

Proceedings

XI CIM

Colloquio di Informatica Musicale

a cura di
Luigi Fiparelli e Fabio Regazzi

Bologna 8 - 11 Novembre 1995

A.I.M.I. Associazione di Informatica Musicale Italiana

Conservatorio di Musica G.B.Martini

Dipartimento di Musica e Spettacolo, Università di Bologna

Progetto MusicaDuemila - Dipartimento dello Spettacolo

Presidenza del Consiglio dei Ministri

Davide Rocchesso
Sound Processing

D.Rocchesso@computer.org
www.dei.unipd.it/english/cs/roc

In copertina:

da un manoscritto di Aldo Piccialli, elaborazione grafica di Emilia Franchini.

Proceedings

Atti

**XI Colloquium on Musical Informatics
XI Colloquio di Informatica Musicale**

a cura di Luigi Finarelli e Fabio Regazzi

Bologna 8-11 Novembre 1995

Davide Rocchesso
Sound Processing

D. Rocchesso @ computer.org
www.dei.unipd.it/english/sac/roc

Istituto di Filmologia - via del Guasto, 7
Conservatorio di Musica G.B.Martini - piazza Rossini, 2
Aula Absidale di Santa Lucia - via de'Chiari 23/a

A.I.M.I. Associazione di Informatica Musicale Italiana

Conservatorio di Musica G.B.Martini

Dipartimento di Musica e Spettacolo, Università di Bologna

**Progetto MusicaDueMila - Dipartimento dello Spettacolo
Presidenza del Consiglio dei Ministri**

COMITATO SCIENTIFICO

- Giacomo Borelli (Università di Bologna)
• Antonio Chiarugi (Università di Genova)
• Roger Charles Berg (Carnegie Mellon University, Pittsburgh)
• Giovanni De Poli (Università di Padova)
• Giandomenico Belotti (Università di Milano)
• Marco Leonetti (CERN - The European Organization for Nuclear Research, Geneva)
• Alan Meirman (Universität Regensburg, Regensburg)
• Paolo Pizzetti (Università di Napoli)
• Stephen Tully (University of Western Ontario, London)

ORGANIZZAZIONE

- Comitato Organizzatore: Nino Di Stefano (AIAV), Giacomo Borelli (Università di Bologna), Antonio Chiarugi (Università di Genova), Giandomenico Belotti (Università di Milano), Marco Leonetti (CERN - The European Organization for Nuclear Research, Geneva), Alan Meirman (Universität Regensburg, Regensburg), Paolo Pizzetti (Università di Napoli).

CONTRIBUZIONI

- Contributi scientifici: Giacomo Borelli (Università di Bologna), Antonio Chiarugi (Università di Genova), Giandomenico Belotti (Università di Milano), Marco Leonetti (CERN - The European Organization for Nuclear Research, Geneva), Alan Meirman (Universität Regensburg, Regensburg), Paolo Pizzetti (Università di Napoli).

COLLABORATORI

- Collaboratori: Giacomo Borelli (Università di Bologna), Antonio Chiarugi (Università di Genova), Giandomenico Belotti (Università di Milano), Marco Leonetti (CERN - The European Organization for Nuclear Research, Geneva), Alan Meirman (Universität Regensburg, Regensburg), Paolo Pizzetti (Università di Napoli).

INTRODUZIONE

Mario Baroni

Dipartimento Musica e Spettacolo
Università degli Studi di Bologna

Per la prima volta nella sua storia ormai più che ventennale il Colloquio di Informatica Musicale organizzato dall'AIMI approda quest'anno a Bologna. La circostanza non è casuale e può richiedere un breve commento, se non altro per ricordare che la città non è mai stata un centro di produzione di musica informatica. Questo può stupire in un contesto così ricco d'iniziative musicali com'è quello del capoluogo emiliano dove fra l'altro esiste presso l'Università un corso di laurea come il DAMS che è un po' un unicum in tutto il territorio nazionale e dove esiste presso il Conservatorio un corso di musica elettronica che è nato fra i primi in Italia. Direi però che l'assenza del CIM da Bologna non deriva da una mancanza di interessi o di vocazioni, ma piuttosto da un diverso accento che qui si è preferito porre sulla funzione dell'informatica come strumento di ricerca e di sperimentazione in campo musicale.

L'informatica musicale ha infatti molte anime, e fra queste c'è anche un'anima per così dire "teorica" che si affianca e fa da bordone alla produzione di musica o di tecnologia. Intendo alludere soprattutto alle teorie musicali che, in un'epoca di importanti trasformazioni com'è quella in cui viviamo, stanno cercando nuove organizzazioni concettuali in grado di spiegare le forme inedite assunte da tante musiche d'oggi. L'analisi del linguaggio musicale, l'analisi del suono, l'analisi dei meccanismi psicologici di produzione e di ascolto, l'analisi dei comportamenti musicali, costituiscono il cuore di questa nuova concettualizzazione e corrispondono ad altrettanti settori d'interesse che stanno alla base del nuovo sapere (o teoria) musicale. E la tecnologia informatica è ormai diventata indispensabile corredo per l'acquisizione di queste conoscenze.

Appunto a questo particolare versante dell'informatica musicale, che è certamente più discreto e più sotterraneo dell'altro al quale siamo abituati, ma che forse non è meno importante, è idealmente dedicato il CIM di quest'anno. Un po' perché le realtà istituzionali e le persone stesse che sono attive a Bologna hanno una particolare tradizione d'interessi e di realizzazioni in questo settore, un po' perché è sembrato ormai opportuno metterne in rilievo la presenza e anche in un certo senso la maturità. Studi sull'analisi della musica elettroacustica, infatti, o in genere delle musiche senza partitura, così come studi sull'esecuzione musicale e sulle componenti emotive dei messaggi sonori, sulle modellizzazioni dei procedimenti dell'ascolto e della composizione, anche della composizione tradizionale, sulla percezione e sugli aspetti cognitivi dell'elaborazione del suono, studi di questa natura sono stati coltivati negli ultimi anni non solo a Bologna, ma a Padova, a Milano, a Firenze, a Genova, a Roma, per non parlare ovviamente dei centri attivi all'estero: a tutti questi il CIM bolognese intende dare un segno non solo di ospitalità, ma di incoraggiamento: intende mettere in rilievo l'importanza della loro presenza.

E' ovvio che tutto ciò non può sacrificare e non sacrifica il programma di quest'anno che, come sempre, ha il suo fulcro nell'hardware e nel software necessari per produrre musica al calcolatore e nelle novità che il CIM, nel suo ruolo di "fiera" biennale di prodotti tecnologici, offre ai musicisti che sono interessati al problema. Vorrei anzi sottolineare a questo proposito come la partecipazione sia stata ancora una volta straordinariamente viva ed entusiasmica. Anche un semplice sguardo ai temi delle sedute può dare un'idea della ricchezza e varietà d'interessi che percorrono oggi la disciplina.

Fra i vari temi proposti non si può fare a meno di notarne uno abbastanza inconsueto nel

nostri colloqui: quello del rapporto fra l'informatica e la produzione radiofonica. Il 1995 è anche anno marconiano e soprattutto a Bologna le celebrazioni del centenario si sono fatte sentire. La tavola rotonda inclusa nel nostro programma non solo vuol proporre un tema di riflessione tutt'altro che secondario per l'informatica musicale e per la stessa storia della tecnologia, ma vuole anche tener desta l'attenzione sul fatto che l'informatica musicale è terra di confine, è disciplina che sta fra le scienze naturali e le scienze umane, e in questo senso può benissimo inglobare discorsi sulla sua propria storia e considerarli parte integrante del proprio programma scientifico. Insomma, il proposito di quest'anno è quello di un'apertura della tecnologia informatica ai temi musicologici, nella convinzione che questo orientamento sia ricco di futuro.

Il programma di quest'anno è nato anche da una collaborazione vivace e attiva fra il Conservatorio e l'Università. Anche questa collaborazione non è casuale, ma è frutto di specifico lavoro: nel marzo del 1994 si è tenuto a Bologna, a cura della SIEM, un convegno nazionale sulle relazioni fra il Conservatorio e l'Università, convegno dedicato a problemi politici e istituzionali di ampio respiro, ma anche al tentativo di configurare in sede locale qualche forma di collaborazione che potesse servire da esperienza per rapporti futuri e forse anche da modello per altre realtà territoriali. L'XI CIM è anche frutto di quel convegno, ed è comunque una prima esperienza di collaborazione nel nostro settore in attesa che fra le due istituzioni possano sorgere ulteriori occasioni di scambio e forse di più stretta e organica integrazione tecnologica.

Ma il CIM si apre anche sotto il segno di un evento improvviso e tragico. Aldo Piccialli, che era uno dei fondatori dell'AIMI e che non era stato assente in nessuno dei precedenti appuntamenti, non sarà più insieme a noi. Ci mancheranno certo il suo lucido apporto scientifico e la rete viva dei contatti che egli aveva saputo tenere in questi anni con i centri di produzione di tutto il mondo, dai più presigiosi istituti europei al Massachussets Institute of Technology dov'egli era di casa.

Ma ci mancheranno anche la sua indimenticabile simpatia, la sua luminosa parlata dialettale, la sua saggezza antica, i suoi scatti di umore; ci mancherà soprattutto il suo sorriso nel quale sembravano incarnarsi le dolcezze, le astuzie e le sofferenze di tutta la storia della città di Napoli. Alla sua memoria abbiamo voluto dedicare la manifestazione di quest'anno.

Bologna, novembre 1995

PRESENTAZIONE

Lelio Camilleri

Presidente

AIMI - Associazione di Informatica Musicale Italiana

Nel presentare questo XI Colloquio di Informatica Musicale permettemi innanzitutto di provare un po' di imbarazzo dato che oltre a presiedere l'AIMI faccio anche parte del comitato organizzatore: è quindi inevitabile che mi senta particolarmente coinvolto in questa iniziativa.

Questa edizione nasce da una collaborazione fra Conservatorio e Università che nella città di Bologna cercano di collaborare il più possibile, soprattutto in quei settori, come l'applicazione della tecnologia in musica, dove la co-operazione è di estrema importanza. L'XI Colloquio di Informatica Musicale è una dimostrazione di come Istituzioni con differenti obiettivi, la formazione del musicista l'una e la formazione del musicologo l'altra, possano collaborare per curare aspetti diversi che però trovano ambedue posto nel convegno: la ricerca compositiva e quella scientifico-musicale.

L'XI edizione del Colloquio presenta un'attenzione più marcata per alcune tematiche di carattere specificamente musicale, e non poteva essere diversamente viste le Istituzioni che lo organizzano. Due sessioni lo sono in special modo, e sono la sessione speciale sull'analisi della musica elettroacustica e quella sui rapporti fra il medium *radio* e la musica elettroacustica.

Per quanto riguarda l'analisi della musica elettroacustica, essa è una tematica emergente (e sempre più consolidata) che oltre a interessare coloro i quali studiano questo repertorio, ha applicazioni sia didattiche, come materia di studio nei corsi di musica elettronica dei Conservatori, che metodologiche per gli altri settori analitici. Una sessione di questo tipo trova la sua naturale collocazione in questa edizione dato che Bologna è, per molti versi, la "capitale" analitica italiana.

Anche il tema *radio e musica elettroacustica* ha una relazione importante con gli avvenimenti odierni. La realizzazione di un ciclo di radio-film prodotti dalla RAI, infatti, ha dato nuova vita ad una rapporto, quello fra musica elettroacustica e radio, che si è dimostrato molto fecondo negli anni passati. Questa sessione vuole quindi presentare, attraverso interventi qualificati, una profonda riflessione fra avvenimenti del passato e del presente.

Mi sono soffermato su queste sessioni solo perché rappresentano un po' la particolarità di questa edizione del colloquio. Ovviamente, anche le altre presentano aspetti di notevole interesse, con un massiccio contributo di interventi di ricercatori di altri paesi. Anche questo aspetto, come rilevavo nella presentazione del Colloquio precedente, va sottolineato. Il Colloquio di Informatica Musicale, fondato come convegno della comunità italiana è ormai un convegno europeo, o meglio internazionale. La vitalità della ricerca italiana nell'informatica musicale ha creato un importante momento di confronto in cui si riconoscono molte realtà di altri paesi.

Ciò si riflette anche nella parte più propriamente musicale del Colloquio, nei concerti. È importante sottolineare come, rispetto al Colloquio precedente, il numero delle composizioni selezionate sia quasi raddoppiato, merito dell'alto numero e della qualità dei contributi musicali ricevuti.

Vorrei concludere questa breve presentazione con dei ringraziamenti che, tengo a precisarlo, non sono assolutamente di circostanza. Vorrei innanzitutto ringraziare gli autori, sia dei contributi scientifici che di quelli musicali, il Dipartimento di Musica e Spettacolo dell'Università di Bologna, nella persona di Mario Baroni, per molti aspetti dell'organizzazione e il direttore del Conservatorio di Musica G.B. Martini, maestro Carmine Carrisi, per il suo sostegno all'iniziativa.

Un caldo ringraziamento va ai membri dei comitati scientifico e musicale per il prezioso lavoro svolto, alla Presidenza del Consiglio dei Ministri Progetto MusicaDuemila e alla segreteria operativa CIDIM-MusicaDuemila per il contributo dato alla pubblicazione degli atti.

Un particolare ringraziamento alla collaborazione di Annamaria Morini e Alberto Caprioli per l'aiuto nella realizzazione delle esecuzioni strumentali.

Infine, un sentito ringraziamento ai curatori di questo volume, Luigi Finarelli e Fabio Regazzi, per il lavoro svolto con una grande dedizione e scrupolo professionale.

Bologna, novembre 1995

INDICE

M. Baroni <i>Introduzione</i>	I
L. Camilleri <i>Presentazione</i>	III

Scientific Session: Computer Assisted Composition

C. Goldman, D. Gang, S. Rosenschein <i>NetNeg: A Hybrid Architecture for Polyphonic Music</i>	3
J. Leach <i>Algorithmic Composition as Gene Expression Based upon Fundamentals of Human Perception</i>	7

Scientific Session: Digital Signal Processing (I)

Z. Settel, C. Lippe <i>Musical Applications Using Real Time Frequency Domain Signal Processing</i>	13
A. Di Scipio <i>Real-time Polyphonic Time-shifting of Sound with Interactive Systems</i>	19
I. Prignano <i>Sintesi di eventi sonori complessi per mezzo di iterazioni funzionali</i>	23
F. Fontana, D. Rocchesso <i>A New Formulation of the 2D-Waveguide Mesh for Percussion Instruments</i>	27

Special Session: Electroacoustic Music Analysis

A. Di Scipio <i>Riflessioni sull' analisi della musica elettroacustica e informatica. Esperienze e spunti teorici</i>	33
E. Pancaldi <i>Analysis of 'Continuo' (1958) by B. Maderna</i>	37
A. Cipriani <i>Problems of methodology: the analysys of 'Kontakte'. Thinking in terms of stories</i>	41
F. Giomi, M. Ligabue <i>Forme e strutture nella musica elettroacustica tra gli anni Cinquanta e Sessanta</i>	45

Report from Laboratories

- T. Itagaki, D. Nunn, D. Phillips, D. Batjakis, A. Purvis, P. Manning
Activity Report from Durham Music Technology 51

- I. Popovic, R. Coifman, J. Berger
Towards a Unified Representation of Sound and Analytical Structure in Music 55

- N. Bernardini, A. Vidolin
The Centro TEMPO REALE: a presentation 59

Scientific Session: Musical Analysis by Computer

- M. Annunziata, P. Caianiello, M. De Amicis
Minimal Description-Length and Musical Analysis 63

- E. Cambouropoulos
A Computational Model for Discovery of Parallel Melodic Passages 67

- D. Gang, D. Lehmann
An Artificial Neural Net for Harmonizing Melodies 71

- E. Rapoport
FFT Analysis and Emotional Expression in Opera and Lied Singing 75

Scientific Session: Digital Signal Processing (II)

- P. Vettori
Fractional ARIMA Modeling of Microvariations in Additive Synthesis 81

- G. Evangelista, S. Cavaliere
Karplus-Strong Parameter Estimation 85

- G. Borin, G. De Poli
A Hammer-String Interaction Model for Physical Model Synthesis 89

Special Session: Music and Multimedia Performance

- N. Beranardini, A. Vidolin
*'Noms des airs' di Salvatore Sciarrino.
Una musica algoritmica di trasformazione sonora* 95

- A. Vidolin
Il Live Electronics in Canti d'inquietudine di Andrea Molino 99

- S. Sargentì
*Software per la definizione di strumenti nella musica elettronica:
analisi e realizzazione di 'Mikrophonie I' e 'Solo' di Karlheinz Stockhausen* 103

- T. Belletti, G. Galizia, G. Haus
MediaSynth. A System for the Synthesis of Multimedia Scores based on Petri Nets 107

G. Cospito, R. Guida <i>Le problematiche della creazione di opere multimediali: composizione e performance nelle esperienze realizzate presso il L.I.M. dell'Università degli studi di Milano</i>	111
A. Camurri, R. Chiarvetto, A. Massari, R. Rossi <i>The HARP/Vscope Interactive Dance Music System</i>	115
E. Vaccarino <i>Gli amori di Tersicore e Silicio</i>	119

Special Session: Radio and Electroacoustic Music

F. Giomi <i>Un percorso tra musica elettroacustica e radio italiana: 47 anni di Prix Italia</i>	125
N. Scaldaferrri <i>La radio e la nascita della musica elettroacustica</i>	129
L.Bianchini, S.Tamaro, M.Lupone, E.Palandri <i>Il progetto CRM per una drammaturgia dell'ascolto radiofonico e del suo spazio virtuale</i>	135

Scientific Session: Acoustics and Psychoacoustic

P. Toivainen <i>Optimizing Self-Organizing Timbre Maps: The Effect of Auditory Images and Distance Metrics</i>	141
N. Boatin, G. De Poli, P. Prandoni <i>Timbre Characterization with Mel-Cepstrum: a Multivariate Analysis</i>	145
D. Rocchesso, W. Putnam <i>A Numerical Investigation of the Representation of Room Transfer Functions for Artificial Reverberation</i>	149

Scientific Session: Computer Music Systems

P. Fossati, G. Haus, A. Minetti <i>Newton MusicKit. Scrittura musicale con pen - computer</i>	155
R. Bresin, N. Orio <i>A Gesture Interface Controlled by the Vocal Tract</i>	159
R. Bresin, G. De Poli, R. Ghetta <i>A Fuzzy Approach to Performance Rules</i>	163
K. Ng, D. Boyle, D. Cooper <i>Automated Optical Music Score Recognition and its Enhancement using High-level Musical Knowledge</i>	167

Special Session: Feature Extraction

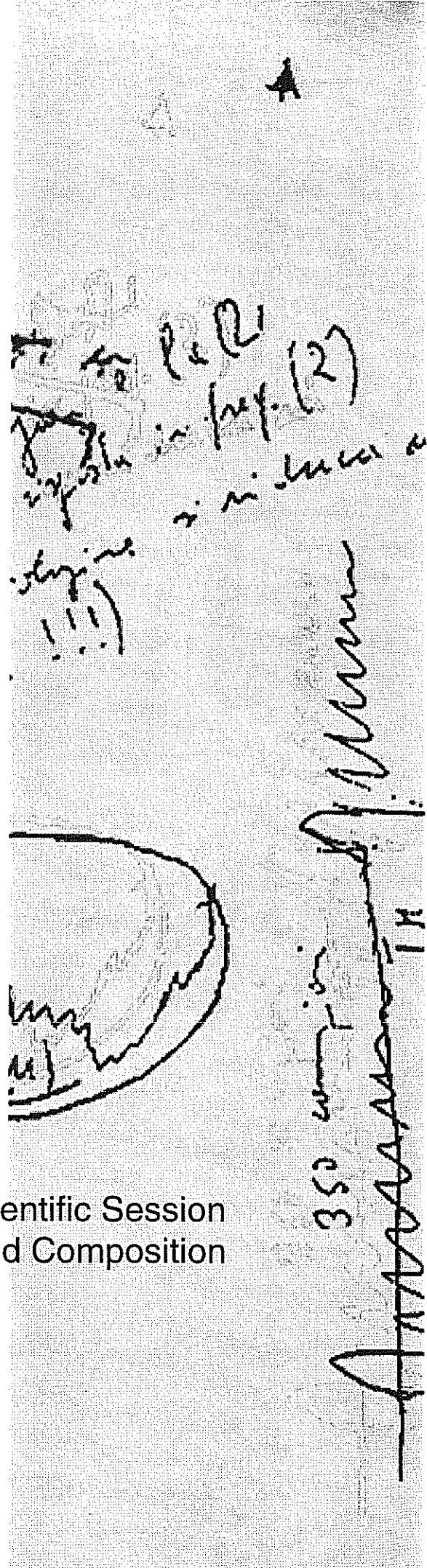
M. Leman, F. Carreras	<i>Simulation of Listening to Bach's Wohltemperiertes Klavier</i>	173
D. Alghi	<i>Estrazione in tempo reale di parametri di controllo da segnali audio complessi</i>	177
A. Ballista, G. Frazzini, G. Haus	<i>SYRINX. Conversione automatica di processi audio analogici in processi MIDI: visualizzazione e confronto</i>	181
A. Bertoni, R. Bresin	<i>Real-Time Musical Rhythm Tapping: a Neural Networks - Numerical Algorithms Hybrid System</i>	185
C. Cioffi	<i>Soul: un sistema per la classificazione di segnali musicali basato su reti neurali</i>	189
G. Allasia, A. Camurri, M. Leman	<i>A Hybrid System for the Analysis of Musical Signals: HARP/TCAD</i>	193

