideology*, Cambridge University Press, Cambridge, 1998.

[16] B. Roubeau, N. Henrich, M. Castellengo, Laryngeal Vibratory Mechanisms: The Notion of Vocal Register Revisited, *Journal of Voice*,Vol. 23, No. 4, pp. 425-438, 2009.

[17] C. Sadolin, *Complete Vocal Technique*, CVI Publications, Copenhagen, 2008.

[18] O. Schindler: *La voce. Fisiologia, patologia clinica e terapica,* Piccin Nuova Libreria S.p.a., Padova, 2010.

[19] J. Sundberg: *The science of the singing voice*, Northern Illinois University Press, Illinois, 1987.
[20] J. Sundberg, M. Thalen: What is‘‘Twang”?, *Journal of Voice*, Vol. 24, No.6, pp.654-660, 2010.

HIS ENGINE'S VOICE: TOWARDS A VOCAL SKETCHING TOOL FOR SYNTHETIC ENGINE SOUNDS

Stefano Baldan
Department of Architecture and Arts,
Iuav University of Venice
stefanobaldan@iuav.it

Stefano Delle Monache
Department of Architecture and Arts,
Iuav University of Venice
sdellemonache@iuav.it

Luca Comanducci
Department of Computer Science,
University of Milan
luca.comanducci@studenti.unimi.it

ABSTRACT

We present a work in progress on a sound design tool for automotive applications, which allows the designer to rapidly prototype synthetic engine sounds through the use of vocal imitations. A physics-based model for engine sound synthesis, with real time parametric control based on timbral descriptors extracted from the vocal signal, is under development. This work can be considered as a preliminary study in the larger context of the SkAT-VG EU project, whose goal is to find ways to exploit and develop the use of voice as a sketching and prototyping tool in sound design practices.

1. INTRODUCTION

The automotive industry is investing a lot of energy to carefully design every sound produced by a car. In many existing car models, the engine related sounds in the passenger compartment are electronically added through loudspeakers, and designed in order to enhance the overall brand and driving experience [1]. Beyond pedestrian safety aspects, this is a crucial issue for electric vehicles, which are inherently silent [1]. Moreover, several electronic engine sound devices are available on the market to turn one own's city car into a sport or muscle car, such as a Lamborghini or a Pontiac [2]. Nonetheless, design tools for sonic sketching and fast prototyping are still missing. Engine sound simulators are often very complex to operate and time-consuming. They require some effort to produce sets of sound that can be easily compared and refined, in the early stage of the design process.

Humans are very familiar with the many voices of vehicles. Engines purr, crank, hum, whir, grumble, snarl, growl, chortle, sputter, and stutter. Even naïve listeners can easily reproduce stationary and rev sounds with their voice, and communicate effectively big and small cars, trucks, vans or motorbikes [3]. Children do it as well, when they are engaged in manipulating toy cars. Moreover, passionates are real experts in appraising and understanding engine sounds. They are even able to distinguish and mimic the specific sound of exhausts and engines' configurations. A quick query on Youtube shows a huge variety of technics and tutorials on how to make very realistic imitation of car sounds [4].

Given these premises, goal of the EU project SkAT-VG [5] (Sketching Audio Technologies using Vocalizations and Gestures) is to find ways to exploit and develop the use of voice [2], as a sound sketching and prototyping tool, in sound design practices. Research in SkAT-VG proceeds along three directions:

1. improving our understanding on how sounds are communicated through vocalizations and gestures, in terms of cross-modal relationships of kinematic gesture and sound features [3, 4, 5],

2. looking for physical relations between vocal sounds and sound-producing phenomena,

3. designing tools for converting vocalizations and gestures into parametrized sound models.

Automotive applications represent a powerful design scenario, which is currently under investigation in the context of SkAT-VG research.

In this paper we describe the work in progress regarding the development of a tool for sketching synthetic engine sounds, starting from the vocalizations of the sound designer. The paper is organized as follows: in section 2 several existing approaches to the synthesis of engine sounds are described and compared, section 3 describes in detail the architecture of our synthesis model, section 4 explains how the synthesis model is controlled by vocal imitations, and finally section 5 concludes the discussion and provides some suggestions for further developments.

[1] http://www.youtube.com/watch?v=HoEDLvQZg5I: The Audi e-Tron sound. Audi is developing the synthetic sound signature for its future electric cars.

[2] http://www.youtube.com/watch?v=uc3AFlaSa2o, http://www.soundracer.se/?p=98&p2=128: an example of electronic engine sound device.

[3] For instance, in the "Car Talk" radio show, people call and vocalize the sound that their car is making to describe a problem to the hosts. http://www.cartalk.com/.

[4] https://www.youtube.com/watch?v=_m6icEz7jII: A detailed explanation on how to make a realistic vocal imitation of a V10 engine sound, by means of a beer can. The main characteristic of the V10 sound is the 3rd major interval between low and middle frequency components.

[5] http://www.skatvg.eu/.

Copyright: ©2014 Stefano Baldan, Stefano Delle Monache, Luca Comanducci. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

2. APPROACHES TO ENGINE SOUND SIMULATION

Existing combustion engine sound simulators can be roughly split in two main groupings. In sample-based approaches, wavetable techniques are exploited to manipulate existing engine sounds recorded at specified RPM levels. Typically, sound samples are looped, cross-faded and pitch-shifted according to control information such as revolutions per minute, throttle and gear. This approach is largely used in video games. Even with relatively simple implementations it is possible to achieve a high level of realism [6]. However, the main drawbacks are that good simulations require the recording of huge banks of sound samples and that sound design is heavily constrained by the timbral space of the samples loaded.

In procedural audio approaches, sound is generated starting from the computed description of the characteristics of the sound producing event, i.e. the mechanics and configuration of the engine, according to a perceptually-relevant set of rules and control logics applied to live input [6]. In procedural sound, synthesis parameters ideally coincide with the parameters describing the underlying physical process. Analysis - resynthesis techniques of engine sound samples are more flexible and computationally affordable [7, 8], though still constraining the sound design space. On the contrary, physically-informed modeling of synthetic engine sounds allows the exploration of wider spaces with a single model, and a potentially more intuitive parametrization of physical events and dimensions. Nonetheless, most of the work done in this direction is either covered by patents or largely undocumented. One notable exception is Andy Farnell's engine sound model [9], which represents a valuable starting point for the development of our tool.

The sound of a four-stroke engine [7] is largely influenced by the behaviour of the exhaust valve. Its peculiar sound is especially produced during the upward motion of the piston, when gas under a high pressure is expelled during the exhaust stroke. Since the valve is open only in the positive pressure cycle of the piston and every other cycle, the sound can be represented by the positive half cycle of a cosine wave, once every two full cycles.

In Farnell's prototype, depicted in figure 1, the four cylinders are shifted in phase by a factor of 0.25 and slightly modulated in frequency by lowpass-filtered noise to take account of mechanical irregularities. In addition to the output of the four cylinders, a simulation of the exhaust system is realized by means of a delay chain split in four quadrants. Each quadrant receives inputs from other parts of the model and feedback from the quadrants, in order to represent the new excitations that happen at different times in the exhaust pipe. A parabolic function simulates the whole engine vibration, while three overtone generators and two FM operators are introduced to enrich the final sound. All of these elements can be separately controlled, together with the engine speed and audio mixing controls.

[6] `http://www.html5rocks.com/en/tutorials/casestudies/racer-sound/`: a nice html5 implementation of an engine sound model for the web-based game Racer.

[7] `http://en.wikipedia.org/wiki/Four-stroke_engine`.

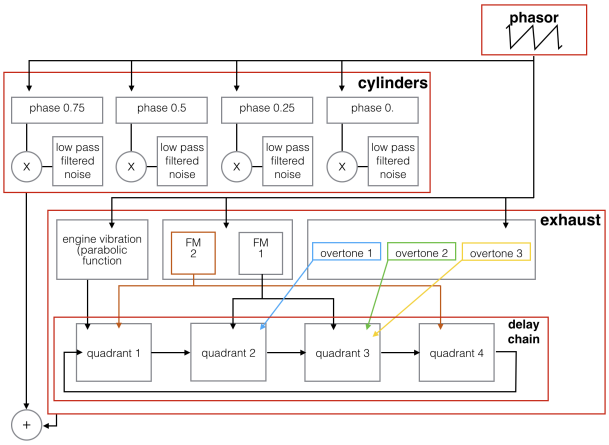


Figure 1. Layout of Andy Farnell's engine model.

Although these parameters allow a great amount of accuracy in controlling many acoustic details, the model results quite difficult to master. Indeed, some parameters have little physical meaning, sometimes with hardly predictable changes in sound.

From a spectral viewpoint, the sound texture of a combustion engine can be roughly described by two main components:

1. a stationary, harmonic tone, whose fundamental frequency is dependent on the revolutions per minute of the crank shaft, the engine vibrations and exhaust resonances,

2. a non-stationary, noisy component mostly produced by turbulences in the exhaust gases, and several mechanical interactions.

By coupling few and intuitive vocal techniques it is possible to effectively mimic a wide range of engine sounds: lip buzzes, palate grind, and rolling tongue [10] serve to render and enhance the stationary and pitched component of the engine. In addition, open tubes and even cans applied to the mouth are useful tools to augment the vocal sound and simulate the exhaust resonances. In this respect we moved from Farnell's implementation in order to have a simpler, yet more controllable model and tie the vocal sketches to the real time prototyping of synthetic sounds.

3. THE FOUR-STROKE ENGINE SOUND MODEL: A WORK IN PROGRESS

Figure 2 shows the diagram of our four-stroke engine model, which is currently implemented as Max/MSP patch. In our model, the combustion chambers of the engine cylinders are modeled as resonant cavities by means of comb filters. The positive half cycle of the cosine wave representing the opening of the exhaust valve is used to modulate the amplitude of both the cylinder output and the feedback of the delay line (i.e., the resonant cavity). The delay time of the comb filter changes accordingly to the volume of the combustion chamber, that is the delay time is longer when the exhaust valve opens, and shorter when the valve closes.

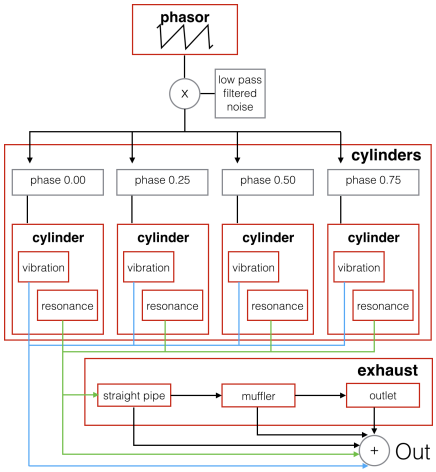


Figure 2. General layout of our engine model.

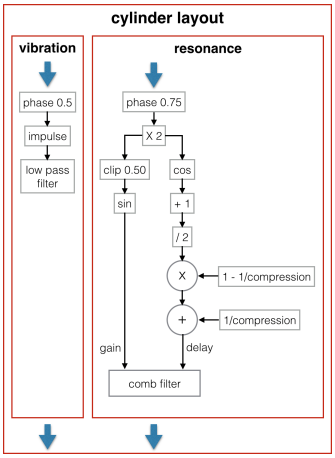


Figure 3. The cylinder model.

The delay time range is conditioned by the cylinders size and their compression ratio parameters. The compression ratio corresponds to the maximum volume of the combustion chamber, namely when the piston is at the bottom of its course, divided by its minimum volume, namely when the piston is at the top. Higher compression ratios result in a sound slightly richer in harmonics, due to the greater variation in the delay time of the comb filter and consequentially in the range of excited frequencies. The constructive and destructive interferences introduced by the comb filter results in a harmonically rich sound. Compared to Farnell's model, this solution works around the need to add the overtones generator module.

Instead of using a parabolic function, the engine vibration caused by the ignition of the fuel during the third stroke is modeled through lowpass-filtered impulses. Similarly, in the exhaust system shown in figure 4, the multiple quadrants layout is replaced by a sequence of three blocks, namely a straight pipe, a muffler and a shorter pipe serving as exhaust outlet. The two plain pipes are modeled by means of comb filters, whose delay time is directly proportional to the length of the tube and inversely proportional to the speed of sound. The final output is also delayed by the same amount of time, in order to simulate a sound wave passing through the whole length of an open tube. The muffler is made of three tubes of different length, placed in parallel. A second order lowpass filter simulates the effect of noise-absorbing glass wool of the car muffler.

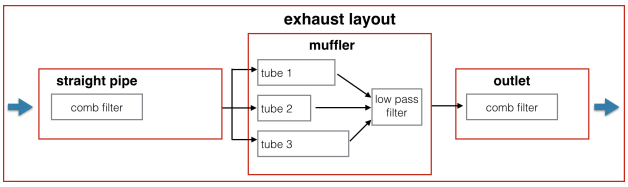


Figure 4. The exhaust model.

Their lengths are mutually prime, trying not to have frequencies fully enhanced or cancelled by more than one tube, and modulated by a small amount of white noise, an unconventional solution which adds some buzz and non-stationarity to the sound.

Finally, the exhaust system is controllable in size and resonance (i.e., exhaust length, resonance, silencer buzz and cut off). Each block of this model (i.e., explosions, exhaust valves, main exhaust tube, muffler, and exhaust outlet) can be mixed, thus providing the effect of listening to the car sound from different pick up points.

4. VOCAL CONTROL OF ENGINE SOUNDS

Our goal is to prototype synthetic engine sounds, starting from sketches produced with the voice, in an environment flexible enough to have an immediacy of communication and comparison [11]. The vocal sketch sets a landmark in the sound design space of the model, and provides the designer with an engine sound configuration that can be further tweaked. For this purpose, several low-level audio descriptors [12] are exploited to describe the salient timbral qualities of the vocal sound. Extracted features are used to drive the control parameters, such as RPM, cylinders size, exhaust length and resonance. At this stage, the choice of the descriptors used is based on empirical observations, in terms of producing meaningful effects from the viewpoint of the physical dimensions of the sound model. The next subsections introduce the audio descriptors and their associations with the model control layer.

4.1 Fundamental pitch: revolutions per minute

Internal combustion engines commonly found in cars, motorbikes and other vehicles operate at frequencies which fall in the audible range. The periodic succession of explosions, the gases from the exhaust system and the revolving mechanical parts generate an almost stationary tuned sound. In a four-stroke engine each cylinder fires once every two full revolutions of the crank shaft. A short burst of sound is generated due to the ignition of the fuel and to the sudden change of pressure in the exhaust system, when combusted gases are expelled. If the cylinders are equally spaced in phase, which is often the case, the bursts will happen at regular intervals across each revolution. Therefore the number of RPM and the number of cylinders in the engine have a direct impact on the fundamental pitch

of its stationary sound. This relation is described by the equation:

$$f = \frac{RPM}{60} * \frac{n_{cylinders}}{2} \quad (1)$$

that is RPM is equal to:

$$RPM = \frac{f * 120}{n_{cylinders}} \quad (2)$$

A pitch detection algorithm applied to the vocal sound, is used to estimate the fundamental frequency, and feed the RPM control of the engine. Our implementation is based on the harmonic histogram algorithm available in the Zsa [8] descriptors library, for Max/MSP [13]. A first order low-pass filter, with a cutoff frequency empirically set at 300 Hz, is placed before the detection algorithm, in order to reduce the weight of higher harmonics. A median filter is applied to the estimated fundamental frequency to remove outliers in the detection, yet preserving sudden changes of pitch in the mimicked sound.

4.2 Spectral centroid: engine size and exhaust silencing

The peculiar modes of vibration of resonant objects and cavities depend, among other things, on their geometry and size. In general, the spectral energy of the sound produced by a big resonating object is concentrated at lower frequencies, whereas a small object resonates at higher frequencies. The same concept applies to the vocal sound production: vibrations coming from the vocal folds are selectively modulated by the resonant cavities of the vocal tract, according to their variable configuration and size. Expanding the oral cavity produces vowels with formants at low frequencies, conversely vowels produced by shrinking the oral cavity show formants at higher frequencies.

The point around which spectral energy concentrates can be found calculating the *spectral centroid* c, namely the average of all the frequencies f weighted by their corresponding amplitudes a, divided by the sum of the amplitudes:

$$c = \frac{\sum_{k=0}^{N-1} a(k) f(k)}{\sum_{k=0}^{N-1} a(k)} \quad (3)$$

This value effectively describes the perception of *brightness* for a particular sound [14], independently of its fundamental pitch [15].

We exploit the spectral centroid to control the brightness of the synthetic sound. Indeed the brightness of the engine sound is dependent on the size of the virtual engine, exhaust system included, and the actual configuration of the silencer (i.e., the cut off frequency). Therefore, a spectral centroid computed on the bandwidth of 300-3000 Hz is used to keep only the formants produced by the modulation of the vocal tract. This descriptor maps onto the size parameter. In practice, the centroid works as a scale factor of the delay time of the waveguides in the engine model. The frequency cut off of the silencer, instead, is manipulated according to a second spectral centroid calculated on the full bandwidth, in order to take in account the information coming from lip buzzes and pops, tongue flops, sibilant noises, and so forth.

[8] http://www.e--j.com/index.php/what-is-zsa-descriptors/.

4.3 Spectral spread: exhaust resonance

Sometimes, resonating objects are conceived and tuned to vibrate at a given frequency. The oscillators found in musical instruments such as strings, reeds, metal bars are a significant example. In these objects, the spectral energy is highly concentrated around the main mode of resonance. Concerning vocal production, vowel sounds can also be characterized by to the salience of one or more formants.

The degree of concentration of the spectral energy around a single value is effectively described by the *spectral spread* s, namely the standard deviation of the spectral values around the centroid c:

$$s = \left(\frac{\sum_{k=0}^{N-1} a(k) \left(f(k) - c\right)^2}{\sum_{k=0}^{N-1} a(k)} \right)^{1/2} \quad (4)$$

For this purpose, the spectral spread of the mimicked sound is used to control the gain coefficient of the comb filters which model the exhaust tubes, thus affecting the amount of resonance introduced by the exhaust system.

4.4 Mean absolute bin differences: buzz and roar

The internal combustion engine typically delivers thousands of explosions per minute, producing a large amount of vibrations and air movement. Most of the blocks forming the exhaust system are especially designed for reducing the resulting noise, by means of appropriate silencing devices.

Modern road legal vehicles are equipped with very efficient silencers which reduce both acoustic and air pollution at the cost of a slight decrease in the efficiency of the engine. In racing contexts these devices are often modified or removed completely, to maximize the engine performance instead. The resulting engine sound is louder and more aggressive, with buzz and roar noises in foreground. Vocal imitations of these kinds are achieved by exaggerating the rough and noisy effect of sound techniques, such as tongue vibration (e.g., used to pronounce the letter *r*), or the palate grind (e.g., used to imitate guns and explosions).

To extract these vocal sound characteristics, we first took into account the amount of overlap of salient frequencies (spectral peaks) across critical bandwidths [16]. However, this approach did not provide coherent controls on the silencer pipes, because of the monophonic nature of the human voice. Moreover, the perceived roughness in mouth sounds is conveyed by irregularities in the time domain rather than by beating among various frequencies. The next attempt was to consider the mean absolute difference of the global energy among adjacent spectrogram frames (*spectral flux*), as roughness descriptor. Finally, we

considered the inverse of the mean absolute bin difference of a single spectrogram frame (l) as a measure of the flatness of the spectrum:

$$l = \frac{\sum_{k=0}^{N-1} a(k)}{\sum_{k=1}^{N-1} |a(k) - a(k-1)|} \quad (5)$$

The underlying logic is that a clean vocalization has many distinct peaks in the spectrogram, that is the spectral derivative shows many sudden changes and as a consequence large absolute differences among its bins. Conversely, a noisy vocal sound has a spectral energy spread across more frequencies, with a flatter first spectral derivative and smaller absolute differences. Therefore, the computed flatness is used to modulate the delay times in the silencer pipes block.

5. CONCLUSIONS AND FUTURE WORK

The voice-controlled engine sound synthesizer is implemented as Max/MSP [9] patch.

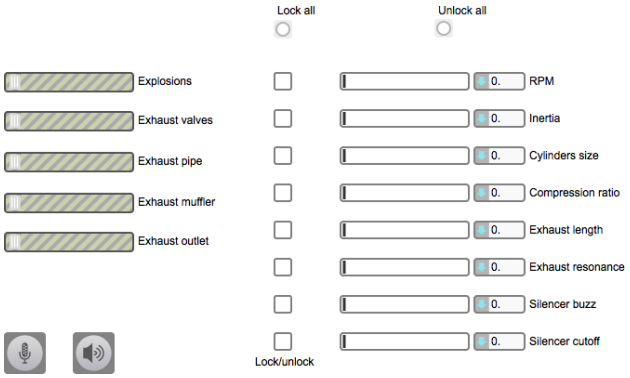


Figure 5. The graphical user interface for the synthesis model.

Being the development of the tool part of a wider study on sonic sketching practices, we expect to improve the audio feature extraction layer, the sound model architecture, and the user experience. At this stage, the current user interaction only allows to specify the parameters configuration through vocal sounds, and freeze it for further manual fine-tuning. Still, the prototype is flexible enough to create on the fly small and big engine sounds, city, sport and muscle car-like sounds, and motorbikes. Following the workshop on "Sonic sketches, voice and gestures in sonic interaction design" [10] [17], we plan to run a workshop on sonic sketching for future car sound. We are especially interested in further exploring the exploitation of typo-morphological attributes (i.e., high-level descriptors of spectro-temporal dynamics, such as shape, irregularity, direction, and size) of sound [18] as vocal sketching patterns, in the specific design scenario of car sound. On one hand, the interactive sketching tool will serve to validate the goodness of use of these emerging patterns, in terms of their coupling with both audio features extraction for vocal control and control parameters of the sound model. On the other hand, we are interested in finding ways to unfold the sound design space and provide the user with a creative and expressive environment. One hypothesis is to replace the engine explosions block with other kinds of sounds. In addition, the resulting engine sound could be convolved with other vocalizations, provided by the designer. Vehicle architecture and designs are expected to change with the introduction of new electric drivetrain systems [19], how will sound contribute to affect the shape aspect of future cars?

[9] www.cycling74.com.

[10] http://www.sampl-lab.org/index.php/Bozzetti_sonori,_voce_e_gesto_nel_design_dell%27interazione_sonora.

6. ACKNOWLEDGMENTS

The authors are pursuing this research as part of the project SkAT-VG and acknowledge the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 618067.

7. REFERENCES

[1] R. Schirmacher, "Active design of automotive engine sound," in *Audio Engineering Society Convention 112*, Apr 2002.

[2] I. Ekman and M. Rinott, "Using vocal sketching for designing sonic interactions," in *DIS '10: Proceedings of the 8th ACM Conference on Designing Interactive Systems*, (New York, NY, USA), pp. 123–131, ACM, 2010.

[3] G. Lemaitre, A. Dessein, P. Susini, and K. Aura, "Vocal imitations and the identification of sound events," *Ecological Psychology*, vol. 23, no. 4, pp. 267–307, 2011.

[4] G. Lemaitre and D. Rocchesso, "On the effectiveness of vocal imitations and verbal descriptions of sounds," *The Journal of the Acoustical Society of America*, vol. 135, no. 2, pp. 862–873, 2014.

[5] B. Caramiaux, F. Bevilacqua, T. Bianco, N. Schnell, O. Houix, and P. Susini, "The role of sound source perception in gestural sound description," *ACM Transactions on Applied Perception (TAP)*, vol. 11, no. 1, p. 1, 2014.

[6] A. Farnell, "Behaviour, structure and causality in procedural audio," in *Game sound technology and player interaction concepts and developments* (M. Grimshaw, ed.), pp. 313–329, New York, NY, USA: Information Science Reference, 2010.

[7] S. A. Amman and M. Das, "An efficient technique for modeling and synthesis of automotive engine sounds," *IEEE Transactions on Industrial Electronics*, vol. 48, no. 1, pp. 225–234, 2001.

[8] Y. Ban, H. Banno, K. Takeda, and F. Itakura, "Synthesis of car noise based on a composition of engine noise and friction noise," in *ICASSP'02*, pp. 2105–2108, 2002.

[9] A. Farnell, *Designing Sound*. Cambridge, MA: MIT Press, 2010.

[10] F. Newman, *MouthSounds: How to Whistle, Pop, Boing and Honk for All Occasions... and Then Some*. New York: Workman Publishing Company, 2004.

[11] M. Aramaki, C. Gondre, R. Kronland-Martinet, T. Voinier, S. Ystad, *et al.*, "Thinking the sounds: an intuitive control of an impact sound synthesizer," in *Proceedings of the International Conference on Auditory Display*, pp. 119–124, 2009.

[12] G. Peeters, B. Giordano, P. Susini, N. Misdariis, and S. McAdams, "The timbre toolbox: Audio descriptors of musical signals," *Journal of the Acoustical Society of America*, vol. 130, no. 5, pp. 2902–2916, 2011.

[13] M. Malt and E. Jourdan, "Zsa. descriptors: a library for real-time descriptors analysis," *Sound and Music Computing, Berlin, Germany*, 2008.

[14] J. M. Grey and J. W. Gordon, "Perceptual effects of spectral modifications on musical timbres," *Journal of the Acoustical Society of America*, vol. 63, no. 5, p. 14931500, 1978.

[15] E. Schubert and J. Wolfe, "Does timbral brightness scale with frequency and spectral centroid?," *Acta acustica united with acustica*, vol. 92, no. 5, pp. 820–825, 2006.

[16] P. Daniel, "Psychoacoustical roughness," in *Handbook of Signal Processing in Acoustics*, pp. 263–274, Springer New York, 2008.

[17] S. Delle Monache, S. Baldan, D. A. Mauro, and D. Rocchesso, "A design exploration on the effectiveness of vocal imitations," in *accepted for publication in Proceedings of 11th SMC Conference on Sound and Music Computing*, 2014.

[18] A. Merer, M. Aramaki, S. Ystad, and R. Kronland-Martinet, "Perceptual characterization of motion evoked by sounds for synthesis control purposes," *ACM Transactions on Applied Perception (TAP)*, vol. 10, no. 1, p. 1, 2013.

[19] M. Luccarelli, D. T. Matt, P. Russo Spena, and M. Lienkamp, "Purpose design for electric cars. parameter defining exterior vehicle proportions," in *Conference on Future Automotive Technology. Focus Electromobility*, 2014.

LA LIBRERIA OPEN MUSIC OMCHROMA DOCUMENTAZIONE ONLINE

Luca Richelli
Sound and Music Processing Lab Conservatorio "C. Pollini" - Padova
lucarichelli@gmail.com

SOMMARIO

OMChroma [1] è una libreria per la sintesi digitale integrata dentro l'ambiente di aiuto alla composizione OpenMusic [2]. È una generalizzazione di Chroma, un sistema sviluppato da Marco Stroppa all'inizio degli anni Ottanta al Centro di Sonologia Computazionale dell'Università di Padova e in seguito riscritto in Lisp ed integrato in OpenMusic a partire dal 2000 da Carlos Agon e Marco Stroppa [3].

Quest'articolo illustra le caratteristiche salienti della documentazione online di OMChroma [4], scritta dall'autore per incarico dell'Institute de Recherche et de Coordination Acoustique/Musique (IRCAM) di Parigi.

1. INTRODUZIONE

Esistono altre librerie per l'integrazione di Csound in OpenMusic, tra cui *Om2Csound* realizzata da Karim Haddad, Mikhail Malt e Laurent Pottier nel 1999 [5] e *om4Csound* sviluppata da Mauro Lanza dal 2008 [6]. Tutte e tre le librerie si basano sul motore di sintesi Csound [7] con il fine di integrare le potenzialità di sintesi sonora di Csound all'interno di OpenMusic, ma differiscono nell'impostazione concettuale.

Om2Csound è finalizzata solamente alla compilazione dei file *score* da utilizzare direttamente con Csound autonomamente. *Om4Csound* offre invece un sistema che integra il software di sintesi (Csound) all'interno del software di controllo usufruendo dei vantaggi della programmazione grafica ad oggetti (object-oriented visual programming language).

OMChroma condivide alcune caratteristiche con alcuni sistemi dedicati alla sintesi sonora, progettati intorno agli anni Novanta, come ad esempio *Formes* [8], *Common Music/Common Lisp Music* [9] e *Patchwork/PWGL* [10], il cui scopo è la creazione di un ambiente integrato per l'elaborazione di strutture di controllo per la sintesi sonora.

Copyright: © 2014 Luca Richelli. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

2. LA MATRICE DI OMCHROMA

Sebbene allo stato attuale sia implementato l'interfacciamento solamente con Csound, OMChroma permette la rappresentazione dei parametri di controllo della sintesi indipendentemente dal motore di sintesi attraverso una matrice bidimensionale.

In Csound, come nei precedenti *Music N* software [11], il processo di sintesi è diviso in due parti: un'*orchestra* (file .orc) [12] contenente uno o più *instrument* che realizzano il digital signal processing, e una *score* (file .sco) [13] contenente i parametri necessari all'*orchestra* per la sintesi seguendo una lista temporale (*timeline*) di singoli eventi. La comunicazione tra l'*orchestra* e la *score* avviene tramite i *p-field* [14] gestibili a piacere dall'utente, ad eccezione dei primi tre riservati dal sistema (*instrument number*, *start-time e duration*).

In OMChroma la stesura dell'orchestra consiste nell'inizializzare, all'interno dell'ambiente grafico, un'orchestra di Csound tradizionale, ovvero un file di testo. L'inizializzazione creerà una *user class* specifica per ogni strumento dell'*orchestra*: per esempio l'orchestra "myorchestra.orc" contenente tre strumenti genererà le *user class* "myorchestra-1, myorchestra-2 e myorchestra-3". Poichè il primo *p-field* delle *score* indica il numero strumento e le *user class* di OMChroma indicano già uno strumento determinato, gli slot per l'inserimento dei dati (*p-field*) delle *user class* saranno pari al numero dei *p-field* dichiarati nell'orchestra meno 1 (ovviamente il primo *p-field*).

Per quanto riguarda la *score* le classi di OMChroma rappresentano i dati attraverso una matrice ruotata rispetto alla *score* di Csound. Nella *score* Csound ad ogni riga corrisponde un evento singolo con tutte le sue componenti mentre in OMChroma ogni riga contiene tutti i dati riguardanti il medesimo *p-field*.
Prendendo ad esempio la seguente *orchestra* di Csound che realizza un'oscillatore sinusoidale controllato in ampiezza (*p-4*) e frequenza (*p-5*)

```
instr 1
iamp = ampdb (p4)
ifreq = p5
ifn = 1
asig oscil iamp, ifreq, ifn
out asig
endin
```

e la *score*

p-1	p-2	p-3	p-4	p-5
instr. #	init time	dur	amp	freq
1	0	3	0	440
1	3	1	-6	660
1	5	2	-12	880

Tabella 1. Score Csound

La corrispondente *user class* in OMChroma rappresenta i dati nel modo sequente:

p-2	e-del	0	3	5
p-3	dur	3	1	2
p-4	amp	0	-6	-12
p-5	freq	440	660	880

Tabella 2. Matrix OMChroma

3. OMCHROMA USER MANUAL

3.1 System Configuration and Installation

"System Configuration and Installation" riporta le informazioni necessarie per l'installazione della libreria OMChroma e sugli altri componenti necessari al corretto funzionamento della libreria all'interno di OpenMusic oltre ai link utili.

3.2 Getting started

In OMChroma gli elementi disponibili per la creazione della programmazione visuale (*patch*), sono rappresentati da contenitori (*box*) differenti, suddivisi in *class* (classi) e *function* (funzioni).

In OMChroma, come in OpenMusic, le classi sono delle strutture di dati, mentre le funzioni si riferiscono alla funzioni sottostanti che eseguono una determinata operazione. Per chiarire: in OpenMusic *TEXTFILE*, un buffer di testo, è una classe, mentre *OM+*, l'operazione addizione, è una funzione. Il capitolo passa quindi in rassegna, con l'ausilio della classe ADD-1 (sintesi addittiva) e della funzione *synthesize*, le procedure di editing di base.