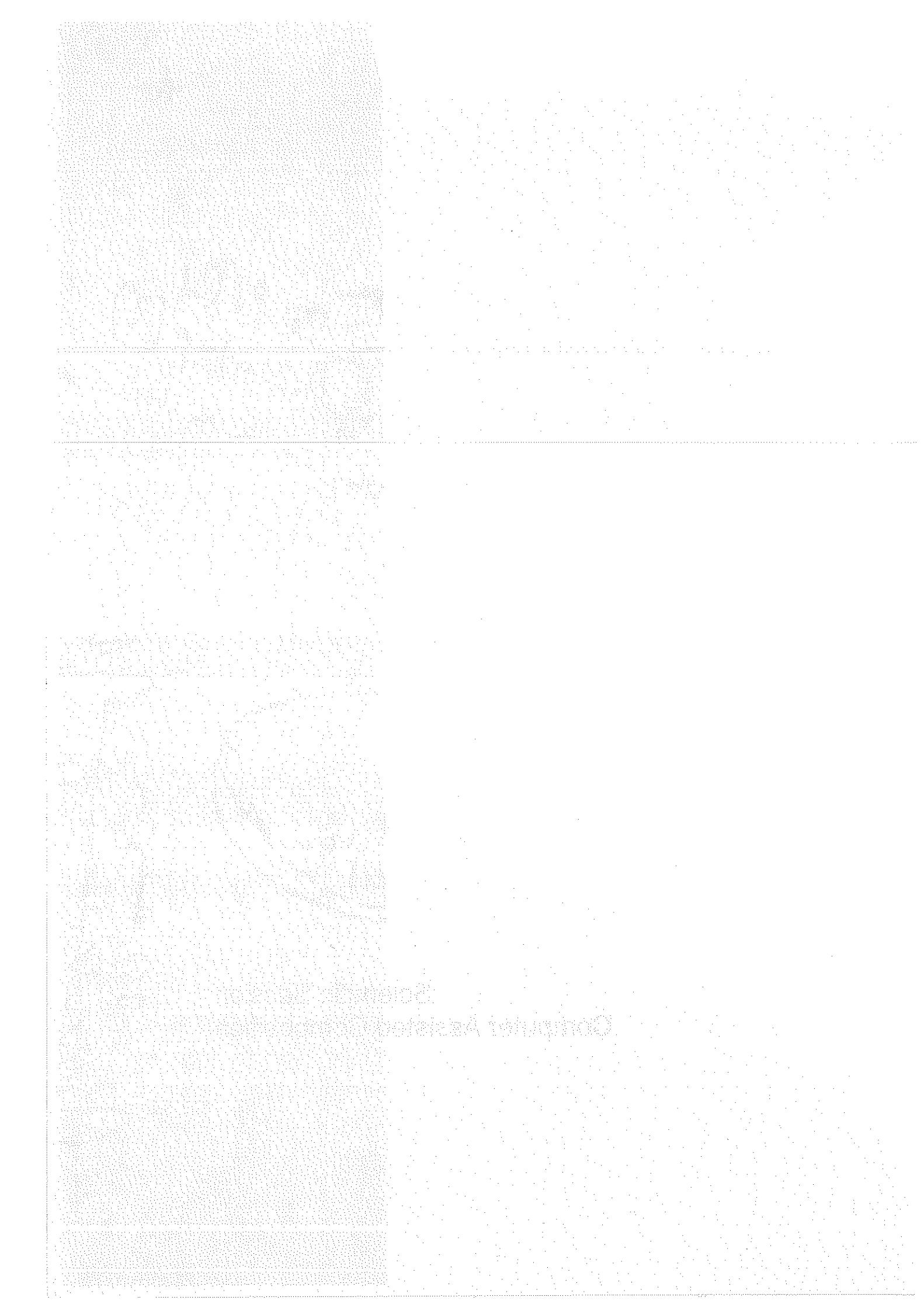
Poster and Demo

G. Cospito, M. Pennese	<i>Ambiente integrato di composizione assistita dal calcolatore e di sintesi del suono: una libreria Patchwork per il controllo di strumenti CSound</i>	199
P. Comerford, L. Comerford	<i>Driving a Musical Sound Synthesis System with a Digital Signal Processor: A Design Exercise</i>	201
A. Katrami, T. Ingall, L. Comerford	<i>A Multimedia HyperText Environment in Computer Music</i>	203
P. Gena, C. Strom	<i>Musical Synthesis of DNA Sequences</i>	205
A. Inzoli, I. Pighi	<i>Promenade.</i>	
	<i>Un Sistema Software per la Sincronizzazione di partiture e Processi MIDI</i>	207
A. Chandra	<i>TrikTraks: Structural Waveform Transformations with Composed Paths</i>	211
F. Degrassi	<i>Un software per la gestione in concerto di dispositivi per il campionamento e l'elaborazione del suono</i>	213
D. Rossi	<i>Attenuazione numerica dei disturbi nei segnali audio</i>	215
P. Torrente, V. Colombo, V. Lombardo	<i>An Environment for Music Performance</i>	219

R. De Tintis	
<i>GRAINS: a Software for Real-Time Granular Synthesis and Sampling Running on the IRIS-MARS Workstation</i>	221
M. Giachi	
<i>Progetto e realizzazione di un sistema di sintesi di un suono in tempo reale nel dominio della frequenza</i>	225
C. Rosati, F. Armani, G. Nottoli, R. Santoboni	
<i>Visual Composing System: a new Approach to Compositional Real-Time Systems</i>	227
P. Sartor, A. Vidolin	
<i>Fuzzy Control of Sound Synthesis Parameters</i>	231
P. Principi	
<i>Un'indagine sulle regole preferenziali di raggruppamento di Lerdahl e Jackendoff</i>	233
D. J. E. Nunn, A. Purvis, P. D. Manning	
<i>Graphical Display of Musical Information</i>	235
M. Malt, J. Baboni Schilingi	
<i>Profile; libreria per il controllo del profilo melodico</i>	237
A. Belladonna	
<i>HyperCSound: ambiente grafico di ausilio alla sintesi del suono ed alla generazione di eventi sonori complessi</i>	239
I concerti dell' XI CIM	243

Scientific Session
Computer Assisted Composition





NetNeg: A Hybrid System Architecture for Composing Polyphonic Music

Claudia V. Goldman, Dan Gang, Jeffrey S. Rosenschein

Institute of Computer Science, Hebrew University
Givat Ram, Jerusalem, Israel

Abstract

There are musical activities in which we are faced with symbolic and sub-symbolic processes. This research focuses on the question whether integrating a neural network together with a distributed artificial intelligence approach has any advantages in the music domain.

In this work, we present a new approach for composing and analyzing polyphonic music. As a case study, we began experimenting with first species two-part counterpoint melodies. Our system design is inspired by the cognitive process of a human musician. We have developed a hybrid system composed of a connectionist module and an agent-based module to combine the symbolic and sub-symbolic levels to achieve this task. The network produces aesthetic melodies based on the training examples it was given. The agents choose which are the next notes in the two-part melodies by negotiating over the possible combinations of notes suggested by the network.

1 Introduction

Whenever we are involved in any musical activity we are faced with symbolic and sub-symbolic processes. While listening, our aesthetic judgment is not necessarily achieved by following explicit rules. Applying a learning mechanism has been shown to be a convenient (and appropriate) way to overcome the explicit formulation of rules. Nevertheless, some of these aesthetic processes might have a symbolic representation and might be specified by a rule-based system.

In many musical styles, the composer needs to create different sequences of notes that will be played simultaneously. Each sequence should follow some aesthetic criterion, and in addition the sequences should sound appropriate when combined. This overall composition is the result of many interactions among its components. The musician achieves his overall result by compromising between the perfection of a single component and the combination of sequences as a whole. Thus, in this activity there is a constant tradeoff between the quality of a single sequence versus the quality of the combined group of sequences.

In this work, we are interested in aesthetic melodies produced by Neural Networks, while at the same time investigating how Distributed Artificial Intelligence (DAI) methods can be applied to solve the aforementioned tradeoff. The case study we have chosen to experiment with is the polyphonic vocal style of the sixteenth century; more specifically, we start with an investigation of two-part counterpoint

first species. We are currently implementing a neural network that learns how to produce each part of these melodies. We then regard the different components of the melodies as agents (in the DAI sense). These agents negotiate with one another to maximize the global utility of the whole system, which for our purposes should be interpreted as the global quality of the composition.

The main purpose of this work is to build a hybrid system architecture for composing polyphonic music. Composing such music is a human activity requiring intelligence, and thus an appropriate problem for AI research. This system is also useful as an application for composition and analysis.

We have decided to begin experimenting with two-part counterpoint melodies for two reasons. First, this is one way that humans acquire the skills to begin writing polyphonic music. Second, this is a relatively simple problem, and thus it is easier to train the network and verify its results. In spite of that, our current system can serve as a basis for further investigation of more complex musical problems.

2 The System Architecture

The focus of this work is on developing a system that is able to produce new two-part counterpoint melodies. We combine the symbolic and sub-symbolic levels to achieve this task. The overall design of our system reflects the cognitive process that a musician undergoes in producing such compositions.

2.1 The Human Musician's Approach

Usually, one part of the melody (*cantus firmus*) is given, and an upper or lower part is added to it. Both parts consist of whole notes. Issues that have to be considered regard the beauty of the melody itself, and the beauty of the combination of both parts. The second part should have the same aesthetic characteristics as the *cantus firmus*. In order to achieve this, the musician is involved in a cognitive process similar to a negotiation process. He has to compromise between the melodies notes by choosing from among the permitted notes those that are preferable.

2.2 The Computational Approach

In our system, we create both counterpoint parts dynamically. Our program, *NetNeg*, is composed of two main sub-systems. One sub-system is implemented by a modified version of Jordan's sequential neural network [Jordan86]. The second one is a two-agent model based on DAI.

2.2.1 The Connectionist Subsystem

Each part of the melody is produced independently by a neural network. Todd [Todd89] has already suggested a sequential neural network that can learn a sequence of melody notes. Currently, our neural network is based on the same idea.

The state units in the input layer and the units in the output layer represent the pitch and the contour. The state units represent the context of the melody, which is composed of the notes produced so far. The output unit activations vector represents the distribution of the predictions for the next note in the melody for the given current context. The role of the plan units is to label different sequences of notes. In the generalization phase, we can interpolate and extrapolate the values of the plan units so as to yield new melodies.

The actual input to the network is given in the plan units. At each step, the net is fed with the variable values in the state units together with the original plan units values. These values will cause the next element in the sequence to appear in the output layer and it will be propagated as feedback into the state units. The current values of the state units are composed of the previous values multiplied by a decay parameter and the current output values.

2.2.2 The DAI-Based Subsystem

We consider each part (i.e., voice) of the melody as an agent. Each agents goal is to compose his melody by choosing the *right notes*. On the one hand, he has to act according to the aesthetic criteria that exist for his voice; at the same time, he has to compose the voice in a manner compatible with the other voice-agent such that both together will result in a two-part counterpoint.

At every time unit in our simulations, each agent receives from the network a vector of activations for

all the notes among which he can choose. Were the agent alone in the system, he would have chosen the note that got the highest activation from the neural network, meaning that this note is the one most expected to be next in the melody. But, in order to compose species, both agents will have to negotiate [Rosenschein and Zlotkin94] over all the other possible combinations to obtain a globally superior result.

In principle, each agent can suggest any of the n possible notes received from the network. Not all of these combinations are legal according to the rules of the species. Moreover, there are specific combinations that are preferred over others for the current context. This idea is expressed in this module by computing a utility function for each pair of notes. In this sense, the goal of the agents is to agree on the pair of notes that is legal and has also achieved the maximal utility value among all other options.

3 Implementation

3.1 The Connectionist Module

We have implemented the Connectionist module in Planet [Miyata91]. We built a three-layer sequential net, that learns series of notes. Each series is a one part melody (i.e., *cantus firmus*). Each sequence of notes is labeled by a vector of plan units. The net is a version of a feedforward backpropagation net with feedback loops from the output layer to the state units. In the generalization phase, the net will be able to produce new *cantus firmi* by extrapolating and interpolating the values of the plan units.

The state units and the output layer can represent the notes in different ways. In this implementation, we choose to represent the notes as a vector of 19 units. The first eight units encode the pitch. The next nine units represent the intervals between the notes. The last two units describe whether the movement of the melody is ascendent or descendent. The intervals and the movement units describe the contour of the melody. These units together with the pitch units complement each other and result in a significant representation.

In the current state of our work, we have focused on two aspects. On the one hand, since we have trained the net with one part melodies, the range of the notes is very limited. Therefore, we needed to formulate rules in order to locate the second part relative to the first one. On the other hand, we are interested in two part melodies, in which the interval between any pair of tones might be up to a tenth. In order to extract more information out of the output units activations, we want to combine the pitch activations with the interval activations. We have mapped the activations of the output units into a vector of thirteen activations corresponding to the notes in more than an octave and a half. This mapping is based on the last

tone of the new melody built so far. If the activation of the ascendent movement unit is higher than the activation of the descendent movement unit, then we change each component in the 13-length vector as follows: first, we copy each pitch activation to the correspondent pitch component in the 13-length vector. Then, we find the interval between the previous chosen note and the note we are updating. Finally, we add the activation of this interval unit to the correspondent pitch component in the 13-length vector. Similarly when the descendent movement activation is higher than the ascendent movement activation.

Each agent receives the 13-length vector, and influences the context with their agreement. Then, the network will predict another output vector given this new context and the initial values of the plan units. This process continues sequentially until the melodies are completed.

It will be interesting in the future to investigate a modified model, where the agent-based component of the system influences the training phase. Thus, we could take into consideration other constraints that the agents could suggest, based on their rules, to enrich the learning abilities of the network.

3.2 The Agent-based Module

We have implemented the agent module using the Mice testbed[Montgomery et al.92]. Both agents are involved in a dynamic process in order to create their correspondent melodies. At each time unit, each agent gets a vector of 13 activation values corresponding to the activations of the tones in more than an octave and a half. This vector is the result of a mapping applied to the output vector of the network as has been explained in Section 3.1. Then the agents start a negotiation process at the end of which a new note is added to each of the current melodies. Each agent sends to the other all of its notes, one at a time, and saves the pair consisting of his note and the other agents note that is legal according to the first species style rules and that has yielded the maximal utility so far. At the end of this process the pair that has achieved the maximal utility is chosen. Both agents feed the network with this result as the current context so that the network can predict the next output. Each agent, then, receives a new input based on this output and the negotiation step is repeated until the melody is completed.

We define the system utility function to express the rules given by Jeppesen in [Jeppesen92]. A pair of notes is considered legal according to the following rules:

- 1 The intervals between pairs of notes in the two part melodies should not be dissonant.
- 2 There should be perfect consonance in the first and last place of the melody.

- 3 Unison is only permitted in the first or last place of the melody.
- 4 Hidden and parallel fifths and octaves are not permitted.
- 5 The difference between the previous and the current interval (when it is a fifth or an octave) should be two.
- 6 The interval between both tones cannot be greater than a tenth.
- 7 At most four thirds or sixths are allowed.
- 8 If both parts skip in the same direction, none of them will skip more than a fourth.
- 9 In each part, the new tone is different from the previous one.
- 10 We prefer not to have more than two perfect consonants in the two part contra-punct, not including the first and last notes.

The utility function is heuristic. Its values will be determined according to whether the pairs of notes are legal or illegal, and whether they are more preferred or less preferred. For example, the value of the function will increase when there is contrary motion. Thus, the system can produce contrary motion although the network might have suggested movement in the same direction.

4 Experiments

4.1 Running the Net

In the training phase, the network learned to reproduce four melodies that were taken from[Jeppesen92]. We have tested the performance of the network with different number of hidden units. The result we present regards to a net with 15 units in the hidden layer.

In the generalization phase, given a specific vector of plan units, the network will predict a vector of activations in the output layer. This vector can be interpreted as the distribution of expectations for the next note in the melody taking into consideration the contour of it. We have tested the net with different plan units vectors to create many new melodies (cantus firmus).

4.2 Running NetNeg

We have chosen two different plan vectors for the net that will output the notes for each agent. We run the net, each time with the correspondent plan vectors, and mapped their outputs to two different thirteen activation values. Then, we run the dai-based module with these inputs. The agents negotiate over the different pairs of possible combinations, computing for each the system utility. Finally, the agents will agree upon a legal pair of notes that has yielded the maximal utility or might decide that no combination is legal, given the previous note in the melody. In our current case, the nets are fed with the agents

agreement and the system continues to run. This process is executed until the two-part melodies are completed. Currently the length of the melodies is fixed.

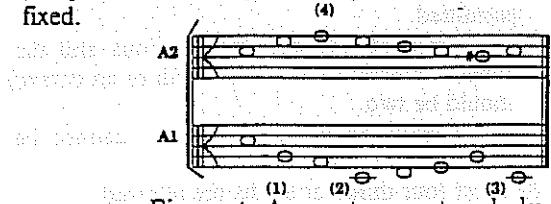


Figure 1: A new two-part melody

A melody that resulted from an experiment we performed with two different plan vectors ((0.8 0 0.8 0) and (0 1 0 1)) is given in Figure 1. From Figure 1, we can observe that the system gives aesthetic results, quite appropriate for the species style we have experimented with. Both melodies are consistent with the intervals constraints. Nevertheless, there is a contour problem as pointed in (1) and (2) in A1's melody in Figure 1. According to Jeppesen [Jeppesen92] it is preferred to descend by a step and then perform a descending skip. After a descending skip, we are expected to have a compensating ascending movement. In (3), we prefer to approach the last note by a step. A2's melody is perfect with regard to the contour. There is a single climax as shown in (4).

5 Summary

We have presented a novel computational approach for composing two-part counterpoint melodies. Our system design has been inspired by the cognitive process of a human musician, and the hybrid system is intended to broadly reflect that process.

We start by building a system composed of a connectionist module and an agent-based module. The agents decide upon the notes in both melodies. The neural network predicts the expectation for every note to be the next one in the melody. The agents then negotiate over the possible combinations of notes until they reach an agreement on notes that, added to the melody, will most greatly benefit the system. Our system is still under development. The current state of the system consists of a sequential neural network that is able to produce Dorian melodies. The agents can read their input based on the networks output and produce their result, that comprises the input data for the network. We are developing more natural ways to combine both modules. In their negotiation, the agents follow the rules taken from Jeppesen[Jeppesen92]. The system utility function considers the activations of the current notes, whether this pair is legal or not and it prefers contrary motion. We presented preliminary results of the system, namely the two-part counterpoint melodies produced by the program.

Acknowledgments: the discussions with Daniel Lehmann are gratefully acknowledged.

References

- K. Jeppesen: *Counterpoint The Polyphonic Vocal Style of the Sixteenth Century*. Dover, New York, 1992.
- M.I. Jordan: "Attractors dynamics and parallelism in a connectionist sequential machine", *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, Hillsdale, NJ.,1986.
- Y. Miyata: "A User's Guide to PlaNet Version 5.6", Computer Science dept., University of Colorado, Boulder, 1991.
- T.A. Montgomery, J. Lee, D.J. Musliner, E.H. Durfee, D. Damouth, Y. So: "MICE Users Guide", AI Lab. dept. of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan, February 1992.
- J. Rosenschein, G. Zlotkin: *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers*. MIT press, Cambridge, Massachusetts, 1994.
- P.M. Todd: "A connectionist approach to algorithmic composition", *Computer Music Journal*, Vol. 13, 1989 pp.27-43

The authors addresses:

Institute of Computer Science
The Hebrew University
Jerusalem, 91904 Israel
phone&fax: 972-2-658-5353 / 5439
email:{clag,dang,jeff}@cs.huji.ac.il

Department of Psychology
University of California at Berkeley
Berkeley, CA 94720 USA
Phone: 510-643-1370
Fax: 510-643-1371
E-mail: clag@berkeley.edu

Dept. of Computer Science
Hebrew University of Jerusalem
Jerusalem, 91904 Israel
Phone: 972-2-658-5353
Fax: 972-2-658-5439
E-mail: dang@cs.huji.ac.il

Dept. of Psychology
University of California at Berkeley
Berkeley, CA 94720 USA
Phone: 510-643-1370
Fax: 510-643-1371
E-mail: jeff@berkeley.edu

Algorithmic Composition as Gene Expression Based upon Fundamentals of Human Perception

Jeremy L Leach

School of Mathematical Sciences, University of Bath
Bath, BA2 7AY, England

Abstract

In this paper we present a theory relating the function of visual perception to the organisation of structure in music. We explain this structure and give examples where present in existing works. By separating the concepts of style and quality we are able to represent the information contained in a piece of music as a set of uncorrelated data. Using the above theory we present a generative algorithm which expands the data into a musical instance in an analogous way to the process by which genetic information is transformed into physical structure.

1 Introduction

This century has seen many attempts to analyse and formalise aspects of musical structure (Bent & Drabkin 1987). In some cases this work has led to systems capable of automatically assigning structural analyses to pieces of music (Jackendoff & Lerdhal 1983). More ambitiously though, others have attempted to produce formal systems which can automatically generate music (Todd 1991). Most of the latter have had only limited success. The reason for this, I would suggest, is that they appeal to no fundamental understanding of the function of music. Although by no means a trivial task, I believe that the development of such an understanding - even if incomplete - is of the utmost importance. Without this, we cannot even define objectively what the goal of algorithmic composition should be. In this paper, therefore, I attempt to explain the function of music which, I believe, is related to the way the brain has evolved in order to process sensory information.

It has often been said that music mimics the way natural processes change with respect to time (Pietgen & Saupe 1988, Leach & Fitch 1995). However, even if this is true, it must still be asked why music brings us so much pleasure. We know that the human brain has evolved to process sensory information derived from the kind of changes which occur in nature. By making relationships between events using generic versions of them to predict the future we give ourselves a survival advantage. This has been commented on by other authors (Baddeley 1993). I would suggest that the reinforcement of these

relationships could be linked to the sensation of pleasure. If this is so then it might explain why most widely acclaimed music is so rich in melodic, rhythmic and thematic relations. Thus, in order to create good music we must ask ourselves how these relationships can be formed and what the perceivable auditory phenomena are, such that relationships can be formed between them.

2 Visual Perception

In order to make associations between entities, it is necessary to perceive each entity as being distinct in some way from the others. Now, in the physical

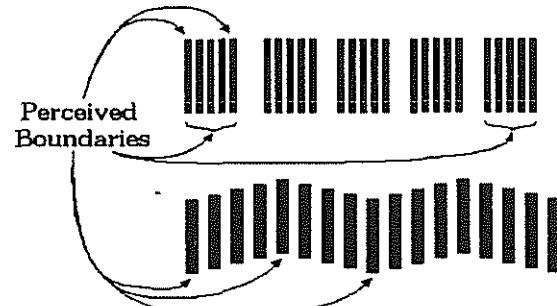


Figure 1: *Boundaries and discontinuities perceived in groups of regions.*

world, matter is more often than not grouped together in some form. Our visual perception has evolved to make use of this, and in general, objects are identified by detecting boundaries or discontinuities around otherwise homogeneous regions in the visual field. This fact was demonstrated by Hubel & Wiesel (1962) who discovered neurons in the cat's brain which are

activated at boundaries or discontinuities. Furthermore, it is also known that boundaries can be perceived at a higher "hierarchical" level and serve to separate groups of discrete regions (see figure 1).

This type of separation was observed by the Gestalt theorists. Both of the above types of perception can be seen to be evident in music. For example, the musical term "note" is useful precisely because it refers to an auditory stimulus, more often than not with a reasonably constant amplitude, and almost invariably with a constant pitch. Its boundaries with respect to amplitude are the attack and decay. If, however, there is little change in amplitude between notes then a rapid change in frequency serves as a transition. It is indeed this fact that allows us to represent music as a discrete sequence of symbols (i.e. the musical score). This is significant when you consider that performed music is in actuality a continuous phenomenon with respect to time, and there is no physical reason why it should fall so readily into groups.