Ogni classe di sintesi è anche detta evento, ed ogni evento ha un determinato numero di elementi, gli input degli eventi sono detti *slot*. I primi quattro *slot* di ogni classe sono detti globali e visualizzati in blu. Gli altri slot, il cui numero è specifico per ogni classe, sono detti addizionali e visualizzati in rosso.

Una delle caratteristiche rimarchevoli dell'ambiente di programmazione di OMChroma è il polimorfismo: ogni tipologia differente di dati (numeri singoli, liste, *Break Point Function*, nomi di funzione, *lambda function*) viene processata secondo la sua caratteristica agevolando notevolmente la programmazione e il flusso dei dati.

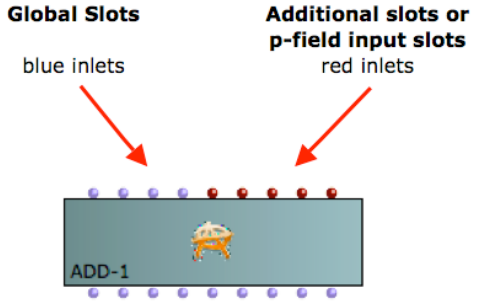


Figura 1. La classe ADD-1

3.3 Managing GEN function and sound files

In Csound è fondamentale la conoscenza e l'utilizzo delle *f-table*, vettori di valori a virgola mobile, generati dalle *GEN Routine* [15]. In OMChroma la gestione delle funzioni è uno degli aspetti cruciali per l'ottimizzazione del rendering audio e per sfruttare completamente le possibilità compositive di OpenMusic.

La libreria permette infatti un duplice impiego delle funzioni: locale e globale. Anche se è possibile ottenere lo stesso risultato usando sia le funzioni locali che quelle globali, il flusso di programma segue due percorsi differenti. Le funzioni globali vengono inizializzate solo una volta ed indirizzate tramite il loro numero univoco di assegnazione. Tutte le istanze di sintesi che faranno riferimento a questa funzione verranno indirizzate "fisicamente" alla stessa funzione.

Il funzionamento logico è quindi identico alla creazione di *f-table* all'interno della score di Csound. Le funzioni locali vengono ri-inizializzate per ogni singolo evento di sintesi, anche se la funzione rimane invariata. Di conseguenza è possibile un uso più flessibile a scapito di un maggior utilizzo di memoria ed una minore efficienza del rendering audio.

Il capitolo termina con la rassegna dei vari modi con cui la libreria può leggere i file audio.

3.4 Predefined Classes

OMChroma mette a disposizione 31 differenti classi predefinite con implementate le principali tecniche di sintesi. Lo scopo delle classi predefinite non è quello di comprendere tutte le tecniche di sintesi (impresa ovviamente impossibile), ma di mostrare all'utente una panoramica di strategie per la costruzione delle proprie classi (vedi in seguito 3.6 Creating a new Class) e di mettere a disposizione delle classi "pronte per l'uso" per incominciare a comprendere il funzionamento dell'ambiente di programmazione e le sue peculiarità.

Tecnica di sintesi	Elenco Classi
Additive Synthesis	ADD-1, ADD-2, ADD-3, ADD-A1
Buzz Synthesis	BUZZ-1, BUZZ-2, BUZFL-1
Frequency Modulation	FM-1, FM-2
Formant Wave Function (FOF)	FOF-1, FOF-2, FOF-3, FOF-4, FOF-A1, FOF-A4

Granular Formant Wave Function (FOG)	FOG-1
Karplus-Strong	PLUCK-1, PLUCK-2
Random Amplitude Modulation	RAN-1, RANFL-1
Sampler	SMPL-1, SMPL-2, SMPL-A1, SMPL-A2, SMPL-3, SMPL-4, SMPL-5, SMPL-6
Subtractive Synthesis	SUB-1
Wave Shaping	WSHP-1
Hybrid Models	SNARE-1

Tabella 3. Elenco delle classi predefinite

Sono allo sviluppo ulteriori classi, con il progetto di creare anche una sezione con quelle create dalla comunità degli utenti: OMChroma User Group (http://forumnet.ircam.fr/user-groups/).

3.5 User-fun

La *user function* è un potente mezzo per applicare una funzione lambda [16] ad ogni componente delle classi di OMChroma. In molti casi è possibile ottenere lo stesso risultato attraverso la programmazione delle funzioni di OpenMusic, ma la *user function* permette un maggior livello di astrazione ed una migliore efficienza di programmazione. La *user function* è una funzione che permette di effettuare una serie complessa di operazioni su tutta la matrice contenente le singole istanze di digital signal processing senza dover necessariamente procedere secondo la successione temporale degli eventi. Facendo riferimento al processo di sintesi di Csound, si ha la possibilità di intervenire con una funzione complessa per modificare, filtrare aggiungere, togliere, duplicare ogni singola istanza (*i-statement*) della *score* prima del rendering audio.

3.6 Creating a new Class

Questa è sicuramente la risorsa più grande della libreria. Anche se è possibile utilizzare OMChroma solamente con le classi predefinite, è fondamentale che l'utente abbia la possibilità di implementare le proprie classi (*user class*), che equivale alla scrittura delle proprie orchestre con Csound. Il procedimento non è particolarmente complesso: l'attenzione principale deve essere rivolta all'inizializzazione dei *p-field* utilizzati in funzione polimorfica (vedi 3.2 Getting started).

3.7 Multichannel processing

Tutte le classi predefinite sono monofoniche. Le *user class* possono ovviamente avere un'uscita multicanale utilizzando gli appositi *opcode* di Csound, ma c'è un modo più elegante e sofisticato per la gestione della disposizione spaziale dei file audio: l'utilizzo della libreria OMPrisma [17] ora integrata in OMChroma attraverso la funzione *Chroma-Prisma*.

OMPrisma è una libreria per il controllo dei processi di spazializzazione che permette di sviluppare complesse morfologie spaziali attraverso una descrizione simbolica dei parametri di sintesi. È dunque possibile tenere separati il processo di sintesi e quello di spazializzazione, permettendo la realizzazione di rendering alternativi in funzione del numero e delle possibili disposizioni degli altoparlanti previsti per la diffusione sonora.

4. SVILUPPI FUTURI

Il completamento della documentazione è attualmente allo stadio progettuale e prevede:

- Una breve introduzione alle tecniche di sintesi utilizzate nelle classi predefinite,
- L'estensione delle classi predefinite con l'inclusione di tecniche di sintesi granulari,
- L'aggiunta, alle classi predefinite, di una sezione di *classi avanzate* includendo anche quelle proposte dalla comunità degli utenti,
- Un capitolo dedicato al *Vertical Pitch Structure (VPS)* [18]: strutture polimorfiche che permettono di rappresentare i materiali spettrali come frequenze assolute, relative o in relazione simbolica ad una *frequenza-pivot*. Sono strutture che raccordano l'approccio compositivo armonico/melodico e quello numerico /spettrale,
- Un capitolo dedicato ai *Chroma-model*: strutture astratte di dati parametrici e formali per la sintesi, costituite sia da liste di segmenti temporali (time structures) che da sequenze di *Vertical Pitch Structure*,
- Un capitolo dedicato all'importazione delle analisi spettrali effettuate con Audiosculpt [19] o altri software che utilizzano il protocollo Sound Description Interchange Format (SDIF) [20] per la codifica dei dati.

5. CONCLUSIONI

La diffusione ed utilizzo di qualsiasi tipo di piattaforma software dipende, oltre che dalla validità intrinseca del progetto stesso e della sua realizzazione, anche dall'esistenza di una documentazione approfondita che permetta agli utenti di comprendere facilmente il funzionamento globale dell'ambiente e le funzioni di ogni singolo componente. Anche se la documentazione relativa ad una parte dei componenti di OMChroma non è ancora pronta, lo stato attuale del manuale permette all'utente di arrivare ad un buon livello di conoscenza della libreria e di essere in grado di sviluppare autonomamente le proprie *user class*: ovvero la risorsa principale di OMChroma.

La scelta di un'edizione online permetterà un aggiornamento continuo e costante del manuale, anche e soprattutto a seguito delle osservazioni della comunità degli utenti.

6. BIBLIOGRAFIA

[1] C Agon, J. Bresson, and M. Stroppa: “OMChroma: Compositional Control of Sound Synthesis” Computer Music Journal, 35:2 pp. 67–83, 2011.

[2] G. Assayag, C. Rueda, M. Laurson, C. Agon, and O. Delerue, "Computer Assisted Composition at Ircam : PatchWork & OpenMusicComputer" Computer Music Journal, 23:3, pp. 59-72 1999.

[3] C. Cadoz, A. Luciani, and J.-L. Florens: “Responsive input devices and sound synthesis by simulation of instrumental mechanisms: the Cordis system” Computer Music Journal, 8:3, pp. 60-73, 1984.

[4] http://support.ircam.fr/docs/om-libraries/omchroma/co/OMChroma.html visita 21 settembre 1014

[5] K. Haddad: "OpenMusic Om2Csound, Bibliothèque de modules de generation de scores pour Csound", version 1, IRCAM Centre Georges Pompidou, Paris 1999.

[6] M. Lanza, G. Verlingieri, and N. Biagioni: "La libreria OpenMusic om4Csound. Introduzione e progetto di documentazione" Atti del XVIII CIM Colloquio di Informatica Musicale Torino - Cuneo pp. 166-172, 2010

[7] B. Vercoe, "The Canonical Csound Reference Manual", Version 6.00.1, MIT http://csounds.com/manual/html/

[8] X. Rodet, and P. Cointe: "Formes: Composition and Scheduling of Processes". Computer Music Journal, 8:3, pp. 32-48, 1984

[9] H. Taube: "Common Music: A Music Composition Language in Common Lisp and CLOS". Computer Music Journal, 15:2, pp. 21-32, 1991

[10] M. Laurson, M. Kuuskankare, and V. Norillo: "An Overview of PWGL, a Visual Programming Environment for Music". Computer Music Journal, 33:1, pp. 19-31, 2009

[11] C. Roads: "The Computer Music Tutorial". Cambridge, Massachussetts: MIT Press, 1996

[12] R. Boulanger: "The Csound Book. Perspectives in Software Synthesis, Sound Design, Signal Processing, and programming".Cambridge, Massachussetts: MIT Press, pp. 7-8, 2000

[13] Ibid. pp. 8-9

[14] Ibid. pp. 14-15

[15] R. Bianchini, and A. Cipriani: “Il Suono Virtuale. Sintesi ed Elaborazione del Suono – Teoria e Pratica con Csound”. Seconda ed. Roma, ConTempo, pp. 9-15, 2001

[16] Bresson, C. Agon, and G. Assayag: "Visual Lisp/CLOS Programming in OpenMusic" Higher Order and Symbolic Computation vol. 22 pp. 81-111, 2009

[17] M. Schumacher, and J. Bresson: "Compositional Control of Periphonic Sound Spatialization" Proc. of the 2nd International Symposium on Ambisonics and Spherical Acoustics, May 6-7, Paris, 2010

[18] M. Stroppa: "Structure, Categorization, Generation and Selection of Vertical Pitch Structures (VPS). A Musical Application in Computer Assisted Composition" IRCAM Document, 1988 http://articles.ircam.fr/textes/Stroppa88a/index.pdf

[19] C. Diatkine: “AudioSculpt 3.0 User Manual”. 2011 http://support.ircam.fr/docs/Audio Sculpt/3.0/co/AudioSculptguideWeb. html

[20] D. Schwarz, and M. Wright: "Extensions and Applications of the SDIF Sound Description Interchange Format" International Computer Music Conference Berlin, pp. 481-484, 2000

PER UN'ANALISI DI IMPROVVISO STATICO DI CARMINE EMANUELE CELLA

Elisabetta Capurso
betticap@hotmail.com

SOMMARIO

La composizione *Improvviso statico* per sassofono contralto e live electronics di Carmine Emanuele Cella è l'argomento fondamentale della nostra analisi. Scritta nel settembre -ottobre 2012, eseguita il 13 ottobre dello stesso anno a Emufest, il Festival di musica elettronica del Conservatorio S. Cecilia di Roma, la composizione è dedicata al sassofonista Enzo Filippetti e a Giorgio Nottoli, quest'ultimo curatore artistico per molti anni del Festival stesso. L'analisi i propone anzitutto di collocare la figura di Carmine Emanuele Cella nel contesto storico della musica classica contemporanea, al fine di metterne in rilievo le peculiarità della scrittura compositiva in relazione anche alla produzione musicale attuale.
Quindi l'indagine analitica mette in rilievo la doppia formazione musicale e scientifica del compositore: la colleganza di scuola con Aurelio Samorì al Conservatorio di Pesaro, gli studi e la laurea in filosofia logica conseguita all'Università di Urbino, il lungo lavoro come ricercatore svolto negli ultimi anni presso l'Ircam, l'istituto di ricerca musicale di Parigi. Ma, al di sopra di ogni considerazione, l'analisi si propone di mettere in evidenza le riflessioni importanti che sono alla base della scrittura di Carmine Cella: la natura dicotomica della musica, il parametro del timbro come fondamento del senso della musica.

1. INTRODUZIONE
Carmine Emanuele Cella : Biografia

<In un mondo in cui si chiede sempre più specializzazione e settorialità il mio modo di essere é sempre stato trasversale. La mia base di conoscenza è tesa all'unione tra discipline apparentemente diverse come la musica, la matematica e la teoria dei segnali.> [1]
In tale modo - con una citazione che gli appartiene - Carmine Emanuele Cella definisce la propria personalità di compositore e di ricercatore in ambito scientifico.
Nato a Urbino nel 1976, Carmine E. Cella è un interessante rappresentante, fra i compositori attuali, di quella tendenza tipica del nostro tempo a fare coesistere molteplici esperienze intellettuali e ad inserirle in contesti di formazione della propria individualità e di produzione artistica diversi.

Copyright: © 2014 This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use,d istribution, and reproduction in any medium, provided the original author and source are credited.

L'avventura ha inizio a Pesaro e a Urbino. Al Conservatorio 'G. Rossini'- noto fra l'altro per essere stato nel corso degli anni la sede di interessanti esperienze innovative soprattutto nel campo della didattica della composizione - Carmine E. Cella si diploma in pianoforte, composizione, musica elettronica. All'Università di Urbino studia e si laurea in filosofia logica. Il dottorato di logica matematica arriva più tardi nel 2011 all'Università di Bologna, con una tesi dal titolo "*On Symbolic Representations of Music*".
Già negli anni 2007-2008 era stato chiamato come ricercatore/sviluppatore nel gruppo di ricerca di Analisi e Sintesi presso l'Ircam, l'Istituto di ricerca musicale, soprattutto elettronica, di Parigi. Una doppia formazione culturale la sua, scientifica e musicale, che apre alla creazione <del suono con il suono>, dei suoni prodotti dal computer e dagli strumenti acustici, anche dalla loro combinazione. Il processo della combinazione si rivelerà, infatti, nel corso degli anni come un tratto tipico della sua personalità artistica, non soltanto dal punto di vista di una ricerca dei possibili accorpamenti, di unione di piani sonori diversi, pure sempre flessibili per le esigenze timbriche, ma anche dal punto di vista di una vocazione generale a gestire processi musicali dal carattere combinatorio. A nostro avviso, in effetti, due riflessioni fondamentali conducono all'approccio combinatorio, due percorsi di pensiero sottendono alla modalità compositiva di Carmine E. Cella.
La prima riflessione conduce l'interesse del compositore nei confronti della composizione musicale alla consapevolezza che la musica presenta al suo interno una natura dicotomica: da una parte esprime delle emozioni, dall'altra si mostra come un sistema formale con delle regole precise. L'attenzione all'aspetto razionalista, alla ricerca della struttura formale collegata al piano delle altezze dei suoni musicali emerge dalle composizioni che si rifanno agli anni della colleganza di scuola con Aurelio Samorì. La gestione delle altezze e la creazione di figure musicali, ossia la collocazione di una regia delle altezze in una sintassi musicale, rappresentano i termini del sostanziale interesse di una grande parte della sua prima produzione musicale.

1 Citazione da "Sintesi di Presentazione per il Concorso S. Fedele" Milano , 2010-2013

La successiva riflessione importante sulla musica arriva con una diversa maturità, collegata anche all'esperienza di studio vissuta con Azio Corghi: il senso della musica risiede essenzialmente nel timbro del suono. Attraverso il lavoro sull'orchestrazione e le sue tecniche si è affinata la componente più razionale che in precedenza guidava la gestione matematica delle altezze dei suoni.
Gli elementi della nuova scrittura sono evidenti nelle opere degli ultimi anni: *Reflets de l'hombre* per grande orchestra e live electronics (2013); *The Manhattan distance* per orchestra (2010 – revisione 2014); *La mémoire de l'eau* (2014) per Ensemble, eseguita dall'EOC alla Casa de Velàsquez a Madrid.

1.1 *Physical-inspired-synthesis*
Modellazione dell'ampiezza, il timbro

Il peso che la ricerca sul parametro timbro occupa nella riflessione sull'attività compositiva emerge in modo particolare dalle numerose pubblicazioni a carattere didattico che Cella ha dedicato ai diversi ambiti musicali e matematici, mettendo in rilievo, attraverso la tensione unitaria dei processi teorici, anche l'unità del processo compositivo. In buona parte esse sono riunite in una raccolta dal titolo significativo "Teaching Resources". Di particolare interesse speculativo risultano essere le pubblicazioni intitolate "Physical- ispired-synthesis", che argomentano a proposito della sintesi ispirata al modello fisico, cioè la sintesi che genera i suoni copiando il sistema meccanico sotto vibrazione e non copiando il suono stesso. La natura stessa della sintesi vuole alcune informazioni, pure brevi. Essa è rappresentata da un sistema eccitatore-risonatore; fra le diverse modalità di sistemi c'è la sintesi modale, implementata in filtri a due poli e pesi, questi ultimi detti pesi modali. Le frequenze e le linee di decadimento sono basate su delle proprietà fisiche, le ampiezze sono determinate dall'interazione di *feedback* fra i risonatori e l'eccitatore. Infatti se un segnale *feedback* è aggiunto all'impulso digitale originale l'eccitazione del segnale stesso produrrà una modellazione temporale e una equalizzazione della frequenza. Ogni filtro reagirà in forma indipendente allo stimolo e assumerà pertanto una differente ampiezza, generando così un particolare **timbro.**

2. L'elaborazione elettronica e *Improvviso statico* per sassofono contralto e live electronics

Improvviso statico per sassofono contralto e live electronics, una composizione del 2012, risponde in modo particolare alle riflessioni fondamentali precedentemente esposte, alle linee di pensiero che sottendono al processo compositivo. La modalità di evoluzione del processo compositivo costituisce pertanto l'argomento principale della nostra esposizione: l'indagine, la ricerca a proposito di tale modalità è strettamente collegata ad una analisi di carattere compositivo fatta sulla partitura stessa. Dalla partitura cercheremo di fare emergere, mettendoli in evidenza, gli aspetti linguistici e formali che caratterizzano la scrittura compositiva di Carmine Cella.
Improvviso statico è un brano ordinato da Emufest 2012, il Festival di musica elettronica del Conservatorio S. Cecilia di Roma. Il brano - la prima esecuzione è avvenuta il 13 ottobre 2012 – è dedicato a Enzo Filippetti e a Giorgio Nottoli, quest'ultimo curatore artistico del Festival stesso.
Carmine Cella ha realizzato la parte del live electronics sul processore personale Mac OSX 10.8.2, corredato della versione 6.1 di MaxMsp. La data che segna la conclusione della partitura è indicata in calce all'ultima pagina, settembre/ ottobre 2012, Strasburgo e Pesaro. La nostra analisi è fatta sulla partitura che porta nella copertina il "copyright" del 2012, non ancora pubblicata a stampa.
Improvviso statico offre immediatamente, ad un ascolto e ad una lettura analitica molto attenti, la chiara percezione che i due aspetti del comporre, quello razionalista, evidente nelle opere della prima maniera di Cella e quello collegato alla ricerca sul timbro, una ricerca costantemente perseguita nelle opere della maturità, si completino e trovino la loro fusione in questo lavoro. La modalità della gestione intervallare delle altezze, rimane ancora oggi l'elemento strutturale
fondamentale della scrittura compositiva; una modalità che non appare rigida, bensì estremamente musicale, a tratti anche 'melodica'.In *Improvviso statico* sono ancora le altezze gli elementi basilari che strutturano la forma musicale, il processo musicale; vi appare però una differenza sostanziale con certi stilemi linguistici che si ritrovano nelle opere giovanili. L'aspetto intervallare, infatti, viene esaltato - talvolta messo anche in contrapposizione - dai numerosi elementi di ricerca timbrica: le piccole note scritte come 'acciaccature', i segni di dinamica ravvicinati, la presenza fondamentale del live electronics, le cui "*patches"* sono elaborate con il programma MaxMsp in maniera magistrale. La compresenza dei diversi elementi produce nel parametro intervallare una singolare mutazione: la connotazione fondamentale di frequenza cede ad una rappresentazione timbrica che definisce così l'altezza del suono come parametro del timbro; ne deriva una sorta di "altezza-timbro".
Sembra essere questa la ricerca fondamentale perseguita dal compositore per definire il carattere della sua scrittura musicale. L'affermazione è sorretta sia dallo studio attento con cui è stato esplorato lo sviluppo del pensiero compositivo di Carmine Cella secondo le linee indicate da alcune opere, in particolare dalla sua produzione teorica, sia dal lavoro sostanziale di elaborazione analitica, testuale, svolto sulla partitura di *Improvviso statico.*

2.1 Le frequenze

L'indagine analitica parte dalle prime due sezioni A e B, esemplificative per gli aspetti formali e linguistici; assieme alle sezioni C, D, E compongono la struttura totale del brano.

In una seconda fase l'analisi prenderà in esame il processo di elaborazione elettronica , la "*concert-patch"*, contestualmente al rapporto di interazione dello strumento acustico, il sassofono, con il live electronics. Si evidenziano sull'istante alcuni elementi di sostanziale importanza. Infatti il carattere delle frequenze, indicato dalla persistenza di intervalli morbidi per scelta sonora - la terza maggiore e minore, il loro rivolto di sesta - e quello delle durate, del tempo indicato da una coesistenza di suoni lunghi, di suoni contati in secondi oppure in unità metriche definite dal metronomo, non sempre danno luogo ad uno sviluppo del processo compositivo, ancorché ad una logica di causa-effetto nel passaggio da una sezione all'altra; appaiono invece come degli elementi di scrittura abbastanza statici, pure se fondamentali per la costruzione della composizione.
Il carattere di staticità intervallare e temporale è presente già nel **Grave** iniziale della sezione A. Sotto l'aspetto strutturale e sonoro essa mostra infatti la sua fondamentale realtà di "forma-timbro", una realtà che va attribuita certamente al suono innovativo per il timbro del sassofono, ma ancora più al processo di elaborazione del live electronics a cui sono sottoposte le nove micro-strutture della sezione stessa.
Allo stesso modo si evidenzia la scarsa dinamicità del parametro della frequenza o della forma nella sezione B, dedicata allo strumento solista. E' chiara la persistenza su due singole frequenze, il cui colore e timbro mutano invece in maniera quasi prismatica a causa di micro - variazioni di toni, di dinamiche differenziate, di attacchi del suono in ribattuto o in tremolo. Contestualmente il timbro e l'elaborazione timbrica delle frequenze strutturano brevi elementi fraseologici, formali. Come non notare infatti la cadenza di frase generata dall'intervallo di terza minore con le due corone, cui conduce l'iniziale processo di tensione timbrica? Oppure la sorprendente lunga linea melodica che struttura formalmente la seconda parte della sezione e che porterà, attraverso un accelerato convulso segnato dai molti effetti di *slap,* di *subtone* ,di *ampio vibrato* del sassofono alla improvvisa cadenza finale? La scrittura compositiva mostra di essere sostanzialmente una scrittura di tradizione; oppone però il carattere di scrittura innovativa per la ricerca che il compositore svolge nell'ambito di una semantica musicale essenzialmente timbrica. Fondamentale è il ruolo di protagonista che il sassofono è chiamato a svolgere sotto questo aspetto.
A questo punto della nostra analisi conviene osservare come il compositore ricerchi anche l'unità formale, nel passaggio da una sezione all'altra, attraverso l'utilizzazione di elementi essenzialmente strumentali, come fossero dei dati di sutura timbrica e dinamica. La sezione C mutua, pertanto, dalla precedente sezione un elemento di scrittura tipico di molti strumenti musicali, in particolare degli strumenti a fiato: il ribattuto. Sempre in *fff,* prima con un *flatterzunge*, quindi con alcuni trilli sulla medesima nota. L'esplorazione timbrica valuta in maniera intelligente sia le modalità di emissione sonora in *subtone* quanto l'opposizione delle dinamiche, oppure la novità timbrica offerta dall'esecuzione di un doppio trillo. Esso appare per la prima volta nel corso della partitura, non innovativo dal punto di vista grafico, ma evocativo di un momento cadenzale per il gesto musicale di elemento concluso, come di una scatola chiusa. Possiamo mettere in evidenza come anche nella sezione D l'elaborazione timbrica sia presente sul singolo suono, formato però dal bicordo *mi naturale - la*. Caratterizzato da attacchi diversi del suono, da dinamiche variate, esso si trova all'interno di una serie di più o meno brevi scale di acciaccature a toni interi: la percezione dell'ascolto segnala la presenza di una collocazione spaziale di natura timbrica. Il processo di variazione continua coinvolge anche l'ultima sezione E, che re-introduce il movimento del ***Grave*** dell'inizio della partitura. La struttura generale riconquista in questa maniera la sua valenza storica, e risolve l'unità formale nell'unità della forma musicale circolare.
Benché l'ultima sezione sia la più estesa delle cinque sezioni per le dimensioni formali, non vi si ritrovano degli aspetti linguistici che non siano stati già indagati in precedenza. Pur tuttavia è possibile mettere in evidenza ancora due elementi di un certo interesse per un'operazione di modalità analitica: la presenza, nelle ultime due battute, di una breve parte denotata da una scrittura libera e dal live electronics polifonico (vi appare "lux aeterna" di Ligeti) e la frase melodica data al sassofono, ripetuta *ad libitum*, in cui la linea di canto è formata da una figurazione intervallare di sole altezze. E' una memoria della scrittura musicale segnata prevalentemente dalla gestione delle frequenze?

2. 2 Il tempo, il ritmo

Il parametro del tempo, pure collegato alla struttura delle singole sezioni, aderisce al criterio di una sufficiente dinamicità e libertà del divenire dei diversi istanti.
Tuttavia la relazione del tempo con delle strutture formali di limitata estensione potrebbe indurre a segnalare, in un momento di analisi, una certa uniformità anche del divenire temporale. L'osservazione critica pone in evidenza, però, come anche il contesto delle relazioni di certi parametri con altri parametri nel processo musicale – soprattutto se si tratta del parametro strutturale- rientra alla fine nell'idea musicale fondamentale della scrittura di Carmine Cella: il limitato interesse nei riguardi di una idea globale dello sviluppo, a favore di "variabili" di limitata entità che, proprio come dati parziali, istanti del tempo, possono assumere il valore di un maggiore interesse musicale.
Anche certe soluzioni ritmiche, talvolta ripetitive, ricorrenti, sembrano accentuare il carattere di staticità del processo compositivo. Tuttavia l'uniformità dei parametri sonori, quando è ricercata soprattutto come un elemento ideale della scrittura musicale, non può dirsi equivalente ad una dichiarazione di povertà del pensiero musicale; può definirsi invece come un ulteriore parametro sonoro, originale per i tanti fattori di evoluzione interiori, ed assumere il valore di una semantica musicale personale, di una poetica propria.

2. 3 Il timbro

In *Improvviso statico* il parametro sonoro del timbro presenta la maggiore dinamicità possibile e il maggiore

interesse musicale possibile, perché sorretto e amplificato dal processo della variazione continua.
E' sufficiente, per una conferma, rivolgere l'osservazione analitica alla prima pagina della partitura. Il suono *sol* con il quale lo strumento acustico apre lo spazio sonoro della composizione porta l'indicazione "*breath, non vibrato"*, ossia solo "fiato, non vibrato"; si allarga quindi in un accordo di frequenze multi-fonico, infine in un accordo multi- fonico caratterizzato da un "ampio regolare vibrato". Le mutazioni sono nella sostanza di portata minima, sufficienti però a dare vita al passaggio veloce di un colore del suono all'altro, come in una situazione di rifrazione prismatica. Il suono *sol* sale quindi alla seconda maggiore: in pochi secondi del tempo si evidenziano sul secondo suono un crescendo dal *piano* al *forte,* ancora al *piano*, quindi una figurazione ritmica a velocità crescente di suoni armonici, variati dal brevissimo *slap* del sassofono che chiude la sezione A. L'articolazione timbrica strumentale è in relazione contestuale con l'articolazione timbrica del *live*. La dimensione generale del timbro, che si accompagna a quella di una dinamica segnata scrupolosamente nei particolari, viene così esaltata.

3. Concert- Patches *Improvviso statico*

L'interfaccia della 'Concert-patch' di *Improvviso statico* presenta quattro colonne: vi sono indicati gli elementi di esecuzione del live electronics.
Nella prima colonna sono presentati: **Checklist, BCF 2000 router, Init e Reset, Audio, Score**.
Nella seconda colonna sono presentati i **Processors** (gli effetti): **Harmonizer, Players, XSynth, Combsyn**, **Multidelay, Reverb**.
Nella terza colonna sono presentati i comandi del **Tempo** e della **Console.**
Nella quarta colonna è presentata la **I/O Matrix**, il Reset, altri comandi di cancellazione e di Reset della Matrix.
La **Checklist** (è possibile vederla sia in *presentation mode* che in *patcher mode*) rappresenta il controllo della lista; è necessaria per verificare quali sono gli elementi dell'esecuzione. Pertanto va seguita negli elementi particolari per potere eseguire correttamente la *Patch.* In particolare: *I brevi incisi* e *I buoni avvisi.*
I buoni avvisi: quando si controlla lo stato DSP bisogna verificare che il segnale Vector Size sia abbastanza largo (512). Bisogna verificare che l'interruttore Audio sia fuori; che Wi-Fi e Bluetooth siano ambedue fuori; che l'hard disk non sia mai in modalità spenta.
Per fare una nuova scena è necessario: premere Reset nella *patch* principale; configurare i Processori, la Matrix, i microfoni; premere il "Current status button" sulla Scena Manager; scrivere la *subpatch* della partitura e aggiungere il nuovo messaggio; dare la lista.
La BCF 2000 Router presenta la grande *patch* della Master control; quindi la *subpatch* degli effetti. Entriamo nelle due *subpatches* del controllo degli effetti: nella *subpatch* Multidelay è il controllo dei ritardi, dei filtri, dei *feedbacks,* dei modulatori, del *chorus*, dell'alimentazione; nella *subpatch* del Combsyn control è il controllo dei diversi parametri. Non è fondamentale per uno studio analitico della Patch-concert. Nella interfaccia vi sono ancora le singole *patches* di Inizio e Reset. Nella *subpatch* di Reset sono i Processors: 1.2.3. Players, XSynth, Combsyn, Multidelay, Harmonizer, Reverb. Dopo le Keys e l'Audio c'è la *Score:* vi si entra in *patcher mode* e si vede la partitura elettronica del brano. La *Score* è l'elemento centrale della elaborazione elettronica, sia per la ricchezza dei segnali elettronici messi in gioco, sia per la struttura compositiva che vi sottende. All'inizio della *Score* il compositore commenta espressamente che questa è la partitura elettronica della composizione.
Ora analizziamo brevemente la Sezione A; la riteniamo infatti un modello di base, sotto l'aspetto compositivo, per un processo interessante di elaborazione dei modelli elettronici. Entriamo nella *subpatch* e vediamo raffigurate in sintesi le elaborazioni elettroniche delle nove *micro-patches*; le elaborazioni elettroniche sono indicate anche nella partitura cartacea. Per esempio, se consideriamo la scena A e apriamo la *sub patch "r A.*_0_rev" vediamo diversi aspetti di riverbero, con il livello di *input* del microfono a - 6. Dentro la sub patch "r A._ 1_ harmon" c'è Harmonizer- enable con le diverse modalità di *harmonizer:* il messaggio accende l'armonizzatore [chord 1], e un riverbero di 4 secondi. L'esempio successivo indica un processo di elaborazione dei segnali molto interessante: nella *sub patch* "r A.3_harmon_rand3" c'è 1 harmonizer - enable 1 con 1 harmonizer – random - pitch - range 3, cioè +- 3 semitoni. L'elaborazione elettronica delle singole *patches* continua fino ad A.9 , quando il Player 2, già presentato nella subpatch A.8, esce di scena.
La Sezione B è la scena dedicata al sassofono solista. Ne osserveremo le peculiarità nel capitolo successivo.

4. L'interazione del sassofono con il live electronics

L'interazione della *Concert-* P*atch* con lo strumento che suona dal vivo è molto stretta: in effetti una grande parte della *Patch* è una elaborazione del sassofono. Senza l'esecutore non si potrebbe ottenere un grande risultato. Vi sono tuttavia alcune scene - lo afferma il compositore stesso – in cui l'indicazione dei *Player*s rende non indispensabile la presenza del sassofono.[1]
La ricerca analitica mette in evidenza come i due soggetti fondamentali della struttura compositiva siano in un rapporto di reciproca relazione : la *Patch* con il sassofono e il sassofono con il sistema interattivo fornito dal programma MaxMsp. L'eccezione alla regola è data dalle sezione B, in cui il sassofono suona come solista, protagonista assoluto della parte forse più interessante sotto l'aspetto compositivo della partitura. La concezione formale proviene dalla scrittura di tradizione: quella della cadenza,del pezzo di bravura affidato allo strumento solista. La forma è davvero originale, anche perché la sezione di 'assolo' strumentale è posta quasi all'inizio della composizione, subito dopo la Sezione A.

1 Comunicazioni personali con l'autore

In questa maniera la Sezione A viene ad assumere la funzione di introduzione alla composizione.
Nella Sezione B risiedono gli elementi fondamentali della scrittura: la persistenza dei suoni singoli, variati nel timbro come in un prisma geometrico, degli intervalli di terze per lo più minori, di seconde, anche di microtoni. Insomma un sistema intervallare significativo per certi aspetti, tuttavia non eleggibile - per la carenza di un processo evolutivo nel tempo - a elemento strutturale della forma.
La struttura formale è definita invece da una semantica il cui *senso* – inteso esattamente come significato- è espresso dal timbro.

5. CONCLUSIONI

Sintesi di pensiero o concertazione fra lo strumento acustico e la tecnologia elettronica?
E' nostra convinzione che lo strumento preso dalla tecnologia elettronica non riesca a raggiungere una sintesi, sul piano del pensiero compositivo, con lo strumento acustico. Il pensiero è l'*inventio,* l'idea prima: non ha delle regole precostituite e può vivere anche del suo *status* di pensiero. Soltanto a posteriori l'idea prima si trasforma in un codice, in un linguaggio reale, quale che sia. Il "manufatto" tecnologico deriva invece da una programmazione precisa, da un progetto che ha delle regole stabilite, tanto è vero che ogni errore di esecuzione del progetto stesso viene notificato. Per di più esso si codifica non in un linguaggio, ma in una pluralità di linguaggi (analogico, digitale), sempre linguaggi della macchina. A nostro avviso il compositore fa ricorso, nella ipotesi di superare la dualità fra l'idea compositiva espressa da uno strumento acustico e il linguaggio di uno strumento elettronico, al processo della combinazione, utilizzando il parametro più astratto del suono, il timbro. Il timbro è infatti il parametro meno fisico, meno obiettivo del suono, certamente il più evocativo. La chiave di rappresentazione sta nei termini della combinazione. Ma la combinazione non è un fattore di sintesi né sul piano del suono né sul piano del pensiero. Esiste una soluzione a questo problema? Intanto non lo si consideri un dilemma, ma si considerino le due situazioni, della sintesi e della combinazione, come delle realtà di idee diverse, il cui riscontro è sempre avvertibile nel momento in cui vengono a fronte. Una soluzione è possibile a nostro avviso, e consiste nel sollevare la situazioni duale in uno stato metafisico, meta-reale: in questa zona franca della nostra mente, nel mondo delle idee, può avvenire la soluzione della sintesi.

6. BIBLIOGRAFIA

[1] Carmine E. Cella: "Sulla struttura logica della musica" *Rivista umbra di musicologia* n. 48, Perugia 2005/1, pp. 3 - 57

[2] Carmine E. Cella: "Logical Foundation of Music," *Séminaire MaMuX Ircam,* Salle O. Messiaen, 14 Gennaio 2006

[3] Carmine E. Cella: *Modellazione digitale del pianoforte* Teaching resources, 2010, slides

[4] Carmine E. Cella: *Sul tempo musicale e sulla sua percezione* Teaching resources, 2011

[5] Carmine E. Cella: *Physical-inspired synthesis a tutorial* Roma, 22 Novembre 2012

THE HYPERSAMPLER BASED ON FEATURES EXTRACTION; A BRIDGE BETWEEN THE PLAYER AND INSTRUMENT PARADIGMS

Marco Marinoni
Conservatorio "G. Verdi" di Como
marco.marinoni@virgilio.it

ABSTRACT

This paper concerns the hypersampler implemented for my piece *Il grifo nelle perle nere* for piano and hypersampler, composed in 2008 and premiered during the 54th Venice Biennale's International Festival of Contemporary Music, in the framework of XVII CIM – Colloquium on Music Informatics.

The hypersampler involves a real-time synthesis engine based on processes of feature extraction as an alternative to hyperinstruments' physical control paradigm. Features are derived from the performance of a traditional musician on an acoustic instrument – a piano – and are used as a control for the mapping between the instantaneous power spectrum of the acoustic instrument's sound output (the musical dynamics performed by the pianist) and real-time synthesis engine's parameters.

1. INTRODUCTION

The definition of "hyperinstrument" used here is consistent with that proposed by Machover [6], which I then proposed a possible extension [10]. According to Tod Machover, the basic concept of a hyperinstrument is to "take musical performance data in some form, to process it through a series of computer programs, and to generate a musical result" [6]. The hyperinstrument, in its most simple meaning, as it has been conceived for the first time in 1987 for the work *Valis* [9], is based on musical instruments able to provide a great variety of solutions that musicians play on the computer. The simplest method is through an instrument similar to an existing conventional one, such as a keyboard or a percussion. The hypersampler developed for *Il grifo nelle perle nere* [11] implements a keyboard instrument that be-comes hypertext of another keyboard instrument, the piano.

The parameter that really interested for long time Machover's research is rhythm. In a live performance, this can mean the musicians are required a greater precision than is normally demanded; this may also involve a higher degree of rhythmic complexity, and the creation of delicate relations of synchronicity that would be difficult to play without the aid of computer. However, a theoretical approach to hyperinstruments and their development cannot help but account the dimension of the live performance. Even in its interactions with technology, music is an art that is based on performance and interpretation, so "the 'brain' of a hyperinstrument is the computer system that monitors musical data from the input instrument, redefines the controls on that instrument, and acts in accordance with its programmed musical knowledge" [6]. In this sense, in the piece that underpins this paper, the gestural expressiveness related to pianistic musical dynamics triggers a sonification of the interpretative data. This occurs through a delicate process of feature extraction aimed to the construction of a virtual instrument that is informed in real time by a traditional instrument. The virtual instrument retains its own identity and all the features of a musical instrument in its own right, including the permeability to interpretative data.

Feature extraction is intended by Bullock [2] as "a form of data processing that takes a set of values and returns a more compact representation of those values. The compact representation is called a feature, and the initial set of values could be referred to as the input vector. The process of feature extraction is a form of dimension reduction, because it involves the mapping of an input vector of dimension N onto an output scalar or vector that has dimensionality that is smaller than N". As Bullock stated, a hyperinstrument system based on feature extraction can "minimise the number of prosthetic elements, and provide a seamless sense of interaction for the performer where sound becomes both the source of control and the means of gaining auditory feedback. Using sound as a medium for interaction removes the requirement for sensors, switches and other physical controllers in order to convey gestural information and performer intention".

This approach is consistent with that proposed by Machover in pieces like *Sparkler* (2001) [8], where Machover, Jehan and Fabio developed a hyperinstrument system (an acoustic instrument-plus-laptop combination) aimed at expanding the expressive power of traditional instruments and performers by placing mi-

Copyright: © 2014 Marco Marinoni. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

crophones within the orchestra to capture all the instruments, which was then analyzed with a laptop and processed "to shape and manipulate a complex electronic 'aura' that was added live to the orchestral sound" [5].

From an interpretive and perceptual point of view, the hypersampler can be regarded as a minimal (two units) Interconnected Musical Network (IMN) intended according to the definition proposed by Weinberg [13], who defines IMNs as "live performance systems that allow players to influence, share and shape each other's music in real time". Weinberg also states that "only by constructing electronic (or mechanical) communication channels among players can participants take an active role in determining and influencing, not only their own musical output, but also their peers'. For example, consider a player who while controlling the pitch of his own instrument also continuously manipulates his peer's instrument timbre. This manipulation will probably lead the second player to modify her play gestures in accordance with the new timbre that she received from her peer". Both the sensor-based (multimodal) and the feature-extraction-sound based approaches are aimed at developing an interactive network (the hybrid double instrument called the hyperinstrument system) able to combine gestural characteristics of musical interpretation and real-time sound processing into a "constantly evolving collaborative musical product".

In 1992 Rowe [12] proposed two distinct models of interaction in live electronic music: systems based on *player par adigm*, which provide a musical presence with a personality and a behavior of its own and systems based on *instrument par adigm*, which extend and augment the human performance through direct response to input which is generated by the performer via sound or physical control. One possible way of overcoming the limitation of these two paradigms is represented by the sensor-based approach: it stands on the ground of most of the hyperinstruments. The sensor-based approach investigates the correlation between musical and physical gesture and sonic output through the use of sensors that can be attached to the acoustic instruments and/or performers; their outputs are then scaled and routed into live controlled sound processing algorithms. In the sensor-based hyperinstrument systems, in which "a sensor converts physical energy into electricity in the machine, and may therefore be called the 'sense organ' of a system" [4], physical gesture is closely coupled with the audio output but "availability of existing gestural controllers is limited and new controllers can be expensive or time-consuming to develop" [2]. In 2001 Jehan proposed a system developed in Max/MSP[1] which combines audio feature extraction, timbral mapping and synthesis in the context of live electronics performance whereby "continuous changes in articulation and musical phrasing" lead to "highly responsive sound output" [3]. The system developed by Jehan included real time mapping of extracted sound features and sonification of rescaled data in order to get completely new material generated by the performance on traditional instruments.

The research at the basis of the hypersampler started from the purpose of developing a hyperinstrument system intended as an IMN able to overcome the limitations of Rowe's player-instrument paradigm: a hybrid instrument not including the sensor-based approach, developed following the *instrument-player c ontinuum model* proposed by Bullock [2] in 2008 (see Figure 1) that extends Rowe's player paradigm and instrument paradigm and takes account of Jehan's approach to real-time synthesis engines based on the extraction of perceptual features.

The hypersampler includes a piano, a master keyboard (e.g. EDIROL PCR-M1) or MIDI controller (e.g. KORG NANOKONTROL 2) the computer and the technical equipment needed for the implementation of live-electronics. The software environment has been entirely developed in Max/MSP.

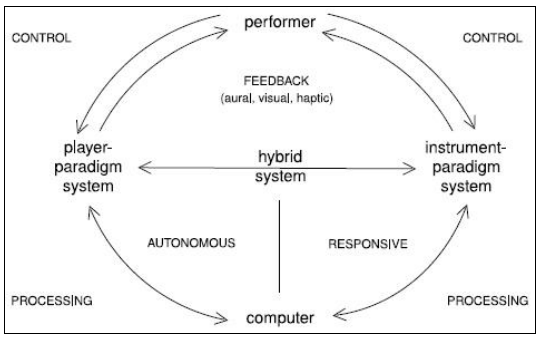


Figure 1: Bullock's instrument-player continuum model.

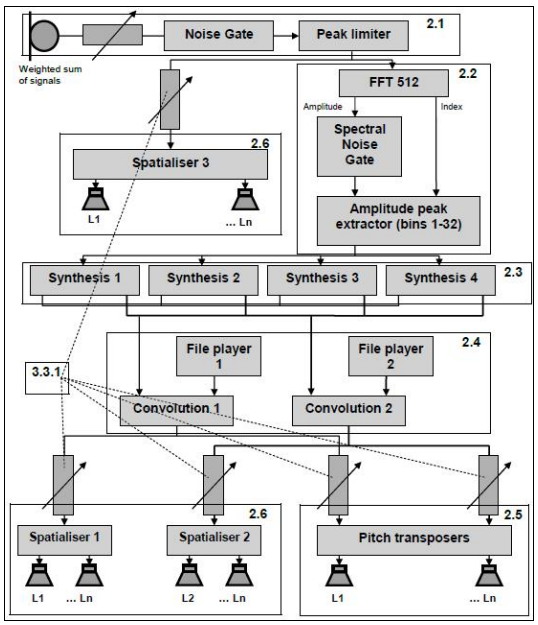


Figure 2: General overview of the hypersampler environment.

[1] http://cycling74.com/products/max/

The instrumental signal is captured using three microphones and then mixed according to different percentages in one monophonic signal. The monophonic signal is routed to the units of processing (see Figure 2), which include: first stage of treatment by noise gate and peak limiter aimed at reducing the dynamic range and making the sound materials more easily treatable; second stage of treatment by processes of feature extraction (analytic level) and transformation of the data so obtained; third stage of treatment by synthesis (synthetic level), informed by the data coming from the previous stage; fourth stage of treatment by real-time convolution (two .wav files are read in real time by buffers; the materials included are recorded sounds of violin, viola, cello and double bass performing held notes produced by various traditional and not traditional performance modalities, such as natural harmonics close to the bridge, "spazzolato", vibrato with bow on the fingerboard, "grattato", etc., covering all frequency ranges and dynamics) of output materials from the synthesis modules, which is a completely arbitrary choice aimed to confer an instrumental identity (the spectral image of a string ensemble in this case) to the hyperinstrument; finally, fifth stage of treatment by pitch transposers and sound projection by spatializing matrices controlled in real time by the hypersampler performer. The instrumental signal processed by first stage of treatment is projected (transparent amplification) too by matrices controlled in real time. For the aims of this paper we will mainly focus on second and third stages of treatment, which form the hypersampler's engine.

It is necessary in this regard to separate what technically defines and implements the theoretical conception of the hypersampler (see paragraphs 2.1 and 2.2), moving from the definitions taken as a starting point and switching to the objectives set out above (see Chapter 1), as opposed to what concerns its implementation in the musical piece, which is not discussed in detail here. We must consider, however, that some parameters and compositional choices cannot be separated from what pertains to the theoretical system, as a piece of music is a unity of design, implementation, and technical realization.

The choices about the envelopes, the spectral content of the files relating to the convolution, the particular type of jitter of frequency and amplitude realized through the modules Clock, the choice to use frequency transposition in the domain of time (here called "detune" and "frequency shifting"), the parametric correlation curves and the spatialization system form part of that required implementation of the idea that underlies the sampler, in order to obtain a musical result that realizes this idea, according to Machover's statement that "the final component of the hyperinstrument is the musical result" [6].

Il grifo nelle perle nere was written in 2008 for the "Concerto per Ipertastiere" included in XVII CIM – Colloquium on Music Informatics. The first performance took place in Venice at the Concert Hall of Palazzo Pisani on Wednesday October 16th 2008 H 5pm, during the 54th Venice Biennale's International Festival of Contemporary Music, with the following performers: Davide Tiso, piano; Marco Marinoni, hypersampler; Alvise Vidolin, sound direction.

2. HYPERSAMPLER'S ENGINE

In this section the typologies of sound processing are described, specifying the data and the variables essential for the realization of the hyperinstrument system. For each processing, the values of the parameters and their significance within the performance are indicated, identifying the ones intended to be controlled real-time by the live-electronics performer.

Jehan's assumption that "the timbre of a musical signal is characterized by the instantaneous power spectrum of its sound output" [3] and Machover's statement according to which "the 'brain' of a hyperinstrument is the computer system that monitors musical data from the input instrument, redefines the controls on that instrument, and acts in accordance with its programmed musical knowledge. These programs range from the highly deterministic (one physical event is tied to a particular musical result), to the computationally flexible (musical result chosen by particular context, and modified according to the specific musical gesture received), to the truly intelligent (analysis by rule of performed music and the machine-choice of an appropriate response)" [6] represented the starting point for the development of the hypersampler's synthesis engine.

The typology of sound tracing developed for *Il grifo nelle perle nere* integrates the approach of Jehan with that of Jensenius, which identifies three types of sound tracing: "focusing on sound-production, timbral features or temporal development" [4]. The feature extraction process implemented here uses the third type of sound tracing.

2.1 Spectral noise gate – Amplitude bin extractor

The output signal from peak limiter unit is analyzed using a length N FFT of $\sim x_m$ to obtain the STFT at time m:

$$\tilde{x}'_m(e^{j\omega_k}) = \sum_{n=-N/2}^{N/2-1} \tilde{x}'_m(n) e^{-j\omega_k nT} \qquad (1)$$

where $\omega_k = 2\pi k f_s/N$, and $f_s = 1/T$ is the sampling rate in Hz. The STFT *bin number* is k. $N = 512$. Then each FFT bin $\sim x'_m(e^{j\omega k})$ was converted from rectangular to polar form to get bin k's instantaneous amplitude.

$$A_k(m) \triangleq \left|\tilde{x}'_m(e^{j\omega_k})\right| \qquad (2)$$

As shown in Figure 3, only the first 32 bins are used and in particular only the amplitudes of bins that exceed a threshold, in order to cut the residual nondeterministic components of the sound in addition to the deterministic harmonic components. The signal so obtained is then filtered using a second order low-pass filter so as to obtain a low-frequency control signal. That signal is finally 'converted' in Hertz multiplying it by an appropriate conversion factor and sent to the peak extractor unit which identifies the maximum value sent out to the synthesis units by means of the trigger command T-R which is controlled in real time.

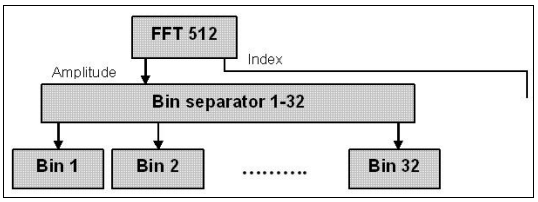


Figure 3: FFT analysis.

In Figure 4 the operations concerning the extraction of the parameter amplitude in one bin and its translation to a frequency scale are described.

The value of the parameter "threshold", that is the minimum amplitude value of single bins sent to the low-pass filter, must be so as to neatly cut the ground noise without compromising or altering the spectromorphological peculiarities of the analyzed signal.

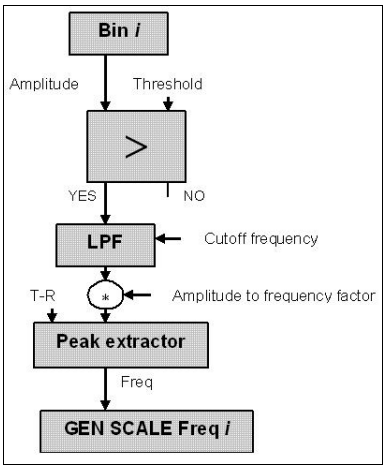


Figure 4: Translation of bin *k*'s instantaneous amplitude to a frequency scale.

The value of the parameter Cutoff frequency of the low-pass filter, is approximately set to 0.4 Hz.

The value of the parameter Amplitude to frequency factor, that is the conversion factor, must be determined in a way that the maximum output values do not exceed the number 4000 and the minimum values never lower the value 20. The Grain Generation Scale (GEN SCALE) is composed of the 32 frequency values so obtained.

2.2 Synthesis

The synthesis engine includes four clock-controlled Grain Generator Units, as shown in Figure 5. In Figure 6 the implementation of a Grain Generator is described.

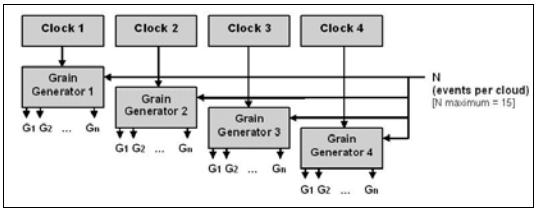


Figure 5: Synthesis module.

The four Synthesis units require four different waveforms (W). In the case of implementation using Max/MSP it is suggested to use the object *gen* (linear b.p.f. wavetable generator) included in *PeRColate – A collection of synthesis, signal processing, and video objects* by Dan Trueman (Princeton University) and R. Luke DuBois (Columbia University)[2] ported from real-time cmix, by Brad Garton and Dave Topper.

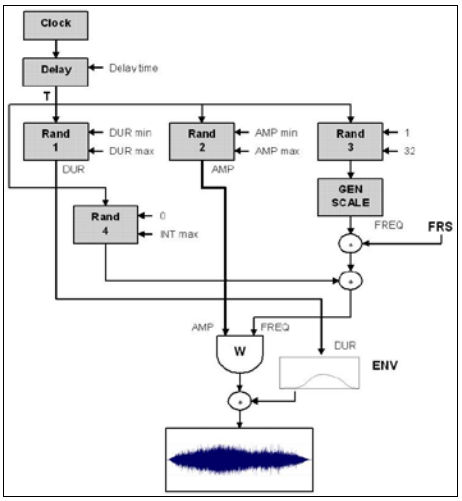


Figure 6: Synthesis engine: Grain Generator unit.

Three different typologies of envelope (ENV) applicable in a mutually exclusive way to the grains generated by the four Synthesis units are required.

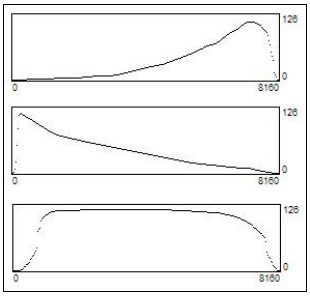


Figure 7: Envelopes which apply to the grains.

[2] http://music.columbia.edu/percolate/

The selection of an envelope is controlled in real time during the performance, with interpolation time equal to 6 seconds. The three types of envelope are shown in Figure 7.

The module RAND 1 controls the parameter Grain Duration, by generation of random floating-point numbers comprised between the minimum value DUR min and the maximum value DUR max. The module RAND 2 controls the parameter Grain Amplitude, by generation of random floating-point numbers comprised between AMP min and AMP max. The module RAND 3 generates random integers between 1 and 32, determining the Grain Generation Frequency among the 32 possible frequencies generated by the module GEN SCALE which form the Grain Generation Scale. The Grain Generation Scale must be changed many times during the performance using the command T-R (see Figure 4). The performer decides, according to his interpretation, how many times the scale is changed during the performance and when, according to the musical score. The module RAND 4 controls the variance of the parameter Transposition Interval expressed in semitones and cent, which causes random variation of the Grain Frequency around the original value, comprised between 0 and the value INT max = 1 semitone, 27 cent. The parameter Frequency Range Shifting (FRS) controls the transposition interval *n* (in Hertz) applied to the grains so that the grains' frequency is modified as shown by the formula:

$$Freqfin = (Freqinit)*n \qquad (3)$$

The value of parameter *T* (delay time) of the Delay unit is comprised between 0 and 12700 ms, and is controlled in real time, as well as *n*. The module Clock (see Figure 8) implements the following parameters.

CT1 = clock time [ms]
Vmin1 = minimum randomly generated number [int]
Vmax1 = maximum randomly generated number [int]
IT1 = interpolation time [ms]
CT2 = clock time [ms]
Vmin2 = minimum randomly generated number [int]
Vmax2 = maximum randomly generated number [int]
IT2 = interpolation time [ms]
CT3 = clock time [ms]

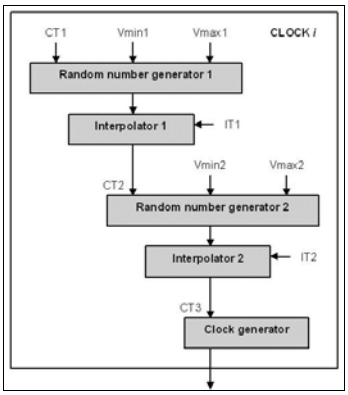


Figure 8: The module Clock.

3. DISCUSSION

In *Il grifo nelle perle nere* a virtual instrument is coupled with a traditional instrument, giving rise to a hybrid between mechanical and computer, using data extracted from the musical interpretation of the pianist to control an independent virtual system, which meets the requirements of a hyperinstrument and realizes the statement of Machover according to which the goal of a hyperinstrument would be "to produce music of unprecedented subtlety, complexity, richness, and expressive power that is intimately, but not obviously, linked to the original intent of the performer/composer" [6].

Machover's approach towards "double" and "triple instruments"[3], in which two or more people are playing a single hyperinstrument, is not unlike the one that underpins *Il grifo nelle perle nere*, where a "double instrument", the hypersampler, is controlled, at different levels, by the pianist and by the keyboard performer: the first, by playing the instrumental dynamics, affects a number of parameters including the choice of the pitch scale from which the synthetic sounds are generated by the computer; these sounds, processed by convolution, are controlled in real time by the second, which in turn can change the number of sounds produced at the unit of time, their density, their positioning within the virtual space, providing the first performer a new musical material on which interact, in a continuous and fertile creative feedback mechanism, since double instrument performers "must relate their musical gestures not only to the resulting sound as in traditional instruments, but also to the gesture and sound of the other performer" [6].

Consistently with Machover's assertions about the importance of the conceptual simplicity of the interface, this system is easily understood by the performer, who has a chance to become aware about the specific relationship of causality (semi-deterministic and bound to the interaction with the live electronics performer) that binds his actions to the production of the sound output by the system and, through a period of practice, refine his performance. In this way, the system is partially controllable by the instrumental performer, which can achieve a level of control over the music that is even greater than it has in general. The computer does not play a part isolated. The performers have the opportunity to check the results and to take on more roles from a musical point of view, depending on the particular di-

[3] A *double instrument* is conceived for two musicians that play together on separate physical controllers (one of those can be an acoustic instrument) to breed a hybrid instrument "so that each musician can influence certain aspects of the music, but both players are required to perform in ensemble to create the entire musical result" [6]. For example, in Machover's *Towards the Center* [7] the keyboard player controls the overall sound spectrum –the partials, the harmonic series, the spectromorphologic qualities of sound – while the percussionist controls the behavior of each partial, like a microscope where one observer acts on a smaller portion (controls more extended parts) while another observer acts on a greater portion (controls smaller parts, internal to the parts controlled by the other observer).

rection they decide to give the performance from time to time.

The relationship control / independence between the performers is mediated by the hypersampler, which assumes the role of *double instrument* (a *hybrid instrument* that is partly physical instrument and partly virtual) formed by two performers that work together to control a complex instrument, each of which controlling only part of the final result. This instrument also includes the "partial and expected *unpredictability*" [10] which was mentioned earlier as a distinctive feature of each instrumental practice, traditional or contemporary.

The feature extraction process which is implemented on the traditional instrument does not imply structural change: this increases the level of reproducibility, not bound to the context or the availability of specific technologies, however, placing a question of theoretical order: is the difference between what we call generally live-electronics and what we call a hyperinstrument linked to the use of technologies such as sensors etc., as Machover and the MIT researchers seem to say, or is it a difference of higher order (multiple instruments, different in nature, acoustic and electronic, with specific performers who play performances interconnected, according to Weinberg's theory, which form a single hybrid instrument, equipped with its own identity, qualitatively different from the sum of the identities of the individual instruments involved: the hyperinstrument system) and the hyperinstruments are but a subset of the broader category of live-electronics?

4. CONCLUSIONS AND FUTURE WORK

In the light of findings from the experience with *Il grifo nelle perle nere*, and assuming with Benzon that music is "a medium though which individual brains are coupled together in shared activity" [1], the hypersampler can be truly considered as a basic Interconnected Musical Network (composed of two units), that is a "live performance system that allow players to influence, share, and shape each other's music in real-time" [13], being also a feature extraction driven *double instrument* in which an acoustic instrument is "complemented by delicate electronics played and transformed by a keyboard-with-laptop [...] creating shifting textures that 'fuse' the various instrumental lines" [5].

From the musician's perspective, the hypersasmpler behaves intuitively and predictably. The control features of the traditional instrument are used "as input for the model of a different instrument" [3] which is perceptually meaningful. Choices related to the technical implementation appear to be adequate and able to provide the hyperinstrument with a sufficient degree of freedom and individuality, to ensure the possibility of a true musical interpretation.

Future work will include algorithms that extract more control features and the extension of this approach to different instruments such as flute and violin, in order to be able to advocate a generalization of results, with regard to the connection between mapping of input signal and commands to control parameters to audio output.

5. REFERENCES

[1] W. L. Benzon: *Beethoven's Anvil: Music in Mind and Culture*, Basic Books, New York, 2001.

[2] J. Bullock: "Implementing audio feature extraction in live electronic music", Ph.D. dissertation, Birmingham Conservatoire, Birmingham City University, 2008.

[3] T. Jehan: "Perceptual synthesis engine: An audio-driven timbre generator", M.S. thesis, School of Architecture and Planning, Massachusetts Institute of Technology, Cambridge, 2001.

[4] A. R. Jensenius: "Action — sound. Developing methods and tools to study music-related body movement", Ph.D. dissertation, Department of Musicology, University of Oslo, Oslo, 2007.

[5] T. Machover: "Dreaming a New Music", *Chamber Music*, Vol. 23 No. 5, pp. 46–54, 2006.

[6] T. Machover: *Hyperinstruments: A Progress Report*, Technical report, MIT Media Laboratory, Cambridge, 1992.

[7] T. Machover: *Towards the center*, Score, Editions Ricordi, Paris, 1989.

[8] T. Machover: *Sparkler*, Score, Boosey and Hawkes, New York, 2001.

[9] T. Machover: *Valis: an opera in two parts*, Score, Editions Ricordi, Paris, 1987.

[10] M. Marinoni: "Comporre per gli iperstrumenti. Il sistema-iperstrumento come agente intermediatore tra l'esecutore strumentale e i processi di trasformazione/generazione del suono", Ph.D. dissertation, Conservatorio di Musica "B. Marcello", Venezia, 2007.

[11] M. Marinoni: *Il grifo nelle perle nere*, Score, ArsPublica Edizioni Musicali, Camino al Tagliamento, 2009.

[12] R. Rowe: *Interactive music systems: machine listening and composing*, MIT Press, Cambridge, 1992.

[13] G. Weinberg: "Interconnected Musical Networks – Bringing Expression and Thoughtfulness to a Collaborative Group Playing", Ph.D. dissertation, School of Architecture and Planning, Massachusetts Institute of Technology, Cambridge, 2003.

THE HYPERVIOLIN IN *DALLA SUA ORBITA*

Marco Marinoni
Conservatorio "G. Verdi" – Como
marco.marinoni@virgilio.it

ABSTRACT

Dalla Sua Orbita is a piece for hyperviolin written in 2007 and revised in 2011 in which the hyperinstrument system produces modifications of sound and structure starting from incoming data. The musical data is encoded through processes of feature extraction and preprocessed by a limiter, then routed to algorithms for sound transformation/generation, which produce a musical result which varies, according to process intensity, from morphologically independent sound material (e.g. sound materials generated by the granulators [Patch 5] and pitch transposers [Patch 3]) to a reinforcement of the starting instrumental materials (spatialisation), timbre modifications (ring modulation with distortion of negative sideband output signal [Patch 2], live-sampling subsequent reading of fragments varying rate [Patch 4]).

The instrumental performer must be aware of how the hyperinstrument system interprets his actions instant by instant, so to provide significance to his technical and interpretational choices, in connection with the performance of the live-electronics interpreter.

The mapping is directly applied to the traditional musical instrument, implemented by specific hardware designed by professor Matteo Ricchetti and realized by InfoMus- Lab of Genoa.

1. INTRODUCTION

I composed *Dalla Sua Orbita* in the spring of 2007 specially for the "Primo Concorso di Composizione per Iperviolino, where it has been awarded the First Prize. The première took place in Genoa at the Auditorium Casa Paganini on Wednesday October 17th 2007 H 9pm, during the Festival Paganiniana 2007, with the following performers: Diana Jipa, hyperviolin; Marco Marinoni, live-electronics and sound direction.