Once again, but this time on a higher level, we find that auditory perception mimics the visual. In a study into pitch groupings (Deutsch 1982) it was found that memory for pitches of notes was strongest if they occurred "close" to the beginning or end of a melodic sequence.

So far though, we have not discussed the kind of relationships that can be formed between regions, once identified as being distinct. Clearly, the trivial relationship of "being different" is possible when regions share nothing in common apart from the fact that they *are* regions. However, the more similar a number of regions are to each other, the more our attention is drawn to the differences between them. This is because each region perceived is compared against already-formed memory models. Small differences between regions are ascribed to any contextual differences which are present. This is useful, since in the natural world, this allows a thinking organism on future occasions to predict such differences given the context of the region in question (again, further discussion in Baddeley 1993).

Forming relationships of this kind are important since differences which depend upon particular contexts account for a large part of the organisation in the natural world. The fact that the brain is good at recognising "similar" forms and relating only the differences to the context saves us from re-learning structures in their entirety.

Most music exhibits relationships of this kind. The context is of course in this case often temporal. The difference in context between two melodic sequences can be simply the fact that the second

succeeds the first. Some examples illustrate the point.



Figure 2(a): *Joseph Haydn, Sonata In G Major Hob. XVI:G, first movement, bars 1-4 and 8-11.*



Figure 2(b): *C.P.E.Bach, Sinfonie in D Major, H.663, first movement, bars 1-4, Second Violin.*

In figure 2(a) we recognise the beginning of the second sequence so we expect the ending to be the same. However, it is different and so this difference is attributed to the temporal context of "being the second sequence". 2(b) gives another example of the same phenomenon.

More rarely, a musical theme will feature a first and second part which share identical ends, but have different beginnings. In this case, although we don't recognise the beginning of the second part, we can "retrospectively" attribute it to a difference in temporal context once the end of the second part has been identified. This ability to be retrospective about perceived temporal information only serves further to demonstrate the closeness between visual and auditory perception (in the sense that temporal location can become time independent through use of memory much as spatial location is naturally time independent).

3 Separation of Quality and Style

So far, we have demonstrated a number of things; all of which suggest that the way we perceive visual stimuli is similar to the way we perceive structures in auditory stimuli. This kind of perception allows us to group regions in the spatial or temporal domain and assign properties to them. These groupings *can* be hierarchical. The relationships that we form between regions can either be simple or be between *components* of these regions involving contextual differences.

Now, if we create an algorithm which generates relationships between temporal regions conforming to the theories above, then we are saying that this is an objective definition of quality

in music; i.e. this kind of organisation in auditory stimuli is what makes the stimuli aesthetically pleasing and allows it to be termed "music".

However, for a composition to exist, it remains to identify the actual relationship *instances* that we wish to convey to the listener. Clearly though, if the aesthetics are not affected by the actual instances then they merely amount to a definition of the *style* of the piece.

Since different audiences like different styles it would seem reasonable to suggest that an individual's preferred style(s) is dependant on his or her socialisation.

As far as I am concerned therefore, these instances can be chosen for a specific audience or, indeed, chosen at random.

4 Formalisation of a Generative Algorithm

My basic approach was to construct an algorithm which could, by a process of subdivision, expand an initial starting unit into a complete composition with relationship instances explicitly defined by a given data set.

The first problem was to decide how to represent any intermediate stage between the starting region or "unit" and the complete work. If we must represent regions of stability with properties such as pitch, timbre or amplitude, then it would make sense to use such a region as a basic generic unit. Since this would be the only data structure that we use, it would mean that between every pair of regions there must be an *implicit* transition.

The second problem was to decide whether a parent region should be kept after it has been subdivided into a number of children, which would therefore, after the composition is complete, leave a complete tree structure describing the transformation from start region to finished composition.

I decided not to do this and opted for an intermediate representation which would only contain regional units with properties describing the surface phenomena.

The rationale behind this was that higher level groupings are perceived by a listener implicitly through the surface level organisation. There is no need to have "higher level" parent regions which embody concepts which may or may not be perceivable by means of the surface.

However, during construction of a piece, it may be necessary to know how a region should behave with respect to higher level temporal contexts. This means that in addition to surface properties, each region must keep a *record* of its ancestors. These ancestors would no longer exist at that point in the growth of the composition, but the record

can be referenced to see where the region used to belong in the previously existing hierarchy.

I shall now describe how the transformation takes place. If a region is to undergo a subdivision process, it does so according to the appropriate rule from a rule base. This rule base is the data set and is therefore an encoding of the relationships to be conveyed. These rules are of the form '**a->b**'. '**b**' takes the form of a number of symbols which represent the regions into which '**a**' must be subdivided. Each symbol has a unique integer to identify it and this integer is preceded by either a "**B**" (which stands for boundary region) or a "**R**" which represents a region *between* boundary regions.

For any "**B**" region we may have a rule which defines its subdivision thus: '**Bx -> Ba Rb Bc**' which means that **Bx** will be transformed from a boundary region into two smaller boundary regions, **Ba** and **Bc**, with a homogeneous **Rb** region separating them.

For any "**R**" region we may have a rule which defines its subdivision thus:

'**Rx -> Ra (Bb Ra)*n**' where for example, if **n** is equal to 3 then **Rx** will be transformed into **Ra Bb Ra Bb Ra**.

This means that a region in between two boundary regions may subdivide to express the fact that it is itself composed of **n+1** smaller identical regions separated by **n** identical boundary regions.

In addition, rules exist that dictate which properties should be conveyed by a region. These are of the form '**a := x1 x2**', where **x1** denotes the property, and **x2** the value which the property will hold.

Pitch relationships can be defined through a third type of rule; '**Bx + By := z**'. Pitch relationships can only be defined between boundary regions since a homogeneous "**R**" region consisting of many subregions may contain multiple pitches and hence cannot convey a single pitch.

Contextual relationships can only be created by referencing a region's ancestral record. We can appeal to these symbols which define the region's hierarchical context. We explain this by way of an example.

Rule Base:-

```
Ba -> Bb Rc Bd  
Bb -> Be Rf Bg  
Bd -> Be Rf Bg  
Be -> Bh Ri Bj  
Bb Bg -> Bk Rl Bm  
Bd Bg -> Bn Ro Bp
```

Transformation Sequence:-

a been slightly left towards east or horizontal **Stage 0**.
b been slightly right towards west or vertical
a.a.a been right towards west relatively from Stage 0.
b.c.d been a rotation of 90 degrees to **Stage 1**.
c been right towards west or vertical
a.a.a.a.a.a.a been right towards west relatively from Stage 0.
b.b.b.c.d.d.d been a rotation of 90 degrees to **Stage 1**.
e.f.g been right towards west relatively from **Stage 2**.
e.f.g been right towards west relatively from **Stage 2**.
a.a.a.a.a.a.a.a.a.a.a.a.a been right towards west relatively from Stage 0.
b.b.b.b.b.b.c.d.d.d.d.d.d been a rotation of 90 degrees to **Stage 1**.
e.e.e.f.g.g.g been right towards west relatively from **Stage 2**.
e.e.e.f.g.g.g been right towards west relatively from **Stage 2**.
h.i.j been right towards west relatively from **Stage 2**.
k.l.m been right towards west relatively from **Stage 2**.
h.i.j been right towards west relatively from **Stage 3**.
n.o.p been right towards west relatively from **Stage 3**.

In the transformation sequence each vertical column of alphabetic symbols at each stage represents a region. Each bold symbol represents the surface form of a region and the other symbols above them represent the ancestral record. At stage 1, rule 1 transforms '**a**' into '**b c d**' (note the "**B**"'s and "**R**"'s have been removed for brevity). At stage 2 both '**b**' and '**d**' are transformed into '**e f g**' by rules 2 and 3. Lastly, rules 5 and 6 appeal to the context of once being '**b**' or '**d**' to transform '**g**' into either '**k l m**' or '**n o p**'. Thus the "rule base" we have devised is essentially a hierarchically context sensitive grammar in the sense that the production rules and relationships depend not only on the region's current form, but also on where the region has *come from*.

5 Analogy to Genetic Expression

On the simplest level, the system invites comparison to a biological system from the way in which a work "grows" from a fundamental "cell" to a "complete" composition. The subdivision process is reminiscent of cell subdivision in an organism and, the complete rule base is accessible by all regions during generation, much as all cells contain a full copy of the genetic code for their biological system.

The analogy goes further though. Since we use a "hierarchically" context sensitive grammar, a region's further development can depend heavily upon its ancestral history, and hence depend upon its temporal position. The ancestral history is not apparent just by examining the surface. This kind of context sensitive development can be seen during the development of an organism where, for example, some cells may develop into one structure such as a heart, and others into another structure such as a brain. A more subtle example is the development of say, the hands in a human. In this case the left hand develops an opposite symmetry to the right. Clearly though, some genetic code is shared by both hands since genetic

mutations can result in the same defects appearing in each hand.

6 Conclusion

The algorithm mentioned above has been implemented and is capable of producing works such as that of figure 3. At present, instances of the rule base are determined automatically by iteratively choosing a production rule at random and splitting it by inserting extra rules as necessary between its left and right hand side.

Ultimately, the theories' worth can only be judged by the quality of the music they produce. The works produced to date have been subjectively judged by colleagues to be of reasonable quality and certainly musical.

A more directed approach could allow the listener to choose one rule base from a number of mutations to be used in the next generation. This would constitute an evolutive system and allow the user to converge on desired styles.



Figure 3: A small melody composed by the algorithm.

References

- I.Bent,W.Drabkin:** *Analysis*, Macmillan Press 1987.

R.Jackendoff, F.Lerdahl: *A Generative Theory of Tonal Music*, Cambridge, Massachusetts, USA: MIT Press, 1983.

P.Todd: *Music and Connectionism*, MIT Press, 1991.

H.Pietgen, D.Saupe: *The Science of Fractal Images*. Berlin: Springer-Verlag, 1988.

J.L.Leach, J.P.Fitch: "Nature, Music and Algorithmic Composition.", in *Computer Music Journal* 19(2):pp. 23-33, 1995.

A.Baddeley: "Working Memory and Conscious Awareness", in *Theories of Memory*, Edited by A.Collins *et al.*, Lawrence Erlbaum Associates Ltd., 1993.

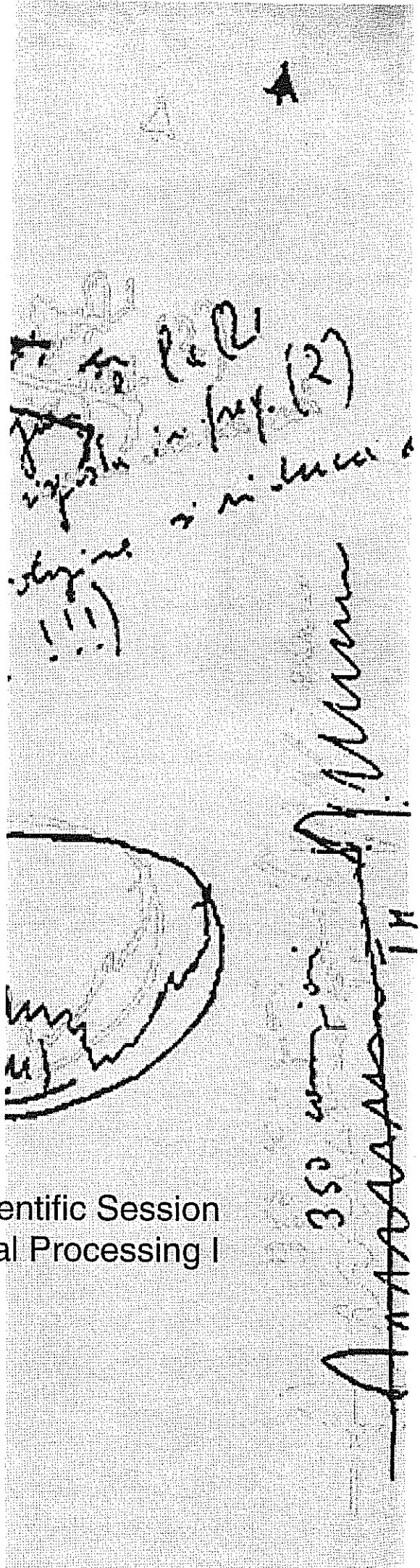
D.Deutsch: "Grouping Mechanisms in Music", in *The Psychology of Music*, pp. 99-134, Academic Press Inc. 1982.

D.H. Hubel, T.N. Wiesel: "Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex", in *J. Physiology* 1962 Vol 160, pp. 106-154.

The author can be contacted at the address on the first page, or more preferably, by e-mail at:

ill@maths.bath.ac.uk

Scientific Session
Digital Signal Processing I



John L. Lewis

Musical Applications Using Real-time Frequency Domain Signal Processing

Zack Settel
Lysis : Ircam
1 place Igor Stravinsky
75004 Paris, France
settel@ircam.fr

Cort Lippe
University of Buffalo: Music Dept.
222 Baird Hall: Box 604700
Buffalo, New York 14260-4700
lippe@acsu.buffalo.edu

Abstract

This paper presents real-time musical applications using the IRCAM Signal Processing Workstation which make use of FFT/IFFT-based resynthesis for timbral transformation in a musical context. The paper continues on the author's work, introduced at the X Colloquio di Informatica Music, and features improved and new algorithms, techniques and applications. The paper focuses extensively on real-time phase vocoding and discusses new techniques for high quality time expansion (stretching) of sounds. Additionally, the authors discuss filtering applications, which permit the separation of noised and non-noised signal components, amplitude-independent de-noising, the use of dynamic range processing in cross-synthesis, and granular synthesis in the frequency domain. As with our previous work, emphasis is placed on providing musicians with potentially greater influence on their sound transformations, based on their interpretation, and/or choice of musical material. Many of the applications presented by the authors feature low-dimensional control of sound generators and (highly responsive) signal processors, whose output can reflect, or exaggerate subtle changes in the input signal.

The ability to work in the frequency domain in real time allows for kinds of empirical learning and understanding, which have not been generally possible hitherto. The Ideas and musical applications discussed in this paper, offer composers an intuitive approach to timbral transformation in electro-acoustic music, and new possibilities in the domain of live signal processing.

Key words

convolution, time stretching, cross-synthesis, FFT/IFFT, ISPW, Max, dynamic range processing, de-noising, granular synthesis (frequency domain)

1. Introduction

The Fast Fourier Transform (FFT) is a powerful general-purpose algorithm widely used in signal analysis. FFTs are useful when the spectral information of a signal is needed, such as in pitch tracking or vocoding algorithms. The FFT can be combined with the Inverse Fast Fourier Transform (IFFT) in order to resynthesize signals based on analyses. This application of the FFT/IFFT is of particular interest in electro-acoustic music because it allows for a high degree of control of a given signal's spectral information (an important aspect of timbre) allowing for flexible, and efficient implementation of signal processing algorithms.

This paper presents real-time musical applications using the IRCAM Signal Processing Workstation (ISPW) [Lindemann, Starkier, and Dechelle 1991] which make use of FFT/IFFT-based resynthesis for timbral transformation in a compositional context. Taking a pragmatic approach, the authors have developed a user interface in the Max programming environment [Puckette, 1988] for the prototyping and development of signal processing applications intended for use by musicians. Development in the Max

programming environment [Puckette, 1991] tends to be simple and quite rapid: digital signal processing (DSP) programming in Max requires no compilation; control and DSP objects run on the same processor, and the DSP library provides a wide range of unit generators, including the FFT and IFFT modules. Techniques for filtering, cross-synthesis, noise reduction, and dynamic spectral shaping have been explored, as well as control structures derived from real-time signal analyses via pitch-tracking, envelope following, noise gating, and signal compression and expansion[Lippe & Puckette 1991]. These real-time musical applications offer composers an intuitive approach to timbral transformation in electro-acoustic music, and new possibilities in the domain of live signal processing that promise to be of general interest to musicians.

2. The FFT in Real Time

Traditionally the FFT/IFFT has been widely used outside of real-time for various signal analysis/re-synthesis applications that modify the durations and spectra of pre-recorded sound [Haddad & Parsons 1991]. With the ability to use the FFT/IFFT in real-time, live signal-processing in the

spectral domain becomes possible, offering attractive alternatives to standard time-domain signal processing techniques. Some of these alternatives offer a great deal of power, run-time economy, and flexibility, as compared with standard time-domain techniques [Gordon & Strawn 1987]. In addition, the FFT offers both a high degree of precision in the spectral domain, and straightforward means for exploitation of this information. Finally, since real-time use of the FFT has been prohibitive for musicians in the past due to computational limitations of computer music systems, this research offers some relatively new possibilities in the domain of real time.

2.1 Algorithms and basic operations

All of the signal processing applications discussed in this paper modify incoming signals and are based on the same general DSP configuration. The DSP configuration includes the following basic steps: (1) transformation of the input signals into the spectral domain using the FFT, (2) operations on the signals' spectra, (3) and resynthesis of the modified spectra using the IFFT. Operations in the spectral domain (FFT data in the form of rectangular coordinates) include applying functions (often stored in tables), convolution (complex multiplication), addition, taking the square root (used in obtaining an amplitude spectrum), noise gating (of both frequency and amplitude), and compression and expansion. Differences in the choice of spectral domain operations, kinds of input signals used, and signal routing determine the nature of a given application: small changes to the topology of the DSP configuration can result in significant changes to its functionality. Thus, we are able to reuse much of our code in diverse applications. (For example, though functionally dissimilar, high-resolution filtering and subtractive synthesis differ only slightly in terms of their implementation.)

3. Applications

3.1 Complex spectral envelopes and cross synthesis

A static spectral envelope used in a simple filtering application or in subtractive synthesis can be drawn by hand or obtained through signal analysis. In cross synthesis a continuously changing (dynamic) filter can be created by doing an FFT analysis of a signal (signal B) and extracting its spectral envelope, or amplitude spectrum, which describes how another signal (signal A) will be filtered. Thus, the pitch/phase information of signal A and the time

varying spectral envelope of signal B are combined to form the output signal. Audio signals produced by standard signal processing modules such as frequency modulation (FM) are of particular interest for cross synthesis because they can produce rich, easily modified, smoothly and nonlinearly varying spectra [Chowning 1973] which can yield complex time varying spectral envelopes. Other standard signal processing techniques, such as amplitude modulation (AM), additive synthesis, or a band-pass filtering offer rich varying spectral information using relatively simple means, with few control parameters. One of the advantages of using standard modules is that electronic musicians are familiar with them, and have a certain degree of control and understanding of their spectra.

It should also be noted that interesting transformations can be produced by simply convolving signal A's spectrum with signal B's spectrum (keeping both the phase and amplitude information of signal B as well as signal A). In this case, the phase (frequency) and amplitude information from each signal figures in the output signal.

3.2 Mapping qualities of one signal to another

Musically, we have found that in some cases, the relationship between signal A and signal B can become much more unified if certain parameters of signal A are used to control signal B. In other words, real-time continuous control parameters can be derived from, for instance, signal A, and used to control signal B. For example, the pitch of signal A can be tracked and applied to signal B (an FM pair) to control the two oscillators' frequencies. Envelope following of signal A can yield expressive information which can be used to control the intensity of frequency modulation (FM index) of signal B.

3.3 Frequency dependent spatialization

In the spectral domain, the phases of a given signal's frequency components can be independently rotated in order to change the component's energy distribution in the real and imaginary part of the output signal. Since the real and imaginary parts of the IFFT's output can be assigned to separate output channels, which are in turn connected to different loud-speakers, it is possible to control a given frequency's energy level in each loud-speaker using phase rotation.

3.4 Band-limited energy dependent noise-gating

In the spectral domain, the energy of a given signal's frequency components can be independently modified. Our noise reduction algorithm is based on a technique [Moorer & Berger, 1984] that allows independent amplitude gating threshold levels to be specified for each frequency component in a given signal. With a user-defined transfer function, the energy of a given frequency component can be altered based on its intensity and particular threshold level. This technique, outside of being potentially useful for noise reduction, can be exaggerated in order to create unusual spectral transformations of input signals by emphasizing or masking specific components.

3.5 Band-limited energy dependent noise-gate

In the spectral domain, the energy of a given signal's frequency components can be independently modified. Our noise reduction algorithm is based on a technique [Moorer & Berger, 1984] that allows independent amplitude gating threshold levels to be specified for up to 512 frequency band-limited regions of a given signal. With a user-defined transfer function, the energy in a given frequency range can be altered based on its intensity and a user-specified threshold level. This technique, besides being potentially useful for noise reduction, can be exaggerated in order to create unusual spectral transformations of input signals, resembling extreme chorusing effects. Using non-linear transfer functions, it is possible to modify the relative intensities of the input's frequency components, allowing for example, masked or less important components to be emphasized and brought to the aural foreground.

3.6 Band-limited frequency dependent noise-gate

Similar to the noise-gate described above, this module functions independently of gain; its output depends on the stability of the frequency components of the input signal. Using a technique borrowed from phase-vocoding [Gordon & 1987], time-varying frequency differences of components in a given band-limited region of the spectrum are used to determine the stability of those components. Pitched components in the input signal tend to be stable and can thus be independently boosted or attenuated.

3.7 Improvements to frequency domain-based time expansion and compression techniques

Useful techniques for sound manipulation in the frequency domain are proposed by the phase vocoder [Dolson 1986; Nieberle & Warstat 1992]. One particularly useful application of this technique, popularly referred to as time stretching, allows a sound's length to be modified independently of its pitch (frequency content). However, this application has two particular shortcomings which have been revealed in practice—especially when a sound's duration is lengthened. The first problem is that transient components, normally quite brief, become "unusually long". The second problem is that noise components tend to become "less agitated" since their rate of change is not preserved during time stretching. These shortcomings can be clearly heard for example, when time stretching speech. A spoken text rapidly loses intelligibility when: A) transients are protracted; consonants are no longer recognizable and loose their functional role as articulators, and B) noise components are "de-agitated"; sibilant components such as "sh", loose their distinctive "windy" noisy quality.

In order to overcome these two problems, we have made the following additions to our phase vocoding algorithm:

- 1) Selective time stretching, which acts only on non-transient portions of the signal.
- 2) A resynthesis stage that approximates the original rate of change (with respect to frequency) of noise components, using statistical approximations.

4. Combining processing modules

The modules described above may be placed in series in order to perform multiple processing operations on a sound. Since most of the cost of processing occurs in time/frequency domain conversion (FFT and IFFT), modules operating in the frequency domain may be ganged together with great efficiency, requiring no additional steps of conversion.

4.1 Dynamic range processing with graphic EQ

The 512 band filter module described above may be easily connected to another module, providing the additional possibility of high resolution filtering.