The research at the basis of *Dalla Sua Orbita* was aimed to transform some aspects of instrumental technique, such as the position, pressure and motion of the bow on the strings or the motion of the left hand along the fingerboard following positions, fingering and gestures indicated in the score, in encoding engines for the hyperinstrument system, in order to make the musician not only responsible for the performance of the instrumental part but also, through his interpretative choices, directly and significantly involved in the building of the process of transformation/generation of sound and of the spatialisation paths of sound materials through the diffusion system.

Copyright: © 2014 Marco Marinoni. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



Figure 1. Bracelet for the encoding of the parameter right wrist's tilt angle.

The right hand holds the bow. The bracelet (see Figure 1) fastened to the wrist encodes the parameter right wrist's tilt angle, detecting the tilt angle of the bow on the violin (indicated in the score as PARAM 1).

The left hand shifts along the fingerboard. A microled (see Figure 2) is placed under the scroll of the violin and encodes the parameter distance of left hand from the sensor (position of the hand on the fingerboard, indicated in the score as PARAM 2).



Figure 2. Microled for the encoding of the parameter distance of the left hand from the sensor.

The analogic data is converted in number values on scale MIDI 0÷127 and then sent to the processing units.

The instrumental signal is recorded by a super-cardioid dynamic microphone [indicated in the score as MIC 1] and a condenser cardioid microphone for close recording [indicated in the score as MIC 2]. Sound materials are sent to the processing units through a digital audio interface (MIC 1 to Patches 1, 3, 5; MIC 2 to Patches 2, 3, 4).

2. LIVE-ELECTRONICS

The processing unit implements five algorithm chains for sound processing.

Patch 1 [0'00' – 0'30''] (see Figure 3): includes eight pitch transposing modules PTR. Each unit receives audio signal from MIC 1 processed by a limiter/dynamic compressor and routes it to a specified loudspeaker, so to obtain a spatialisation of the sound materials which includes interconnected pitch and length fluctuations controlled by the performances of both the musician wearing the hyperviolin and the live electronics performer. The features of a PTR unit are shown in Figure 4.

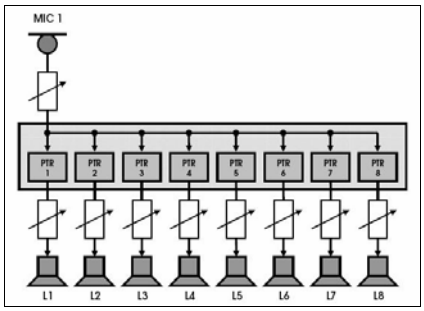


Figure 3. Patch 1.

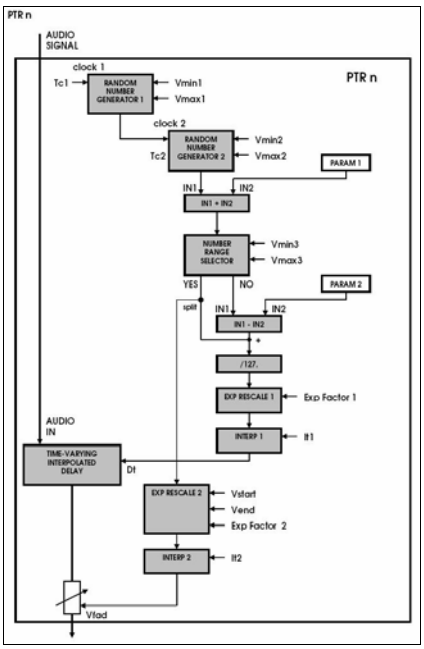


Figure 4. PTR module.

Patch 2 [0'30'' – 1'45''; 4'15'' – 5'30'']: audio signal from MIC 2 processed by a limiter/dynamic compressor is sent to a single sideband ring modulator (FREQUENCY SHIFTER module). Only the negative sideband is considered for further processing. The signal is split and routed directly to the spatialisation unit SPAT 2 x 8 and to a distortion unit (DISTORTION module). The signal from the DISTORTION module is then processed by FFT FILTER which provides equalization aimed to remove hiss and noise brought into the process by the distortion. Audio signal from FFT FILTER is then routed to the spatialisation unit, which is PARAM 1 and PARAM 2 sensitive.

Amplitude controls 1 and 2 are performed in real time by the live-electronics performer.

The algorithms included in Patch 2 are shown in Figures 5-9.

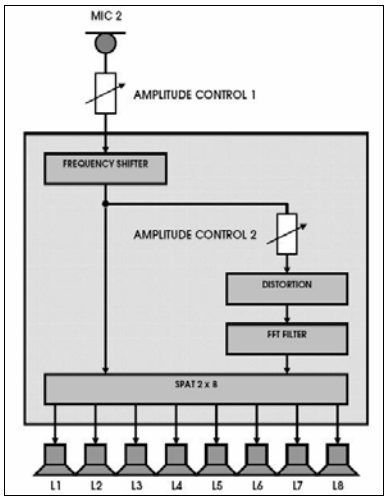


Figure 5. Patch 2.

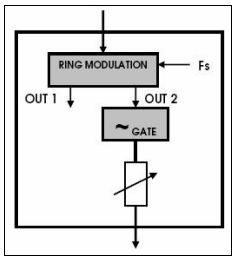


Figure 6. Frequency shifter module.

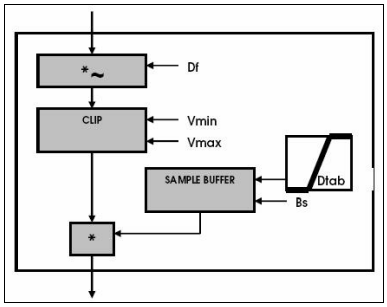


Figure 7. Distortion module.

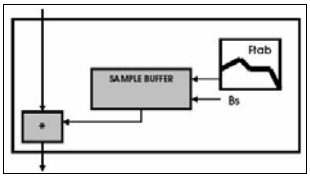


Figure 8. FFT filter module.

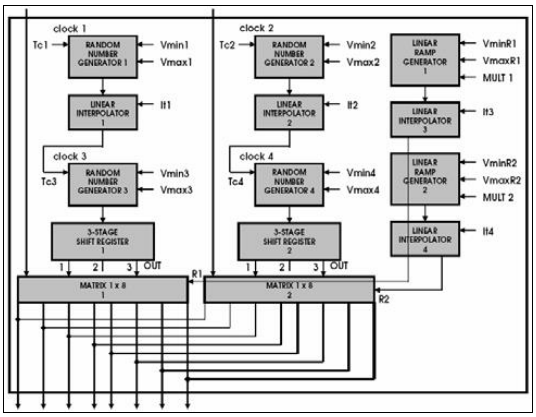


Figure 9. Spat 2x8 unit.

Patch 3 [1'55'' – 3'00''; 6'05'' – 6'32''] (see Figure 10): fifteen Pitch transposition modules (see Figure 11) in which PARAM 1 and PARAM 2 determine the range of values for the transposition factor.

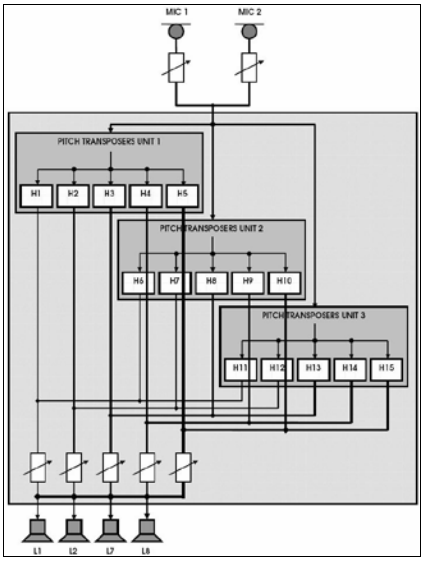


Figure 10. Patch 3.

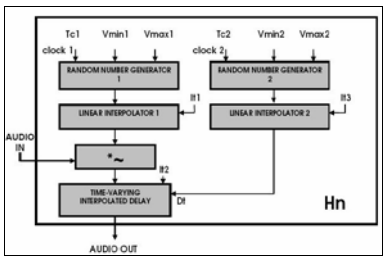


Figure 11. Pitch transposition module Hn.

Patch 4 [3'00'' – 3'45''; 5'50'' – 6'05''] (see Figure 12): audio signal is recorded in a buffer and then read by four engines graphically controlled in real time dragging the pointer inside a window: the engine reads the selected part. The velocity of reading depends on the length of the window: reading slower or faster a fragment causes pitch transposition. Inverse proportionality is suggested. The sound materials out from READ modules are routed to four spatialisation units PTR (see Patch 1).

Patch 5 [3'55'' – 4'15''] (see Figure 13): eight granulation units (see Figure 14) in which PARAM 2 determines the fluctuations of minimum grain duration values and PARAM 1 the fluctuations of the maximum grain increase values calculated on the basis of the grain duration values.

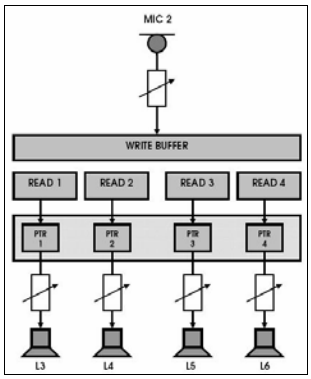


Figure 12. Patch 4.

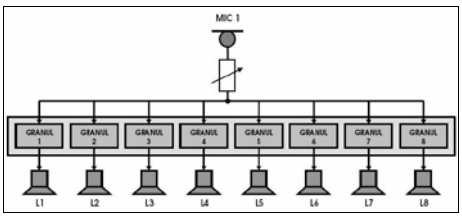


Figure 13. Patch 5.

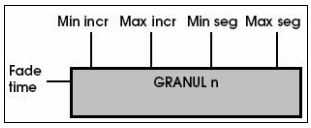


Figure 14. Granulation unit n.

For what concerns the right hand, the parameters are:

- bow direction;
- bow velocity;
- bow pressure;
- bow position;
- bow inclination;
- bow portion.

Different parameters are controlled by the live-electronics performer, in order to assure a real "four hand" performance.

The parameters controlled by the live-electronics performer are:

- input level for microphones' signal;
- output level from the five sound processing algorithms;
- input level to the distortion unit and output level from ring modulator and from distortion unit (Patch 2);
- output levels from stereo dynamic compressor (Patch 1);
- fade time for the spatialisers.

The sound materials performed by the musician are specified in the score [7], where the assignment of the patches to the sections of the piece is indicated.

3. DIFFUSION

The diffusion system requires eight independent channels and a sub-woofer. The diffusion point sources are placed around the audience, orientated toward the audience as shown in the Sound Projection Scheme (see Figure 15). The processing algorithms distribute the sound materials to the eight diffusion sound sources as explained below:

- Patch 1, 2 and 5 to 360° diffusion (L1-8);
- Patch 3 to front diffusion (L1, L2, tilt 0°) and rear diffusion (L7, L8, tilt 180°);
- Patch 4 to side front diffusion (L3 tilt 270°, L4 tilt 90°) and side rear diffusion (L5 tilt 270°, L6 tilt 90°).

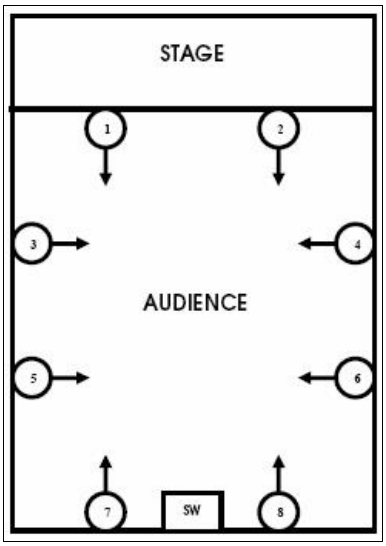


Figure 15. Sound Projection Scheme

4. THE INSTRUMENTAL MATERIALS

The choice of musical materials to be included in the score, especially the alternation of two opposite typologies of musical events (point-dynamic and linear-static), was aimed to support the processes of sound transformation/generation and their differentiation (diegetic function) with regard to the position on the timeline of performance (mimetic function). The experience of a musical time shared between the violinist and the live-electronics performer, in which events of different types find a common ground of evolution pertains to the character of *mimesis*, while the course of the musical materials from time to time generated or processed through the separate gestures of the two performers, in their particularity and different parametric connotations pertains to the character of the *diegesis*. The reference to an archaic time shared allows the occurrence of that particular type of interaction that Weinberg described as an Interconnected Musical Network where "live performance systems that allow players to influence, share, and shape each other's music in real-time" [11]. Together, the two connotations of *mimesis* and *diegesis* intensionally inform a number and quality of sound objects, which organize the spectro-morphological texture of the composition.

The musical materials are mainly constituted by:

- Point musical objects featuring variable pitch but undergoing to crystallization and characterized by micro-variation processes and iterative rhythm structure; the range is generally high or extremely high (Typology 1, see Figures 16-18);
- simple (one line) or stratified (two or more lines) linear musical objects featuring the possibility of internal rhythmic subdivision and variable range (Typology 2, see Figures 19, 20).

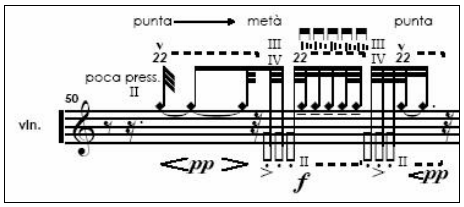


Figure 16. Typology 1; example 1.

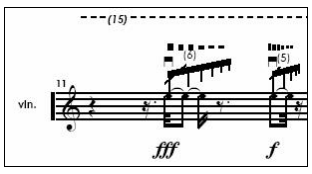


Figure 17. Typology 1; example 2.

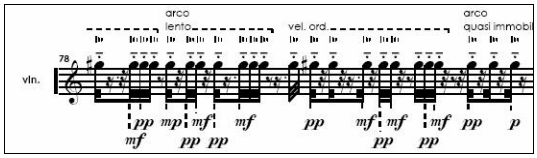


Figure 18. Typology 1; example 3.

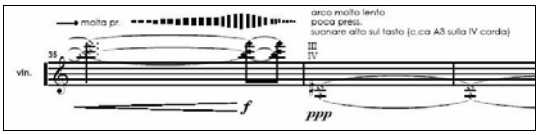


Figure 19. Typology 2; example 1.

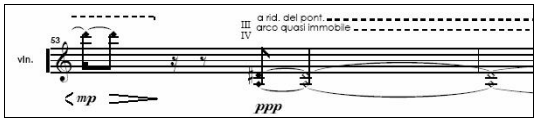


Figure 20. Typology 2; example 2.

For what concerns linear objects, the degree of intelligibleness of pitch varies on a continuum axis including white noise (e.g. bow on the bridge and strings muted with the left hand, 0'00'' – 0'30'', see Figure 21), single note (e.g. bow tremolo as light as possible close to the bridge at 2'40'', see Figure 22), bichord (e.g. low range natural harmonics with bow almost motionless close to the bridge at 3'30'', see Figure 23), trichords (e.g. high range natural harmonics with light bow in ordinary position around 2'00'' and maximum bow pressure with high distortion of the sound resultant at 2'24'', see Figures 24 and 25).

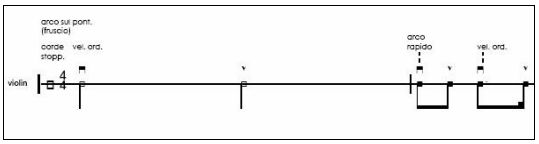


Figure 21. Bars 1, 2.

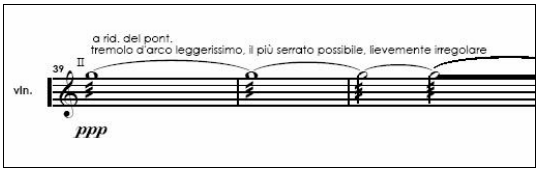


Figure 22. Bars 39-41.

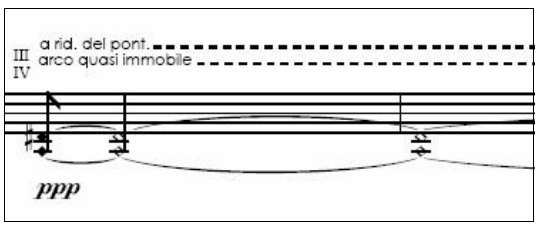


Figure 23. Bars 53, 54.

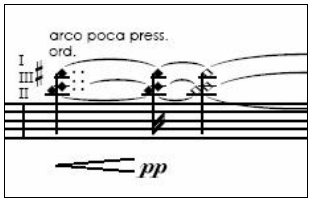


Figure 24. Bar 29.

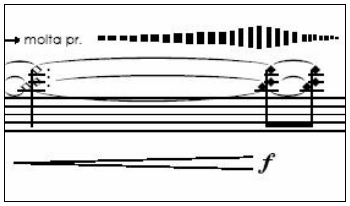


Figure 25. Bar 35.

5. DISCUSSION

From the point of view of language, it is useful to refer to a hyperinstrument as a hypertext constructed from an acoustic instrument, such as the violin. In this sense, the hyperinstrument is a "multidimensional structure simultaneously organized in a network of a multiple interconnections" (my translation) [1]. This conception of the hyperinstrument is consistent with that of Interconnected Musical Network, where "the use of technology […] pushes the tension between structure and process music further into an experience where predetermined rules and instructions, combined with improvised interdependent group interactions, lead to evolving musical behaviors, giving a new meaning" [10]. For further discussion of this issue, please refer to my contribution about the hypersampler in *Il grifo nelle perle nere* [8].

The choice of instrumental materials was made taking into account Cadoz's "Instrumental Gesture Typology" [2] which is based on gesture's function. Cadoz distinguishes between Excitation, Modification and Selection Gestures; Excitation gestures (see Section 4, Typology 1) can be Instantaneous (sound starts when gesture finishes: see Typology 1; example 3, Figure 18) or Continuous (gesture and sound coexist: see Typology 1; example 2, Figure 17); Modification Gestures (see Section 4, Typology 2) can be Parametric (continuous variation of a parameter: see Typology 2; example 1, Figure 19) or Structural (modification is related to categorical difference: see Typology 2; example 2, Figure 20); Selection Gestures describe the category of micro-variation (see Typology 1; example 3, Figure 18) which plays the role of parameter of formal organization.

The hyperviolin system developed for *Dalla Sua Orbita* is a multimodal interface controlled by both "the perceptual features […] extracted from the audio stream of the controller instrument" and the musical gestures: an "*interface* between a musical sound controller and a musical sound output of arbitrary timbre" [4] which

gives meaning to the musical gestures of the performers. It includes "high-level extensions toward integration of gesture and audio processing, aimed at music performance analysis and expressive information processing" [3] which effectively measures "aspects of the physical performance that describe the interaction of the player with the instrument" providing "possibilities for the real-time alteration of acoustic or electric sound though gestures familiar and learned by the player" [12].

The role of the technology implemented by the hyperviolin is consistent with that suggested by Tarabella and Bertini, according to which sensors" should leave the body completely free from whatsoever electrical and/or mechanical link with the main system, for giving the performer(s) maximum freedom of gesture and to hide technology away from the stage" [9]. No other gesture is required in addition to the violinist's musical gestures implicit in the performance of the score [7], through which it becomes possible to "evoke and control a greater – even unlimited – range of sounds and textures by simply interpreting the music in the most natural way"[6]. Provided that the parameters involved in a violin performance are many more of those taken into consideration in this research, and often difficult to quantify, the decision to focus on two macroscopic parameters such as PARAM 1 and 2 was a forced choice to "refine this multitude of possibilities into a new and meaningful unity, and the beautiful and tenuous fragility, and expressive possibility, of the new instrument, once it exists" [5].

6. CONCLUSIONS

In the light of the performances accumulated in recent years, including one by Carlo Lazari which took place on July 7, 2011 in the SMC – Sound and Music Computing Conference 2011 in Padova, we can say that *Dalla Sua Orbita* can be considered one successful effort to combine research in technology with the study of musical expression.

As previously specified, the effort was to match the parameters related to the musician's expressive gestures to those related to computer processing of instrumental material, in order to build a chain of meanings around the movement and energy that characterize the expressiveness of a violinist during a performance. This effort provided results from time to time changing, even in relation to the architectural space in which the performance took place, and significantly related to the design and implementation of the hyperinstrument system.

This is consistent with Machover, according to which a hyperinstrument must be able to capitalize on the continuous feedback mechanism that is created during performance, assuring "that intuitive musical intent is transmitted throughout the system" [5] and coherent with Weinberg, who emphasizes the flexibility to allow a true interaction "to stir the musical output into unpredictable directions, leading to an experience that is based on evolving and dynamic social contexts" [11].

Future developments will include the attempt to experience the extension of this paradigm to other more complex hyper-instrumental combinations.

7. REFERENCES

[1] A. Antinucci: "Summa hypermedialis. Per una teoria degli ipermedia" *Sistemi intelligenti*, Anno V, No. 2, 1993.

[2] C. Cadoz: "Instrumental gesture and musical composition", *Proceedings of the 1988 International Computer Music Conference*, pp. 1-12.

[3] A. Camurri, P. Coletta, C. Drioli, A. Massari and G. Volpe: "Audio Processing in a Multimodal Framework", *Proceedings of the International Conference 118th AES – Audio Engineering Society Convention*, Barcelona, 2005.

[4] T. Jehan: "Perceptual synthesis engine: An audio-driven timbre generator", M.S. thesis, School of Architecture and Planning, Massachusetts Institute of Technology, Cambridge, 2001.

[5] T. Machover: *Hyperinstruments: A Progress Report*, Cambridge (MA), The MIT Press, 1992.

[6] T. Machover: *Dreaming a new music*, Cambridge (MA), The MIT Press, 2006.

[7] M. Marinoni: *Dalla Sua Orbita*, Score, ArsPublica Edizioni Musicali, Camino al Tagliamento, 2011.

[8] M. Marinoni: "The feature extraction based hypersampler in Il grifo nelle perle nere: a bridge between player and instrument paradigm", *Proceedings of the ICMC SMC 2014*, pp. 772–779.

[9] L. Tarabella and G. Bertini: "Original gesture interfaces for live interactive multimedia performances", *Proceedings of the IV Journées d'Informatique Musicale/JIM'97*, pp. 41-45.

[10] G. Weinberg: "Interconnected Musical Networks: Toward a theoretical framework", *Computer Music Journal*, Vol. 29(2), pp. 23-39, 2005.

[11] G. Weinberg: "Interconnected Musical Networks – Bringing Expression and Thoughtfulness to a Collaborative Group Playing", Ph.D. dissertation, School of Architecture and Planning, Massachusetts Institute of Technology, Cambridge, 2003.

[12] D. Young: *New Frontiers of Expression Through Real-Time Dynamics Measurement of Violin Bows*, The MIT Press, Cambridge (MA), 2001.

PIANOSPACE: MODELLI MORFOLOGICI VIRTUALI E MAPPE COMPOSITIVE

Luca Richelli
Sound and Music Processing Lab
Conservatorio "C. Pollini" - Padova
lucarichelli@gmail.com

Sergio Canazza
Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Padova
canazza@dei.unipd.it

SOMMARIO

L'installazione PianoSpace [1], presentata il 27 settembre 2013 a Padova durante la Notte dei Ricercatori (iniziativa promossa dalla Commissione Europea che dal 2005 fa incontrare i ricercatori con il grande pubblico in differenti città europee), è il risultato della ricerca di un nuovo approccio alle potenzialità strumentali del pianoforte finora limitate dalle caratteristiche morfologiche del sistema tastiera-mano.

L'installazione si inserisce nel filone di ricerca inerente all'implementazione di "iper-strumenti" avviata da alcuni anni dal Sound and Music Processing Lab (SaMPL) del Conservatorio di Padova in collaborazione con il Centro di Sonologia Computazionale (CSC) del Dipartimento di Ingegneria dell'Informazione (DEI) dell'Università di Padova.

Il sistema sperimentale è costituito da un Disklavier, un sistema di motion capture interfacciati e pilotati tramite un computer attraverso l'ambiente di programmazione grafica Max. L'esecutore indossa un paio di guanti dotati di 4 led all'infrarosso e, muovendosi in uno spazio delimitato da 8 telecamere, suona il Disklavier.

PianoSpace intende volutamente allontanarsi dalla tradizione pianistica per esplorare le potenzialità sonore del pianoforte e un sistema di controllo cinetico nello spazio attraverso la creazione di molteplici Modelli Morfologici Virtuali e differenti Mappe Compositive.

1. INQUADRAMENTO STORICO

Il pianoforte, strumento a corde percosse da martelletti mediante una tastiera secondo la classificazione classica degli strumenti musicali [2], da un punto di vista morfologico è un sistema meccanico di produzione sonora controllato da tasti. Ad ogni tasto corrisponde l'emissione di una sola nota o frequenza prodotta da un minimo di una corda ad un massimo di tre corde accoppiate. I tasti sono organizzati secondo una disposizione lineare in successione cromatica (intervallo di semitono) dal tasto che controlla la nota più grave a quello che controlla la nota più acuta, dall'estrema posizione di sinistra (La0 27.5 Hz) all'estrema posizione di destra (Do8 4189 Hz) per un'estensione totale di 88 note.

La dinamica delle note è in rapporto diretto con la velocità di abbassamento del tasti: maggiore è la velocità, maggiore è l'ampiezza (volume) del suono emesso. La durata dell'abbassamento del tasto controlla la durata dell'evento sonoro.

La tecnica esecutiva pianistica tradizionale è legata alle possibilità delle mani come mezzo di controllo della produzione sonora dello strumento mediato dalle caratteristiche morfologiche e fisiche delle mani stesse; possibilità ma anche limiti: dieci dita possono controllare simultaneamente solo un numero parziale dei tasti dello strumento e con una relativa indipendenza ritmico-dinamica. Le dimensioni della tastiera permettono inoltre, ad una mano di medie dimensioni, di raggiungere generalmente l'estensione di un'ottava (13 tasti consecutivi), con un limite massimo di una decima maggiore (17 tasti consecutivi) superato, di poco, in casi rarissimi.

2. IL SISTEMA SPERIMENTALE

Per esplorare le potenzialità intrinseche del pianoforte è stato ideato un sistema composto da un pianoforte ibrido (Disklavier Yamaha), un sistema di optical motion capture (PhaseSpace) e un software per la gestione dei dati (Max).

Il Disklavier [3], strumento commercializzato dalla Yamaha dal 1986, è essenzialmente un pianoforte ibrido che usa dei solenoidi elettro-meccanici e dei sensori ottici per un controllo meccanizzato della martelliera del pianoforte, permettendo di arrivare ai limiti esecutivi fisici "teorici" dello strumento. Si deve comunque precisare che anche il Disklavier presenta dei limiti esecutivi che permettono sicuramente di controllare un numero maggiore di tasti e con una velocità decisamente superiore rispetto alle capacità di ottimo pianista, ma vi è comunque un limite di risoluzione temporale degli eventi eseguibili dovuto a fattori meccanici intrinsechi allo strumento e al protocollo MIDI di trasmissione dati.

Il PhaseSpace [4] è un sistema di optical motion capture utilizzato da alcuni anni al Centro di Sonologia Computazionale (CSC) del Dipartimento di Ingegneria dell'Informazione (Università di Padova) per il tracciamento del movimento dell'esecuzione strumenta-

Copyright: © 2014 Luca Richelli, Sergio Canazza. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

le tradizionale; il suo impiego come sistema di controllo ha permesso la realizzazione di iper-strumenti (musicali) per l'allargamento delle potenzialità sonore degli strumenti tradizionali mediante live electronics.

La gestione del flusso di dati provenienti dal PhaseSpace per il controllo in tempo reale del Disklavier è affidata a Max [5], un linguaggio di programmazione visuale per musica e il multimedia sviluppato da Miller Puckette all'IRCAM a metà degli anni Ottanta e ora commercializzato dalla Cycling '74.

La figura seguente illustra il flusso dei dati del sistema sperimentale.

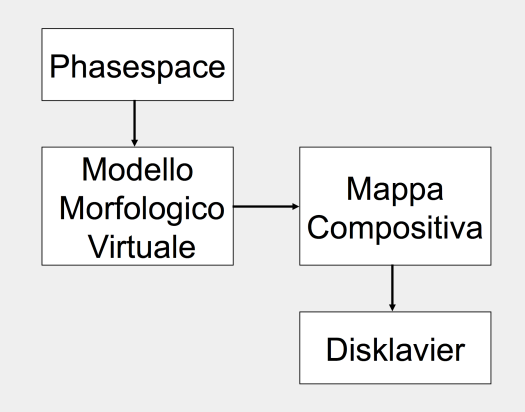


Figura 1. Flusso dati del modello sperimentale

Il PhaseSpace rileva e codifica i dati di posizione delle mani dell'esecutore attraverso le tre coordinate spaziali *x, y* e *z* ed il parametro rotazione (*R*). I 4 parametri sono ricavati dall'integrazione delle singole coordinate spaziali di ognuno dei 4 marker all'infrarosso applicati ai guanti. La scelta di utilizzare più marker è per aumentare l'affidabilità del sistema anche in caso di momentaneo oscuramento di qualcuno dei sensori.

Ogni 10 millisecondi viene inviato un pacchetto di dati di posizione al computer che gestisce, attraverso l'ambiente di programmazione visuale Max, il Modello Morfologico Virtuale e la Mappa Compositiva inviando al Disklavier, attraverso il protocollo MIDI [6], i parametri esecutivi relativi a: *note on/off, key velocity e damper pedal on/off.*

3. MODELLI MORFOLOGICI VIRTUALI E MAPPE COMPOSITIVE

Alla base dell'esplorazione delle potenzialità implicite dello strumento vi è la realizzazione di molteplici Modelli Morfologici Virtuali che rovesciano il paradigma esecutivo classico in cui il movimento delle mani si deve adattare alle caratteristiche fisico-morfologiche della tastiera, in cui le note sono disposte cromaticamente in ordine ascendente da sinistra verso destra e la dinamica è controllata dalla velocità con cui viene abbassato il tasto. Invece, grazie al Modello Morfologico Virtuale, il controllo dello strumento è svincolato dalle caratteristiche fisico-morfologiche della tastiera.

È possibile immaginare un numero infinito di Modelli Morfologici Virtuali con un numero variabile di parametri di controllo, anche se rimane ovviamente il limite oggettivo di quanti siano i parametri umanamente controllabili in tempo reale e con quale modalità.

Il Modello Morfologico Virtuale implementato sperimentalmente utilizza 4+4 parametri di controllo assegnati alle due mani dell'esecutore.

La tabella seguente riporta la corrispondenza tra i movimenti nello spazio dei sensori e i relativi parametri del Modello Morfologico Virtuale.

Movimento spaziale	Parametro controllato nel Modello Morfologico Virtuale	
	Guanto destro	Guanto sinistro
Sinistra-Destra (*x*)	Pivot Note	Keyboard Configuration
Alto-Basso (*y*)	Note on/off	Damper Pedal on/off
Avanti-Indietro (*z*)	Pattern Chord	Key Velocity
Rotazione (*R*)	Density	Pattern Chord Key Velocity Distribution

Tabella 1. Esempio di Modello Morfologico Virtuale

Le Mappe Compositive trasformano i dati del Modello Morfologico Virtuale in parametri esecutivi per il Disklavier. Questa trasformazione implica ovviamente delle scelte rilevanti da un punto di vista sia stilistico, che esecutivo determinando il materiale sonoro risultante e le possibili articolazioni e il senso musicale ed espressivo dell'esecuzione.

La figura 2 mostra la parte preliminare di realizzazione delle Mappe Compositive effettuata nell'ambiente grafico di aiuto alla composizione OpenMusic [7], per creare e verificare l'algoritmo compositivo basato su differenti *Keyboard Configuration*, riassemblamenti virtuali della disposizione dei tasti della tastiera del pianoforte, scelti mediante una *Pivot Note*. Il numero di note suonate contemporaneamente è in funzione del parametro *Density* in relazione al *Pattern Chord* selezionato.

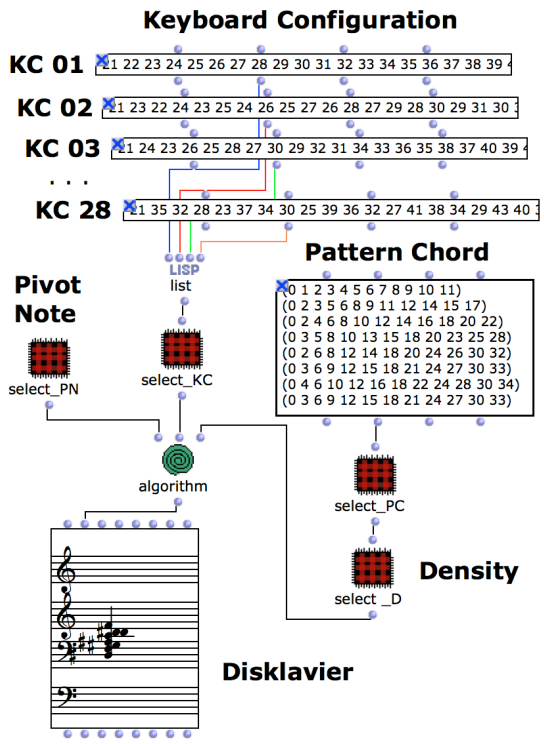


Figura 2. Mappa Compositiva

4. CONCLUSIONI E SVILUPPI FUTURI

Nel corso dei secoli l'evoluzione degli strumenti musicali si è sviluppata parallelamente e in simbiosi con l'evoluzione della musica. Nella diversità tra le varie culture la relazione tra gli strumenti musicali e la musica per essi concepita, è sempre stata molto stretta in un circolo virtuoso in cui nuovi strumenti necessitano di nuove musiche e nuove musiche hanno bisogno di nuovi strumenti. Questo circolo virtuoso se da un lato ha portato alla formazione di un repertorio specialistico per ogni strumento musicale, e la letteratura pianistica ne è l'esemplificazione per antonomasia, da un altro punto di vista ha limitato le possibilità potenziali di ogni strumento musicale a causa dei limiti esecutivi e del peso della tradizione.

I Modelli Morfologici Virtuali e le Mappe Compositive intendono essere, oltre che una ricerca teorica, anche uno stimolo a pensare al pianoforte come strumento di emissione sonora a prescindere dai limiti pianistici tradizionali e un invito a concepire nuova musica ripensando e inventando nuove modalità esecutive.

5. BIBLIOGRAFIA

[1] https://www.youtube.com/watch?v=1NCc7spSU14 Ultima visita: 20 settembre 2014

[2] A. Basso: "Strumenti musicali (classificazione)", *Dizionario Enciclopedico Universale della Musica e dei Musicisti (DEUMM) Il lessico, volume IV,* pp. 432-440, UTET, Torino, 1984

[3] http://usa.yamaha.com/products/musical-instruments/keyboards/disklaviers/ Ultima visita: 18 settembre 2014

[4] http://www.phasespace.com/ Ultima visita: 18 settembre 2014

[5] http://cycling74.com/products/max/ Ultima visita: 18 settembre 2014

[6] R. Guérin: MIDI L'interfaccia digitale per gli strumenti musicali, Apogeo, Milano, 2003

[7] G. Assayag, C. Rueda, M. Laurson, C. Agon, and O. Delerue, "Computer Assisted Composition at Ircam : PatchWork & OpenMusicComputer" *Computer Music Journa*l, 23:3, pp. 59-72 1999.

UN NUOVO APPROCCIO ALLA FORMA COMPOSITIVA NELLE INSTALLAZIONI SONORE : [re]BO[u]NDS E LA FORMA IN TEMPO REALE

Giulio Colangelo
Conservatorio "L. Refice" di Frosinone, ZKM
giuliocolangelo@gmail.com

Valerio De Bonis
Conservatorio "L. Refice" di Frosinone, ZKM
nastromagnetico@gmail.com

SOMMARIO

In questo articolo si affronta una discussione riguardante un particolare approccio estetico all'installazione sonora come strumento di composizione in tempo reale. Si discute nello specifico del ruolo del compositore che, nell'opera installativa, non rinuncia alla gestione di una forma compositiva ben definita e delle risorse informatiche atte ad agevolare tale operazione. Si considera inoltre il compositore contemporaneo come artista totale, desideroso di coinvolgere il fruitore in performance intermediali e quindi capace di progettare e realizzare nuovi strumenti, unici interpreti della propria idea compositiva.

Si presenta dunque l'installazione/performance [re]BO[u]NDS, sviluppata durante un periodo di residenza presso lo ZKM di Karlsruhe nel 2012, e l'utilizzo di un originale sistema elettromeccanico: un esecutore dal vivo in grado di leggere e interpretare autonomamente una composizione elettroacustica.

1. L'ESPERIENZA DELL'INSTALLAZIONE

Senza dubbio dal XX secolo in poi il mondo dell'arte è stato fortemente condizionato e ispirato dalla tecnologia che lo circonda. Nel corso degli anni la musica, grazie alle sperimentazioni d'avanguardia, ha conquistato un posto all'interno di musei, gallerie e spazi espositivi. Da ciò è affiorata la necessità di una nuova forma, una nuova opportunità di presentazione dell'arte, che si è definita attraverso l'installazione sonora. Essa mira a essere un'opera d'arte che non si esaurisce in un "oggetto d'arte"; vuole essere un ambiente, un luogo dove il pubblico partecipa, o meglio completa l'opera attraverso "l'esperienza dello spazio"[1, 8].

La volontà da parte dell'artista di coinvolgere attivamente il visitatore, affinché raggiunga un maggiore livello di empatia con l'opera d'arte, lo induce a considerare un nuovo modo di relazionarsi alla stessa. L'obiettivo è evadere dalla consueta fruizione stimolando lo spettatore verso una fruizione "espansa".

Copyright: © 2014 Valerio De Bonis, Giulio Colangelo. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

1.1 Installazione Sonora d'Arte

Le installazioni sonore hanno la grande capacità di coinvolgere totalmente il visitatore in un ambiente "aumentato". Tale ambiente tiene conto del rapporto che l'opera intende instaurare col fruitore attraverso modalità e forme espressive differenti (percorsi o architetture sonore, disposizione di punti sonori nello spazio, strumenti acustici o sculture etc.).

L'opera musicale ha ovviamente adattato la propria forma espressiva a questo nuovo modo di fruire; lavori interattivi, intermediali e "spaziali" hanno trasformato il rapporto che si può instaurare con l'opera d'arte. Il compositore/artista non può ignorare che nuovi spazi e tempi di ascolto significano anche nuovi spazi e tempi di fruizione dell'utente, un differente coinvolgimento sensoriale e quindi un nuovo modo di concepire la propria musica.

"L'installazione sonora d'arte è, tra le opere musicali, quella che più di ogni altra evidenzia le caratteristiche innovative di fruizione suddette. La sinergia che può instaurare con l'utente, con l'ambiente e l'architettura, con i materiali e le forme scultoree, con lo scorrere del tempo e degli eventi, le permettono di offrire al pubblico un insieme correlato di esperienze che acuisce la sensibilità per il segno artistico ed espressivo e stimola la conoscenza." [7]

Molti compositori contemporanei, aperti al concetto di installazione d'arte, hanno cominciato a concepire opere musicali realizzate e destinate appositamente ad esse. Tuttavia il concetto di installazione è spesso legato all'idea di oggetto d'arte che vive di vita propria e che si esplica all'interno di una esibizione che non ha una durata ben definita.

Installazioni sonore adattive (gestite tramite architetture informatiche capaci di considerare la componente ambientale e le sue variazioni) ed evolutive (capaci cioè di evolvere autonomamente nel tempo) viceversa tangono il concetto di suono in divenire facendo trasparire, all'interno di una forma ben più rarefatta e in parte aleatoria, le intenzioni del compositore. Questione fondamentale a questo punto riguarda la comprensione della forma compositiva e quindi del senso dell'arcata musicale, non avendo come punto di riferimento un inizio e una fine e quindi un tempo ben determinato.

Tenendo in considerazione però che è sempre lo spettatore a essere l'elemento più importante, il protagonista che giustifica e dà concretezza a qualsiasi rappresentazione artistica, possiamo fare questa riflessione: nel momento stesso in cui osserva, elabora e interpreta, l'utente dà vita all'opera e per lui l'opera vive solo in quel preciso momento, definendone sempre il carattere di un'esperienza finita.

"The musical work is fully defined only at execution time depending on context available resources, and only after appropriate user intervention. Nevertheless, the self-adaptation must ensure the character of the music desired by the artist, called style. […]" [6]

"La musica esiste solo fintanto che dura l'ascolto, come Dio finché dura l'estasi. L'arte suprema e l'essere supremo hanno questo in comune: dipendono interamente da noi" [4]

1.2 Installazione Performativa

Considerando che già nella metà del Novecento i compositori sperimentarono su generi che possiamo definire ibridi, posti a metà tra installazione e composizione musicale, ad esempio le installazioni di Stockhausen da lui stesso definite "Raummusik"[1], [1, 2] allo stesso modo il compositore moderno può avvicinarsi alle installazioni sonore con la volontà creativa che lo contraddistingue, considerando l'installazione sonora come un nuovo strumento, o meglio ancora come un nuovo interprete: sculture e architetture informatiche capaci di essere installazioni, in quanto autonome, ma a servizio di una forma compositiva predeterminata.
Il musicista, divenuto così anche inventore e artista totale, crea un'esperienza percettiva capace di coinvolgere il pubblico sia dal punto di vista musicale che visivo: viva scultura intermediale.

1.3 Artigianato Tecnologico

In tanti anni di musica e di ricerca (tecnica prima e tecnologica poi) artisti, e se vogliamo anche artigiani, hanno dimostrato estremo interesse per l'innovazione e l'estensione degli strumenti sonori fino alla creazione di nuovi prototipi.
Si pensi ad esempio al violino aumentato o al "Magnetic Resonator Piano"[2] o ancora all' "Electromagnetically Actuated Vibraphone"[2], strumenti questi che associano alle tradizionali funzionalità l'applicazione di sensori o componenti elettromagnetici in grado di captare e trasmettere ad un elaboratore informazioni di carattere sonoro o interpretativo. Essi tuttavia conservano tutti i limiti fisici dell'interprete umano. Non sono nuovi gli strumenti che hanno ovviato a questa problema facendo a meno della figura dell'interprete o limitandone fortemente le funzioni. Si faccia riferimento ai famosi organi a rullo o organi di barberia (antesignano del computer in quanto macchina programmabile) capaci di integrare nei giri di manovella di un unico esecutore le possibilità polifoniche di una piccola formazione orchestrale, o al noto "Disklavier"[3],totalmente automatizzato. L'assenza dell'esecutore, in realtà, può essere considerata una "decurtazione" o una mancanza, ma di fatto molte opere utilizzanti strumenti sonori elettronici si rivelano più interessanti di quanto si possa immaginare, aprendo la percezione dell'ascoltatore a nuovi percorsi o metodi di fruizione.

Dagli anni '90 a oggi diversi compositori hanno sperimentato con la scrittura per questi strumenti, ma quando si passa dalla semplice estensione di uno tradizionale alla costruzione ex-novo di un artefatto sonoro, allora le possibilità espressive e musicali conducono a risultati inaspettati e sorprendenti. La realizzazione di tali opere musicali avviene attraverso differenti architetture, perlopiù elettromeccaniche, che generano suoni spesso ripetitivi avvicinandosi sempre più al concetto di spazio sonoro statico e quindi allontanandosi dal concetto di evento sonoro che evolve nel tempo.

L'elemento della ripetitività dunque ci porta ad osservare come questi lavori siano formalmente distanti da un intento compositivo. La struttura formale sembra rispondere all'unico comando d'inizio e fine, piuttosto che a una serie controllata d'istruzioni nel tempo. Ci troviamo davanti a sculture sonore che fondano proprio sull'automatismo la loro struttura, che appare immune da variazioni formali.

2. [re]BO[u]NDS

La nostra intenzione è stata quella di progettare un interprete elettromeccanico in grado di leggere una partitura di comandi e di essere estremamente preciso nel suo lavoro. L'opera non rinuncia alla dimensione compositiva unicamente per palesare un effimero, se pur complesso, meccanismo di precisione. Quest'ultimo è strutturale di un'arcata musicale ben definita: un meccanismo reso umano dal compositore.

[re]BO[u]NDS è una installazione/performance intermediale, uno spettacolo totale che cerca di raggiungere il fruitore sia dal punto di vista sonoro che visivo. Non si presenta come oggetto sonoro autonomo: l'esigenza è stata quella di non lasciare l'istallazione a se stessa ma di controllarla in ogni momento al fine di esprimere la nostra intenzione formale, utilizzando eventi di tipo tensivo e risolutivo, tecniche consolidate quali accelerandi e rallentandi, variazioni dinamiche, oggetti sonori improvvisi, silenzi, eventi figurativi e complesse

[1] Musica spaziale o di ambiente. Stockhausen, in alcune composizioni, introduce una particolare disposizione spaziale degli strumentisti e del pubblico, considerando perfino il loro movimento. Ciò, come stesso egli dichiara, influenza non solo l'esecuzione del brano ma anche il suo modo di concepire e comporre l'opera, valutando differenti tempi di fruizione. La composizione *Ensemble* (1967), ad esempio, è una performance di quattro ore in cui i musicisti suonano in automobili separate, incrociandosi solo parecchi chilometri al di fuori della città. È un esperimento che l'autore definisce con la parola 'Raummusik'. Cfr. K. Stockhausen, *Texte zur Musik*, vol.3, 1963 – 1970.

[2] C4DM.

[3] Yamaha corporation.

trame tessiturali al fine di incuriosire il fruitore e sorprenderlo nella visione della scultura sonora in azione.

2.1 Sinossi e Concept

Esplorare la dimensione spaziale di una reazione a catena che evolve da un piccolo cambiamento iniziale fino a un grande disordine indotto. Un flebile battito d'ali di una farfalla può causare enormi catastrofi naturali. Allo stesso modo il sottile suono di una singola goccia d'acqua può trasformarsi in un immenso oceano.

La nostra ricerca parte dall'indagine sulle dinamiche di un caos indotto. Il compositore, in quanto demiurgo, deve maneggiare la materia, immergersi in questo "caos liquido", canalizzarlo nei cavi audio e negli altoparlanti allo scopo di interpretarlo.

Lo sforzo metafisico è quello di produrre questa "confusione", assecondando le sue energie, gravitazioni, e forze interne che gestiscono il processo di formazione di una struttura autonoma, come un cristallo. Vogliamo interpretare ciò che sembra apparentemente complesso e renderlo umano.

L'opera d'arte non è creazione dal nulla. L'arte (o l'apparenza) cristallizza per un attimo il caos in forma, rendendolo mondo." [3]

Dodici altoparlanti (Figura 1) sono posti in modo da creare uno spazio sonoro totale. Forze immaginarie, considerate da noi come "attrazioni", collegano i diffusori come cellule di una stessa struttura, disegnando immaginari legami molecolari (molecular "bonds") all'interno dello spazio. Il suono originario, come embrione di un effetto domino, rimbalzerà ("rebound") in tutte le direzioni attraverso tutti i punti sonori. Dal suono "cristallino" della goccia d'acqua alla "cristallizzazione" del disordine indotto dalla sua caduta.

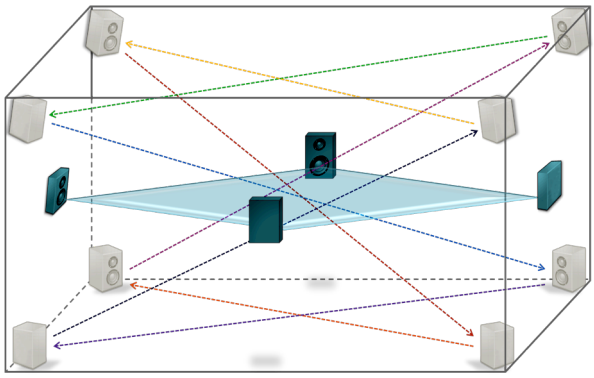


Figura 1. I "rimbalzi" spaziali nello schema originale.

2.2 Procedura dell'Installazione

[re]BO[u]NDS è stata realizzata durante un periodo di residenza artistica presso il dipartimento IMA dello ZKM (Zentrum für Kunst und Medientechnologie) di Karlsruhe.

Essa presenta una doppia personalità: 1) composizione/performance live multicanale per un "dripper" (Figura 2) comandato, elaborazione sonora in tempo reale e luci sincronizzate (ca.13'); 2) installazione d'arte sonora multicanale;

Al centro della sala si sviluppa una struttura verticale composta da un lungo cilindro di vetro riempito d'acqua, poggiato su una base, al di sotto del dripper. Quando una goccia d'acqua cade all'interno del cilindro il suono della goccia verrà captato ed elaborato in tempo reale da una complessa architettura informatica che sovraintende anche alla spazializzazione.



Figura 2. Il *dripper*; elettrovalvola comandata a 12V

2.3 L'architettura di [re]BO[u]NDS

L'architettura può essere semplicemente schematizzata (Figura 3): due unità di calcolo lavorano in sincrono per l'erogazione controllata e la successiva elaborazione sonora di gocce d'acqua. Il suono generato e processato viene infine spazializzato su un panorama sonoro a otto o dodici canali. Tutta la catena segue una gerarchia ben precisa e tutto il sistema opera in piena autonomia.
L'opera, presentata come esibizione dal vivo, dà inoltre la possibilità al compositore di diventare performer, formando un ulteriore anello della catena di elaborazione, capace di incidere sul risultato sonoro finale.

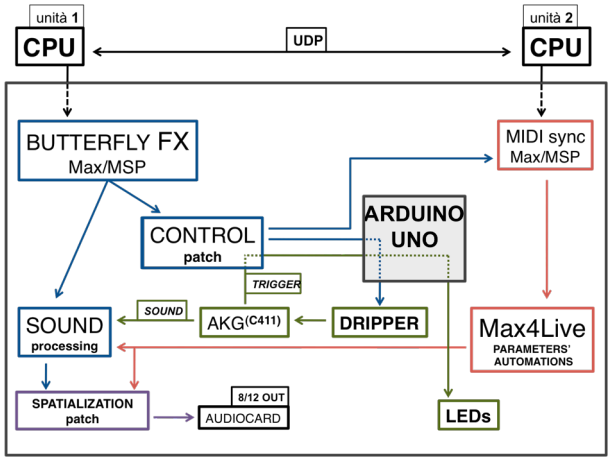


Figura 3. Schematizzazione della struttura informatica.

L'unità n.1 è probabilmente il punto di partenza o se vogliamo il primo fondamentale anello della struttura; essa è dotata dell'ambiente di programmazione Max/MSP (vers. 5) sul quale è sviluppata la patch "Butterfly FX" (da un confronto con la successiva versione del programma risulta evidente una migliore prestazione

e velocità di calcolo, forse dovuta alla quantità minore di risorse destinata alla grafica). Quest'ultima consta di una "control patch", che sovraintende allo svolgimento temporale della composizione riservando le prime risorse all'invio del comando che aziona il dripper tramite la lettura di una partitura informatica, e di un'altra addetta ai moduli di elaborazione sonora.

Il comando che viene inviato dall'unità n.1 giunge al secondo elemento di gestione della catena: l'*unità n.2* ("slave" della prima) che gestisce le automazioni dei parametri di elaborazione del suono. Per questa e altre funzioni è stato valutato opportuno l'utilizzo del software *Live 8 (Ableton)* comprensivo di ambiente *Max4Live*. Ci preme precisare che la scelta è stata motivata dalle difficoltà incontrate utilizzando gli oggetti "table", "itable" e "coll" [5] che, direttamente in Max/MSP, non hanno garantito un'agile e veloce lettura in contemporanea di numerose tabelle di dati. Per contro un sotware come *Ableton Live* è pronto a un'ottima gestione della lettura nel tempo di numerosi array.

Le due unità sono collegate tramite protocollo di rete UDP (istantaneo grazie al fatto che mantiene meno informazioni di controllo e non gestisce il riordino di pacchetti IP). Quella che può sembrare un'unità maggiormente dedicata ai suoni e ai vari processi audio è, come già anticipato, assegnataria del controllo dello strumento erogatore d'acqua: tramite la scheda elettronica "Arduino Uno" (con microcontrollore ATmega328 e velocità di clock a 16 MHz) il comando digitale binario proveniente dall'unità n.1 viene trasformato in variazione di tensione. Una tensione di 5 volt eccita un dispositivo elettronico a relè (innestato su un'ulteriore scheda elettronica sviluppata per l'uso) in grado di convogliare i 12 volt (in corrente continua) necessari al dripper. Di conseguenza ogni elemento temporale segnato in partitura equivale a un'apertura e chiusura di un'elettrovalvola che genera la caduta di piccole quantità d'acqua.

La particolarità interessante di tale meccanismo è che si riesce a variare la frequenza delle gocce, nonché la dimensione delle stesse. La velocità massima che si raggiunge è di una goccia ogni 35ms circa. Inoltre, dilatando il tempo tra l'apertura e la chiusura dell'elettrovalvola, si ottengono gocce di lunghezza e peso variabile. Si può facilmente immaginare cosa accade se si lascia l'elettrovalvola costantemente aperta!

Proseguendo la catena o, se vogliamo, seguendo la caduta verticale di ogni goccia ci s'imbatte nel cilindro contenitore d'acqua; al di sotto di quest'ultimo è posto un microfono a contatto (serie C411 della AKG) ed una serie di LED pronti all'accensione.

Cosa accade quando una goccia tocca la superficie dell'acqua? Il suono viene captato dal trasduttore il quale a sua volta comunica questa variazione di pressione sonora all'unità n.1. Visivamente è possibile osservare l'impulso ricevuto dal microfono grazie alla serie di LED luminosi nascosti sotto il cilindro di vetro. Essi si comportano rispondendo istantaneamente al comando ricevuto. Questo effetto di variazione di tensione nel tempo viene ricreato grazie alla modulazione di larghezza d'impulso (PWM) che permette l'utilizzo di una uscita digitale dalla scheda Arduino (funzionante a tensione di lavoro pari a 5V) per simulare un comportamento simile ad una sinusoide: il risultato è un abbassamento o innalzamento di luminosità grazie al passaggio veloce in uscita da 0 a 5 volt.

L'elaborazione elettronica del suono della goccia prende vita da questo momento in poi. Il suono del microfono giunto all'unità n.1 viene contemporaneamente ripartito tra le due unità. In questa maniera la seconda riceve il comando di leggere istruzioni disposte sulla timeline del software Live, e queste informazioni vengono tempestivamente rimandate alla prima unità. Come precedentemente detto è qui che avviene la vera e propria elaborazione del suono e successiva spazializzazione.

La patch "Butterfly FX" dispone di moduli per la modulazione d'ampiezza, per la riverberazione, per i ritardi e per l'equalizzazione. Ma soprattutto ha tre moduli per la sintesi granulare nei quali buffer viene memorizzato il suono reale della goccia. Ognuna di queste viene mantenuta in memoria e all'occorrenza "spezzettata" in tantissimi minuscoli "grani" a loro volta diffusi sul panorama multifonico.

A proposito della spazializzazione, il processo di passaggio dei suoni su ogni singolo canale è stato anch'esso gestito in forma di istruzioni disposte con molta precisione nel tempo: la composizione, dunque, controlla anche l'elemento spaziale con particolare attenzione ai legami che si materializzano tra un diffusore e l'altro.

Nella fase centrale della composizione subentra anche un altro tipo di elaborazione realizzato attraverso una catena di ritardi che determina un lungo momento di aggregazione di singole gocce (denominata all'interno della patch come "chain reaction"). In questa lenta fase, in cui il suono si muove lentamente da un altoparlante a un altro, si passa dalla chiara percezione della posizione dei suoni nello spazio alla graduale e completa invasione di un oceano sonoro che "aggredisce" il fruitore da tutte le direzioni.

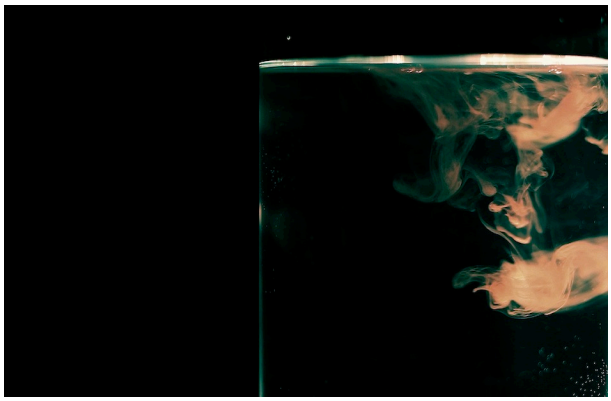


Figura 4. Video dimostrativo di [re]BO[u]NDS (© G. Lancellotti)[4]

[4] *vimeo.com/loxosconcept/rebounds*

La complessa architettura appena descritta è stata valutata con attenzione cercando di ottenere una catena gestibile in tutti i suoi singoli anelli. L'utilizzo di due computer, inoltre, rende possibile l'intervento della figura del compositore nel caso in cui l'opera venga eseguita in tempo reale, permettendo di intervenire al momento su alcuni parametri dell'elaborazione sonora.

2.4 Audiomazioni

Nel tentativo di alleggerire l'architettura eliminando l'unità n.2, cioè quella che utilizza il sequencer, si è ricercata una soluzione per affidare tutta la struttura informatica a un solo computer in grado di elaborare e variare i parametri autonomamente. A questo proposito ci piace porre in evidenza un oggetto external di MSP sviluppato presso l'Università di Musica di Basilea da Thomas Resch e Benedikt Schiefer : "note~".[5] È un valido strumento per l'ambiente Max/MSP che integra le tradizionali caratteristiche di un sequencer a nuove e dinamiche funzionalità. Al pari di "note~" (v0.8 beta) l'external "rs.delos"[6], progettato da Roby Steinmetzer, fornisce la possibilità di editare agilmente tracce MIDI, tuttavia con la considerevole pecca di non disporre di una modalità di importazione di MIDI file. Inoltre c'è da aggiungere che entrambi gli oggetti consentono la sola gestione di eventi MIDI, disposti lungo un piano roll orizzontale, trascurando apparentemente i dati derivanti dai "control change" (CC con valori da 0 a 127. Analogamente si comporta l'oggetto nativo "detonate".

Vista l'impossibilità di avere una lettura stabile in Max/MSP, si è provato a "trasdurre" tutte le automazioni e i comandi MIDI in semplice segnale audio. Per meglio comprendere questa opzione si faccia riferimento al noto dispositivo elettronico VCO[7] che permette di controllare, con una tensione elettrica applicata, una oscillazione periodica. Nel nostro caso, tutti i parametri e i comandi vengono controllati in tensione grazie alla lettura di una forma d'onda precedentemente registrata ed accessibile con l'oggetto "sfplay~". In questo modo viene organizzata una lettura simultanea di diverse tracce audio utilizzate come controllo dei parametri.
Risulta evidente che la lettura delle diverse forme d'onda dovrà avvenire in maniera simultanea ed alla medesima velocità e anche il passaggio da dati MIDI a segnale audio necessita della stessa accortezza.

Talvolta la lettura dei numerosi dati MIDI inviati contemporaneamente comporta degli spiacevoli (sebbene non molto frequenti) inconvenienti: comandi non perfettamente sincroni, brevi ritardi nell'invio, errori nella ricezione. Con l'utilizzo di segnale audio per il controllo dei parametri, questi inconvenienti non si presentano a scapito, c'è da dirlo, di una maggiore dose di calcoli cui sottoporre il computer, ma che può facilmente essere ridotta diminuendo la risoluzione in bit delle audiomazioni (una profondità di 8 bit darebbe già 256 steps di risoluzione). La fase di registrazione prevede il passaggio da numero intero a campione audio tramite l'oggetto "sig~". A seguito di un'opportuna scalatura (da range 0/127 al dominio d'ampiezza -1/+1) è possibile quindi registrare e rileggere, attraverso l'oggetto "snapshot~", come automazione un qualsiasi evento sonoro.

2.5 [re]BO[u]NDS ~ expanded media

In occasione della prima esecuzione assoluta presso lo ZKM [8], è stata realizzata una versione di [re]BO[u]NDS ampliata nella sua struttura.
Sono stati aggiunti altri due cilindri di vetro così da ottenere, in totale, tre elettrovalvole, tre sistemi di luci a LED e tre contenitori cilindrici muniti ciascuno del suo microfono a contatto. La composizione è stata inoltre estesa a ventidue minuti, espansa e arricchita da suoni preregistrati e "triggerati" dalle gocce d'acqua e spazializzata in un panorama multifonico di quarantatré diffusori (disposti a cupola all'interno del *Kubus* dello ZKM). È facile immaginare che la tecnologia utilizzata per l'intera installazione ha subito delle modifiche soprattutto per quanto riguarda l'architettura di comunicazione tra le due unità. La connessione UDP, utilizzata nel primo progetto, è stata sostituita interamente dal protocollo MIDI; in questo modo ogni messaggio, ogni nota e ogni comando sono inviati attraverso dei canali standard MIDI. Ciò ha dato maggiore stabilità all'intera architettura, sempre composta da due unità operative, che gestisce l'installazione. Inoltre è stata valutata la necessità di smaltire il lavoro affidato a una sola scheda Arduino aggiungendone un'altra uguale dedicata solamente ai circuiti dei LED.

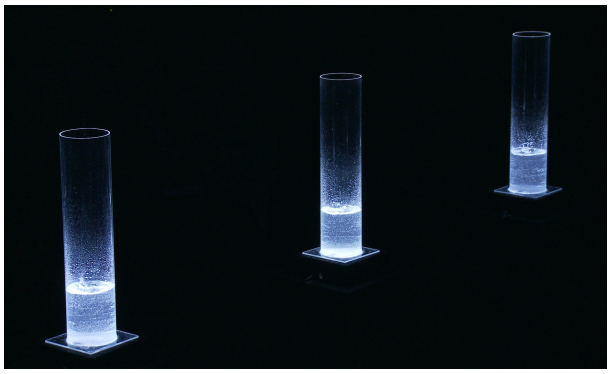


Figura 5. [re]BO[u]NDS - live performance presso lo ZKM. Evento "Visuality" (© ZKM - visual media)[9]

CONCLUSIONI

Una precisa e premeditata disposizione degli eventi e degli elementi nello spazio e nel tempo conduce forzatamente alla realizzazione di un prodotto ben determinato nella durata; probabilmente efficace ma finito.

Per ovviare a questo problema senza rinunciare alla potenza del risultato compositivo, bisognerebbe realizzare un ibrido che si ponga a metà strada tra un meccanismo autonomo e una composizione ben definita. Si

[5] noteformax.net
[6] arts.lu/roby/index.php/site/maxmsp
[7] Voltage Controlled Oscillator

[8] Produced at IMA - "Visuality" (Kubus, 15/05/2014)
[9] *vimeo.com/loxosconcept/reboundsexpanded*

prenda come esempio [re]BO[u]NDS: utilizzando più dripper e programmando una serie di figure musicali ben precise per ognuno di essi si riuscirebbe, con uno studio stocastico, a combinare ritmicamente, in maniera quindi non del tutto casuale, i loro comportamenti. In questo modo il compositore deciderà il suono, il timbro, il carattere e gli sviluppi della forma compositiva pur non conoscendo quando essi effettivamente si presenteranno.

BIBLIOGRAFIA

[1] G. Albert, *'Sound Sculptures' e 'Sound Installations'* in AAA.ATAC, Fabrizio Serra editore, Pisa/Roma, pp. 41-66, 2010.

[2] G. Albert, *Rapporti tra opera e fruitore nelle origini dell'installazione sonora* in ACT - Zeitschrift für Musik & Performance, Thurnau, 2012/4.

[3] W. Benjamin, *Affinità elettiva di Goethe* in *Angelus Novus*, Enaudi, Torino, p. 219, 2008.

[4] E. Cioran, *Confessioni e Anatemi*, Adelphi, Milano, p. 42 ,2007.