5. Input scaling and additional control parameters for cross-synthesis

When two input signals of similar intensities and spectral distributions (eg.: two singers' voices) are convolved, the resulting spectral distribution can be quite different. In such a case, strong "low-mid-range" components typically become much louder, while weaker higher frequency components virtually disappear. For example, when convolving a signal by itself, the resulting energy of a given component is its square, as shown below:

$$\text{resulting energy} = \text{signal} * \text{signal}$$

By scaling the input or output signal(s), the "original" spectral distribution can be preserved. If in the above example, we take the square-root of the output signal, a given component's energy is now the logarithmic mean of the energy of its input signals, and not the square. In the case above, we recover the original input signal.

$$\text{resulting energy} = \sqrt{\text{signal} * \text{signal}} = \text{signal}$$

5.1 Including dynamic range processing in cross-synthesis algorithms

The algorithms for cross-synthesis discussed above can be expanded to include additional processing units, that allow for scaling and provide more control parameters. A compressor/expander based on the energy dependent noise-gate mentioned earlier is applied to one of the two input signals before convolution. The choice of compression/expansion functions determines the way the energy in the output signal will be scaled; certain choices (such as a square root function) help preserve the spectral distribution of the input signals. Thus, two input signals of similar intensities and spectral distributions will combine to form a spectrum whose components' relative energies remain little changed. By choosing a compression or expansion function that boosts or cuts the energy of weaker components in one of the input signals, the degree of spectral intersection of the two inputs can be specified. The compression or expansion ratio parameter (or intensity of compression or expansion) provides dynamic control over the "degree of spectral intersection" of the two input signals. This parameter is particularly useful when cross-synthesizing dissimilar sounds such as voice and percussion.

6. Future Directions

The authors are currently working on alternative methods of sampling and granular synthesis that operate the frequency domain, based on real-time phase vocoding [van der Heide, 1994]. At present we are able to modify a sound's spectrum and duration independently, and are working towards being able to perform pitch transposition independently of the spectral envelope (formant structure), thus allowing one to change the pitch of a sound without seriously altering its timbral quality. Additionally, we are exploring techniques for smooth sample looping and cross-fading between sounds.

7. Summary

With the arrival of the real-time FFT/IFFT in flexible, relatively general, and easily programmable DSP/control environments such as Max, non-engineers may begin to explore new possibilities in signal processing. Though our work is still in its early stages, we have gained some valuable practical experience in manipulating sounds in the spectral domain. Real-time convolution can be quite straightforward and is a powerful tool for transforming sounds. The flexibility with which spectral transformations can be done is appealing. Our DSP configuration is fairly simple, and changes to its topology and parameters can be made quickly. Control signals resulting from detection and tracking of musical parameters offer composers and performers a rich palette of possibilities lending themselves equally well to studio and live performance applications.

Acknowledgments

The authors would like to thank Miller Puckette, Bennett Smith and Stefan Bilbao for their invaluable technical and musical insights.

References

- Chowning, J. 1973. "The Synthesis of Complex Audio Spectra by means of Frequency Modulation" Journal of the Acoustical Society of America 21(7), 526-534.
- Dolson, M. 1986. "The phase vocoder: a tutorial", Computer Music Journal, 10(4), 14-27

Gordon, J. and Strawn J. 1987. "An introduction to the phase vocoder", Proceedings, CCRMA, Department of Music, Stanford University, February 1987.

Haddad R, and Parsons, T. 1991. "Digital Signal Processing, Theory, Applications and Hardware", New York: Computer Science Press

Heide van der, E. 1993. Private communication.

Lindemann, E., Starkier, M., and Dechelle, F. 1991. "The Architecture of the IRCAM Music Workstation." Computer Music Journal 15(3), 41-49.

Lippe, C. and Puckette, M. 1991. "Musical Performance Using the IRCAM Workstation", Proceedings of the 1991 International Computer Music Conference. San Francisco: International Computer Music Association.

Moorer and Berger, 1984. "Linear-Phase Bandsplitting: Theory and Applications", Audio Engineering Society (preprint #2132), New York: 76th AES convention 1984.

Nieberle, R and Warstat, M 1992. "Implementation of an analysis/synthesis system on a DSP56001 for general purpose sound processing", Proceedings of the 1992 International Computer Music Conference. San Francisco: International Computer Music Association.

McNally, G. W., 1984. "Dynamic Range Control of Digital Audio Signals." Journal of the Audio Engineering Society, Vol. 32, 1894.

Puckette, M. 1988. "The Patcher." Proceedings of the 1988 International Computer Music Conference. San Francisco: International Computer Music Association.

Puckette, M., 1991. "FTS: A Real-time Monitor for Multiprocessor Music Synthesis." Music Conference, 420-429, San Francisco: International Computer Music Association.

Smith, Bennett K., 1994 Personal communication.

1970. Comisión de la Asamblea

Provincial que se reunió en

la Provincia de La Rioja

en la ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

En su reunión se acordó:

que el Presidente de la Asamblea

de la Provincia de La Rioja

se presentara a la reunión

de la Comisión de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

que el Presidente de la Asamblea

Provincial que se reunió en la

ciudad de San Salvador de Jujuy

el día 19 de junio de 1970.

Real-time Polyphonic Time-shifting of Sound with Interactive Systems

Agostino Di Scipio

Laboratorio Musica e Sonologia, Università di L'Aquila

Via Vetoio, 67100, L'Aquila, Italy

tel: +39 862 433892 / 433130 email: lms@vxscqaq.aquila.infn.it

Abstract

This paper describes methods of real-time granular processing of sound adopted in recent compositions by the author. Discussed are also the interactive control structures available and/or specially designed for those compositions using two computer music systems, namely KYMA-CAPYBARA33 (at LMS) and the PODX (at Simon Fraser University). Special emphasis is given to polyphonic processes of time-shifting and to recursive granulation. Beside some technical details, the paper also shows the role of such strategies in the particular context of microcomposition.

1 Introduction

In this paper I would like to describe methods of granular processing of sound utilized in two recent works of mine, *Hybris* and *Essai du Vide. Schweigen*. In both cases I utilized time-shifting of sound (altering the sound duration without - in principle - side-effects in the frequency domain) by means of granular processing techniques.

1.1 Granular time-shifting as a technique for microcomposition

The particular time-shifting methods discussed here should be seen as instances of micro-time sonic design, or *microcomposition* (with a term of the electroacoustic music jargon). The common feature of the many different approaches of microcomposition lies in the possibility of manipulating microtemporal relations among myriads of minimal sonic quanta. For such a reason, these methods are often operated by means of compositional control structures which instantiate global strategies of statistical nature. I assume, however, that any strategy of microcomposing deals with a specific problem domain: in which way minimal, time-limited units should be *composed* in order to give rise to high-level structural properties. This is a problem of perception (Bregman, 1990, p.118 and *passim*) and, at the same time, a problem of music theory - hence of computer music software design - the discussion of which involves issues of relevance to the aesthetic potential of electroacoustic and computer music (Di Scipio, 1995; 1994a).

1.2 A notion of interactivity

Another central point in this paper relates to the notion of *interactivity*. In principle interactive control structures embody knowledge about the

continuing exchange between *concept* and *percept* in the course of composition and/or performance, knowledge about the way in which the composer/performer adjusts his/her own actions and goals during the creative process. Interactivity requires the possibility of exerting real-time controls over various parts of a program such that both the sonic and syntactic levels can be accessed by the user in an attempt to establish a link between them. Hence "interactivity" cannot simplistically mean "immediate audible output" - it also means that the user can address his/her action to different rates of change in the musical flow, from audio-rate through event-rate and higher. We should make a distinction between interactive *composition* systems and interactive *performance* systems. Secondly, between interactive *programs* and interactive *programming environments*. Accordingly, there are 4 general classes of interactive music systems (for a more detailed taxonomy, see Rowe, 1993):

	program	environment
composition	1	2
performance	3	4

The distinction between program and environment is primarily a question of the particular computational paradigm at work. A classical notion of computer science is that a *program is a theory*, resulting in a procedural description leading to the solution of a specific problem. The user uses a theory put forth (or relied upon) by the designer (Simon & Newell, 1970). On the other hand an *environment is a meta-theory*, a theory about how to instantiate and verify either new or existing theories. In recent years, this has often taken the form of object-oriented representations.

The following two sections describe my recent experiences with granular processing using a programming environment for composition and live performance (cases 2 and 4, above) and with a

program for interactive composition (case 1). These experiences revealed to me the relevance of interactive controls for perspectives of micro-time sonic design, a most peculiar and fertile terrain for electroacoustic composition (Di Scipio, 1994b).

2 Polyphonic time-shifting

Granular representations of sound (also wavelet representations) lend themselves to techniques operating in the frequency- and the time-domain independently. Various authors have demonstrated theoretical and practical aspects of granular time-shifting of sound (Jones & Parks, 1988; and various articles in De Poli et al., 1992). My interest is not so much for the mere compression or dilation of the duration of sound. Rather, it led me to devise processes possibly functional to the emergence of dynamical gestures and polyphonic textures within and through the sound of instruments played live. Two facets must be taken into account: (2.1) the real-time generation of streams of grains captured from the live sound, and (2.2) the algorithmic control structure which allows me to handle different streams of grains, each with its specific parameters.

2.1 Granulation of live sound

The input signal is continuously recorded into a n second long wavetable memory, wrapping around the same chunk of memory m times. When the cyclic recording starts, the wavetable is also cyclically looked-up and its samples picked-up. The current look-up memory position is itself a function of time, either another wavetable or a virtual slider on the computer screen. A minimum look-up sampling increment is prescribed whose duration in samples equals the grain duration, here ranging from 10 to 70 msec. Hence the pointer to the input memory buffer proceeds linearly only within the grain-time (from the first to the last grain sample), then jumps elsewhere in the buffer as from the function or real-time slider by which it is controlled. If you use a ramp, then the linear direction of time is preserved though duration can be shortened or lengthened. If you use, for example, a gaussian function, then initially the direction of time will be preserved, with an s-shape like accelerando, then it will be reversed, with a reversed s-shape ritardando. There's no prescription as to how the look-up pointer should move through the input memory buffer. Hence not only dilation, but also compression of the sound duration can be achieved.

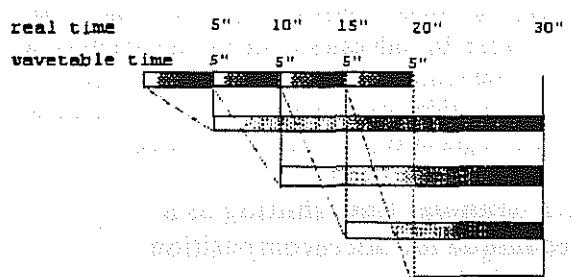
Samples picked-up from the input buffer are enveloped with gaussian or trapezium curves. What I consider a "stream of grain" is actually made of two "granulators" (look-up + envelope + allpass filter), shifted by half grain-duration with respect to one another. The signal of both

granulators is allpass filtered with an average delay time equal to half grain-duration. Intergrain delay and allpass delay are given random values within prescribed limits.

This can be seen as a form of "asynchronous" granulation. However, micro-level time parameters (intergrain delay, grain duration, onset delay) can be made functions of the pitch of the input signal (if any pitch is tracked) and set in synchronous relation to the detected periodicity.

2.2 Algorithmic control in *Hybris*

For the performance of *Hybris* (g-flute, bassclarinet and computer processing, 1994), I designed a control structure which activates the following: the input signal is continuously recorded into a 5" wavetable, wrapping around 4 times; 4 different processes of granulation and time-shifting start at 5", 10", 15" and 20"; each process has its own low-level parameters and time-shift ratio - respectively 5, 4, 3, 2 times slower than real. The look-up is driven by linear ramps (time direction preserved).



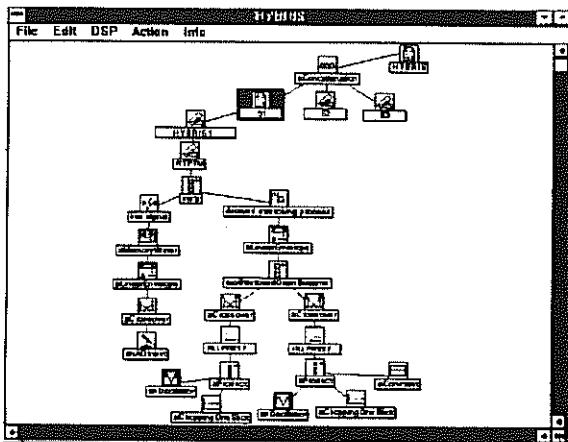
Depending on the shift ratio and the performers' nuances of timing, some input material eventually will not be granulated; this happens when wavetable locations are not looked-up before new material comes in there (the case with large shift ratios). Other material, however, may undergo more than one process of granulation and time-shift; this happens when locations are looked-up by two or more granulators at the same time. A clear picture of the overall process can be drawn only by studying the phase relationship between the write and read pointing functions in relation to the 5" wavetable memory buffer.

The process lasts 30 seconds and represents one single formal unit in the first section of *Hybris* (made of 4 such units). It yields a polyphony of gestures and textures emerging from the instrumental performance. Usually there will be a growing density of sound as the process approaches the end point. The average density will be of a few hundreds grain/sec, depending on the amount of feedback in the allpass filters.

2.3 Implementation in KYMA

At the time of the first performance of *Hybris* (with M.Zurria, flute, and P.Ravaglia, bassclarinet,

Rome, December 1994) the above method had been implemented using the KYMA2.0 software (under MS-Windows) controlling the CAPYBARA DSP system (Scaletti & Hebel, 1991); then it was ported to KYMA4.0 for a recent performance (Montreal, ISEA95). A representation of the HYBRIS1 algorithm is found below, with the KYMA objects (or Sounds), hierarchically ordered from top down.



In this graph both the HYBRIS1 Sound and its subSound, RTPTM (real-time polyphonic time modifications) actually are quite simple and short Smalltalk-80 programs, the first scheduling 4 instances of the second, the second scheduling 4 instances of the INSTR Sound. Together they represent, in effect, the highest level *control structure* by which the variables reputed of primary compositional interest are evaluated during performance.

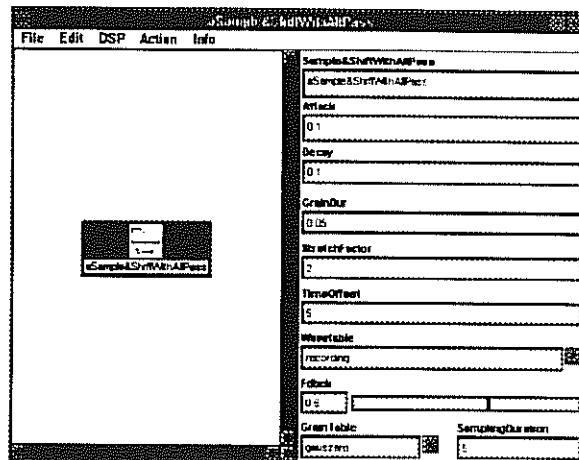
```
"script in HYBRISI"
1 to: 4 do: [ :i | RTPTM start: ((i*30)-30)
seconds.].  
  
"script in RTPTM"
1 to: 4 do: [ :i | instr start: ((i*5)-5)
seconds grainDurl: 0.04 stretchFactor: (6-
i).].
```

A single granulator corresponds to the abstraction level of the INSTR Sound. After debugging and optimizing, I made one only object of the whole substructure of INSTR, a new Sound class aSample&ShiftWithAllPass, with its own icon and parameters - those expected to change during rehearsing and performing (all other parameters being customized). With this new prototype (see graph below), HYBRIS1 could be re-shaped in a more compact and efficient representation.

2.4 Some considerations

A look at the parameters of this new prototype, would show that its functionality is primarily a question of time variables (indeed this is the case with *Hybris'* first section, but wouldn't be with sections 2 and 3, not described here). The sounding

results include phase-related effects especially interesting when the live sounds extend to breathing, hisses, whistles, noise bursts (e.g. tonguing), etc...



As a new class, `aSample&ShiftWithAllPass` features custom spatial trajectories which aid the perceptual segregation of grains in separate auditory streams and the perception of a polyphony of distinct gestures in the stereophonic space - rather than a single "wall" of sound.

Interactivity in *Hybris* is observable at two levels. In the compositional process, the KYMA user interface allowed me to search for the appropriate models of sound transformation. This is *not* just a matter of editing, testing and optimizing the algorithms in a straightforward deterministic manner. It is more like analysing the link among acoustic, perceptual and more abstract, conceptual aspects of the sound processing technique: exploring how the organization of low level details could achieve auditory images of peculiar timbral properties.

At another level, interactivity is paramount in the performance of *Hybris*. The sounding effects of the sound processing method is particularly sensitive to the details of the input signal: the performers cannot always predict the (sometimes dramatic) effects caused by the magnification of rather unperceivable differences in their performance. As he/she is urged to dynamically adapt the performance in timing and timbral qualities, the performer becomes here a source of feedback and self-regulation within a dynamical system. Far from being a just matter of reproducing written symbols into audible sounds, his/her task is more one of self-regulating the whole "performance system" in order to avoid totally uncontrolled results as much as strictly periodic behaviors.

3 Recursive processing

I would like to conclude with a short description of a particular extension of granular processing and

time-shifting, that I found very rich in implications. I refer to it as recursive granular processing. To give the reader a concrete idea of its functionality, I describe how recursive granulation was used in the composition of *Essai du vide. Schweigen*, a solo tape work composed in 1993 while I was visiting composer at Simon Fraser University (Burnaby BC). Among the many programs of B. Truax' PODX set of programs for interactive composition and sound synthesis, the resource I utilized the most was GSAMX, which allows real-time granular processing of sound. I refer the reader to (Truax, 1991; 1994) for a discussion of the technical details.

3.1 Details of *Essai du vide. Schweigen*

I devised a strategy of recursive granulation in two steps: 1) a sampled sound is submitted to granular time-shifting and the result is recorded on hard disk; 2) the time-shifted sound is randomly accessed in order to select short fragments of it, resulting in a new pattern of the selected chunks of sampled sound. The result of step 2 is then passed back through step 1 and time-shifted, then again randomly accessed, and so on... In other words, the strategy involved a nonlinear transformation comparable to the iterated application of two transfer functions to an initial datum:

$$x_{n+1} = f_b(f_a(x_n))$$

x_0 : original source sound; x_n : granulated sound after the n-th iteration; f_a : time-shift function; f_b : random access rule

Similar strategies I had already utilized in *Zeitwerk* (tape music, 1992) and *Kairós* (soprano sax and tape, 1992) (Tisato & Di Scipio, 1993; Di Scipio 1994b). In such works they were to transform synthetic sounds, while in *Essai du vide*, instead, the original source was "concrete" in nature, the sound of scratching with nails and jazz brushes against the lower strings of a harp, followed by a short pause. Currently a real-time implementation of recursive granulation is in the course of realization for two new works (*Lichtung. Variazioni sul ritmo del vento I*, for baritone sax and computer processing, and *II*, for contrabass recorder and tape).

3.2 An exploratory process

The control variables in the realization of *Essai du vide* were: average grain duration, duration range, average intergrain delay (or density, i.e. number of grains/second), intergrain delay range and time-shift ratio. Interactive control in took place via a single line on the screen of the PDP Micro11 computer controlling the DMX-1000 DSP. The control variables could be accessed one at a time during the synthesis process. They could also be

"ramped", achieving a simple time-varying automated control. All steps in the composition involved grains shorter than 50 msec. The time-shift ratio in f_a was cyclically ramped, with an average shift of 30 times slower than real. The random access rule f_b was a Poisson distribution, and the fragments recursively picked-up were 170 msec long. I could explore the audible effects of micro-level modulations with interesting auditory properties due to the changing relationship of phase among overlapping grains, and shape extended structures including both continuous textures of sound and more dynamical gestures, also featuring silent pauses (the magnified effect of the short pause in the source material). The iterated transformations produced a "vaporization" of the original source and brought forth gestural patterns absolutely unpredictable at the outset.

Recursive granulation in *Essai du Vide. Schweigen* provides an example of an operating creative metaphor. It is a method for subtracting energy from the sound, in opposition to methods of granular synthesis: in the former case one puts "quanta of silence into the sound", in the latter "puts quanta of sound in the otherwise silent flow of time". In this way the sense of void, internal silence and solitude which is behind the title, was given both an actual significance and, I think, a chance of being concretely conveyed to the listeners.

References

- Bregman, A. *Auditory Scene Analysis*. MIT Press, 1990.
- De Poli, G., A. Piccialli & C. Roads (eds.) *Representations of Musical Signals*, MIT Press, 1991.
- Di Scipio, A. "Formal Processes of Timbre Composition Challenging the Dualistic Paradigm of Computer Music", in *Proceedings ICMA*, 1994a.
- Di Scipio, A. "Micro-time sonic design and the formation of timbre", *Contemporary Music Review*, 10(2), 1994b.
- Di Scipio, A. "Inseparable models of material and of musical design in electroacoustic and computer music", *Journal of New Music Research*, 24(1), 1995.
- Jones, D. & T. Parks, "Generation and organization of grains for music synthesis", *Computer Music Journal*, 12(2), 1988.
- Rowe, R., *Interactive Music Systems. Machine listening and Composing*. MIT Press, 1993.
- Scaletti, C. & K. Hebel, "An Object-based Representation of Audio Signals", in (De Poli et al.) *Representations of Musical Signals*, MIT Press, 1991.
- Simon, H. & A. Newell, "Information Processing in Computer and Man", in (Pylyshyn, Z. ed.) *Perspectives on the Computer Revolution*. Prentice-Hall, 1970.
- Tisato, G. & A. Di Scipio, "Granular synthesis with the Interactive Computer Music System", in *Proceedings CIM, AIMI/LIM*, 1993.
- Truax, B. *Real-time Granular Synthesis with the DMA-1000 Computer*. Software documentation, 1991.
- Truax, B. "Discovering Inner Complexity. Time-shifting and Transposition with a Real-time Granulation Technique", *Computer Music Journal*, 18(2), 1994.

Sintesi di Eventi Sonori Complessi per mezzo di Iterazioni Funzionali

Ignazio Prignano

Laboratorio Musica & Sonologia, L'Aquila (Italy)

Dip. di Matematica Pura e Applicata,
Via Vetoio, Loc. Coppito,
67100 L'Aquila - Italy
telephone: ++39 862 433130
E-mail: lms@vxscqaq.aquila.infn.it

Abstract

This paper introduces a non-standard sound synthesis method called *functional iteration synthesis*. It is a non-standard technique in the sense that no well-studied acoustic model is implemented. The present model entails the iteration of a parameter dependent map $F(x_0; a_1, \dots, a_m)$, starting with a given x_0 . A sequence of samples can be constructed by fixing the number of iterations, and varying the map parameters and initial value at sample rate. The paper describes the general formalism and then the utilization of a particular difference equation model called the "sine map".

Sounds generated with this method can show dynamical properties ranging from very "active" behavior (spectrally rich transient phenomena, turbulence and noise) to relatively "inactive" behavior (smooth, if not almost flat curves), and unpredictable transitions in between. It is possible to obtain sounds with harmonic spectra too.