[5] A. Cipriani, M. Giri, *Musica Elettronica e Sound Design Vol. 2*, Contemponet, Roma, pp. 407-427, 2014.

[6] A. Gabriele, M. Lupone, P. Inverardi and P. Pelliccione, "Ad-opera: music-inspired self-adaptive systems", in Proceedings of the FSE/SDP workshop on Future of software engineering research, ACM New York, pp 119-124, 2010.

[7] M. Lupone, Interattivo, Adattivo, Evolutivo, Conservatorio di musica "A. Casella" - CRM, Seminario all'Università di Belgrado, 2008.

[8] V. Zuckerkandl, Sound and symbol. Music and external world, Bollingen Paperback, Princeton 1969.

A SONIC ART BOOK (LITTLE RED RIDING HOOD)

Annalisa Metus
Artist and Paper Designer
annalisa.metus@gmail.com

Giorgio Klauer
Conservatory of Music "C. Pollini" Padua
giorgio.klauer@conservatoriopollini.it

ABSTRACT

A Sonic Art Book (Little Red Riding Hood) is the first output of a cross-sectoral research in the field of sonic movables, artist books, sonic interaction design and electroacoustic music. Specifically, A Sonic Art Book (Little Red Riding Hood) is a prototype resulting from an artistic research on narration that has been conducted referring to methods belonging to design practice, over and above creative/artistic attitude.

Little Red Riding Hood has been developed from user experience design: paying particular attention to the visual display and to the choice of sounds, the work has the purpose of inducing a feeling of childlike wonder in the unaware user. Opening the book, a large paper forest pops up; suddenly, the sonic display manifests itself combining sounds in real time, collecting or scattering them across the page in response to the user's movements, inviting to move around and to explore the movable under different points of view.

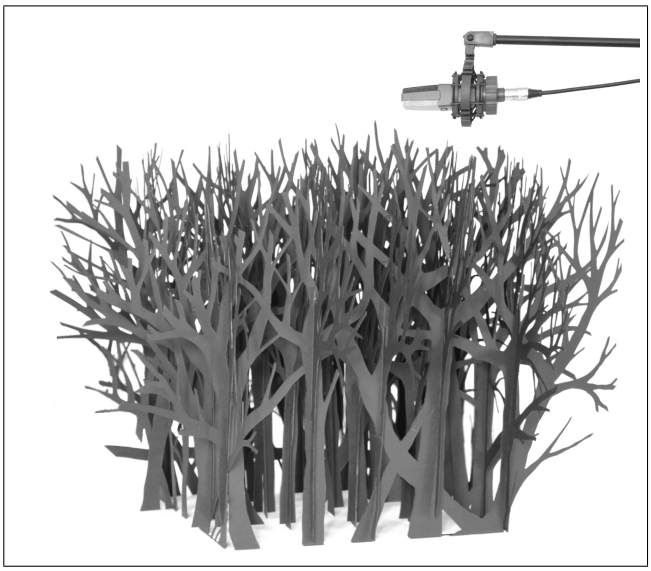


Figure 1. A Sonic Art Book (Little Red Riding Hood)

1. THE SONIC ART BOOK PROJECT

The Sonic Art Book project started with the aim of narrating fairy tales through alternative communication means, yet coherently with the book object, obviating written media and avoiding a predetermined language.

The tradition of artist books lends itself to this kind of research, and even more movable do, being historically an unlimited testing ground for flexibility in incorporating technological solutions and in adapting to fashions without being an exclusive product.

1.1 Artist Books

Artist books are rarely subject to market practices and they historically encourage individual experimentation; self construction is a widespread praxis and reproducibility is not required.

Rather than a confined category, artist books can be better described as a zone of activity where formal means are integrated with thematic or aesthetic issues [1].

The choice of approaching the tradition of artist books legitimizes poetic choices which would otherwise be inconvenient in a pure industrial design and marketing perspective.

1.2 Narrating Fairy Tales

Solely classic and worldwide known fairy tales have been considered, since they are part of the social imaginary and offer an easily detectable *topos* (e.g. the forest in Little Red Riding Hood, the gingerbread cottage in Hansel and Gretel).

The narrative takes place *de facto* leading the user recovering consciousness of data he already knew, suggesting different interpretations or introducing variants that are perceived as such precisely because of the well known structure of the original.

1.3 Movables as a Narrative Tool

The aim of the project is to design a user experience that inspires a sense of childlike wonder - an experience that proves to be magical as may be the narration of a fairy tale for a child.

Respecting the playful halo that surrounds them since nearly three centuries, movable books do not usually synthesize, generalize or make the experience universal; movables try to make the experience as unique as possible: for this reason they are more suitable to the project than illustrated books.

Narration occurs in movables through simulation, which is fulfilled through few unavoidable steps:

Copyright: ©2014 Annalisa Metus, Giorgio Klauer. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

- distancing: the movable is accessible only opening it, and even once opened it is limited in space;
- analogy: the movable implies knowledge of the rules of behaviour allowing the exploration and the comprehension of symbols created by the illustrator or the paper engineer;
- representation: the moment in which the illusion is realized and reality is replaced by fiction.

Large pop-ups [1] create a world apart and physically involve the user, requiring his active participation as a performer, not only as a spectator of an installation.

The peculiar tactile sensation felt by opening a pop-up book is in most cases increased by the sound produced by the paper mechanism itself. Having regard to the exceptional size of Little Red Riding Hood book, in addition to the noise of the paperboard slipping into the correct position, the tactile sensation is increased by the considerable weight of materials, unexpected as the three dimensional volume that is going to be created.

Once opened, the movable behaves as a scenario that can be explored: its large dimensions make it enjoyable and understandable even by spectators.

1.4 Sound as a Narrative Tool

The main strength of adding a sonic display to a pop-up book is the further temporal dimension that is created: as usual with movables, the performer can administer the time it takes to open the book; in the event of more complex movables, the user can unfold different elements at different times.

Once the book has been opened, sounds emerge from the page and move around in space in response to user's behaviour. Sounds are related to movements and gestures in a not obvious way, inviting to a fruition which lasts reasonably in time, in contrast with the immediate recognition of the visual display [2].

Each opening of the book provides a sound characterization of the same *tableau* (i.e. a single "paper scene" summarizing the most relevant features of the tale), as if the story was told in other words, with variants or variations highlighting different details.

Visual and sonic displays have the same relevance and their aesthetics are minimalist in order not to prevail over the other element.

1.5 Experience Design

The main instance of A Sonic Art Book project is to preserve the fact of having to deal with books, not to be confused with generic performative objects or installations.

Also referring to Jakob Nielsen's concept of usability heuristics [3], staying focused on the book makes the affordance very intuitive and instructions (potentially) dispensable: for the interaction to begin, the user has to open the book; to stop, just close it.

[1] Three-dimensional objects which rise suddenly from the page when the book is opened.

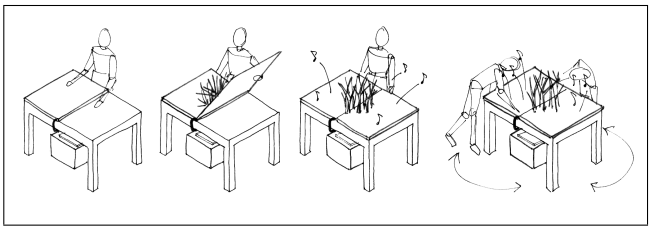


Figure 2. Usage scenario

The phenomenal field [2] varies from user to user, but the choice of renowned fairy tales allows the majority of the users to identify and comprehend the scenarios, making the project A Sonic Art Book addressable both to adults and children.

A typical usage scenario has been imagined:

- users find the movable closed on a table;
- opening the book the user starts the interaction: a large pop-up emerges from the page while the sonic display changes with movements and gestures providing a sonic characterization of the scene;
- closing and opening the book again, the performer switches from one sonic scenario to another;
- the performer moves all around the table where the book lays upon and bends over it to better listen and observe, potentially complexifying the interaction.

It is fundamental that the interaction occurs in a space as silent as possible (e.g. a library), and adequately wide to allow the user to move comfortably around.

2. LITTLE RED RIDING HOOD

The choice of Little Red Riding Hood as first output of the project is based on the interesting implications, for both sound and paper design, of rendering the intrinsic richness and complexity of the *topos* of the forest.

2.1 The Role of the Visual Display

The meeting of Little Red Riding Hood with the wolf has been identified as the most significant moment of the fairy tale. Characters are positioned so that one impedes a clear view of the other, suggesting the user to move around the book and leaving the narrative suspended (Will they meet? Will the wolf immediately devour Little Red Riding Hood?).

Conceived as a single static scenario, the *tableau* enhances the unity of place, while the forest represented in its entirety suggests the existence of an external environment and allows the introduction of an external world, alien to the classic fairy tale.

2.1.1 Paper Design

The technique of orthogonal supporting planes allows the paper forest to open up and fall on the page compactly [4].

[2] Perception of the external world as well as the awareness of one's own thoughts.



Figure 3. Variation on Little Red Riding Hood - cover

Not of secondary relevance, since the pop-up can stand alone without being fixed to the page, maintenance processes both of the pop-up and of the electronics embedded in the thickness of the page are facilitated.

2.1.2 Graphics

The choice of few solid colors is part of the minimalist aesthetics for which elements should be evoked rather than described: characters stand out in the scene, being the only colored elements; the white base lends itself to suggestive shadows.

The rigid geometry of structure has been circumvented depicting the forest with bare trees, maintaining a naturalistic representation of a tangle of trunks.

A demonstration can be viewed on the website `https://vimeo.com/92193054`

2.2 Sonic Interaction Design

The great revolution, not yet accomplished on a commercial scale, is to apply continuous interaction to movables, namely to make sure that user's gestures do not only trigger the effect but control it, greatly amplifying what is already happening regarding the paper interface (the user acts as director and spectator at the same time).

Continuous interaction fosters a certain complexity, leading the user through slightly different sonic scenarios, each one focusing on imaginary variants of the tale (e.g. the story occurs in a winter forest; the meeting with the wolf happens in a city park; Little Red Riding Hood is a teenager).

Realistic sounds make the communication accessible to everyone, changing from being iconic (easily recognizable) to indexical (encouraging mental associations once placed in relation with one another, they facilitate the user's imagination).

Sounds are related to eight categories:

1. Little Red Riding Hood's actions;
2. Little Red Riding Hood's state of mind;
3. wolf's actions;
4. wolf's intentions;
5. animals or people in the forest;
6. landscape;

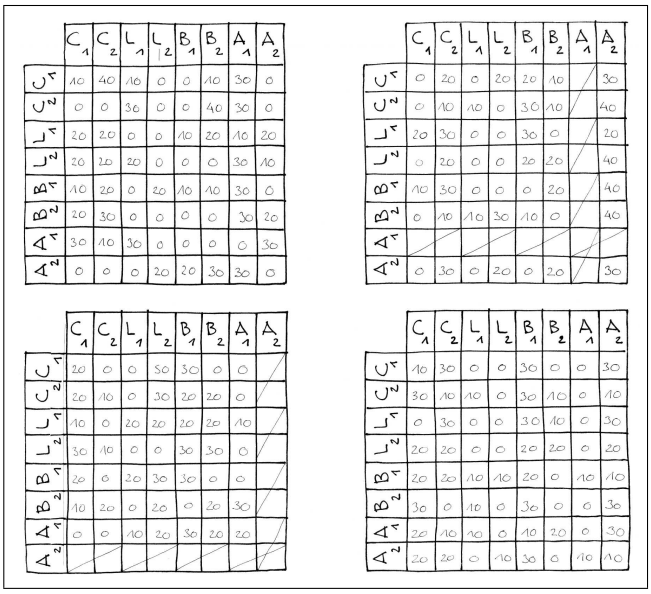


Figure 4. Matrices of transition probabilities governing the composition of different sonic scenarios

7. weather;
8. external elements.

Six scenarios are implemented through matrices of transition probabilities which help interleaving the samples belonging to eight sound categories. A sonic display with varying degrees of unpredictability is composed.

To switch from one scenario to the next it is sufficient to close and open the book again.

The embedding of eight miniature speakers in the book greatly increases the unity and coherence of experience. The very low amplitude of sounds contribute to the sense of privateness, so as to an ecology-aware listening.

3. TECHNICAL NOTES

The size of the open book is 140x100x50 cm. Materials used are black/white/red/grey cellulose paperboard, foamboard, black fabric, straps and tape. One analog electrical circuit is embedded in the foamboard as well as eight high-sensitivity miniature loudspeakers equipped with rare-earth magnets and eight light dependent resistors. Cables run in grooves digged into the foamboard.

The structure of the book consists of 14 disassemblable foamboard panels. The pop-up and the paperboard surfaces are disassemblable as well. This design, which is due to the generic purposes of maintenance and transport, may become in the future a specific additional characteristic.

The artifact is connected to a computer, which contains the sounds to be projected by the movable and performs any necessary algorithm for managing the interaction of the users.

3.1 Embedding of the Electrical Circuit

The main concern in prototyping the artifact were the components to be embedded and the devices to be admitted.

At first sight, the purpose of moving the sound around the surface of the movable might imply a multichannel audio interface, and retrieving data from sensors, a microcontroller to be connected to the computer. Both devices may be embedded in the surface of the page allowing a better user experience. Though, an alternative solution was adopted, envisaging the movable as an extremely simple plug-and-play device containing just the circuitry for sensing and projecting the sound, without any other audio "gear" being placed around. The movable resulted finally a completely passive analog device, being connectable to any computer or digital device providing a 96 kHz audio input/output and a USB port.

Since most of the embedded sound cards in computers and devices are based on chips supporting full duplex analog 96 kHz stereo audio, the idea was to interleave up to eight different audio channels inside the computer and to stream one single amplified analog output into the movable. In this way, the signal has to be demultiplexed in the movable itself before being fed to the embedded loudspeakers. The signals of the embedded sensors have to be multiplexed accordingly, before being fed to the computer as a single analog audio source.

Each of the eight audible streams has thus up to 6 kHz of bandwidth, which is more or less sufficient for the specific purposes of the project. The sensed signal has in turn much more bit depth than the most widespread microcontrollers used in installations, prototypes and artifacts.

The multiplexing-demultiplexing circuit embedded in the movable needs a synchronizing signal which is provided by the computer itself as a second audio output stream. The circuit is powered either by battery or through a USB cable by the computer. In the prototype, both the audio streams and the 5 V supply are fed through one shielded USB cable interfacing the computer with a "fake" USB port placed on the movable's side.

The audio levels are extremely low, for most sound cards do not provide much current/voltage at the headphone output. A cheap external sound card with a high-gain headphone amplifier resolved hitherto the issue. As a future work, a boosting amplifier may be embedded in the artifact, being powered by the same 5 V supply of the mux-demux circuit [3] .

3.2 Sensing and Interaction

Despite of the much more widespread IR sensors, the passive LDRs net provided a surprisingly high accuracy in retrieving the shadows projected by the users moving around the artifact and interacting. The LDRs also had the advantage of being completely invisible under the white paperboard surface of the book. The main disadvantage is that the thresholds implied by the algorithms managing the interaction have to be imposed experimentally or through further learning algorithms (which at the time of writing have not been tested sufficiently).

The bit depth of the digitized signal allows anyway to discriminate the most subtle difference. An advanced peak-picking algorithm permits to correctly estimate the position of up to two people moving around the book and, if the users are quite isolated from each other, up to three people.

In the implemented prototype, a new sound is activated when a new peak is detected beyond the specified threshold; the sound is spatialized continuously according to the peak's displacements - i.e. the sound is "following" the user. Once a peak remains beyond the threshold for a while, sounds succeed one another overlapping more or less irregularly, as if a quite linear story was told. This happens when a user is remaining still and very close to the book's surface.

New sounds may be activated also when a peak, which is already detected, modulates in its amplitude. This happens when a user is continuously bending or moving. Also in this case the spatialization is dynamic but independent from the user's displacements.

The latter interaction is opposite to the former and has the purpose of thickening and complexifying the sonic texture, especially when more than one person is interacting and more peaks are detected.

The activity of the LDRs is monitored also to detect the opening or closing of the book, thus triggering a new matrix of transition probability.

4. FIRST EVALUATIONS AND FUTURE WORKS

The artifact has hitherto been exhibited in conferences such as DRHA 2014 [4] and the joint SMC-ICMC 2014 [5] for several hours a day.

The changing of the lighting due to weather and other factors had no impact on the operation, showing that the simplest technology such as, for instance, light dependent resistors may be actually used for specific purposes like those described in the paper.

As regards the interaction, first evaluations show that several users approach the book with rather deictic gestures towards the wood and the two characters, which is not precisely what one may expect: the context of the exhibition and the knowledge of specific sensing techniques may have furthered that. The low amplitude of sounds encouraged to bend close to the surface of the page for a better perception and localization, which consequently allowed to discover a more fruitful behaviour. This is not, however, generalizable and seems once more to be context-dependent: when the artifact is conceived by the users more as an installation, the concept of telling a story remains in the background - as if the interaction was there for its own sake.

At the time of writing, the work is being presented within an exhibition of movables and artist books in the frame of a historical retrospective on World War I. On this occasion, user experience design is going to be more extensively verified, for the artifact is being presented to a much more broader population.

As a future work, a more effective storytelling will be implemented through alternative ways of real-time audio editing and compositing.

[3] Just the contrary, if the audio power is high enough, no amplification is needed and the mux-demux circuit may be powered by a current obtained from the synchronizing signal.

[4] http://www.drha2014.co.uk

[5] http://www.icmc14-smc14.net

5. REFERENCES

[1] J. Drucker, *The Century of Artists' Books*. New York (NY): Granary Books, 2004.

[2] D. Rocchesso, S. D. Monache, J. Qi, L. Buechley, A. D. Götzen, and D. Cestaro, "Paper mechanisms for sonic interaction," in *Proceedings of the international conference on tangible, embedded and embodied interaction*, 2012.

[3] J. Nielsen and R.L.Mack, *Heuristic evaluation*. New York (NY): Wiley, 1994.

[4] P. Franchi, *Apriti libro! Meccanismi, figure, tridimensionalità in libri animati dal XVI al XX secolo*. Essegi, 1998.

SOME DIFFICULTIES INHERENT IN THE PROJECT OF AN ARTIFICIAL MUSICAL EXPERIENCE

Stefano Lombardi Vallauri
IULM University – Milan
stefano.vallauri@iulm.it

ABSTRACT

In the first part the article sets and discusses from a pragmatic and psychological point of view the foundations of human musical experience. In the second part it considers the difficulties of the construction of an artificial musical experience at the various levels (perception, cognition, emotion, memory and anticipation, subjectivity, and consciousness).

1. HUMAN EXPERIENCE

In order to design an artificial musical experience, i.e. made by a non-human being built by man, we need a preliminary definition of what is a human musical experience, on which the artificial one would model itself (I neglect here the question of another kind of non-artificial musical experience, i.e. natural, but not human, that is the animal one).

I set as a founding category of our reasoning (in an axiomatic way, giving up for the sake of brevity the critical treatment it deserves) the concept of *authentic musical experience*. At the very least, I define it as a listening and a «basic musical understanding» [8: 28] on the part of a subject, who can follow the sound events as they successively occur, grasping their sense and mutual relationships albeit intuitively and at an elementary formal level, as well as catching and responding to their emotional qualities. Therefore I oppose it to what the authentic musical experience is according to other theoreticians, namely an «expert», analytical listening [1: 181-182], or one motivated mainly towards a formal aesthetic appreciation [6].

According to a summary pragmatic distinction, there exist three fundamental kinds of musical activity: composition, execution (improvisation incorporating these two), and listening. Listening is the essential, nuclear musical experience, and indeed it necessarily belongs also to the other activities, whether authentic (I do not consider as an authentic musical experience the exclusively motor execution of a piece, performed without hearing it; or composition as a purely thetic-combinatory work, realised without hearing or at least imagining sounds with the inner ear).

Copyright: © 2014 Lombardi Vallauri. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

On a first plane, in musical listening three kinds of human psychological faculties are involved: perception, cognition, and emotion. They are involved in part separably, but mostly together in an integrated, synergic manner: what produces an emotional state, a psychophysical state in the subject, is something that in the meantime the subject also perceives, and knows as a form of sound. Anyway the integrated involvement of the three kinds of faculty follows a necessary order: there can be no listening without perception; there can be cognitive understanding without emotional reaction; there can be no emotional reaction to music without a cognitive – at least intuitive – understanding of it.

On different planes that intersect the one just considered, other psychological faculties are involved in the experience of musical listening. First, memory and imagination. During listening the grasping of what happens in the present moment always takes place in the interaction with the memory of what has just happened and with the anticipation of what is about to happen (in the Husserlian phenomenological terminology, with «retention» and with «protention» [3: 114-117]). Furthermore, long-term memory is also involved, since the subject is the bearer of a musical and general competence-culture, with regards to which the piece he hears in the particular case is always referred.

Finally, and fundamentally, musical listening involves the psychological faculty of subjectivity, the sense of self: the sense – that the subject himself feels – of being a subject. Subjectivity is not required in all human psychological activities, instead it is never missing in the authentic musical experience, which is always made in reference to an object, the sounds, but by a subject as a subject: it is *I*, irreducibly I, who listen to these sounds, who in a present-extended moment know them at least intuitively and react emotionally.

2. ARTIFICIAL EXPERIENCE

The inseparable link between perception, cognition, emotion, memory and imagination, and self-conscious subjectivity leads to the hypothesis that the project of an authentic musical experience requires the construction not only of an artificial intelligence but also, more specifically and more arduously, of an artificial *consciousness*. The authentic musical experience, which integrates all these faculties in a global act of consciousness, activates man in his way of being more human, different from both a computer and an animal: because, for example, there is

no rational subjective consciousness well-functioning as such without emotion [4], and also there is no precious emotion without subjective consciousness. To date, in fact, there exist computers endowed with rational capacities, but not together with subjectivity; and while the basic emotions (classified as anger, fear, etc.) are probably common also to animals, certain so-called *refined* emotions (still far from being surveyed and classified), including some aesthetic [5] and musical emotions, are exclusively human.

Let us consider – as far as possible, singularly – the various faculties, in the perspective of their possible artificial emulation. Perception of sound, as a physical process, can now easily be engineered: it simply requires an (electro-)mechanical sensor. Indeed, some sensors more sensitive and finer than the human ear do exist. But a pure, neutral perception of sound is not a human thing. Human perception is right from the outset also cognition: what the human sensor, the ear, perceives is never experienced as such by man, at any stage; just at first it is received through psychological categories that make it what it is for the listener (for example, the human listener perceives as a melody with accompaniment and phrasing something that physically, neutrally, is a sequence of different vertical aggregates of different durations). A proper relativism imposes a phenomenological approach to human acoustic perception, because there is no music that even just at the level of the immediately perceived form of sound is what it is for the subject, if not in the cognitively mediated way in which precisely it is for the subject.

The first difficulty in the design of an artificial musical experience lies, therefore, in the emulation of the cognitive mediation of the perceptual datum: the artificial mind must be educated to know sounds and music in the same way the human mind does. Such a difficulty, today not insurmountable, can be overcome through the implementation in the receptive phase of algorithms of grammatical cognition that correspond to the rules through which in the creative phase the «grammar» of a style [2] governs the creation of music itself (a concomitant difficulty, not concerning the sciences of mind but musicology, is in fact the grammaticalisation of styles).

Musical emotion – that is, actually, the integrated plexus perception/cognition/emotion – poses for now, in an attempt of artificial duplication, insurmountable problems. So far we have been able to robotise in part listening as perception and as cognitive grammatical recognition, and even execution and composition, namely also musical activities which require superior and specialist skills; but we have not been able (in fact we are so far from it that the attempt has not been done nor even almost conceived) to robotise that activity, widespread among all common human beings, that is the emotional musical listening. Which is, however, the core of the authentic experience present in all the different musical activities. The mere fact that emotion calls for the human being as a complex mental and *corporeal* whole – endowed not only with a generic physical body, as also a computer is, but properly with nerves, a heart, muscles, a warm excitable flesh: *alive* [10; 7] – constitutes by now an unattainable limit to the efforts of emulation.

Nor we have been able to robotise the other necessary element of the authentic musical experience, the self-conscious subjectivity. In general, regardless of its involvement in a musical experience, the «possibilities of realization» of an artificial consciousness still seem «non-existent» [9: 548]. All the more reason they have to be so in the attempt of its integration within the plexus of the musical experience.

3. REFERENCES

[1] Th. W. Adorno: *Einleitung in die Musiksoziologie* (1962), in *Gesammelte Schriften*, 14, Suhrkamp, Frankfurt am Main, pp. 169-433, 1997.

[2] M. Baroni, R. Dalmonte, and C. Jacoboni: *Le regole della musica: Indagine sui meccanismi della comunicazione*, EDT, Torino, 1999.

[3] C. Dahlhaus: *Musikästhetik*, Laaber-Verlag, Laaber, 1986[4].

[4] A. R. Damasio: *Descartes' Error: Emotion, Reason, and the Human Brain*, Putnam, New York, 1994.

[5] K. Higgins: "Refined emotion in aesthetic experience. A cross-cultural comparison", in R. Shusterman and A. Tomlin (eds.), *Aesthetic Experience*, Routledge, New York, pp. 106-126, 2008.

[6] P. Kivy: *Introduction to a Philosophy of Music*, Clarendon, Oxford, 2002.

[7] M. Leman: *Embodied Music Cognition and Mediation Technology*, MIT Press, Cambridge (MA), 2007.

[8] J. Levinson: *Music in the Moment*, Cornell University Press, Ithaca and London, 1997.

[9] G. Trautteur: "Coscienza naturale, coscienza artificiale, coscienza autentica", *Sistemi intelligenti*, XX/3, pp. 547-550, dicembre 2008.

[10] F. J. Varela, E. Thompson, and E. Rosch: *The Embodied Mind: Cognitive Science and Human Experience*, MIT Press, Cambridge (MA), 1991.

VID/AO
VIDEO IMAGING DIGITAL/ANALOG OPERATOR

Martino Sarolli
Conservatorio "Giuseppe Verdi" Torino
mmsarolli@gmail.com

SOMMARIO

Descrizione del processo di realizzazione di uno strumento per la composizione di immagini animate mediante visualizzazione di segnali analogici controllati digitalmente.

1. INTRODUZIONE

Il progetto nasce dal desiderio di poter sperimentare le possibilità della videosintesi analogica, alla luce delle nuove possibilità di combinazione, interazione e controllo offerte da ambienti di sviluppo come Max/MSP e microcontroller Arduino, con lo scopo di creare un sistema ibrido analogico/digitale per la generazione e manipolazione di immagini animate.

Con il termine "videosintesi" si fa riferimento ad una delle prime tecniche di immaginazione elettronica nate agli albori della video arte (circa 1965-1970) [2,7], caratterizzata dall'impiego di sistemi normalmente utilizzati per la sintesi e l'elaborazione del suono con lo scopo di generare segnali video contenenti variazioni dinamiche di forme e colori. (Figura 1).



Figura 1. Esempio di pattern videosintetizzati dallo strumento – Modulazioni di sinusoidi ed impulsi [1]

Copyright: © 2014 Martino Sarolli. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

[1] Link al video: https://vimeo.com/98926400.

2. TECNICA DI VISUALIZZAZIONE DEL SEGNALE

Individuare un protocollo di trasmissione di segnali video analogici adatto ad essere *hackerato*, ovvero scomposto e modificato secondo le proprie necessità, è stato il primo passo da compiere verso la realizzazione dello strumento.

Il formato video VGA offre le caratteristiche ideali: facilmente scomponibile, ricomponibile, tuttora largamente diffuso e quindi disponibile come porta d'ingresso su schermi e proiettori anche di recente fabbricazione, nonostante qualitativamente ormai superato da nuovi formati ad alta definizione.

Utilizzando due cavi di conversione da VGA a BNC (*Bayonet Neill Concelman* - connettori unipolari a baionetta) connessi tra loro è possibile isolare le cinque componenti del segnale VGA: due per la sincronizzazione con il dispositivo di visualizzazione (V-Sync per la sincronizzazione verticale e H-Sync per quella orizzontale) e tre per il controllo dei colori (RGB – *Red, Green, Blue*).

Mantenendo connesse le componenti di sincronizzazione verticale ed orizzontale di un segnale video proveniente dall'uscita VGA della scheda video di un computer e scollegando le connessioni RGB è possibile interrompere esclusivamente la trasmissione delle componenti per il controllo dei colori, trasformando i connettori RGB del cavo n. 2 (Figura 2) in tre ingressi RGB.

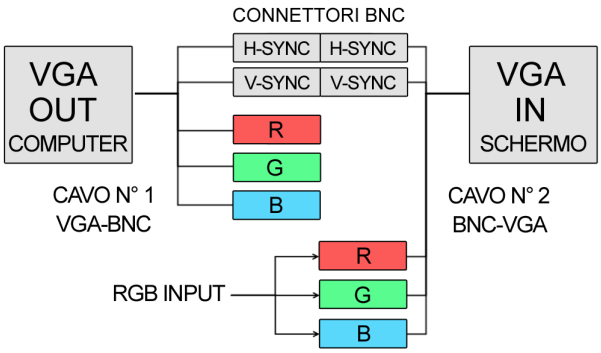


Figura 2. Schema connessioni RGB input.

L'idea alla base dello strumento è quella di generare attraverso tecniche di sintesi digitali e analogiche nuove componenti RGB da inserire all'interno del segnale, sostituendosi a quelle "originali", senza alterare le componenti di sincronizzazione necessarie per rendere il segnale video compatibile con le periferiche di visualizzazione (monitor, proiettori e televisori).

3. GENERAZIONE E CONTROLLO DI SEGNALI VIDEO

Sono previste due modalità di generazione di segnali in relazione alla larghezza della banda frequenziale ottenibile:

- Fino a 96 kHz: segnali digitali generati in ambiente Max/MSP e convertiti in analogico a frequenza di campionamento di 192 kHz
- Da 96 kHz a 1 mHz: segnali analogici generati da oscillatori video controllati digitalmente tramite microcontroller Arduino

Le diverse tipologie ottenute vengono combinate all'interno di un mixer analogico le cui uscite vanno a costituire le componenti RGB del segnale VGA visualizzabile.

L'impiego di moduli analogici controllabili via software permette di realizzare immagini completamente programmabili, memorizzabili e riproducibili, sia automaticamente, realizzando partiture di esecuzione, che in tempo reale attraverso periferiche e superfici di controllo (tastiera, mouse, MIDI controller ecc.).

La presenza all'interno del sistema di un microcontroller sensorizzabile e di convertitori ADC (*Analog-to-Digital Converter*) permette l'acquisizione di informazioni esterne utilizzabili, attraverso differenti modalità di correlazione, come parametri per il controllo delle caratteristiche di forma, colore e movimento delle immagini (Figura 3).

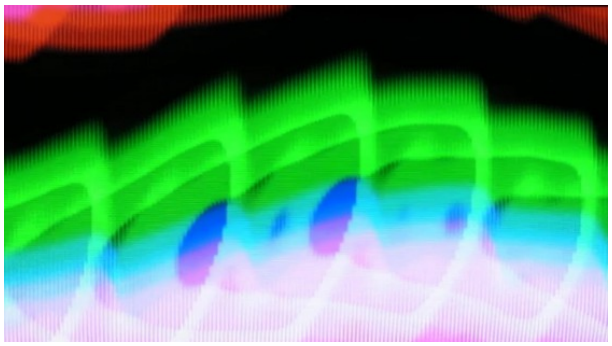


Figura 3. Esempio mapping audio-video [2]

4. SISTEMI DI RIFERIMENTO

La struttura ed i principi di funzionamento dello strumento traggono ispirazione e fanno riferimento principalmente a due sistemi: il *Direct Video Synthesizer* di Stephen Beck (1970) [1,2,5] per la tecnica di videosintesi e il Sistema *GROOVE* di Max Matthews e F.R. Moore [4] per il controllo digitale di segnali analogici.

Il *Direct Video Synth* di Beck utilizza segnali di controllo in voltaggio generati da un sintetizzatore Buchla per pilotare, all'interno di un tubo catodico, i cannoni elettronici RGB che generano i tre fasci di elettroni responsabili della colorazione dell'immagine.