1 Introduzione

Ogni tecnica di sintesi non-standard porta a una rappresentazione microstrutturale del suono nel dominio del tempo discreto (si considerino, ad esempio, le tecniche di sintesi granulare). Ci si chiede allora se sia possibile fare emergere proprietà macrostrutturali particolari operando a livello microstrutturale, cioè legare il momento della sintesi del suono alle idee compositive su larga scala (nel dominio del tempo). Dalla teoria dei sistemi dinamici discreti non lineari e dipendenti da un parametro emergono interessanti proprietà di universalità: una vasta classe di tali sistemi mostrano comportamenti complessi qualitativamente simili, al variare del parametro, indipendentemente dalla forma particolare dei sistemi stessi.

E' naturale lasciarsi attrarre da queste semplici considerazioni e tentare di applicarle ai nostri modelli di sintesi (si veda, ad esempio, l'uso della "mappa logistica" nell'organizzazione dei *grani* in tecniche di sintesi granulare). Nel caso qui descritto si è cercato di sfruttare i

sistemi dinamici discreti come fonte diretta per la sintesi sonora: l'evoluzione del suono è legata, campione per campione, all'evoluzione del sistema al variare dei parametri.

2 Descrizione generale della tecnica di sintesi

Un sistema dinamico discreto non lineare è dato da una funzione che mappa, in modo non lineare, un intervallo della retta in se stesso (che nelle nostre applicazioni dovrà coincidere con l'insieme dei valori dei campioni del segnale audio). La sua evoluzione consiste nell'applicazione reiterata della mappa a un dato punto dell'intervallo, indicato come *dato iniziale*. Scelto un tempo t_n si può studiare l'effetto di una perturbazione del parametro o del dato iniziale sullo stato corrispondente del sistema. Per sistemi non lineari l'effetto può essere notevole e aumentare molto all'aumentare del tempo t_n , cioè del numero di iterazioni della mappa. A tempi grandi la perturbazione del dato iniziale può risultare in una variazione grande o nulla a seconda anche

del tipo di attrattore presente, che, come si evince dal *diagramma di biforcazione* della mappa (cfr. fig. 3), dipende dal valore del parametro, in ogni caso lo stato iniziale viene in un certo senso "dimenticato". Seguendo lo stato del sistema (fissato il tempo) al variare continuo del parametro e del dato iniziale si può ottenere così un segnale interessante con caratteristiche molto dinamiche. Ad ogni traiettoria nel piano cartesiano, rappresentante il parametro e il dato iniziale e che sarà indicato come *spazio delle fasi* (cfr. fig. 4), corrisponderà un segnale che verrà, infine, trasformato in segnale audio.

2.1 Formalismo matematico

Definiamo ora in termini matematici il modello. Consideriamo i seguenti insiemi:

$A \subset \mathbb{R}$ l'insieme dei dati iniziali,

$G \subset \mathbb{R}^m$ l'insieme dei parametri della mappa,

$B \subset \mathbb{R}$ l'insieme dei campioni del segnale audio che sarà generato e il prodotto cartesiano dei primi due:

$$A \times G \subset \mathbb{R} \times \mathbb{R}^m.$$

Sia F una funzione definita come

$$F: A \times G \rightarrow B$$

$$(x, \{a_i\}) \rightarrow F(x; \{a_i\}) \\ (\{a_i\} = a_1, a_2, \dots, a_m)$$

che può essere considerata come una funzione f dipendente da parametri che mappa A in B con gli a_i come parametri. Fissando un insieme di m parametri (un punto di G), abbiamo:

$$f: A \rightarrow B$$

$$x \rightarrow f(x) \\ f(x) \equiv F(x; a_1, \dots, a_m).$$

Specificando una sequenza g_i di punti di G , otteniamo una sequenza di mappe f_i . Ora, se

$$B \subset A$$

possiamo costruire l'iterazione della funzione f - introducendo quindi l'operazione chiamata *iterazione funzionale* - applicando ripetutamente f a se stessa n volte:

$$f^n(x) \equiv f(f(\dots f(x) \dots)) \equiv (f \circ f \circ \dots \circ f)(x)$$

Data una sequenza di dati iniziali $x_{0,i} \in A$, una sequenza di insiemi di parametri $g_i \in G$ e le funzioni corrispondenti f_i , possiamo generare così una serie di campioni; l'*i-esimo* campione è l'*iterata n-esima* di $x_{0,i}$ ottenuta con n applicazioni ripetute di f_i :

$$x_{n,i-1} = f_{i-1}^n(x_{0,i-1}) = = =$$

$$f_{i-1}(f_{i-1}(\dots(f_{i-1}(x_{0,i-1})) \dots))$$

$$x_{n,i} = f_i^n(x_{0,i}) = = =$$

$$f_i(f_i(\dots(f_i(x_{0,i})) \dots))$$

$$x_{n,i+1} = f_{i+1}^n(x_{0,i+1}) = = =$$

$$f_{i+1}(f_{i+1}(\dots(f_{i+1}(x_{0,i+1})) \dots))$$

3 L'iterazione funzionale con la "mappa seno".

Come prima applicazione di questo metodo si è scelta la seguente funzione dipendente da un parametro ($m=1$), che verrà chiamata "mappa seno":

$$F: [-\pi/2, \pi/2] \times [0,4] \rightarrow (-1,1) \\ (x, r) \rightarrow \sin(rx)$$

Per l'*i-esimo* campione da generare, l'evoluzione del sistema dinamico è data dall'equazione:

$$x_{k,i} = \sin(r_i x_{k-1,i})$$

Per $r \geq 1$, l'intervallo $A = [-\pi/2, \pi/2]$ viene mappato dalla funzione $\sin(rx)$ in tutto l'intervallo $[-1,1]$, dove viene confinato il sistema dinamico corrispondente dopo la prima iterazione. Sarebbe quindi inutile scegliere un intervallo 'A' più ampio, data anche la periodicità della funzione seno. Per $r \geq \pi/2$, l'intervallo $[-1,1]$ è mappato in tutto se stesso, ma il sistema rimane confinato solo a destra o solo a sinistra dello zero, finché $r \leq \pi$. Per $r > 4$ il sistema dinamico si ripete e l'intervallo $G = [0,4]$ esaurisce tutte le possibilità. In figura 1 è graficata la seconda iterata della funzione $\sin(rx)$ per quattro valori crescenti di r .

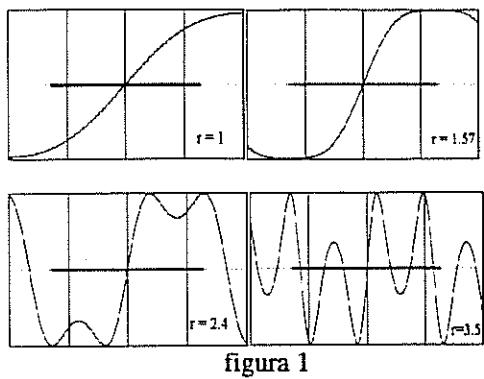


figura 1

E' interessante osservare l'andamento dell'iterata n -esima al crescere di r per x_0 fissato. In figura 2, in alto, sono graficate la 5^a, la 6^a e la 7^a iterata per r che va da 2 a 4; in basso sono graficate la 8^a, la 9^a e la 10^a iterata. Il dato iniziale è fissato a $x_0 = 0.1$. All'aumentare del numero di iterate l'evoluzione del segnale così costruito si fa via via più complessa.

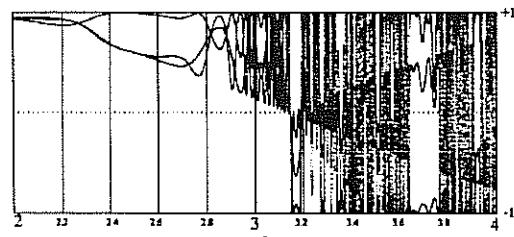
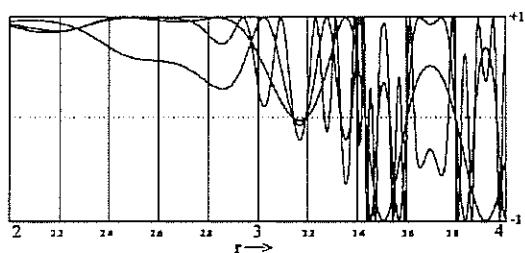


figura 2

Quando il numero di iterazioni è molto elevato il dato iniziale viene "dimenticato" e la sua perturbazione non produce effetti, il sistema "cade" sull'eventuale attrattore e all'aumentare di r l'iterata n -esima traccia in realtà il diagramma di biforcazione che si otterebbe graficando l'evoluzione del sistema dinamico all'aumentare "lento" del parametro. In Figura 3 è graficata la 100^a iterata di $x_0 = 0.1$ per r che va da 2 a 4.

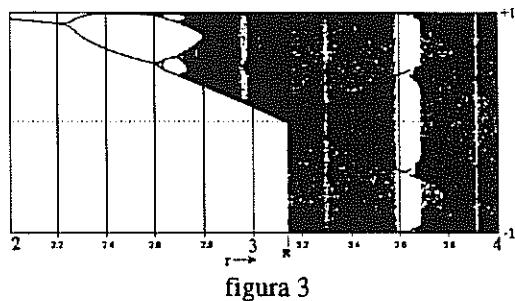


figura 3

4 Traiettorie nello "spazio delle fasi" della "mappa seno"

Dal diagramma di biforcazione della mappa si vede che all'aumentare di r il sistema diventa sempre più complesso esibendo dapprima solo attrattori periodici, poi attrattori più complicati, infine attrattori caotici con numerose e improvvise transizioni tra attrattori di tipo diverso. Osservando lo spazio delle fasi, per un determinato numero di iterate, si capisce che i segnali generati dalle traiettorie in esso dipendono fortemente dalla regione di piano visitata e dal numero di iterate scelto. Nei due grafici 3-D della figura 4 sono rappresentate le superfici dello stato del sistema, dopo 5 e 6 iterazioni rispettivamente, in funzione delle coordinate nel piano delle fasi. La regione dello spazio delle fasi mostrata nei due grafici è: [3, 3.5]x[2, .4].



n-iterate (sample)
 x_0
 r

$x_0: [2, 4]$
 $r: [3.5, 3], inc: .001$
iter: 5



n-iterate (sample)
 x_0
 r

$x_0: [2, 4]$
 $r: [3.5, 3], inc: .001$
iter: 6

figura 4

Ogni traiettoria nello spazio delle fasi genera un segnale secondo il metodo descritto.

Maggiore è il numero delle iterazioni, tanto più "attivo" è il risultato sonoro. Aumentando troppo n , non ne risulterà un segnale più interessante, a causa dell'eccessiva complessità già raggiunta, bensì aumenterà notevolmente il tempo di calcolo!

E' bene esaminare brevemente tre traiettorie particolari:

- fissando x_0 e variando r si potranno ottenere eventi sonori con rapidi cambiamenti strutturali a larga scala (nel tempo);

- fissando r e variando x_0 sarà possibile ottenere numerosi segnali diversi esibenti le medesime proprietà macrostrutturali;

- nei due casi descritti, facendo compiere oscillazioni con frequenze audio alla grandezza variabile, r o x_0 , il suono generato avrà un *pitch*.

Dunque il suono generato con il metodo di iterazione funzionale sarà caratterizzato da fasi di diversa complessità con imprevedibili e improvvise transizioni tra esse, non escludendo casi limite come suoni dotati di spettri armonici e il *rumore*.

5 Conclusioni

Le proprietà descritte per la "mappa seno" sono condivise da una larga classe di sistemi non lineari. La teoria dei sistemi dinamici non lineari e del *caos* dovrebbe guidare nella scelta delle mappe da utilizzare e delle regioni dello spazio delle fasi da visitare, supportando così l'indagine empirica, comunque irrinunciabile. Solo attraverso quest'ultima il compositore può prendere confidenza con il modello non deterministico e pensare una strategia propria per interagire con il materiale sonoro.

L'interesse principale nello studio dei sistemi discreti non lineari, quindi nei processi di iterazione funzionale, come già sottolineato, sta nel tentativo di far cadere la distinzione tra suono e struttura, permettendo al compositore di generare estese trame sonore che sarebbe innaturale considerare e quindi utilizzare e percepire come componenti individuali di strutture musicali a più alto livello.

References

Collet, P. & Eckmann, J.P. (1980) *Iterated Maps on the Interval as Dynamical Systems*. Boston: Birkhauser

De Poli, G. (1981) "Tecniche Numeriche di Sintesi della Musica". *Quaderni del LIMB*, v.1, 12-45

De Poli, G., Piccialli, A. and Roads, C. eds (1991) *Representations of Musical Signals*. Cambridge Ma.: MIT Press

Di Scipio, A. (1990) "Composition by Exploration of Nonlinear Dynamical Systems". *Proceedings ICMC90*, SanFrancisco: ICMA, 324-327

Di Scipio, A. (1994) "Micro-time Sonic Design and the Formation of Timbre". *Contemporary Music Review*, 10(2), 135-148

Di Scipio, A. (1995) "Inseparable Models of Material and of Musical Design" in *Electroacoustic and Computer Music*". *Journal of New Music Research*, 24(1), 34-50

Feigenbaum, M (1980) "Universal Behavior in Nonlinear Systems". *Los Alamos Science*, 1, 4-27

May, R. (1977) "Simple Mathematical Models with Very Complicated Dynamics". *Nature*, 261, 459-67

Pressing, J. (1988) "Nonlinear Maps as Generators of Musical Design". *Computer Music Journal*, 12(2), 35-46

Truax, B. (1990a) "Composing with Real-time Granular Sound". *Perspectives of New Music*, 28(2), 120-134

Truax, B. (1990b) "Chaotic Nonlinear Systems and Digital Synthesis. An Exploratory Study". *Proceedings ICMC90*, San Francisco: ICMA, 100-103

A New Formulation of the 2D-Waveguide Mesh for Percussion Instruments.

Federico Fontana & Davide Rocchesso

Centro di Sonologia Computazionale
Dipartimento di Elettronica e Informatica
Università degli Studi di Padova
Email: roc@csc.unipd.it

Abstract

A new waveguide mesh is proposed for discrete simulation of wave propagation on two-dimensional surfaces. We make use of waveguide junctions having eight branches, six of them lying on the surface itself, one coupling with the air load, and one available for coupling with a digital hammer. The proposed mesh is compared to the traditional rectangular mesh and it shows a better numerical behavior. Particularly efficient implementations can be worked out from the general model.

1 Introduction

The waveguide-filter paradigm has been extensively used for simulating strings and wind instruments [4]. The extension to higher dimensional resonators, like drums, plates, sound boards, or rooms can only be achieved with a certain degree of approximation. For instance, in the case of wave propagation in an ideal membrane, the traveling wave solution involves the integral sum of an infinite number of plane waves traveling in all directions. On the contrary, the waveguide formulation involves propagation along a finite number of directions. A grid of delay lines is formed to discretize the medium, and lossless junctions are used to connect the waveguide branches.

The mesh made of 4-port junctions has been studied [5] and applied to the synthesis of percussion instruments. The major drawback of this mesh is the effect of wave dispersion due to the discretization of propagation along two directions. This effect limits the fidelity of simulation to a low-frequency range. The great advantage of 4-port junctions is that their implementation doesn't require any multiplication. On the other hand, additional ports are needed to couple the junction with an external exciter, like a stick or a hammer, or with a damping element, like the air load. These additional ports clearly prevent from the elimination of multipliers.

The mesh made of 3-port junctions is being studied by Van Duyne and Smith (private communication) and seems to provide a better numerical behavior.

In this paper, we introduce a mesh made of 8-port junctions. Six of the eight branches are laid on the surface to be simulated, one branch is connected to a lumped loss, and the remaining branch is ready to be connected to a lumped model of hammer.

The surface is decomposed into elementary triangles. Different losses can be attached to different nodes of the mesh, thus approximating irregular media. A model of hammer can be attached to any node of the mesh, and effects like scratching the hammer on the membrane, can be simulated.

The membrane decomposition into elementary triangles shows a more uniform and less pronounced dispersion effect than the rectangular mesh. Like in the rectangular case, the waveguide formulation is shown to be equivalent to a finite-difference scheme for the numerical solution of the wave equation.

The performance of the proposed mesh can be improved by eliminating multipliers under certain conditions.

2 The New 2D-Waveguide Mesh

We propose a tessellation of the ideal membrane into triangles, as depicted in fig. 1. For the moment let us assume that there is neither an exciter nor a load of any kind, so that any node in the network can be represented as a 6-port lossless waveguide junction. The branches between two nodes are unit delays, corresponding to a physical

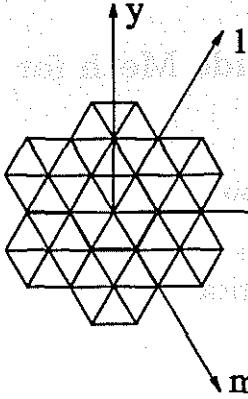


Figure 1: Triangular tessellation of a surface

distance

$$d = c/F_s \quad (1)$$

travelled by velocity waves in one time sample. In equation (1), $F_s = 1/T$ is the sampling rate, $c = \sqrt{T/\sigma}$, where T and σ are respectively the tension and the surface density.

Three axes of symmetry are drawn in fig. 1, and are called x , l , and m . The operations performed for the i^{th} branch of the waveguide junction located at position x, l, m are:

$$\begin{aligned} v_{i,x,l,m}^-(t) &= v_{J_{x,l,m}}(t) - v_{i_{x,l,m}}^+(t) = \\ &\frac{1}{3} \sum_{j=1}^6 v_{j_{x,l,m}}^+(t) - v_{i_{x,l,m}}^+(t) \end{aligned} \quad (2)$$

The mesh behaves well in both the time domain and the frequency domain. The response to an initial stroke is plotted in fig. 2. This plot shows that the wave-front shape is well preserved along all directions, thus indicating that the dispersion error is uniform and low (see section 4). Fig. 3 shows a comparison between the theoretical modal frequencies of a circular membrane [1] and the resonances obtained from our model. The correspondence is very tight.

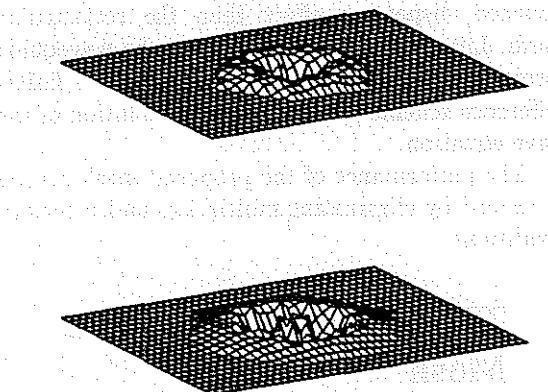


Figure 2: Wave propagation on the mesh

ical modal frequencies of a circular membrane [1] and the resonances obtained from our model. The correspondence is very tight.

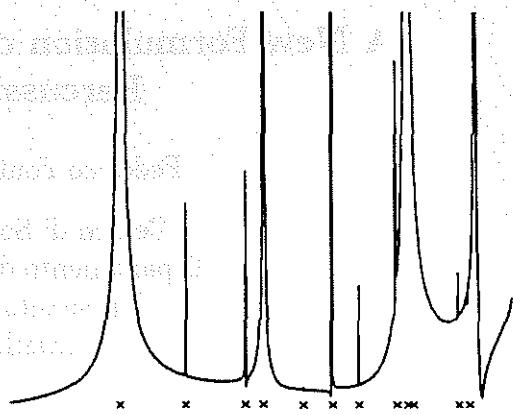


Figure 3: Distribution of resonances

3 Interpretation as a Finite-Difference Scheme

The equivalence between the rectangular mesh and a finite-difference scheme for numerical integration of the wave equation has already been shown [5]. Here we show the equivalence of the triangular mesh to another finite difference scheme working on the same equation.

The wave equation for the ideal membrane is a second-order hyperbolic partial difference equation:

$$\frac{\partial^2 v}{\partial t^2} = c^2 \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (3)$$

where x and y are the two orthogonal axes of fig. 1. The wave equation (3) can be expressed in the derivatives along the x , l , and m axes of fig. 1, obtaining:

$$\frac{\partial^2 v}{\partial t^2} = \frac{2c^2}{3} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial l^2} + \frac{\partial^2 v}{\partial m^2} \right) \quad (4)$$

By noticing that the sum of the inputs to a junction equals the sum of the outputs, and that the input at one port is equal to the output at the opposing port of the adjacent junction at the previous time sample, one can rearrange equation (2) as follows:

$$v_{J_{x,l,m}}(t+1) + v_{J_{x,l,m}}(t-1) = \frac{1}{3} \{ v_{J_{x+1,l+1,m}}(t) + v_{J_{x+1,l-1,m}}(t) + v_{J_{x-1,l+1,m}}(t) + v_{J_{x-1,l-1,m}}(t) + v_{J_{x,l-1,m-1}}(t) + v_{J_{x+1,l,m-1}}(t) \} \quad (5)$$

which is clearly the finite difference discretization of the modified wave equation (4). The bracketed term in the right-hand side of equation (5) is the sum of velocities of the nodes adjacent to the junction x, l, m .

The comparison between (4) and (5) also shows that waves in the mesh travel at a nominal speed $c = \sqrt{1/2}$.

4 Numerical Performance

The main problem of the rectangular mesh is the numerical dispersion error, which causes the high-frequency wavefronts to propagate at the nominal speed along the diagonals, but about a 30 % slower along the main axes.

We have conducted a Von Neumann analysis [3] on our finite-difference scheme (5), as indicated in [6]. Naming ξ_x, ξ_y the spatial frequencies along the normal axes, and d_x and d_y the unit lengths along the axes such that $d_x = d_y = cT$, we have found the following formula for the equivalent normalized speed of propagation $c(\xi_x, \xi_y)/c$:

$$c(\xi_x, \xi_y) = \frac{\sqrt{2}}{2\pi\sqrt{\xi_x^2 + \xi_y^2} T} \arctan \frac{\sqrt{4 - b^2}}{b} \quad (6)$$

where

$$\begin{aligned} b &= \frac{2}{3} \cos[2\pi\xi_x d_x] \\ &+ \frac{2}{3} \cos[2\pi(1/2\xi_x d_x + \sqrt{3}/2\xi_y d_y)] \quad (7) \\ &+ \frac{2}{3} \cos[2\pi(1/2\xi_x d_x - \sqrt{3}/2\xi_y d_y)] \end{aligned}$$

The expression (6) has been plotted in fig. 4, together with the results of Van Duyne and Smith [6].

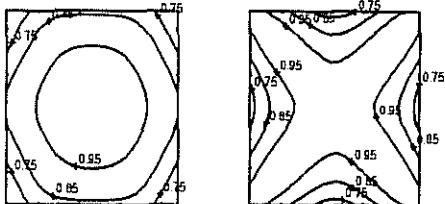


Figure 4: Contour plots of speed of propagation versus spatial frequencies. Left: triangular mesh. Right: rectangular mesh.