Il sistema ibrido *GROOVE* (acronimo di *Generating Realtime Operations On Voltage-controlled Equipment*) sviluppato presso i Bell Laboratories nel 1970, consisteva di una serie di moduli analogici per la sintesi e l'elaborazione del suono, i cui parametri di evoluzione temporale venivano controllati da una combinazione di funzioni algoritmiche, programmate attraverso un calcolatore, e interfacce per il controllo in tempo reale.

5. APPLICAZIONI FUTURE

- Approntare una versione dello strumento adatta all'improvvisazione audiovisiva attraverso il controllo in tempo reale
- Realizzazione di una partitura di esecuzione per una composizione videomusicale automatica
- Installazione audiovisiva interattiva [3,6]

6. BIBLIOGRAFIA

[1] S. Beck: "Beck Direct Video Synthesizer" http://people.wcsu.edu/mccarneyh/fva/B/BeckDirectVideo.html.

[2] T. Dreher: "Video Synthesizers" in *History of Computer Art*, Cap. IV.1.2 http://iasl.uni-muenchen.de/links/GCA-IV.1e.html.

[3] M. M. Gazzano: *Steina e Woody Wasulka Video, Media e Nuove Immagini Nell'Arte Contemporanea*, Fahrenheit 451, Roma, 1995.

[4] L. Spiegel: "Graphical GROOVE: memorial for the VAMPIRE, a visual music system" in *Organized Sounds*, 3(3), pp. 187-191, Cambridge University Press, 1998.

[5] S. Wasulka, W. Wasulka: *Eigenwelt der Apparate-Welt: Pioneers of Electronic Art*, Ars Electronica, Linz, 1992.

[6] W. Wasulka: *The Brotherhood, A Series of Six Media Constructions*, NTT InterComunication Center, Tokyo, 1998.

[7] G. Youngblood: *Expanded Cinema*, (1970), CLUEB, Bologna, 2013.

[2] Link al video: https://vimeo.com/98926401.

NVOICES: GENDY E IL LIVE ELECTRONICS

Giuliano Anzani
Conservatorio di Musica "G. Verdi" di Como
giuliano.anzani@gmail.com

SOMMARIO

nVoices è un'interfaccia di controllo indirizzata al live electronics per *GENDY*, l'algoritmo di sintesi stocastica ideato da Xenakis.

Lo studio effettuato sull'algoritmo si è concretizzato nella sua implementazione in Pure Data e alla successiva progettazione di una grafica di controllo specifica per la manipolazione di *GENDY*, con l'obbiettivo di produrne una versione fruibile e aperta a nuove modifiche e/o migliorie.

nVoices è uno "strumento elettroacustico" composto da un'interfaccia fisica (un controller MIDI) per l'immediata manipolazione dei parametri di sintesi e da un'interfaccia grafica che controlla nel dettaglio la produzione dei suoni.

1. INTRODUZIONE

Nel 1971, al CMAM (*Center for Mathematical and Automated Music*) dell'Università dell'Indiana, Xenakis sperimentò i suoi studi sulla sintesi stocastica testando diversi tipi di random walk.

Nel 1991 Xenakis tornò sulla costruzione di un sistema completamente governato dalle leggi della stocastica e, grazie alla collaborazione del CEMAMu (*Centre d'Étude de Mathématique et Automatique Musicales*) di Parigi, il compositore, insieme all'assistente Marie-Hélène Serra, scrisse il programma di sintesi stocastica dinamica GENDY, acronimo dal francese *Génération* (GEN) e *Dynamique* (DY).

Nello stesso anno egli compose due brani sfruttando questo algoritmo: *GENDY3*, premiato nell'ottobre del 1991 a Montreal in Canada, in occasione dell'*International Computer Music Conference*, e *GENDY301*, premiato nel novembre del medesimo anno a Metz in Francia, al *Journèes de Musique Contemporaine*.

Nel 1994, sfruttando un'estensione dell'algoritmo *GENDY*, Xenakis scrisse un nuovo lavoro dal titolo *S709*, premiato nel dicembre di quell'anno alla Maison de Radio-France.

Il software *GENDY* permette il calcolo di una serie di campioni numerici che vengono di seguito registrati in un file audio. Ciascuna voce calcolata è caratterizzata dai propri parametri di controllo per la sintesi stocastica dinamica e ci possono essere da una a n voci.

Il processo computazionale è suddiviso in due livelli:

- microstrutturale, che comprende l'effettiva sintesi dei suoni;
- macrostrutturale, in cui sono definiti i parametri per forma e computazione audio.

A livello macroscopico si produce una matrice in cui sull'asse verticale giace la distribuzione delle voci generate, mentre sull'asse orizzontale si ha la suddivisione temporale in cui viene collocato un certo numero di sequenze che andranno a costituire l'intera durata del brano. Ogni voce corrisponde ad una singola sintesi sonora, con determinati parametri stocastici, a loro volta gestiti da intervalli di tempo tra suono e silenzio, definiti "campi" (*fields*). La sovrapposizione di un determinato numero di voci determina la configurazione di una sequenza, mentre la giustapposizione in successione di ogni sezione andrà a costituire il corpo stesso della composizione.

Xenakis implementò questa funzione in un programma ausiliario dal nome *PARAGn* (es. PARAG001), una macchina di calcolo per la generazione di parametri stocastici e la loro scrittura in unico file successivamente letto da *GENDY* per l'effettiva sintesi audio.

2. L'ALGORITMO *GENDY*

In *GENDY* la sintesi sonora corrisponde ad una sequenza Z di diverse tabelle di sintesi, in cui ciascuna di esse è variata dalla precedente per mezzo di un processo stocastico (es. *random walk*).

Ciascuna tabella di sintesi corrisponde ad una forma d'onda che viene considerata come il profilo di un poligono aperto, i cui vertici corrispondo ai breakpoints e i lati rappresentano l'interpolazione tra i punti.

Ponendo la forma d'onda su un piano cartesiano avremo: sull'asse delle ascisse gli indici dei campioni di una tabella di sintesi e sull'asse delle ordinate i valori dell'ampiezza che variano tra -1 e 1.

Ciascuna forma d'onda è caratterizzata da un numero costante I di breakpoints e, per garantire la continuità tra le successive forme d'onda, l'ultimo breakpoint $(I-1)$ della forma d'onda z deve corrispondere al primo $(i'=0)$ della forma d'onda successiva $z+1$ (Figura 1), le cui coordinate di ciascun breakpoint i' si ottengono addizionando o sottraendo alle coordinate della forma d'onda precedente un valore aleatorio determinato da un processo stocastico:

$$x_{i,j+1} = x_{i,j} + fx(k) \quad (1)$$

$$y_{i,j+1} = y_{i,j} + fy(k) \quad (2)$$

Copyright: ©2014 Giuliano Anzani. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

dove $fx(k)$ e $fy(k)$ sono funzioni stocastiche che generano valori determinati dall'argomento k. La durata totale D di una forma d'onda z è calcolata sommando i segmenti n_i di interpolazione che si creano tra due breakpoints i e $i+1$. Si avrà:

$$n_i = x_{i+i} - x_i \tag{3}$$

$$d_i = n_i / SampleRate \tag{4}$$

$$D = \sum_{i=0}^{i-1} d_i \tag{5}$$

con
n_i = distanza in campioni;
d_i = distanza in secondi;
D = durata totale espressa in secondi.

Se le coordinate sull'asse x non sono soggette a nessun tipo di variazione, si ottiene un'onda di uguale durata e ampiezze variate non-linearmente. Questo comporterà alterazioni delle componenti spettrali e quindi del timbro ma non dell'altezza. Nel caso in cui siano entrambe le coordinate x e y ad essere variate nel tempo, l'alterazione si produrrà sia a livello spettrale che a livello frequenziale.

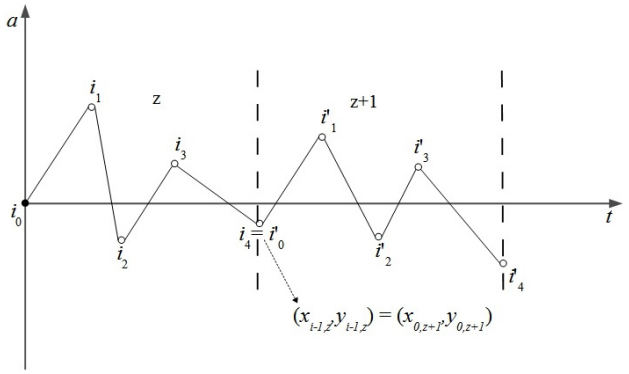


Figura 1. Alterazione dei breakpoints i nelle forme d'onda z e $z+1$.

In *GENDY*, l'articolazione strutturale a livello macroscopico, è definita da una serie di sequenze Z sovrapposte, per cui ciascuna di esse rappresenta una voce di un insieme di n voci.

Sull'asse temporale, ogni sequenza è caratterizzata da porzioni temporali di suono e silenzio, definiti da Xenakis "campi temporali" (*time-fields* [9]). Un campo temporale è definito da due parametri: la durata (sec.) e l'indicatore silenzio/suono (0/1).

La scelta tra suono e silenzio viene determinata computazionalmente tramite una simulazione del processo di Bernoulli che permette di stabilire il successo o il fallimento di un processo aleatorio, dove p è la probabilità di successo che corrisponde ad un numero tra 0 e 1, e q corrisponde al fallimento di probabilità uguale a $1-p$. L'esempio più comune è il gioco della moneta "testa o croce".

La durata dei campi temporali viene stabilita automaticamente attraverso una distribuzione esponenziale:

$$d = -\left(\frac{1}{d_m}\right) \log(1-w) \tag{6}$$

dove w è un numero equi-distribuito tra 0 e 1, d è la durata di ciascun campo sonoro e d_m è la media delle durate. Quest'ultimo parametro viene stabilito per ogni voce in ciascuna sequenza.

3. IMPLEMENTAZIONE DI GENDY IN PURE DATA

La prima versione del programma *GENDY* fu scritta da Xenakis nel 1991 in linguaggio BASIC (eseguito in MS-DOS). Una trascrizione in C++ di tale algoritmo, chiamata *The New GENDYN Program* (eseguibile in Microsoft Windows), fu poi realizzata nel 2000 dal musicologo tedesco Peter Hoffmann [2].

L'implementazione di GENDY ha avuto come punto di partenza la ricostruzione in Pure Data di un sintetizzatore stocastico basato sull'algoritmo di Xenakis, da parte di Gordan Kreković e Davor Petrinović [4].

Allo scopo di ottenere una versione più fedele rispetto al software originale ideato da Xenakis, sono stati modificati due aspetti fondamentali: la gestione delle tabelle di sintesi e il controllo dei parametri. Nella sezione dedicata alla gestione delle tabelle di sintesi, come funzione stocastica, è stato riprodotto il sistema di perturbazione utilizzato da Xenakis (*second order random walk* [5]), che consiste nell'utilizzo di una funzione di distribuzione e una coppia di random walk.

Una random walk consiste in un processo stocastico non-lineare, un esempio può essere l'oscillazione di una particella nell'intorno di un punto di equilibrio. In questo caso possiamo interpretare una random walk come un accumulatore di numeri casuali.

Le funzioni di distribuzione inserite sono le medesime utilizzate dal compositore greco in origine, che sono: Cauchy, esponenziale, logistica, triangolare, uniforme, arcoseno e coseno iperbolico.

Il processo di una *second order random walk* (Figura 2), può essere schematizzato in sei passaggi:

1. viene generato un valore casuale in base alla specifica funzione di distribuzione utilizzata, vale a dire lo *step size* della prima random walk (*RW1*);

2. il valore prodotto viene accumulato con il valore precedentemente generato;

3. il risultato è limitato entro un range prestabilito attraverso una coppia di "barriere elastiche" (*mirror*), il quale diventa lo *step size* della seconda random walk (*RW2*) ;

4. il valore prodotto è sommato a sua volta in un secondo accumulatore;

5. il risultato è circoscritto come per la RW1;

6. quest'ultimo è l'effettivo prodotto della *second order random walk*.

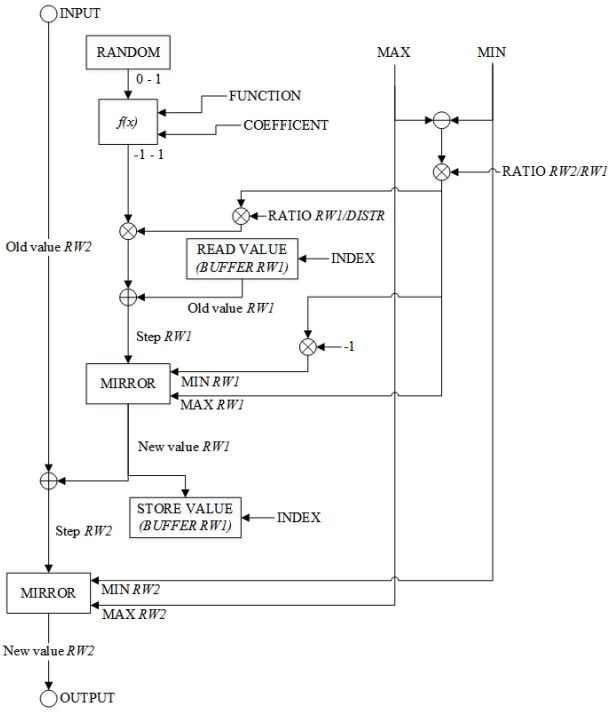


Figura 2. Diagramma a blocchi del funzionamento della *second order random walk*.

I parametri di controllo per la second order random walk sono:

- *function*: permette di selezionare il tipo di distribuzione desiderata (Cauchy, arco seno, coseno iperbolico, logistica, esponenziale e uniforme);
- *coefficent*: controlla il coefficiente della funzione di distribuzione;
- *max*: imposta il limite superiore della funzione *mirror* collegata alla *random walk secondaria* (*RW2*), ossia il valore massimo generato dall'algoritmo;
- *min*: indica il limite inferiore della funzione *mirror*, come per la variabile *max*, determina quindi il valore minimo che può essere generato;
- *ratio RW2/RW1*: è una variabile che varia tra 0 e 1, definisce il rapporto tra il range della *RW2* e quello della *RW1*. Determina il limite superiore e inferiore della funzione mirror connessa alla *random walk primaria* (*RW1*);
- *ratio RW1/DISTR*: è una variabile che varia tra 0 e 1, definisce il rapporto tra il range della *RW1* e quello della funzione di distribuzione $f(x)$. Questo parametro circoscrive la funzione di distribuzione all'interno del range calcolato.

La funzione *mirror*, descritta in Figura 3, ha il compito di mantenere i valori all'interno dell'intervallo stabilito. Viene verificato che il valore x, generato dalla *RW1* o dalla *RW2*, sia contenuto nell'intervallo; nel caso in cui questa condizione non venga soddisfatta, si procede al ribaltamento

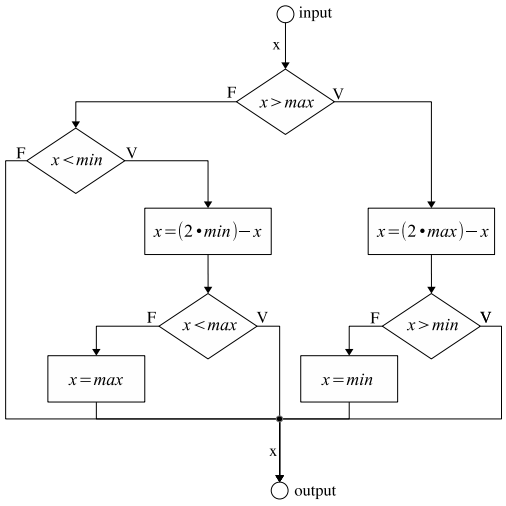


Figura 3. Diagramma a blocchi della funzione *mirror*.

di x all'interno di esso. Nella possibilità ulteriore in cui l'operazione di ribaltamento abbia posto il valore al di fuori dei limiti, si procede alla riflessione di x all'interno di esso, forzando eventualmente il valore all'interno dei limiti per evitare una ricorsione infinita. [1]

Determinata la posizione di ciascun breakpoint, i campioni che vengono a trovarsi tra i due breakpoint sono calcolati tramite una funzione d'interpolazione lineare, attraverso la formula:

$$f(x) = x_a + x_n\left(\frac{y_b - y_a}{x_b - x_a}\right) \tag{7}$$

con $x_a \leq x_n \leq x_b$.

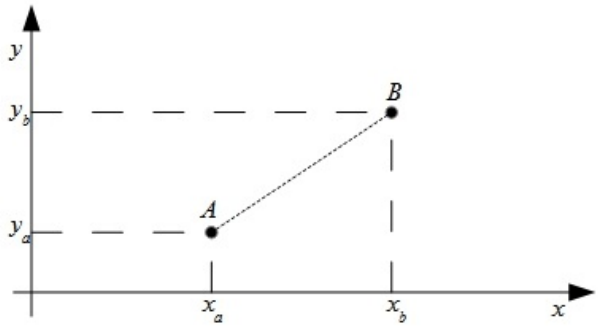


Figura 4. Esempio grafico interpolazione lineare.

4. GENDY NEL LIVE ELECTRONICS

L'implementazione in Pure Data ha portato alla progettazione di un'interfaccia costruita ad hoc per l'algoritmo *GENDY*, denominata *nVoices*, pensata come se fosse un vero e proprio "strumento musicale" utilizzabile nel live electronics.

Oltre alla possibilità di controllare i parametri interagendo direttamente con il software, che però non garantisce

[1] Questo fenomeno in informatica viene definito stack overflow, avviene quando è richiesto una quantità troppo elevata di memoria nello stack, può essere causato da una ricorsione infinita di una funzione o da una variabile eccessivamente grande.

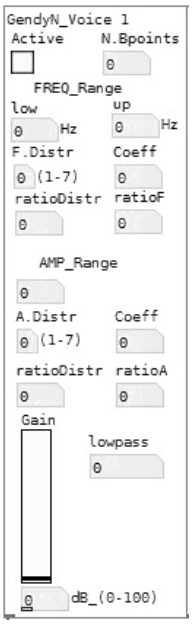


Figura 5. GUI di *GENDY* in Pure Data per il controllo di una singola voce.

un'efficace manipolazione dei dati ai fini di un'esecuzione musicale, è stata aggiunta un'unità fisica di controllo MIDI.

Il numero di parametri utilizzati dal sintetizzatore *GENDY* è stato inoltre ridotto: il totale controllo da parte dell'esecutore costituirebbe un impedimento ad una compiuta interpretazione musicale. Basti pensare che per ciascuna voce, solo per quanto riguarda la sintesi, esistono quindici parametri.

Il passo successivo è stato quello di suddividere le variabili manipolabili in tre gruppi principali, definiti per:

- *altezza*: comprende il range frequenziale e i coefficienti delle due random walks;
- *dinamica*: include l'intervallo di distribuzione dell'ampiezza e i coefficienti delle due random walk;
- *attivazione delle voci*: gestisce il funzionamento delle voci, ossia quando una voce deve o non deve entrare in funzione.

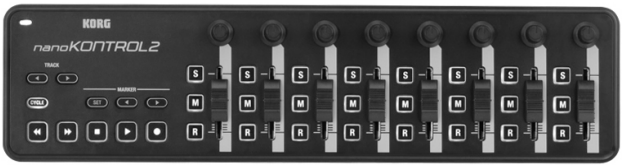


Figura 6. Controller MIDI *nanoKONTROL2 KORG*.

Di seguito viene descritta la progettazione dell'interfaccia fisica sul controller MIDI *nanoKONTROL2 KORG*.

L'interfaccia *nanoKONTROL2* include tre tipologie di controlli: un fader, una manopola rotativa e tre pulsanti per ogni voce. A ogni fader è stato associato il controllo dell'ampiezza di una voce. A differenza dell'algoritmo di Xenakis che utilizzava sedici voci, si è preferito mantenere il controllo per ognuna; in questo caso l'hardware MIDI selezionato permette la gestione contemporanea di otto istanze del sintetizzatore.

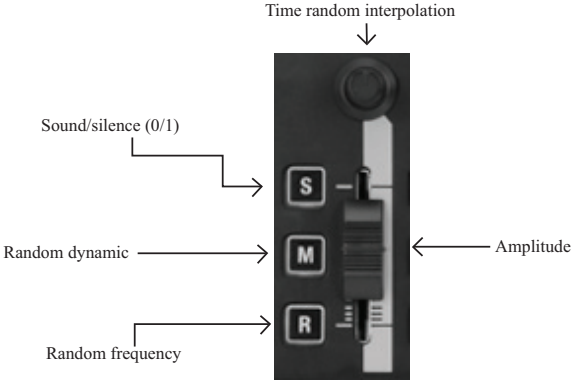


Figura 7. Descrizione dei parametri controllabili per ogni voce nel controller MIDI *nanoKONTROL2*.

Due pulsanti del controller variano le altezze e le dinamiche di ogni voce (tasto "R" per le altezze e tasto "M" per le dinamiche). Ogni parametro è derivato da un processo casuale, prodotto tramite la pressione dei tasti sopracitati, i cui limiti sono stati in precedenza definiti. Al pari di quanto accadeva nell'algoritmo composto da Xenakis, anche qui non interessa avere una definizione microscopica per ogni singolo parametro, ma il controllo del loro insieme. Infine, una manopola rotativa regola il tempo d'interpolazione tra i valori di ampiezza e frequenza generati in ogni singola voce; il range viene stabilito a priori.

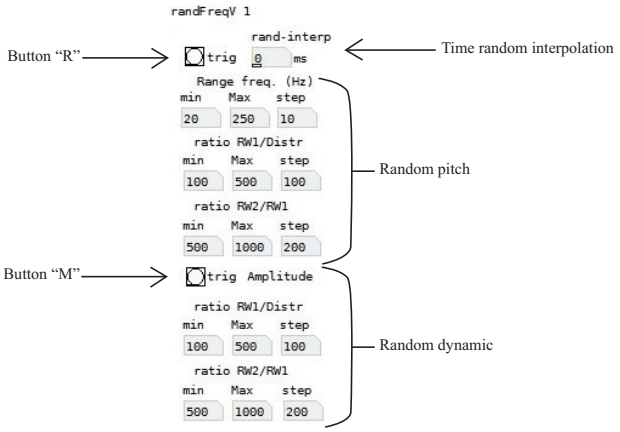


Figura 8. Configurazione del controller MIDI *nanoKONTROL2* nella patch in Pure Data per la generazione casuale dei parametri per una voce.

A livello macroscopico, l'attivazione delle voci è affidata a un'ulteriore serie di tasti: "S" per la singola voce, "Play", "Stop" e "Rec" per controllare le otto voci in contemporanea. "Play" e "Stop" attivano o disattivano tutte le istanze; il tasto "Rec", invece, attiva l'algoritmo responsabile della durata dei campi sonori (*sound fields*), ossia l'attivazione o disattivazione casuale delle voci per una durata d.

Il tasto "Cycle", in combinazione con il pulsante "S" della voce desiderata, permette di separare la relativa voce dall'interazione con l'algoritmo sopracitato, consentendole

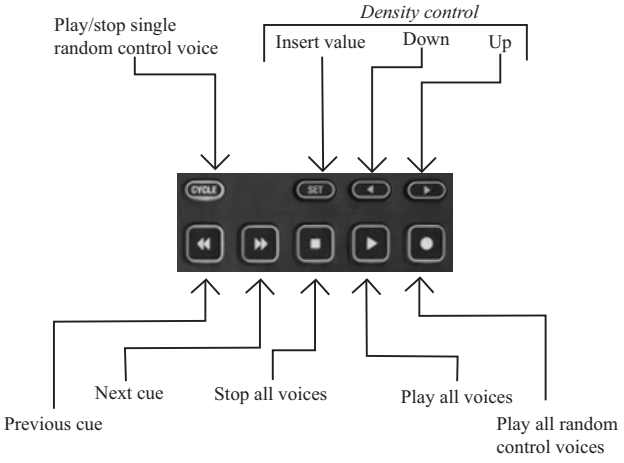


Figura 9. Parametri di controllo per l'attivazione delle voci nel controller MIDI *nanoKONTROL2*.

di rimanere fissa mentre le altre conservano la loro mobilità (gestita dall'algoritmo).

Il pulsante "Set" e le due frecce direzionali, poste alla sua destra, permettono il controllo del coefficiente di densità relativo all'algoritmo di gestione casuale delle voci: la freccia che indica a destra incrementa il coefficiente di un valore pari a 0.1, la freccia che indica a sinistra comporta un decremento della medesima quantità e "Set" inserisce il valore scelto. Il coefficiente di densità varia tra 0 e 1: un valore vicino allo 0 indica una percentuale minore di voci attive, invece un valore vicino all'1 aumenta la probabilità di averne un numero maggiore.

Da ultimo, i tasti "Fast forward" e "Rewind" variano le *cues* dei preset dell'intero software.

Alla spazializzazione del suono (ottofonia) è poi dedicata, in aggiunta all'algoritmo di *GENDY*, un'apposita sezione del software (esterna all'implementazione di *GENDY*) che permette di controllare i movimenti delle voci tramite la definizione di differenti preset (una possibile "partitura" dell'ambiente esecutivo). Questa sezione è stata affidata all'oggetto denominato *vbap*, scritto specificamente per Pure Data, che utilizza il metodo *Vector Base Amplitude Panning* [6] ideato dal finlandese Ville Pulkki, per la disposizione virtuale di una sorgente sonora in un ambiente con n altoparlanti.

5. CONCLUSIONE

In questo articolo si è cercato di riproporre *GENDY* in una nuova chiave di lettura, partendo dalla sua analisi fino ad arrivare alla costruzione dell'interfaccia *nVoices*.

Dal punto di vista musicale, la configurazione dei parametri di *nVoices* ha dimostrato in situazioni live di aver un notevole controllo sonoro sull'algoritmo di sintesi: possono essere generate grandi masse sonore che si deformano fino a ottenere materiale sonoro rarefatto.

Un ulteriore perfezionamento è necessario per calibrare in maniera più equilibrata i parametri variabili in fase di esecuzione e quelli definiti a priori. Sul piano esecutivo, l'utilizzo di controlli multifunzionali amplierebbe l'originalità dei movimenti eseguibili sul controller MIDI, consentendo di modificare più parametri con un singolo gesto e includendo nell'interfaccia fisica un feedback (in questo caso tattile) coerente con l'utilizzo, o meglio ancora, con la musicalità del gesto.

L'interfaccia inoltre farà sì che il sistema di sintesi sonora si comporti come una sorta di strumento musicale, permettendo all'esecutore di interpretare una composizione e di non limitarsi a interagire passivamente con i parametri.

6. BIBLIOGRAFIA

[1] N. Collins, "Implementing Stochastic Synthesis for SuperCollider and iPhone", *Proceedings of the Xenakis International Symposium*, London, 1-3 Aprile 2011.

[2] P. Hoffmann, "The New GENDYN Program", *Computer Music Journal*, 24:2, Summer 2000, pp. 32-38.

[3] P. Hoffmann, *Music Out of Nothing? A Rigorous Approach to Algorithmic Composition by Xenakis*, PhD dissertetion, Technischen Universität Berlin, Berlin, 2009.

[4] G. Kreković and D. Petrinović, "A Versatile Toolkit for Controlling Dynamic Stochastic Synthesis", *Proceedings of the Sound and Music Computing Conference*, Stockholm, Sweden, 2013.

[5] S. Luque, *Stochastic Synthesis: Origin and Extensions*, Institute of Sonology, Royal Conservatory, Amsterdam, 2006.

[6] V. Pulkki, "Virtual Sound Source Positioning Using Vector Base Amplitude Panning", *Journal of the Audio Engineering Society*, vol. 45, n° 6, June 1997, pp. 456-466.

[7] A. Di Scipio, "Da Concret PH a Gendy301 - Modelli compositivi nella musica elettroacusitca di Xenakis", *Sonus - Materiale per la musica contemporanea*, Fascicolo N° 14, Dicembre 1995, pp. 61-92.

[8] M. H. Serra, "Composition and Stochastic Timbre: GENDY3 by Iannis Xenakis", *Perspective of New Music*, Vol. 31, N° 1, Winter 1993, pp. 236-257.

[9] I. Xenakis, *Formalized Music - Thought and Mathematics in Composition*, "Harmonologia Series N° 6", revised edition, Stuyvesant NY, Pendragon Press, 1992.

A KINEMATIC SENSOR TO EXPRESSIVELY CONTROL MUSICAL PROCESSING

Giovanni Costantini

Dep. of Electronic Engineering
University of Rome "Tor Vergata"
Institute of Acoustics and Sensors
"Orso Mario Corbino"
costantini@unirmoma2.it

Massimiliano Todisco, Daniele Casali, Giorgio Nottoli

Dep. of Electronic Engineering
University of Rome "Tor Vergata"
massimiliano.todisco@uniroma2.it
daniele.casali@uniroma2.it
giorgio.nottoli@gmail.com

ABSTRACT

In this paper, we present an innovative gestural sensor interface that allows an electronic music composer to plan and conduct the musical expressivity of a performer. For musical expressivity we mean all those execution techniques and modalities that a performer has to follow in order to satisfy common musical aesthetics, as well as the desiderata of the composer. The proposed sensor interface is able to transform physical parameters in sound synthesis parameters. It is composed by a gestural transducer, that measure motion acceleration and angular velocity, and a mapping module, that transform few physical measured parameters into a lot of specific sound synthesis parameters. In this work, we focus our attention on mapping strategies based on Neural Network.

1. INTRODUCTION

Traditional musical sound is a direct result of the interaction between a performer and a musical instrument, based on complex phenomena, such as creativeness, feeling, skill, muscular and nervous system actions, movement of the limbs, all of them being the foundation of musical expressivity. Actually, musical instruments transduce movements of a performer into sound. Moreover, they require two or more control inputs to generate a single sound. For example, the loudness of the sound can be controlled by means of a bow, a mouthpiece, or by plucking a string. The pitch is controlled separately, for example by means of fingering which changes the length of an air column or of a string. The sound produced is characteristic of the musical instrument itself and depends on a multitude of time-varying physics quantities, such as frequencies, amplitudes, and phases of its sinusoidal partials [1]. The way music is composed and performed changes dramatically [2] when, to control the synthesis parameters of a sound generator, we use input devices such as kinematic, optical or electromagnetic sensors, as well as human-computer interfaces, such as mouse, keyboard, touch screen or gestural control interfaces [3,4].

Copyright: © 2014 Costantini et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

- the quality of a message, in terms of the information it conveys, increases with its originality, that is with its unpredictability;

- information is not the same as the meaning it conveys: a maximum information message doesn't make sense, if any listener that's able to decode it doesn't exist.

A perceptual paradox [6] illustrating how an analytic model fails in predicting what we perceive from what our senses transduce is the following: both maximum predictability and maximum unpredictability imply minimum information or even no information at all. Moreover, musical expressive message is a time variant information which moves between maximum predictability and minimum predictability. Now, it is obvious that the simple one-to-one mapping laws regarding traditional acoustical instruments leave room to a wide range of mapping strategies. In this paper, we discuss a gestural sensor interface. It measures the signals relative to the body movements of a dancer, by means of some kinematic transducers. The acquired signals are then fed to the Neural Network that opportunely controls the sound synthesis parameters, according to the above mentioned gesture mapping strategies.
The paper is organized as follows: first we describe the gestural kinematic sensor and the Neural Network structure, then we describe and illustrate our sensor interface and the mapping strategies adopted; finally, we show a real-time musical application using our sensor interface.

2. GESTURAL TRANSDUCER

The transducer [7] used for the acquisition of kinematic parameters is formed by three mono-axial accelerometers (ACCs) (3031-Euro Sensors, USA) and three rate gyroscopes (GYROs) (Gyrostar ENC-03J-Murata, Japan), that supply motion acceleration and angular velocity and are assembled together and relatively oriented according to an orthogonal reference system. Figure 1 shows the relative orientations of the transducers. The rate gyroscopes measure the components of the angular velocity vector ωx, ωy, ωz of the local frame solid with the body segment where the assembly is affixed. The orientation of the segment is determined by means of the orientation matrix R [8], which is calculated by solving the following differential matrix-equation.