The contour plots show that the triangular mesh provides a very uniform speed of propagation along all directions, while the rectangular mesh is exact along the diagonals, and very inexact along the main axes. The triangular mesh exhibits a high-frequency dispersion error not exceeding 15 % along all the branches, while the rectangular mesh has almost 30 % of error along the axes in high frequency.

For both the rectangular and the triangular mesh, the dissipation error is identically zero at

any spatial frequency, and the Courant-Friedrichs-Lowy stability condition [3] is satisfied in the equality. This has to be expected, since both the meshes are built by interconnection of lossless waveguide junctions and pure delays.

5 Excitation and Air Load

For the purpose of coupling the membrane with an exciter and with a distributed air load, we extend our model in the direction orthogonal to the plane of the membrane. We add two ports to our basic waveguide junction, the upper one responsible for the interaction with a stick or a hammer, and the lower one responsible for the air load.

We model the air load as a second order damped oscillator, whose continuous-time transfer function is:

$$T(s) = \frac{V(s)}{F(s)} = \frac{s}{m_l s^2 + \mu_l s + k_l} \quad (8)$$

where m_l is the mass of air over the junction, μ_l is a resistive parameter depending on the friction existing between air and mesh surface and k_l is the stiffness constant of a linear spring modeling air oscillation. Equation (8) can be rewritten, by defining an arbitrary reference impedance R_l , in terms of incoming and outgoing velocity waves:

$$\begin{aligned} H(s) &= \frac{v^-(s)}{v^+(s)} \quad (9) \\ &= \frac{-s^2 + (R_l/m_l - \mu_l/m_l)s - k_l/m_l}{s^2 + (R_l/m_l + \mu_l/m_l)s + k_l/m_l} \end{aligned}$$

Similarly to the trick used in [7] for the piano hammer, we set the reference impedance to the value

$$R_l = \mu + \alpha m + k/\alpha \quad (10)$$

where α is usually set to the value $2/T$. In this way, when transforming $H(s)$ to the discrete-time domain by bilinear transform, a unit delay may be factored out and there is no non-computable closed loop in the graph.

For the excitation, we have used the model of hammer proposed in [7]. For the excitation port, when it is coupled to the hammer, we have another arbitrary reference impedance R_h , which is set to the value

$$R_h = \frac{m_h k_h \alpha}{m_h \alpha^2 + k_h} \quad (11)$$

where m_h and k_h are respectively the mass and stiffness of the hammer. When the hammer is not in contact with the membrane, the reference impedance is set to zero, thus physically removing one port from the junction.

The junctions which are coupled with a hammer model are time-varying, and therefore we can

not do much to reduce the elementary operations involved with the wave scattering. On the contrary, when the junction do not interact with the exciter, there exists a theoretical possibility to get rid of the multiplications. Rewriting part of equation (2) in a more general form, involving R_l and the membrane impedance $R_0 = \sqrt{\sigma T}$:

$$v_{J_{x,l,m}}(t) = \frac{2R_0 \sum_{j=1}^6 v_{j_{x,l,m}}^+(t) + 2R_l v_{l_{x,l,m}}^+(t)}{6R_0 + R_l} \quad (12)$$

we notice that the computation may be performed with no multiplies if R_0 and R_l satisfy the two conditions $R_l = 2^{n_1} R_0$ and $R_l = 2^{n_2} R_0 - 6R_0$, with n_1 and n_2 positive integers; this only happens for the couple $(n_1, n_2) = (1, 3)$, which implies the solution $R_0 = R_l/2$. This theoretical possibility may be practical if physical conditions guarantee this solution, that is, recalling (10):

$$\sqrt{\sigma T} = 1/2(\mu + \alpha m + k/\alpha) \quad (13)$$

Another theoretical possibility is to consider even R_0 as a nominal impedance, regardless of physical implications.

In practice, the junction computation may be performed with no multiplies inserting a non-physical unit delay in the air-load loop, and assuming that the variables are slowly varying. In this way, the reference impedance R_l is set to half the membrane impedance R_0 , so that all multiplications can be replaced by shifts in fixed-point arithmetic.

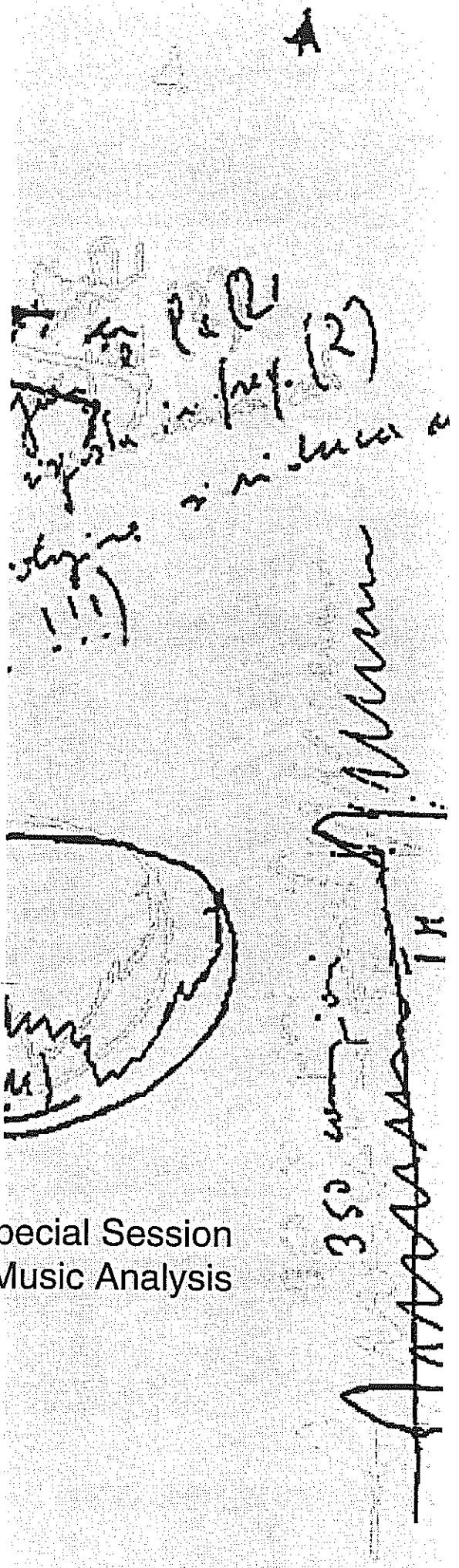
6 Conclusion

In all kinds of simulations, the ultimate goals are efficiency and accuracy. When simulating wave propagation in a two-dimensional medium, these two goals can only be achieved with a certain degree of approximation. Nevertheless, we believe that the triangular mesh presented in this paper is a feasible solution for many physical-modeling applications, since it is much more accurate than the rectangular mesh, and it can be quite efficient when certain (weak) conditions are met.

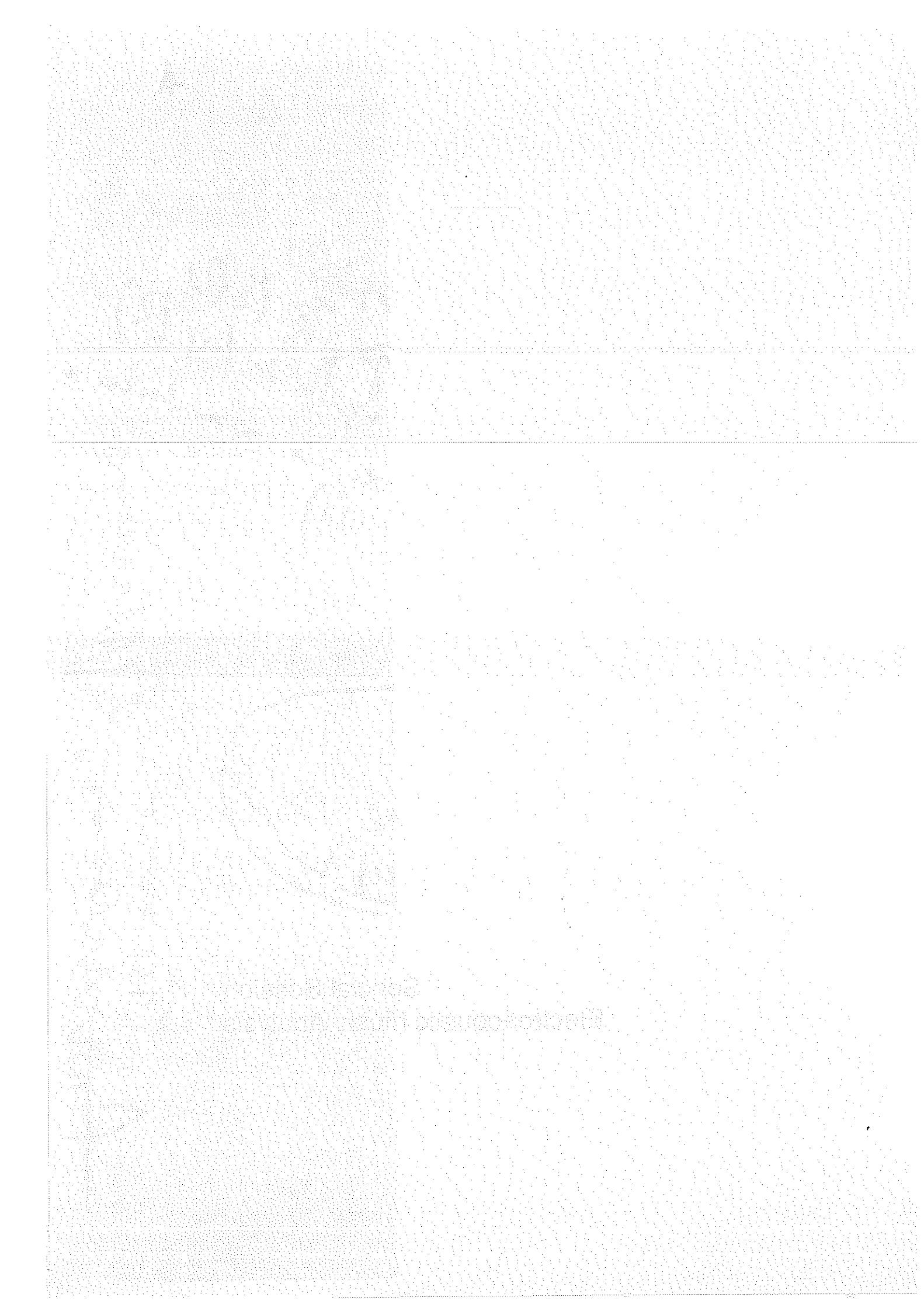
Our ongoing research is trying to apply the triangular mesh to the simulation of real-world percussion instruments and drumming effects.

References

- [1] N.H. Fletcher and T.D. Rossing, *The Physics of Musical Instruments*, Springer-Verlag, NY, 1991.
- [2] T. Kailath, *Linear Systems*, Prentice-Hall, Englewood Cliffs, 1980.
- [3] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, *Numerical Recipes in C*, Cambridge University Press, Cambridge, MA, 1988.
- [4] J.O. Smith, *Physical Modeling Using Digital Waveguides*, Computer Music Journal, vol. 16, n. 4, pp. 74-91, Winter 1992.
- [5] S.A. Van Duyne, and J.O. Smith, *Physical Modeling with the 2-D Digital Waveguide Mesh*, Proc. Intern. Comp. Music Conf., Tokyo, Japan, 1993.
- [6] S.A. Van Duyne, and J.O. Smith, *The 2-D Digital Waveguide Mesh*, Proc. IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics, Mohonk, New York, 1993.
- [7] S.A. Van Duyne, J.R. Pierce and J.O. Smith, *Traveling Wave Implementation of a Lossless Mode-Coupling Filter and the Wave Digital Hammer*, Proc. Intern. Comp. Music Conf., Aarhus, Denmark, 1994.
- [8] S.A. Van Duyne, *Personal communication*, 1995.



Special Session Electroacoustic Music Analysis



Riflessioni sull'analisi della musica elettroacustica e informatica. Esperienze e spunti teorici

Agostino Di Scipio

Conservatorio di Musica "N.Piccinni", Bari
Laboratorio Musica e Sonologia. L'Aquila
email lms@vxscqaq.aquila.infn.it

Abstract

This paper proposes some reflections about the analysis and study of electroacoustic music, summarizing the author's experience in this field. It is observed that methods of historical and hermeneutic analysis are far behind the theoretical and aesthetic reaches of much electroacoustic music. For renewed and specific analytical perspectives to be taken, the musicological observation should address itself towards two specific problem domains: 1) the process and role of *τεχνη* in the artist's work - the world of techniques, technologies, mental representations, conventions, practical tools which captures the artist's relation to the materials and forms of his/her art, thereby enabling us to recognize the musical work as *τεχνημα*. i.e. humanly devised artifact; 2) the complex relation of *sound to form* - i.e. the problem domain of *timbre*.

1. Introduzione

Analizzare una composizione musicale, o un repertorio di composizioni omogenee, implica sempre volgere la nostra attenzione di osservatori verso un quadro di riferimento teorico dal quale deriviamo una metodologia, dei modelli e delle strategie operative. Il problema sostanziale, a mio avviso ancora irrisolto, posto dalla musica elettroacustica e informatica all'osservazione analitica si può riassumere nella domanda: abbiamo una *teoria della musica* (o almeno elementi teorici di base) alla quale (ai quali) volgerci per rendere conto delle condizioni espressive e tecniche proprie di questa musica e delle sue esperienze creative, interpretative e ascoltative?

In queste brevi riflessioni vorrei fermarmi su una convinzione personale: le categorie teoriche e i metodi di analisi della musicologia di stampo storiografico e di stampo ermeneutico promuovono approcci analitici che non sono sempre in grado di caratterizzare in modo approfondito i fenomeni e le esperienze della musica elettroacustica e informatica. Facendo riferimento a concrete esperienze di analisi vorrei mettere in rilievo alcuni nodi teorici, principalmente di carattere cognitivo ed epistemologico, che occorre affrontare se si mira ad un approccio analitico più specifico.

2. Aspetti problematici nelle principali prospettive attuali

Il discorso della musicologia ermeneutica si basa ampiamente sulle capacità di ascolto,

verbalizzazione e sistematizzazione da parte dell'osservatore-analista. Si tratta di un'attività che potremmo dire di "modellizzazione del proprio ascolto" (su base per lo più intuitiva) e di "modellizzazione delle fonti storiche" (su basi certamente più rigorose, derivate dalla storiografia). In alcuni casi, come nella *teoria della ricezione* in ambito di critica letteraria, si mira alla ricostruzione di discorsi e testimonianze - di artisti, critici, pubblico, ecc... - riguardanti l'opera studiata o il lavoro di un artista o gruppo di artisti (Borio & Garda, 1989).

Alla pratica della "modellizzazione dell'ascolto" da qualche tempo si sono prestate basi più rigorose derivate dallo studio dei processi cognitivi e percettivi dell'ascolto musicale. In generale, ciò è stato determinante per lo sviluppo della *musicologia cognitiva*. Troppo spesso, però, tali ricerche manifestano l'acquisizione tacita e acritica di un paradigma epistemologico che privilegia un unico punto di osservazione: quello di un *ascoltatore ideale*. Questa è la prospettiva affermatasi in campo scientifico con la pubblicazione del volume *A Generative Theory of Tonal Music*, di F.Lerdahl e R.Jackendoff (1983). (Lerdahl ha applicato le prospettive delineate in quel volume anche all'analisi di musica atonale, con esiti, a mio avviso, del tutto arbitrari: cfr. Lerdahl, 1988, dove viene analizzato *Le Marteau sans Maître* di Boulez).

Questa impostazione è sembrata particolarmente pertinente all'analisi di musica elettroacustica (Delalande, 1986; Camilleri, 1991; Doati, 1991), poiché infatti permette di rendere segmentabile il flusso delle sonorità e quindi di riscrivere una

"testo" che, in assenza di altre forme di notazione, diventa a sua volta oggetto di descrizione.

Tuttavia in questo tipo di ricerche ci si scontra inevitabilmente con una difficoltà a dar conto in modo pertinente dei principali *fatti estetici* di questa musica, i quali sono partecipi di non trascurabili aspetti concettuali che non giustificano una riduzione alla sola "musica come percepito" - a meno di non voler ingenuamente ridurre la nozione di *estetica* a quella antica di *scienza delle sensazioni*, invece di considerarla come *teoria dell'arte* (nell'accezione della tradizione filosofica occidentale a partire da Baumgarten, cioè dal XVIII sec.) cioè la teoria di come l'uomo esprime in forma creativa la propria visione o "percezione" - del mondo, il proprio *sentire*. I limiti dell'analisi fondata esclusivamente su quegli assunti - quello di un *ascoltatore ideale* e quello della *ricostruzione* di testimonianze e discorsi a proposito di altre testimonianze e discorsi - non possono essere delineati qui in dettaglio. Vale la pena ricordare, però, che il programma scientifico proposto da Lerdahl e Jackendoff, pur rappresentando un modello importante per la psicologia della musica, recentemente è stato più volte messo in discussione (cfr. Laske, 1993; Smolian, 1995). E che, da canto suo, la storiografia musicale, nei risultati che riesce a proporre quando tratta di musica elettroacustica, non offre prospettive di reale interesse: Emmerson (1986) nota che perfino la distinzione tra *musique concrète* e *elektronische Musik* è stata profondamente banalizzata in ricerche di stampo prettamente storico.

Lungi dall'essere solo questioni di metodologia, questi problemi tradiscono la mancanza di un più adeguato tentativo di interpretazione teoretica, richiedendo, inoltre, rinnovate categorie critiche (alcuni elementi, in proposito, sono tematizzati in Di Scipio, 1993; 1995a).

3. Due specifici dominii di osservazione

Nelle mie esperienze di analisi di opere elettroacustiche ho privilegiato due dominii di osservazione specifici - sui quali ci soffermeremo tra breve. La puntualizzazione secondo la quale essi hanno rilievo per me come compositore - e che dunque che sono di relativo interesse per un musicologo - non fa che addurre ulteriore evidenza ad un fatto: la musicologia attuale, anche quella *contemporaneistica*, non riesce a sviluppare strumenti concettuali per accettare le sfide che gli pone la musica elettroacustica. Non siamo di fronte ad un fatto del tutto nuovo: in ogni periodo della storia durante il quale si sono sviluppate le coordinate di una forma d'arte nuova o ampiamente rinnovata, gli artisti sono diventati teorici e la loro arte si è arricchita di elementi

concettuali che sfuggono a chi opera come se invece la prassi consolidata e la relativa teoria fossero immutevole e definitive. Pierre Schaeffer (1966, pp.28-32) segnalava l'*impasse* della musicologia di fronte ad alcuni fatti sostanziali proposti dalla nascita della musica concreta. Koenig, nel 1955, doveva a sua volta illustrare la trasformazione dell'ambiente di lavoro nel quale prendeva corpo la prassi estetica elettroacustica, indicando così implicitamente che proprio lì - nella *studioteknik* - si dovevano cercare elementi utili a caratterizzare il progetto estetico della *elektronische Musik* (Koenig, 1958).

Quanta strada l'accademia musicologica non abbia fatto nonostante quelle (e molte altre) indicazioni iniziali lo si può evincere dal modo del tutto vago e superficiale con cui sono state trattate ancora oggi le produzioni elettroacustiche di alcuni grandi del nostro tempo, come Xenakis e Nono (si vedano, ad esempio, i pochissimi spunti contenuti nelle raccolte curate da E. Restagno, 1987; 1988).

3.1 *Tεχνη* come dominio di analisi

Le rappresentazioni mentali, le conoscenze relative al materiale e alle forme della propria arte, alla relazione tra azione e percezione - questo ed altro è quanto viene catturato nelle procedure tecniche, nelle attrezzature tecnologiche, nelle strategie operative attraverso le quali l'idea viene attualizzata e resa tangibile - ascoltabile. Come è possibile prescindere da una conoscenza degli strumenti attraverso i quali un'artista media tra il suo mondo interiore e quello circostante? Il problema (che altrove ho definito *mancanza di consapevolezza etnomusicologica* - cfr. Di Scipio, 1993) è che per giudicare i risultati quali percepiti - la musica come *oggetto prodotto* - occorre avere nozioni precise su come tali risultati sono ottenuti - sulla musica come *processo, azione*. Il perché - con quali finalità, con quale progetto di interpretazione del mondo - un lavoro è così come appare viene testimoniato ("registrato", dice Laske, 1977) nel modo in cui i mezzi dell'azione vengono adoperati. Il doppio problema tecnologico - il problema della *tecnologia che interviene nel processo creativo* e quello della *relazione tra arte e mondo della tecnologia* (cfr. Borio, 1993; Di Scipio, in corso di stampa) - se seriamente affrontato ci dice che qualsiasi strumento o mezzo di realizzazione cattura tutto un ambito di conoscenze relative al proprio dominio d'azione musicale e ad un più vasto campo di conoscenza teorica (*performance e competence*, con i termini di Chomsky). E, infine, dinamizza lo stesso mondo della tecnica a disposizione dell'arte, mettendone in evidenza limiti e possibili estensioni (a proposito si vedano alcuni contributi raccolti in Di Scipio, 1995b).

Per venire ad esempi concreti: è impossibile cogliere la portata estetica della produzione

elettroacustica di Xenakis senza analizzarne la specifica *τεχνη* gravida di elementi essenziali per comprenderne le visioni immanenti relative al suono, al tempo, all'idea di costruzione, di natura, di *ordine* (Di Scipio, 1995a); è impossibile comprendere la portata costruttiva del caso, a più livelli di struttura, in *Scambi* di Pousseur (nonché il rapporto tra intenzione volta alla forma e elemento costruttivo aleatorio) senza analizzare e verificare "sul campo" - ovvero "in studio" - la portata dei dettagli tecnici (Doati, 1992). Non è possibile valutare le valenze musicali di processi ricorsivi di trasformazione del suono in *Terminus* di Koenig prescindendo dallo studio della catena di trasformazioni attuata con le apparecchiature di studio dal compositore (Ungeheuer, 1994).

Si può parlare a lungo della centralità della percezione nella musica di Risset, ma al momento dell'analisi non se ne può cogliere come l'*atto percettivo* riesca poi a diventare *progetto* se non si analizzano i processi di sintesi ed elaborazione al computer (Di Scipio, 1988). In un saggio di analisi dal titolo *On different approaches to computer music as different models of compositional design* (Di Scipio, 1995c) ho cercato di mettere in evidenza come la diversa impostazione del processo costruttivo in differenti compositori di musica per computer, si rifletta nella diversa significatività musicale ed estetica dei prodotti finali.

Naturalmente questo tipo di impostazione ha bisogno a sua volta di consolidare un quadro teorico di riferimento. Personalmente negli ultimi anni ho prestato molta attenzione a delinearne le componenti e a discuterne la pertinenza in relazione alla molteplicità dei fenomeni della musica elettroacustica.

3.2 Timbro: un nuovo modo relazionale di materiale e forma

L'altro specifico dominio di osservazione che viene spesso disatteso dal discorso musicologico intorno alla musica elettroacustica è quello del *timbro*. Perché oso dire "disatteso" quando invece si sa bene che proprio il timbro è l'elemento centrale di questa musica, quando invece se ne studia la potenzialità di *parametro conduttore di forma*?

Nel contesto teorico e operativo della musica elettroacustica e, ancor di più, in quello della musica al computer, il materiale sonoro non preesiste necessariamente ad alcun atto intenzionale, ad alcuna intenzione formativa: il materiale diventa passibile di apprezzamento e di giudizio, si pone, quindi, come categoria di rilevanza estetica; in quanto deliberatamente progettato e composto, il materiale non è più materiale, si dematerializza (Duchez, 1991) si virtualizza (al pari della forma) - esso è *natura naturata*, ancor prima di essere usato per l'articolazione di ampie strutture formali.

Analizzare come il suono venga composto, quindi, diventa importante quanto analizzare come i suoni vengano disposti nel tempo; significa osservare quale teoria del suono, quale idea di materia sonora venga scelta, attuata, o magari del tutto inventata dal compositore. (Ovviamente esistono produzioni elettroacustiche nelle quali i materiali sonori sono "già dati" - circostanza che, anch'essa, non è possibile trascurare nel momento analitico). E poi, anche, significa vedere *se* e *come* la composizione del suono venga messa in relazione alla composizione delle forme, ai gesti formali di ampio respiro formale.

Vorrei far osservare, inoltre, che lo studio del rapporto materiale/forma in un'opera o in un compositore non è attuabile senza adeguata conoscenza della tecnica che ne permette la attuazione (cioè ci rimanda al dominio di osservazione della *τεχνη*). Ed inoltre - ciò che forse è ancor più importante - che esso è reso necessario da una constatazione che da sola basta a definire l'unicità dell'impresa estetica alla portata della musica elettroacustica. la possibilità di fondere - di confondere - *suono e struttura*, di dire "forma" il materiale sonoro e "materiale" la forma del suono. Il rilievo cognitivo ed estetico di questa possibilità-limite è tutt'uno con la rottura del dualismo (fin troppo ribadito) tra *comporre-i-suoni* e *comporre-con-i-suoni*, ovvero con quel potenziale estetico (spesso rimosso sia dai compositori che dalla critica) abbozzato già alle origini e poi declinato sul fondo della tradizione elettroacustica: il riconoscimento dell'opera musicale come *artificio integrale*, di cui interamente (cioè a tutti i livelli) è responsabile una volontà formativa. Nel 1955 Pousseur scriveva: "A questo punto possiamo ideare strutture musicali che non si astengono più dalle variegate ricchezze del mondo dei suoni, ma che sostituiscono il suono naturale con l'intenzionalità di ciò che è ideato in piena consapevolezza" (1958, p.34, traduzione mia).

Chiunque sa che ogni materiale d'arte è sempre e comunque *cultura*, ovvero è storicamente determinato, sedimentazione delle conoscenze e dei raggiungimenti del passato. Non c'è bisogno di citare Adorno per ricordarlo. Tuttavia il ruolo della musica elettroacustica è stato (ed è) anche quello di tematizzare questa consapevolezza nell'ambito della stessa esperienza creativa: qui davvero il materiale sonoro può diventare quello che Lukács chiama il "non-ancora" della conoscenza dell'artista; per cui questa musica si presenta anche come un'arte di *materiali sonori possibili* (Di Scipio, 1994). La *Gestalt* del materiale è ciò che diciamo *timbro* - la forma del suono. Perciò una "musica di timbri", come prodotto di un'intenzione formativa che interviene a tutti i livelli della struttura musicale, è appunto un *artefatto integrale*: l'arte che si riconosce come definitivo

τεχνημα - oggetto completamente culturale, affrancato da ogni falso appoggio ad una qualsiasi *natura naturans*, esterna all'uomo.

4. Conclusione

Potremmo comprendere un τεχνημα senza comprenderne la τεχνη che lo origina? L'antropologia suggerisce alla (etno-)musicologia che gli strumenti dell'azione (sia quella *utile* che quella *gratuita*) sono la sede stessa di conoscenze e di cultura, di significati e di idee (Schaeffner, 1968). Non è paradossale temere che l'accademia musicologica dei nostri tempi, di fronte al rinnovarsi del pensiero e della prassi musicale, esibisca di una sorta di atteggiamento "etnocentrico" (il vizio della etnomusicologia degli inizi del secolo): giudicare i risultati dell'azione musicale di una cultura lontana secondi i parametri della *propria* cultura.

Il lavoro da svolgere nelle prospettive qui appena delineate è in gran parte tutto da fare (ma si rimanda il lettore ai riferimenti bibliografici); inoltre esso non può che far tesoro anche delle legittime osservazioni critiche che può sollevare. Tuttavia se ritengo urgente lavorare in questo senso non è solo per recuperare il tempo perduto, ma anche (forse soprattutto) perché la mancanza di consapevolezza finisce inevitabilmente con il vincolare l'entusiasmo e la coscienza dei musicisti stessi (compositori e interpreti). Già oggi osserviamo l'avvenuta relativizzazione del mezzo elettronico in rapporto al linguaggio musicale, l'affermarsi (presso musicisti e istituzioni) dell'opinione secondo la quale l'idea, in fin dei conti, possa prescindere dalla tecnica disponibile o possibile, quasi che idea e tecnica non si co-determinassero. Ciò rappresenta, almeno in parte, il risultato dell'assenza di una quadro teorico-critico in grado di dar conto dei caratteri specifici dei repertori elettroacustici, con le inevitabili ripercussioni negative che questa assenza ha (ed ha avuto) sulla diffusione di questa musica e sulla levatura dei suoi progetti artistici.

Riferimenti bibliografici

- Borio, G. "New technology, new techniques. The aesthetics of electronic music in the 1950's". *Interface-Journal of new music research*, 22(1), 1993.
- Borio, G. e M.Garda (a cura di) *L'esperienza musicale. Teoria e storia della ricezione*. Torino, 1989.
- Camilleri, L. "La musica elettroacustica e l'analisi dei processi di ascolto", *Sonus-Materiali per la musica contemporanea*, 3(2), 1991.
- Delalande, F. "En l'Absence de Partition: le Cas Singulier de l'Analyse de la Musique Électro-acoustique", *Analyse Musicale*, 3, 1986.
- Di Scipio, A. *Un'analisi dettagliata di Contours, di J.C. Risset*, CSC, Univ. di Padova, 1988.
- Di Scipio, A. "Models of material and of musical design become inseparable. A study in composition theory", *Atti della International Conference on Cognitive Musicology*, Jyväskylä, 1993. Ristampato col titolo "Inseparable models of material and of musical design", *Journal of new music research*, 24(1), 1995.
- Di Scipio, A. "Micro-time sonic design and the formation of timbre", *Contemporary music review*, 10(2), 1994.
- Di Scipio, A. "Da Concret PH a Gendy301. Modelli compositivi nella musica elettroacustica di Xenakis", *Sonus-Materiali per la musica contemporanea*, 6(2), 1995.
- Di Scipio, A. (a cura di) *Teoria e prassi della musica nell'era dell'informatica*. Bari, 1995.
- Di Scipio, A. "On different approaches to computer music as different models of compositional design", *Perspectives of new music*, 33, 1995.
- Di Scipio, A. "The centrality of tēchne for an aesthetic approach to electroacoustic and computer music", in corso di stampa su *Journal of new music research*.
- Doati, R. "György Ligeti's *Glissandi*: An Analysis", *Interface-Journal of New Music Research*, 20(2), 1991.
- Doati, R. "Il caso filtrato", *Quaderni della Civica Scuola di Musica*, 21-22, 1992.
- Duchez, M.E. "L'évolution scientifique de la notion de matériau musical." In J.B. Barrière (a cura di) *Timbre. Métaphore pour la Composition*, Paris, 1991. Trad.it. "La nozione di materiale nella tecnologia musicale contemporanea". In A. Di Scipio (a cura di) *Teoria e prassi della musica nell'era dell'informatica*. Bari, 1995.
- Emmerson, S. "The relation of language to materials". In S. Emmerson (a cura di) *The language of electroacoustic music*. London, 1986. Trad.it. "La relazione tra linguaggi e materiali nella musica elettroacustica". In A. Di Scipio (a cura di) *Teoria e prassi della musica nell'era dell'informatica*. Bari, 1995.
- Koenig, G.M. "Studio technique", *Die Reihe*, 1, 1958 (ed.or. 1955).
- Laske, O. *Music, Memory and Thought. Explorations in Cognitive Musicology*. AnnArbour, 1977.
- Laske, O. "Music and A.I. Some philosophical reflections", *Atti della International Conference on Cognitive Musicology*, Jyväskylä, 1993.
- Lerdahl, F. "Cognitive constraints on compositional systems". In J. Sloboda (a cura di) *Generative processes in music*, Oxford, 1988.
- Lerdahl, F. e R.Jackendoff, *A generative theory of tonal music*, Cambridge Ma., 1983.
- Pousseur, H. "Formal elements in a new compositional material", *Die Reihe*, 1, 1958 (ed.or. 1955).
- Restagno, E. (a cura di) *Noon*, Torino, 1987.
- Restagno, E. (a cura di) *Xenakis*, Torino, 1988.
- Schaeffer, P. *Traité des object musicaux. Essai Interdisciplines*. Paris, 1966.
- Schaeffner, A. *Origins des instruments de musique*. Paris, 1978. Trad. it. *Origine degli strumenti musicali*, Palermo.
- Smoliar, S. "Parsing, structure, memory and affect", *Journal of new music research*, 24(1), 1995.
- Ungeheuer, E. "From the elements to the continuum: timbre composition in early electronic music", *Contemporary music review*, 10(2), 1994.

Analysis of *Continuo* (1958) by Bruno Maderna

Prof. E. Pancaldi

Course of Electronic Music, "G. B. Martini" Conservatory of Music
Bologna, Italy

Abstract

In the title itself it is contained important information of the excerpt conduction way, that stresses how, more than a dialectic "sound-silence", we have a sound temporal continuity dominating the whole piece.

At a first investigation level the mainly used sound typology is represented the white filtered sound that, in accordance with the major or the minor spectrum range appears or as a rustle or as a tonic sound (with a fluty or vocal "color" according to the circumstances). Against this sound typology is set a sound innerly complex object that, during the excerpt, lives some transformations that stress that inner "turbulence".

To these two sound figures dialectic process other events are added to which can be linked (as for example, the impulsive-percussion sound that introduces a complex sound object again, above mentioned, at 3'38") or put inside the sound continuum as events more or less characterizing a certain section (for example the sounds with a bell-like spectrum at 4'10" and on).

We report a sound objects list which will be referred, for our convenience, to a letter:

- A-white filtered sound
- B-complex granular sound
- C-rolling
- D-third major descent (cucu)
- E-bell/like impulse
- F-percussion impulses
- G-iterative sequence

We must point out that, in some cases, some synthesis have been made and some sound objects have been grouped under an only one name since then morphological properties were quite similar.

The hearing check on listening gives quite important informations of the excerpt structure, pointing out that to the major mass thickness corresponds also the highest point for sound intensity (at 5'22"), a value that won't be reached any more again, except at the end (but with a different dynamic).

The fact that just after this top there is a stop and it begins again from a very low thickness, allows us to establish preliminarily this point as one of the most relevant in connection to the excerpt breaking up. This hypothesis is supported also by the

progressive mass thickness increase from the excerpt beginning to the point in question, sign of a sound "directionality" that finds achievement and conclusion at the top.

An other relief supporting this hypothesis is the observation that, from 5'23" to the end, there is a restriction of the occupied sound band (always by A sounds), opposed to the B sound raid (at a lower-pitched range): so besides a decreased mass thickness it occurs a more infrequent sound events distribution.

Among the several motion typologies included in D. Smalley classification (Smalley, 1986) we have chosen the most suitable ones to define the excerpt principal sound objects.

The characteristic motion typologies are:

- linear
- unidirectional ascent or descent
- curvilinear parabola
- eccentric endogeny
- centric/cyclic vortex
- eccentric conglomerate

The examination of table n.1 gives precious functional informations to a complete structure hypothesis since, if we observe the different motions stratification we can identify a first breaking up at 2'10" where to the series of mainly ascent and descent motions (and to the eccentric endogeny motion of B sound) are added the eccentric conglomerate B sound motion at 2'18" and the linear A sounds motion of low-pitched and medium range. This second section goes on up 5'22" (at 5'04" and on we have centric/cyclic vortex G sounds) where we find the already mentioned excerpt top. Afterwards the elements reductions characterizes the last section (from 5'23" to 8'11").

We must say that the choice of the first segmentation can be also criticized in regard to the fact that the linear motions entrance can be considered as a simple accumulation inside that "gesture" that appears from 0" to 5'22".

Therefore it is true that we notice a considerable gradual fall from 1'50" to 2'07" as if we were at a first impulse-section end. A sound entrance to the low-pitched range, after a section completely distributed in the medium-high region, is also itself a rather characterizing element.

Coming back to the notion of "continuous" motions superposition it is interesting to observe the excerpt tessitura and gestual virtues (Smalley, 1986).

We report D. Smalley classification:

- Gesture
- Texture
- Gesture framing
- Texture setting

In the gesture framing the gestural dimension is prevalent, but the directionality is not so stressed as in the gesture itself; in the texture setting the inner elements are balanced like in weaving but on the whole a certain directionality is shown.

We can notice the tessitura element clear predominance, surely in the first two sections (0"-2'11" and 2'12"-5'22"); in the third the gestural element appears more clearly in some points (6'04", 6'32", 7'32"; 7'43").

The principal breaking up we have got from the previous observation can be, in its turn, divided further; therefore the work deliberate functionality leads us not to "section" up to the improbable. In the second section we notice some sounds that, in one sense, differ from the excerpt sound landscape; we refer to the C rolling sound (3'34") to E bell -like impulses (4'10" and on) but above all to the D third major descent (like a *cucu*) (3'40" and 4'19"), a strange and curious element in the excerpt economy we will find again in the third section at 6'45". This last section, precisely, seems a diffraction of the second; a section where sudden appear fragments previously heard speech (like the short episodes at 6'04" and 6'40" that recall the 5'22" top).

The same final part (7'41"-8'11") presents again the B sound with such an "explosive" value to recall itself at 5'22". We notice two segmentation possibilities:

- we can classify the first section as a *Proposal*¹; but seen the substantial graduality that links it to second (that is from 2'11" to 5'22") it can be considered as a "*Preludiando*", since the sound components real development is occurs in the second section (from 2'11" to 5'22"), the real *Proposal*. The third section, for the analogies with the second (as reported before) can be interpreted as *Evolution*, presenting quickened and modified fragments of previous situations. This quickening stands out at the end mainly (7'41"-8'11") where the closer and closer B explosions bring to a decisive *Conclusion* and in a rapid disappearance.
- We can, on the contrary, consider the first individualized section as the real *Proposal* and the following two as a first *Evolution* (in connection with the timbral thickness) and as a second *Evolution* (in relation with the sound typologies durations).

As far as the conclusive part concerns (7'41"-8'11") we could discuss endlessly: the excerpt for the conclusive rush shortness and for its sudden appearance-disappearance, seems not to finish!

This kind of critical opinion doesn't pretend to be an assertorial judgement, but only a proposal: the points of view can be extremely various and divergent. Therefore if we accept, not definitively, these premises, *Continuo* can be seen as a "window opened over the unexplored sound electroacoustic continuum", a window that, having been closed quickly, doesn't end the experience, leaves the fantasy door open, preludes to new spaces opening, new horizons: it can be seen as a metaphore of the "beginning", already in maturing phase, of the electroacoustical contributing way of B. Maderna.

Note

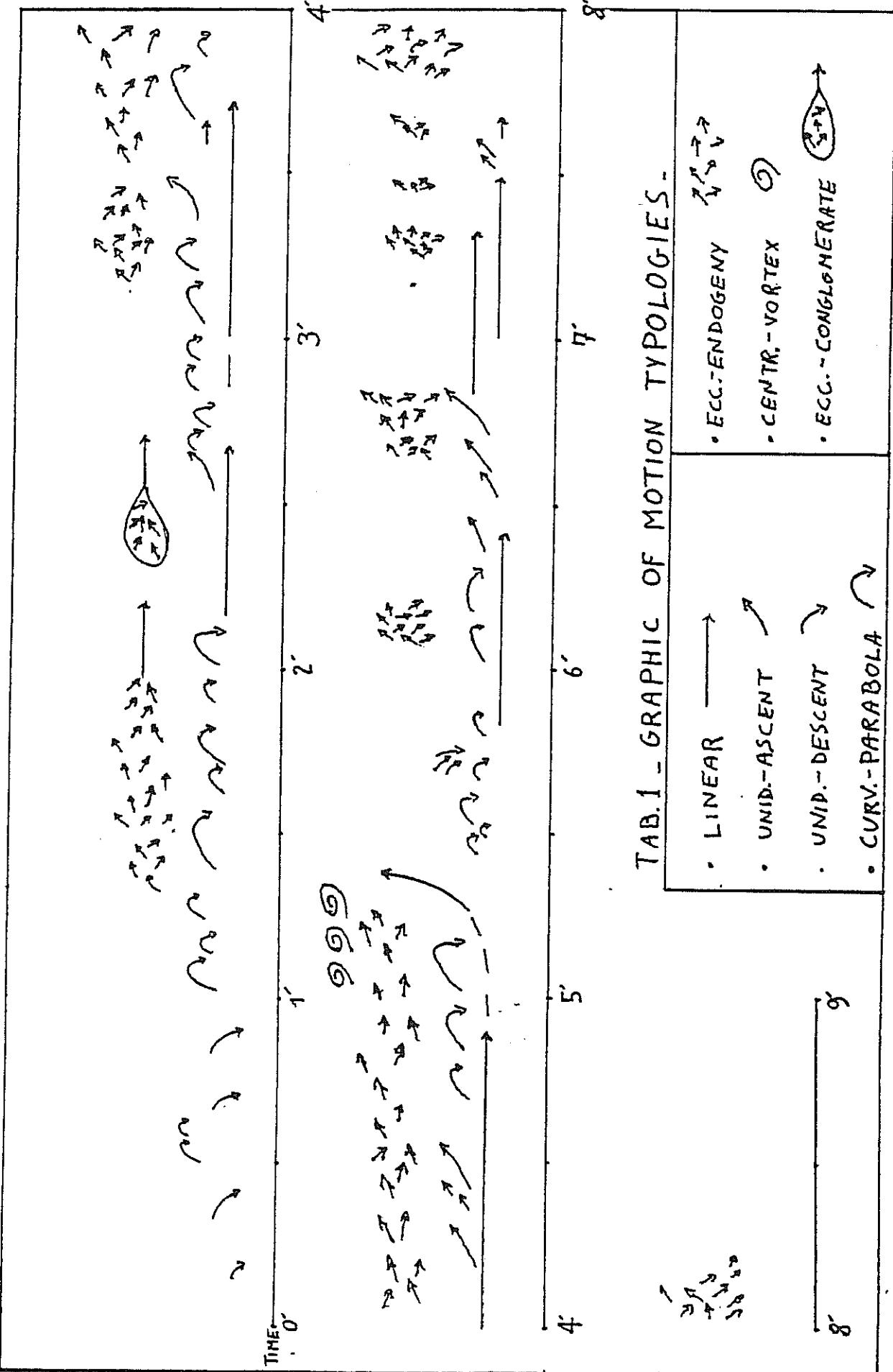
¹ These sound objects give rise to perceptive-discursive units which may be taken to mean Proposal (a simple complete sense impulse), Suspension (when the parameter's general dynamics are static), Evolution (when a parameter varies while the other remain stable) or actual Conclusion (when through different dynamics the end of the expressive discourse is reached).

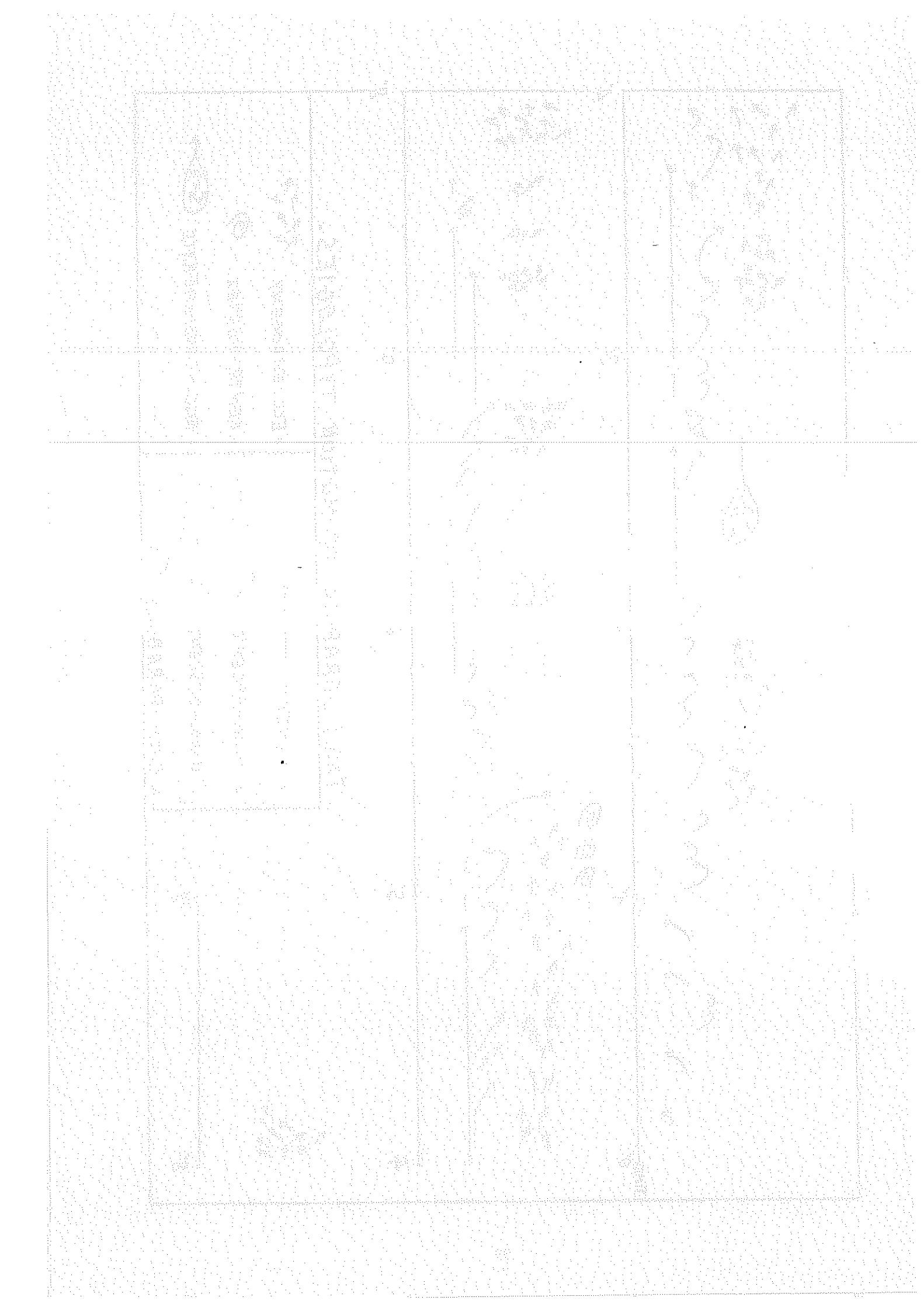
References

- D. Smalley, "Spectro-morphology and Structuring Processes", in *The language of electroacoustic Music*, London, Macmillan press, 1986, pp. 61-93.