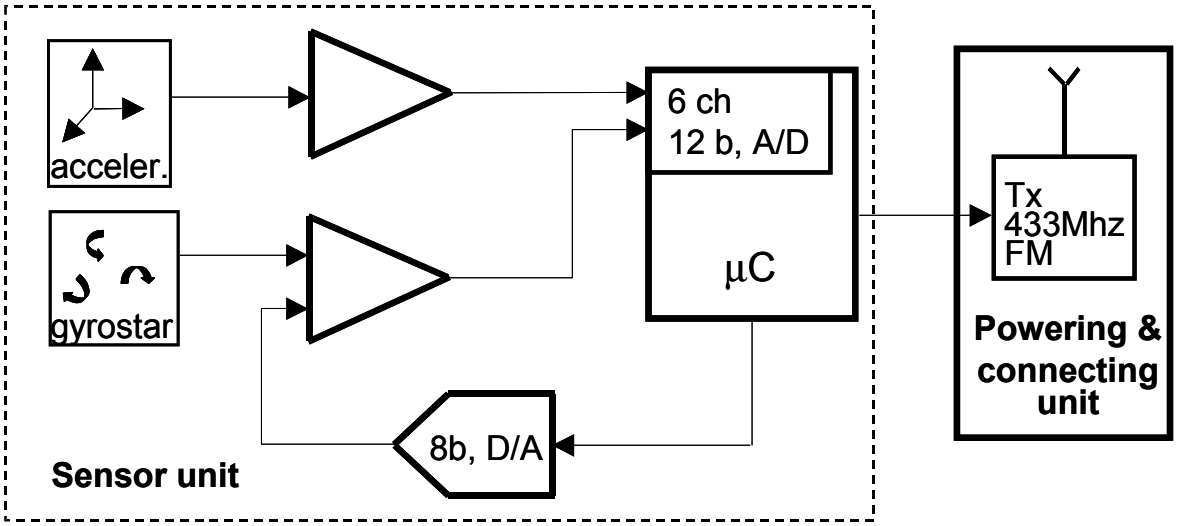


Figure 2. The sensor unit

$$\frac{dR}{dt} = \begin{bmatrix} 0 & -\omega_z & \omega_z \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \cdot R \qquad (1)$$

It was found convenient to express the segment orientation in nautical angles: pitch, roll and rotation (, and). The real instantaneous linear acceleration vector is obtained by equation

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = [R] \begin{bmatrix} a_{x'} \\ a_{y'} \\ a_{z'} \end{bmatrix} - g \qquad (2)$$

where ax, ay and az are the acceleration vectors in the reference system solid with the device.

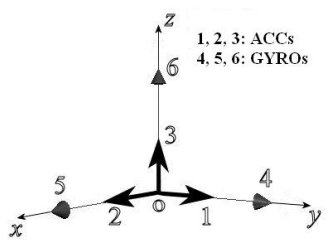


Figure 1. The coordinates of the 3-GYROs & 3-ACCs assembly.

3. KINEMATIC SENSOR

Figure 2 reports the block diagram of the wearable device (WD). It is composed of two separate units: a sensor unit and a powering, connecting unit. The latter contains the battery and the circuitry for communication at a 433 MHz FM radio frequency.

The sensor unit is very small (4x5x2 cm) and light having a weight of 100 g, in order to limit encumbrance to the performer. The circuitry of the wearable device consists of 6 signal conditioning chains (3 for the triaxial accelerometer, 3 for the gyrostars), a microcontroller that features a 12 bit A/D converter and an 8 bit D/A converter.

The WD furnishes acceleration and angular velocity signals, which are low-pass filtered (smoothed) by means of six Sallen-Key cells, amplified and calibrated. Signal processing and calibration parameters have been optimized for this specific task. The cut-off frequency of the smoothing filter was set at 14 Hz and the amplification gain at G=10.

Figure 3 shows the hardware realization of the described WD. The sensor circuitry has been split in two separate boards: one for the rate gyrostar tern and the other for the accelerometer tern.

The circuitry, also shown in Figure 3, was assembled with surface montage technology, taking special care with the positioning of the sensors and their stability, so as to ensure the proper functioning and reliability of the device. A dedicate Testing Equipment was developed in order to calibrate the equipment and for the test bench. Based on the DMC-1410 controller and a step-by-step motor with encoder [9], this equipment permitted an error lower than $2*10^{-2}$ deg in the interval of interest [7].

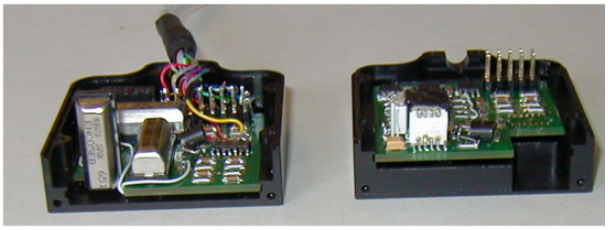


Figure 3. Image of the hardware realization of the 3-GYROs (left) and ACCs (right) assembly

The instrument was calibrated in two phases: one static, the other dynamic. The calibration of the accelerometer tern was carried out during the static phase. The sensor unit was subjected to different g vectors by positioning each of its six faces on a horizontal plane; the output signals were averaged over a 6-second time interval.
The dynamic calibration was done with the Testing Equipment. A plate on which the device is affixed was rotated by means of the Testing equipment in order to impose known rotational time laws with the necessary accuracy. Angular velocities ranging from 10 to 60 degrees/s in both directions were imposed for each of the three orthogonal axes.
In both cases the calibration matrixes were computed by the least squares method.
[CA] is the matrix for the accelerometers channels, [CRG] is the matrix for the rate-gyroscopes channel. Each one of these matrix was obtained solving the following equation

$$[C]=[L][S]T([S][S]T)-1 \quad (3)$$

where [L] is the matrix of the imposed quantities and [S] is the matrix of the assessed quantities.

4. NEURAL NETWORK STRUCTURE AND LEARNING RULE

An artificial Neural Network [10] is a mathematical model for information processing based on a connectionist approach to computation, inspired by the human brain. In a Neural Network model, simple nodes (or "neurons", or "units") are connected together to form a network of nodes. The strength of a connection between a neuron and another is influenced by a weight value. A typical Neural Network is arranged with three layers of neurons: input, hidden, and output layer. In this context, we consider feed-forward architectures only, where the information signal propagates from input layer, through intermediate or hidden layer, to output layer, with no loops back to previous neurons.
This Neural Network is known as Multilayer Preceptron (MLP) or FeedForward Backpropagation Neural Network (FFBPNN), due to its learning algorithm. FFBPNN's input layer and output layer represent the points of contact of the net with the external environment, while the hidden layer contributes to form the non linear relations existing between inputs and outputs.
A key property of a Neural Network is its ability to acquire knowledge by examples. Learning is an iterative process of adjustment applied to the synaptic weights of the network in response to an external stimulus. In particular, we will consider only Neural Networks trained by means of supervised learning: a training set, which contains both the input patterns and the corresponding desired outputs (or target patterns), is presented iteratively to the network with the aim of implementing a mapping that matches the training examples as closely as possible. Weights are iteratively modified through two passages, which represent an epoch (backpropagation algorithm):

1. A pattern input is proposed to the network input and then it is propagated to the network output (forward pass); than the error E as the squared difference between desired and actual output is calculated;

2. The error E is back-propagated (backward pass) and weights are updated according to the formula of gradient descent [10].

After all, a FFBPNN works like powerful mathematical trainable interpolation systems to calculate non-linear functions starting from desired inputs/output relationships. The complexity of interpolating function grows with the number of the neurons in the hidden layer, as well as learning capability. These kinds of structures are simple and effective, and have been exploited in a wide assortment of machine learning applications [10].

5. SENSOR INTERFACE AND MAPPING STRATEGIES

The new virtual musical instrument that we propose is constituted by three components: the sensor unit, the mapping unit and the synthesizer unit. Particularly, the signals supplied by the sensor unit, that realizes the interface between the performer and the system, don't influence directly the parameters that rule the behavior of the sound generators, but they are pre-processed by the mapping algorithm. The chosen sensor, as well as the mapping strategies, by means of which the synthesis parameters are controlled, all influence the way the musician approaches the composition process. In Figure 4, the structure of the virtual musical instrument is shown.
The implementation of musical expressivity is accomplished once we define the correspondence between the n sensor outputs and the m synthesis parameters, that is to say, once we define the right mapping.

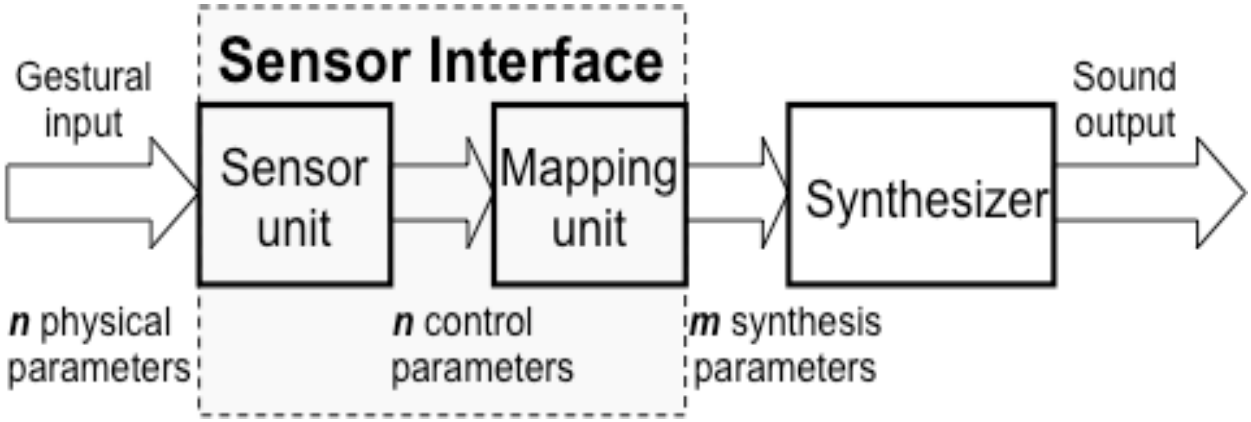


Figure 4. The virtual musical instrument

Let's assume the following concepts:

1. a predictable musical message be associated to an a priori known functional relation between the two hyperspaces and , that is to say, between the set of all the n sensor inputs and the set of all the m synthesis parameters;

2. an unpredictable musical message be associated to a non linear and a priori unknown correspondence between and .

A composer can easily follow the above assumptions by making use of a FFBPNN trained as follows:

1. he chooses the transducer to use, i.e. n physical parameters to measure (in our case, acceleration and angular velocity);

2. by means of the sensor unit, the system carries out the n control parameters;

3. he fixes a point in the n-dimensional hyperspace and he links it to a desired configuration of the m synthesis parameters;

4. he repeats D times step 3., so as to have D n-to-m examples at his disposal; they constitute the training set for the mapping unit;

5. he chooses the Neural Network structure, that means he chooses the number of hidden neurons to use, then he trains the Neural Network.

6. he explores the n-dimensional input hyperspace by moving through known and unknown points, with the aim of composing his piece of music.

6. REAL-TIME MUSICAL APPLICATION

As application of our sensor interface, we have developed a real-time musical composition. The synthesis process was realized by means of the sound synthesizer "Textures 3" [11]. This musical composition has been developed by using the Max/MSP [12] environment. Figure 5 show "Texture 3" standard VST [13] (Virtual Studio Technology from Steinberg ltd) audio plug-in.
The synthesized sound of "Texture 3" is based on granular additive synthesis algorithm. There are seventeen sound synthesis parameters [11], showed in Fig. 5, regarding sliders and knobs, through which we can shape the sound waveform.
The transducer is applied to the barycentre of the body of a dancer and measures the acceleration and the angular velocity of the body postures and gestures. Therefore, we have six control inputs to operate on the seventeen parameters that influence the sound produced by the synthesizer.
We have chosen nine reference points in the six-dimensional hyperspace of the accelerations and the angular velocities. Then, we have trained a FFBPNN with ten neurons in the hidden layer and we have explored the six-dimensional hyperspace. Finally, we have chosen, amongst many others, the movements the dancer has to repeat in order to reproduce the interesting sounds discovered during exploration.
In Figure 6 the slope of a synthesis parameter returned by the output of the, as a function of the input angular velocities ωx and ωy, is shown. The evident points in the graph represent the input/output patterns of our training set.

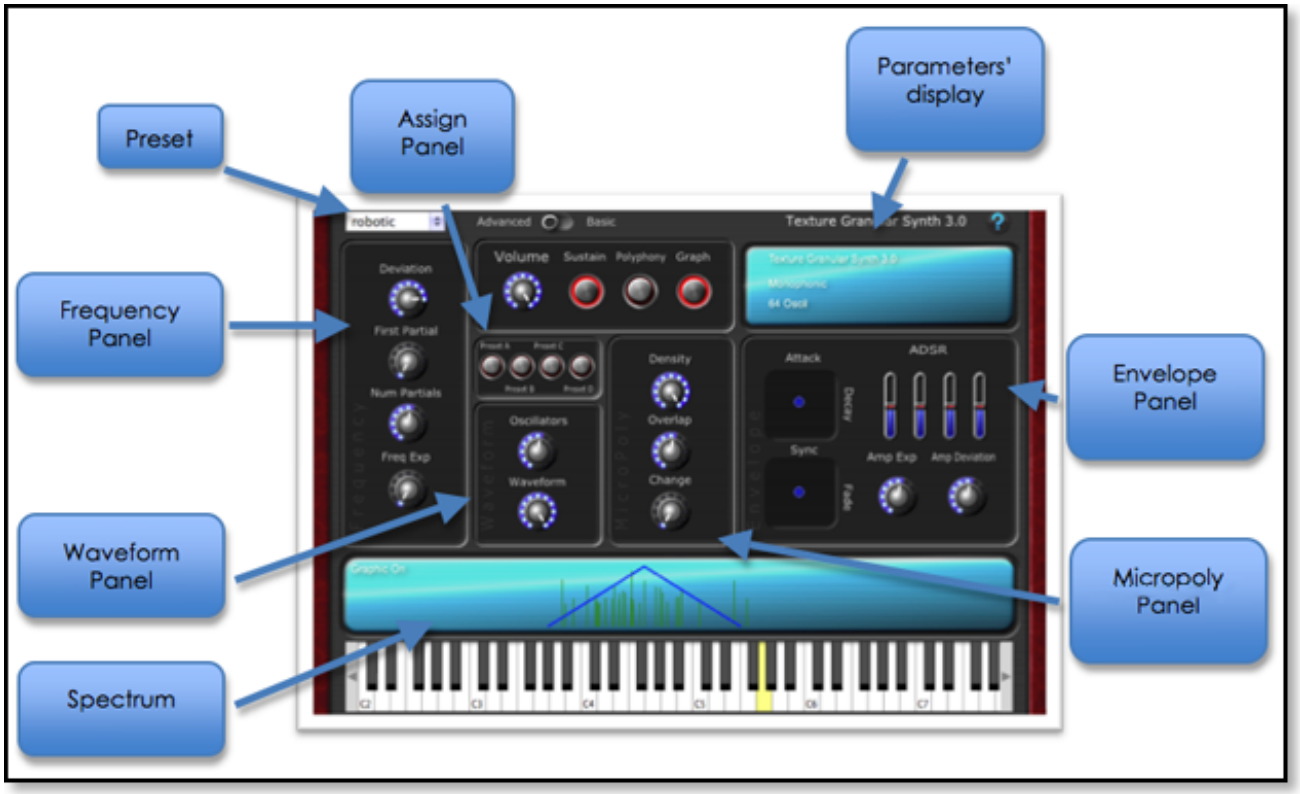


Figure 5. “Texture 3” sound synthesizer

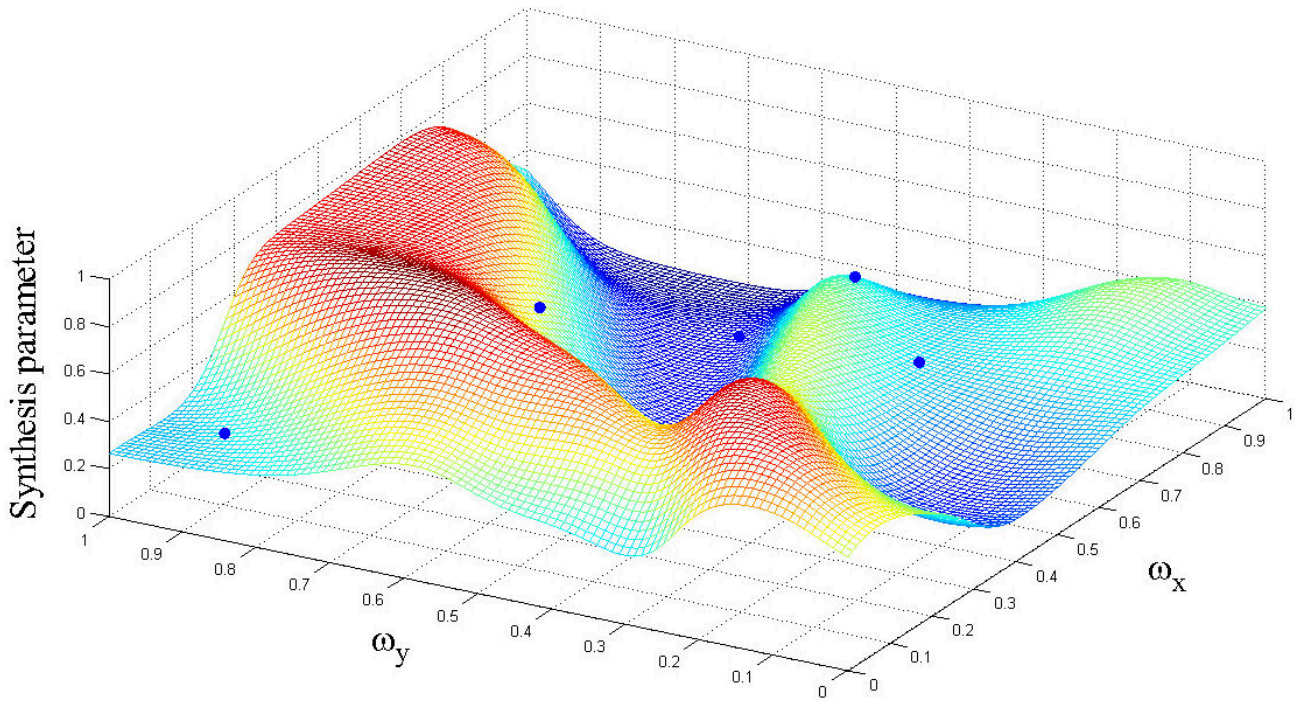


Figure 6. A synthesis parameter as a function of the input angular velocities ωx and ωy

7. CONCLUSION

We have developed a sensor interface for composing and performing expressive musical sound. We direct our attention to common musical aesthetics as a determinant factor in musical expressivity.
The sensor interface we have presented is arranged by a sensor unit that supplies kinematics physical parameters. Particularly, these parameters are motion acceleration and angular velocity processed by a FFBPNN based mapping unit that is able to provide suitable relationships between physical and sound synthesis parameters.
The experiences made by working with our sensor interface have shown that the mapping strategy is a key element in providing musical sounds with expressivity. So, we have defined six composition rules in order that a musician can easily compose his own piece of music with our sensor interface.

8. ACKNOWLEDGEMENTS

The Master in SONIC ARTS at University of Rome Tor Vergata supports this research.
http://www.mastersonicarts.uniroma2.it

9. REFERENCES

[1] Neville H. Fletcher, Thomas D. Rossing, *The Physics of Musical Instruments*, Springer, 2nd edition (July 22, 2005).

[2] Curtis Roads, *The computer music tutorial*, The MIT Press, (February 27, 1996).

[3] Bongers, B. 2000, *Physical Interfaces in the Electronic Arts. Interaction Theory and Interfacing Techniques for Real-time Performance*, In M. Wanderley and M. Battier, eds. Trends in Gestural Control of Music. Ircam - Centre Pompidou.

[4] Orio, N. 1999. "A Model for Human-Computer Interaction Based on the Recognition of Musical Gestures." *Proceedings of the 1999 IEEE International Conference on Systems*, Man and Cybernetics, pp. 333-338.

[5] Abraham Moles, *Information Theory and Aesthetic Perception*, University Of Illinois Press (1969).

[6] Rudolf Arnheim, *Entropy and Art: An Essay on Disorder and Order*, University of California Press (January 29, 1974).

[7] D. Giansanti, G. Maccioni, V. Macellari "The development and test of a device for the reconstruction of 3d position and orientation by means of a kinematic sensor assembly with rate gyroscopes and accelerometers", *IEEE Trans. on Biomedical Engineering*, Vol 52, n. 7, pp. 1271-1277, 2005.

[8] Goldstain H, *Classical Mechanics*, Reading, MA: Addison Wesley (1977).

[9] DMC-1400 *Series Manual By Galil Motion Control*, Inc

[10] Hertz J., A. Krogh & R.G. Palmer, *Introduction to the theory of neural computation*, Addison-Wesley Publishing Company, Reading Massachusetts, 1991.

[11] Giorgio Nottoli, "A sound texture synthesizer based on algorithmic generation of micro-polyphonies", *Proc. of 3nd International Conference "Understanding and creating Music"*, Caserta, December 2003, 11-15. Available on the web at http://www.mastersonicarts.uniroma2.it/ricerca/TexGrSy.html

[12] Cycling74 Max/MSP, documentation available at http://www.cycling74.com/products/maxmsp

[13] Steinberg VST Audio Plug-Ins SDK, 3rd party developer support site at http://www.steinberg.net/324_1.html

"PITCH-CLASS SET", FORTE E DROBISH

Anna Terzaroli
Conservatorio di Musica Santa Cecilia
Roma, Italy
anna.giw@libero.it

SOMMARIO

Lo studio propone e presenta un metodo finalizzato alla fenomenologia della caratterizzazione degli elementi costituenti un "pitch-class set", con evidente rimando alle teorie sostenute da Forte nella seconda metà del XX secolo. Si considereranno gli studi di Drobish e, in particolare, l'assunto del modello a elica per una corretta rappresentazione del "pitch space". L'obiettivo raggiunto costituirà, inoltre, un punto di partenza verso una metodologia di classificazione dell'intera entità "pitch-class set".

1. INTRODUZIONE

1.1 "Pitch-class set", Forte

La teoria matematica degli insiemi ha avuto origine tra il 1874 e il 1897 ad opera di Georg Cantor, è stata oggetto di riflessione compositiva per Anton Webern, quindi ha assunto importanza fondamentale per i compositori seriali dei primi anni Sessanta. Milton Babbitt, dal 1955 al 1972, pubblica una serie di scritti sulla teoria musicale degli insiemi, considerata da un punto di vista compositivo più che analitico. Il contributo più significativo circa l'applicazione della suddetta teoria all'analisi musicale e all'analisi di composizioni atonali non seriali è da attribuirsi a Allen Forte.

Tra il 1964 e il 1973, Allen Forte sviluppa la nozione di "insiemi di classi di altezze" o "pitch-class sets", ovvero insiemi di note, le quali vengono considerate a prescindere dall'ottava di appartenenza (Figura 1 e 2 [1]); inoltre esistono delle relazioni tra gli insiemi, così da permettere la confluenza di più insiemi in "complessi" o "sottocomplessi".

L'importanza delle teorie di Forte è dovuta alla possibilità di evidenziare, anche nell'ambito dell'atonalità, la coerenza interna delle strutture musicali e i legami che intercorrono tra le sezioni [2].

Copyright: © 2014 Anna Terzaroli. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

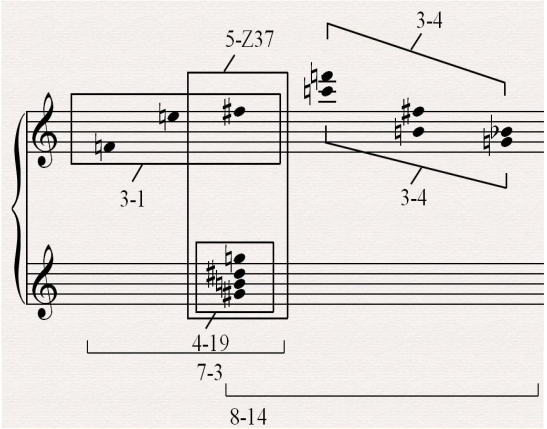


Figura 1. Esempio di segmentazione in insiemi e denominazione degli insiemi.

Name	Pcs	Vector
3-1(12)	0,1,2	210000
3-2	0,1,3	111000
3-3	0,1,4	101100
3-4	0,1,5	100110
3-5	0,1,6	100011
3-6(12)	0,2,4	020100
3-7	0,2,5	011010
3-8	0,2,6	010101

Figura 2. Alcune "prime forms" o "forme primarie" e i relativi vettori intervallari.

1.2 Drobish

Nella teoria musicale, il concetto di "pitch space" è correlato alla necessità di porre in relazione le altezze. Esistono diversi modelli di "pitch space", che, generalmente, utilizzano la distanza tra due o più altezze per indicare il loro grado di parentela (le altezze maggiormente simili saranno più vicine, quelle meno simili, più lontane). In tal senso, un esempio è costiuito dal cerchio delle classi di altezze, che prevede la disposizione di tutte le classi di altezze sulla circonferenza di un cerchio, nell'ordine proprio della scala cromatica. In base alla complessità dei rapporti tra le altezze che vengono presi in considerazione, i modelli possono essere multidimensionali.

Nel 1855 W. M. Drobish [3] suggerì il modello a elica, come rappresentazione del "pitch space". In questo modello la scala cromatica viene distribuita sui livelli di un'elica e ogni singola classe di altezza è idealmente posta su una retta che unisce questa con la stessa classe di altezza presente nelle altre ottave. Il modello a elica è rappresentabile anche tramite un cilindro attorno al quale è avvolto un "pitch space" lineare, anche in tal caso, tutte le altezze in rapporto di ottava tra loro, sono poste lungo una retta ideale.

In base al modello a elica, ogni altezza è costituita da "pitch-class" (che indica la sua posizione nel cerchio delle classi di altezza) e "brightness" o brillantezza (che indica la sua posizione sull'asse verticale).

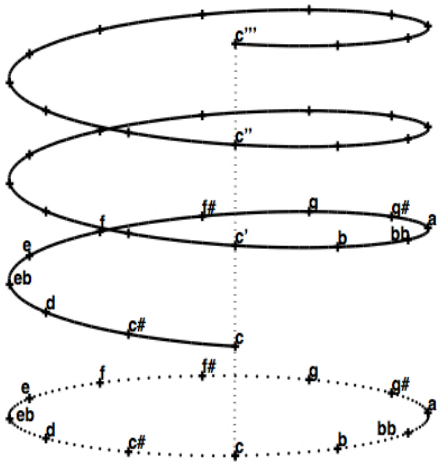


Figura 3. Scala cromatica disposta a elica: una retta ideale unisce tutte le altezze della classe di altezza "c".

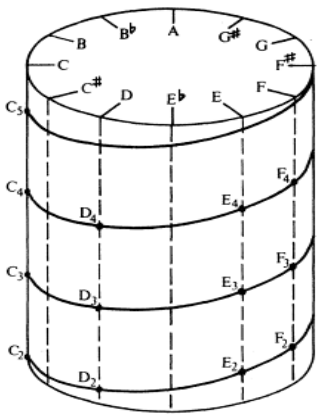


Figura 4. Cilindro con "pitch space" lineare.

2. OBIETTIVI E METODI

Ogni nota musicale ha un proprio spettro armonico, costituito da determinate frequenze componenti, presenti, ciascuna, in una determinata quantità.

Il baricentro dello spettro o "spectral centroid" [4] di ogni segnale complesso, quindi anche di ogni nota musicale, è calcolato come la somma del prodotto di ogni componente frequenziale moltiplicata per la sua intensità, ciò fino all'ultima componente frequenziale presente nello spettro, la somma così ottenuta è divisa per la somma di tutte le intensità presenti nello spettro.

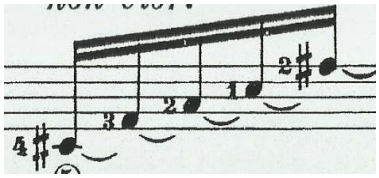
$$\mu_s = \frac{\sum_i f_i I_i}{\sum_i I_i} \qquad (1)$$

Il risultato di (1) è una frequenza (misurata in Hz) il cui valore viene riscalato tra 0 e 1 e costituisce un indice di "brightness" o brillantezza di quella nota musicale.

Questo calcolo, con la relativa assegnazione dell'indice, viene applicato a ogni elemento costituente un dato "pitch-class set"; l'indice, nel tempo di esistenza della nota, subirà delle variazioni, per tenere conto delle quali è opportuno mutare la rappresentazione dell'indice da un singolo valore numerico ad una funzione continua nel tempo.

Tutte le funzioni continue nel tempo, ciascuna propria di ogni elemento costituente un determinato "pitch-class set", vengono riunite in un grafico, che mostra l'andamento dell'indice di "brightness" dell'intero "pitch-class set" in esame. Ogni "pitch-class set", quindi, può essere rappresentato, oltre che dai numeri forteiani indicanti le classi di altezze, anche da un grafico.

Si consideri, a scopo esemplificativo, un insieme costituito da cinque note (come mostrato in Figura 5) tratto dal brano per chitarra "Nunc" di Goffredo Petrassi.



5-22 [0,1,4,7,8]

Figura 5. Particolare di "Nunc" per chitarra di G. Petrassi (1971) e corrispondente "prime form".

L'insieme è stato inizialmente individuato tramite segmentazione, quindi alle note costituenti sono state applicate le operazioni necessarie al fine di trovare la forma primaria corrispondente, che (come mostrato in Figura 5) risulta essere la 5-22. Tale forma primaria non esprime informazioni sull'ottava di appartenenza di ogni nota costituente. In Figura 6 viene proposto il grafico relativo alla brillantezza del "pitch-class set" della Figura 5.

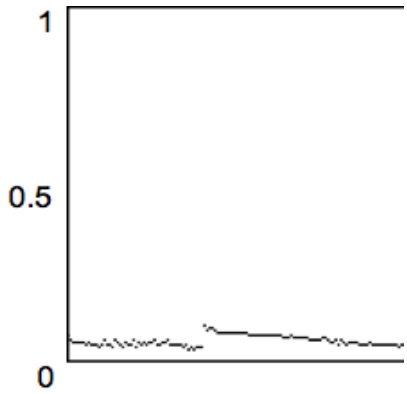


Figura 6. Il grafico mostra l'andamento dell'indice di "brightness" dell'insieme della figura precedente.

Tramite l'utilizzo di questa tipologia di rappresentazione, è possibile rilevare anche altre caratteristiche dell'insieme.

3. CONCLUSIONI

L'introduzione del parametro "brightness" nella teoria degli insiemi di A. Forte può essere utile nell'attività compositiva e analitica, in quanto rende maggiormente chiara e visibile la brillantezza di ogni insieme, restituendo la possibilità di appezzare non solo le classi di altezze in uso, ma anche la loro ottava di appartenenza.

Inoltre, il guardare agli insiemi dal punto di vista della brillantezza, potrà condurre ad una ulteriore classificazione di questi ultimi, i quali si potranno associare (parallelamente alla costituzione di "complessi" e "sottocomplessi") e confrontare anche in base al nuovo parametro.

4. BIBLIOGRAFIA

[1] A. Forte: *The Structure of Atonal Music,* Yale University Press, New Haven, London, 1973.

[2] I. Bent, W. Drabkin: *Analisi Musicale,* EDT, Torino, 2008.

[3] M. W. Drobish: "Über musikalische Tonbestimmung und Temperatur," *Abhandlungen der Mathematisch-Physischen Klasse der Sächsischen Akademie der Wissenschaften,* pp. 3–120, 1855.

[4] G. Peeters, S. McAdams, P. Herrera: "Instrument sound description in the context of MPEG-7," *Proceedings of the International Computer Music Conference (ICMC2000)*, 2000.

[5] W. Aures: "Berechnungsverfahren für den sensorischen Wohlklang beliebiger Schallsignale," *Acustica* Vol. 59, pp. 130–141, 1985.