Address

Prof. E. Pancaldi
via J. F. Kennedy, 20
40054 Budrio Bologna, Italy





Problems of methodology: the analysis of *Kontakte*

Thinking in terms of stories

Alessandro Cipriani

Electroacoustic Music Professor at the 'V.Bellini' Musical Institute of Catania
A.C. - Via Voghera, 7 00182 Roma
Tel. 39-6-7024309
e-mail a.cipriani@agora.stm.it

Abstract

The non-linear sequences of the evolutionary process of *Kontakte*, interpreted through the concepts of ontogeny, phylogeny, context and relevance as formulated by Gregory Bateson, become a model for the analysis of the piece.

1 The difficulty of finding an appropriate model for the analysis of *Kontakte*. The genesis of the piece.

An analysis of *Kontakte* can last years and years because of its complexity. The subject of my study is the form of this piece. From the very beginning *Kontakte* seemed to me a world full of contradictions, as if the story of its development wasn't linear at all. We must consider that in the mid 50's, time of the crisis of integral serialism, (preceding the realization of this piece) the strong ideas about the possibilities of 'pure' electronic music (correspondences between material and form and between acoustic micro-structure and musical macro-structure etc.) became weaker for reasons we won't mention here. Therefore the apparent non-linearity of the genesis of *Kontakte* is consistent with a time when many postulates were being discussed.

One of the most striking things one notices while approaching *Kontakte* is its contradictory genesis. The initial project of this piece was based on *IMPROVISATION*; the fundamental criterion of sketches and plans is the *SERIALIZATION* of the transformation values (*Veränderungsgrade*) ; the realization score reveals the *EXPERIMENTAL CHARACTER* of the research in the electroacoustic field; we can also find *ALEATORY ASPECTS* bound to the techniques used in the studio work (impulse techniques, etc.); the modifications of the original structure that Stockhausen made to finish the work in time for the performance, and the analysis of the 'sound text', all reflect compositional strategies bound to music *TRADITION*; statements about *MOMENTFORM* being the basic idea on which the work was built by J.Harvey and other musicologists

are also well known. Therefore one of the first problems one encounters is to find a model for the analysis that does not reduce that complexity and allows us to understand the real nature of the piece.

2 Finding a model

A possible answer to that problem comes from the simple observation that those different aspects are all part of the genesis of the piece, of its 'story'. Gregory Bateson , in *Mind and Nature, a necessary unity* writes: *I would assume that any A is relevant to any B if both A and B are parts or components of the same 'story'[...] Without context, words and actions have no meaning at all. This is true not only of human communication in words but also of all communication whatsoever, of all mental process, of all mind, including that which tells the sea anemone how to grow and the amoeba what he should do next.[...] the fact of thinking in terms of stories does not isolate human beings as something separate from the starfish and the sea anemones, the coconut palms and the primroses. Rather, if the world be connected, if I am at all fundamentally right in what I am saying, then thinking in terms of stories must be shared by all mind or minds, whether ours or those of redwood forests and sea anemones. Context and relevance must be characteristic not only of all so-called behavior (those stories which are projected out into "action"), but also of all those internal stories, the sequences of the building up of the sea anemone. Its embryology must be somehow made of the stuff of stories. And behind that, again, the evolutionary process through millions of generations whereby the sea anemone, like you and me, came to be- that process, too, must be of the stuff of stories. There*

must be relevance in every step of phylogeny [that is, the race history of a species], and among the steps.

Going back to my analysis of *Kontakte*, the study of its different aspects found a unifying context through the research of that 'relevance' mentioned by Bateson. I didn't try to unify in a forced way what is non-linear in the piece. As in an evolutionary process, *every evolutionary step is an addition of information to an already existing system. Because this is so, the combinations, harmonies, and discords between successive pieces and layers of information will present many problems of survival and determine many directions of change.* (Bateson). This was a model for me to proceed during the analysis and also a key to understanding the evolution of the piece.

3 An outline of the fundamentals of my work of analysis on *Kontakte*

3.1 Sketches and plans

The study of some sketches and of the general plan of *Kontakte* drawn by Stockhausen revealed that at this preliminary level Stockhausen's idea about the piece was quite far from *Momentform*. I found a double palindromic structure of all *Veränderungsgrade* (the way parameters change) and *Freiheitsgrade* and the tripartition of the durations (number 3 is often used at all levels). The relation among all the parameters is built with a symmetrical logic and a numerological/serial strategy.

3.2 the 'sound text' of the electronic version,

In this analysis I took into account:

- 1) Prevailing of Gesture or Texture (as formulated by Denis Smalley) or intermediate levels in each section.
- 2) Prevailing dynamics and possible spatial 'rotation' in each section.
- 3) Morphology of sound objects.
- 4) Tendency to variability or homogeneity of timbre within each section.
- 5) Prevailing timbral typology based on the 6 groups mentioned by Stockhausen (Metal-Noise, Metal-Sound, Wood-Noise, Wood Sound, Skin-Noise, Skin-Sound) in those sections where this frame of reference is relevant. The analysis is not only based on listening but also on a comparison with the performance score. I found some criteria and formal strategies within the piece that are used several times and with the same characteristics:

3.2.1 Directional Sections

Use of homogeneous timbres, crescendos, spatial rotation and *Gesture Framing* model in directional sections heading toward dynamic-timbral-spectral 'apices' (see 3.2.2).

3.2.2 'Apices'

Spectral space plenitude (co-presence of low and high frequencies), *ff* or *fff* dynamics, tendency to inharmonicity and *Gesture Framing* model.

3.2.3 Extinction of the directionality

Clear diminuendo dynamics (sometimes together with the use of delay), textural, graduated continuant (or with a very long decay) sounds, homogeneous timbre and *Texture* model.

3.2.4 Static Fragmentary Sections

Timbral variability, overlapping of sounds, lack of pauses, non-extreme dynamics, *Texture Setting* model in the 'static fragmentary' sections.

3.2.5 Directional Fragmentary Sections

Contrasting dynamics, use of pauses, apparent inconsistency of the sound objects, non-overlapping of sounds *Gesture* or *Gesture Framing* models.

3.2.6 Pseudo-instrumental Cadenzas

Timbres that tend to imitate percussion instruments, tendency to organize sounds in phrases or clusters, use of dynamics that are close to an instrumental model, *Texture Setting* model.

I identified, following the above mentioned criteria, 5 macro-sections each containing its own 'apex', except for the last one that carries a 'coda' function. In those macro-sections one can clearly recognize formal functions and behaviors that are repeated each time. Some of the compositional strategies here have a very strong relation with music tradition (e.g. directionality towards climax and then extinction of directionality, cadenzas etc.). A clear-cut division between macro-sections is not completely consistent with the flow of the piece, of course, (in some cases transitions are smooth etc.) so durations are indicated as reference. Roman numbers refer to sections as indicated in the score by Stockhausen.

MACROSECTION ALPHA (duration: 7'2,8")

- I directional fragmentary
- II transition
- III directional-apex-extinction

MACROSECTION BETA (dur.: 4'16,8")

IV static fragmentary

V directional-

VI/VII-A (1) -apex in three phases-extinction

MACROSECTION GAMMA (dur.: 10'10,3")

VII-A (2) transition

VII-BCDE static fragmentary

VII-F pseudo-instrumental cadenza (skin)

VIII abstraction-graduated spatial passage

IX-A pseudo-instrumental cadenza (wood)

IX-BCDE static fragmentary

IX-F directional-

X -apex-extinction-transition

MACROSECTION DELTA (6'15,6")

XI directional-

XII double interrupted apex

XIII-ABC prolongation of the apex-extinction

FINAL MACROSECTION (6'46,3")

XIII-DE directional fragmentary

XIII-F pseudo-instrumental cadenza - transition

XIV/XV-A coda (suspended directionality)

XV-BCDEF/XVI directional ending-extinction

3.3 The instruments and tape version

3.3.1 The graph

I elaborated a graph representing, within sections that are proportional to their duration, the presence/absence of the 27 instruments of *Kontakte* for electronic sounds, piano and percussion in relation to the 7 general types of sounds of the tape part mentioned by J.Harvey (this idea of 7 types of sounds was then criticized in my work). The instruments were also grouped in six types, following the indications of Stockhausen and Heikinheimo).

From the analysis of this graph one can notice that the distribution of the instruments along the piece is not based on the division in six instrumental groups. This distribution is rather homogeneous among the groups. I also compared the above mentioned division in macro-sections with the results of the graph:

3.3.2 Formulation of an hypothesis

My idea after having analyzed the piece for instruments and tape is that the construction of its form is based on the formal articulation of the tape part, the function of the instruments being to constitute a wide timbral constellation that relates to it in various ways. The function of the tape part would be that of a formal frame of reference and a means of spectral fusion in relation to the timbral constellation

of the instruments. This hypothesis is based also on some considerations:

- a) Stockhausen realized a tape version but didn't write any version for instruments only, so instruments have no autonomy from the tape. 'Contacts' take place between instruments and tape or within the tape
- b) in the version for instruments and tape the latter essentially remains the same, in comparison with the tape version. That means that for Stockhausen the presence or the absence of the instruments didn't cause any need for a change of the formal functions within the tape. They can be obscured, confused, put in the background or in evidence through fusion or contrast by the intervention of the instruments, but without their weaving the piece would have no meaning.

3.3.3 Study of some examples of relation

between instruments and tape in *Kontakte*

On the basis of this hypothesis I studied some different functions that are given to the instruments in relation to the electronic sounds. The form, considered as a 'perceived event' (following Dahlhaus' concept) is modified by these 'contacts' between instruments and tape mainly at the articulation level within the sections, much less at the macrostructural level. The most evident of those mentioned functions are:

- a) timbral 'expansion' and 'underlining' of the electronic sounds
- b) completion of phrases, as in a *Klangfarbenmelodie*
- c) relative autonomy of the instruments in some moments when their function separate from that of the electronic sounds
- d) 'mixed' function: co-presence of function b) and c)
- e) 'underlining' of the formal function and of the timbres of the electronic sounds
- f) pseudo-instrumental interaction (during the pseudo-instrumental cadenzas of the electronic sounds).

4 The projection of the non-linearity of the genesis onto the form of the piece

The analysis, based on the above mentioned criteria, and the consideration of the genesis of the piece and of the historical point of view, can confirm that Stockhausen in this piece synthesizes many different, sometimes contrasting ideas and compositional strategies. He seems to accept influences from tradition, alea, serialism and numerology, as well as from the concept of *Momentform*, from improvisation

and experimental practice. Some of these influences became weaker because of inevitable methodological conflicts or because of practical problems: how could experimental practice and non-deterministic control have co-existed with plans of the entire piece that were so rigidly determined at the beginning? Heikinheimo refers to the aleatory aspects of some techniques used at the time, (e.g. impulse techniques, use of delay, filters, transpositions), demonstrating the strong role of perception in composing. This can be one of the reasons why many plans were abandoned or radically changed. One possible hypothesis is that Stockhausen, after having prepared those plans, found that the best results came out of experimental research (precisely documented in the realization score) and that those serial plans couldn't be easily re-modeled on the base of those results. A methodological conflict: some parameters (e.g. some durations of sections) remain unchanged, some others have completely different values. This is only an example that metaphorically or directly pertains to the *problems of survival* and the *directions of change* Bateson writes about, referring to living species and their phylogeny and ontogeny (the latter is the process of development of an individual, that is, embryology plus all the changes that are determined by environment and habits). *Kontakte* has often been described as a Momentform piece. This idea certainly reduces its complexity and often the simplest answer isn't the only one. I think that an analysis of this piece is incomplete if it doesn't include also the aspects bound to its 'ontogenesis', though this can't be described in a precise manner from a chronological, 'epigenetic' point of view. The non-linearity of the genesis, of the stories of *Kontakte*, 'projects' itself onto the form of the piece. The 'polyhedral' quality of the piece that arises from the analyses, lets me think that the 'contacts' in the piece don't pertain only to the relation 'between electronic and instrumental sound-groups' or 'between self-sufficient, strongly characterized moments' or 'between various forms of spatial movement' (as Stockhausen has said). I could trace, in fact, 'contacts' between various compositional experiences that find their own relatively incomplete, problematic synthesis in the piece. That synthesis appears to me to be an extremely stimulating one in our time because of our present plurality of 'languages' and aesthetic perspectives. In this study I accepted the comparison between different analytical categories and guiding lines to explore the different facets and stories of the piece but I didn't forgo a model, that, in its relation to the concept of 'ontogenesis', turned out to be intrinsically unitary and non-linear at the same time.

References

- Cipriani, A. *Due tesi complementari sulle due versioni di Kontakte di K. Stockhausen* - Thesis in Electronic Music Conservatorio di S. Cecilia 1993
- Cipriani, A. "Toward an Electroacoustic tradition?" *Proceedings of the International Computer Music Conference*, 1995, pp. 5-8 - Banff (Canada)
- Harvey, J. *The music of S.*, London 1975 pp. 87-90
- Heikineimo, S. "The electronic music of K.S." *Acta Musicologica Fennica* 6 Helsinki 1972 pp.115-221
- Kirchmeyer, H. *Zur entstehungs- und Problemgeschichte der 'Kontakte' von Karlheinz Stockhausen* introduzione al disco Wergo 60009 Prima edizione
- Maconie, R. *The works of K.S.* Oxford University Press London 1976 pp.134-146
- Smalley, D. "Spectro-morphology and Structuring Processes" in *The Language of Electroacoustic Music*, Emmerson S. (ed.) Macmillan London 1987 pp.61-93
- Smalley, D. *Spectromorphology: Explaining Sound-shapes* (awaiting publication)
- Stockhausen, K. "L'unità del tempo musicale" in AA.VV. *La musica elettronica* a cura di H. Pousseur Feltrinelli MI 1976 pp.150-160
- Stockhausen, K. "Musica elettronica e musica strumentale" in AA.VV. *La musica elettronica* a cura di H.Pousseur Feltrinelli MI 1976 pp.45-51
- Toop, R. "S.'s Electronic works:Sketches and Worksheets from 1952-1967" in *Interface X*, 1981 pp.183-189
- Stockhausen, K. *Kontakte für elektronische Klänge, Klavier und Schlagzeug* Aufführungspartitur Universal Edition London 5.Auflage 1974
- Stockhausen, K. *Kontakte Elektronische Musik Realisationspartitur - Aufführungspartitur* Universal Edition UE13678 London 1968
- Stockhausen, K. *Kontakte* for electronic sounds pf. and percussion -A.Kontarsky (piano), Ch.Caskel (percussion), CD Stockhausen Verlag N°6 1993
- Stockhausen, K. *Kontakte* Electronic music CD Stockhausen Verlag N°3 1992

Forme e strutture nella musica elettroacustica tra gli anni Cinquanta e Sessanta

Francesco Giomi, Marco Ligabue

Divisione Musicologica CNUCE-C.N.R.
Firenze - Italia

Abstract

The paper deals with the comparative analysis of six electroacoustic pieces from the early Fifties to the early Sixties. The analysis has been developed by taking into account the formal and structural levels of pieces organization and following the aesthetic-cognitive methodology formulated by the authors and mainly based on perception.

1. Introduzione

La relazione si pone come obiettivo l'analisi comparata di una serie di lavori elettroacustici composti tra il 1953 ed il 1961.

Le diverse istanze compositive che attraversano quegli anni, e particolarmente quelle legate all'esperienza elettroacustica, mettono apertamente in discussione i criteri organizzativi del materiale sonoro, sottoponendolo a modalità interpretative di natura eterogenea tanto sul piano qualitativo quanto su quello relativo ai concetti formali e strutturali. Peraltra, i molti laboratori di musica elettronica che iniziano ad operare nel periodo, ospitando compositori di varia provenienza culturale e geografica, agiscono spesso come lenti focali nei confronti dell'uso di materiali o di particolari indirizzi compositivi.

Per quanto un lavoro analitico che voglia darsi più esaustivo richieda il confronto con un più ampio repertorio, ci è sembrato comunque interessante tentare la comparazione su di un gruppo di brani più circoscritto, concentrando l'attenzione su lavori brevi di compositori di rilievo:

- K. Stockhausen, *Studie II* (1953), WDR Colonia
- B. Maderna, *Notturno* (1956), RAI Milano
- F. Evangelisti, *Incontri di fasce sonore* (1956/57), WDR Colonia
- P. Schaeffer, *Etude aux sons animés* (1958), RTF Parigi
- G. Ligeti, *Artikulation* (1958), WDR Colonia
- J. Tenney, *Analog I-Noise study* (1961), Bell Lab. N.J. USA

Già questo repertorio, nella sua essenzialità, permette una valutazione delle diverse modalità con cui la forma, pur nella brevità, si esprime, mettendo in rilievo tutta

una serie di peculiarità compositive che proprio tramite il confronto vengono ad assumere valore. Anche l'analisi strutturale interna delle composizioni consente di mettere in luce aspetti interessanti dal punto di vista della diversificazione dei materiali sonori, per modalità d'uso e per qualità.

L'indagine analitica viene condotta mediante l'approccio estesico-cognitivo sviluppato dagli autori e per la cui descrizione si rimanda ai testi in bibliografia. Il metodo, basandosi su fattori di carattere prevalentemente percettivo, si presta all'analisi della musica elettroacustica e consente di mettere in luce risultanze particolari del testo sonoro che possono emergere solo attraverso un'analisi dell'ascolto condotto sui brani.

2. Forme a confronto

Tre pezzi presentano forma unitaria, ciascuna realizzata attraverso modalità diverse. Stockhausen realizza tale unitarietà attraverso una omogeneità relativa del materiale, organizzandolo senza perseguire apparenti direzionalità e ponendo a chiusura un evento particolare - diverso da tutti gli altri - che marca il momento finale. A questo livello il procedimento seriale consente al compositore di realizzare le proprie intenzionalità estetiche sul piano dell'unitarietà formale.

L'interezza della forma è realizzata invece da Ligeti attraverso la distribuzione lungo tutto il pezzo di materiale disomogeneo, montato con modalità del tutto orientate ad un discorso fraseologico: la sua organizzazione risulta spostata dal piano formale a quello delle catene sintagmatiche. Per tal motivo certe direzionalità, riscontrabili come vedremo sul piano strutturale, appartengono al livello discorsivo dei sintagnmi senza investire il piano formale.

Nel brano di Tenney il materiale risulta molto omoge-

neo, ancor più che in *Studie II*, ed è organizzato per mezzo di un comportamento graduale con un coerente e progressivo processo di accumulazione. Nessun evento si staglia sugli altri e la coda finale riscontrata a livello strutturale non può essere attribuibile al livello formale dato che al suo interno i parametri non cambiano in maniera sostanziale, al di là di un lampante quanto improvviso salto dinamico.

Due invece sono i pezzi che presentano una organizzazione formale di tipo A - B.

Nel lavoro di Maderna la strutturazione formale è ottenuta attraverso una diversificazione dei caratteri: melodico vs armonico. Si ha l'impressione che il compositore utilizzi un trattamento "strumentale" degli eventi elettronici, imponendo alla prima sezione materiale tonico, fatto di sequenze ritmiche di incisi "melodici"; nella seconda sezione vengono aggiunti elementi di rumore: gli oggetti sonori, alquanto allungati e con dinamiche sensibilmente disomogenee, risultano sovrapposti, imprimento un comportamento "armonico-polifonico" a questa sezione.

La suddivisione formale dello studio di Schaeffer è ottenuta mediante un'organizzazione di carattere narrativo del materiale. La sezione A sviluppa una propria autonomia grazie al suo schema interno di natura circolare, la cui coerenza a questo livello è data dal ritornare di un evento particolare di natura melodica che apre e chiude la sezione. In B invece prevale una situazione di mutazione continua del materiale con caratteristiche di maggior articolazione.

La forma della composizione di Evangelisti è tripartita: su questo piano si assiste alla strutturazione più complessa che, al di là degli intenti seriali, dimostra intenzioni di carattere narrativo. Il materiale è peculiare per ognuna delle tre sezioni e risulta disomogeneo solo in A, l'unica zona che per andamento può richiamare l'influenza dello *Studie II* di Stockhausen. Il comportamento fra le tre sezioni si differenzia notevolmente in conseguenza di strutture ritmiche specifiche.

Per uno schema complessivo delle suddivisioni formali si veda la figura 1.

3. Strutture a confronto

Salta subito agli occhi che anche all'ulteriore passo analitico del livello strutturale *Artikulation* rimane neutro nella sua integrità: non sono infatti evidenti processi di strutturazione tali da giustificare una suddivisione precisa della composizione. Semmai i criteri strutturali permettono di evidenziare due zone particolari dove un qualche movimento di struttura potrebbe aver luogo, nel senso che si riscontrano due brevi momenti (quasi simmetrici) dove si nota l'intensificazione di alcuni parametri (particolarmente dinamica, tensione e moto)

che non paiono però sufficienti a determinare una vera e propria suddivisione. L'organizzazione vera e propria si attua a livello dei sintagmi sonori, organizzazione che pur avendo una sua generica ripercussione a livello strutturale non consente di essere trattata in maniera più approfondita su questo piano.

In due dei brani si riscontra invece una parziale suddivisione in sottosezioni.

Nel primo caso, *Incontri di fasce sonore*, Evangelisti lascia inalterate le zone B e C, peraltro brevi, dividendo in quattro la prima sezione formale attraverso un'alternanza costante di moto e densità.

In *Notturno* Maderna non suddivide la prima sezione formale ma sembra attribuirle una funzione analoga a quella che Evangelisti dà alla sua sottosezione strutturale a1 che presenta un carattere introduttivo. Se nel caso del compositore romano tale carattere è realizzato attraverso la presentazione del materiale, in *Notturno* si assiste ad una esposizione melodica delle frasi - riprese poi anche nel corso della composizione - che, conferendo una morfologia ad arco alla sezione A, garantiscono l'integrità formale-strutturale. La sezione B si sviluppa invece in tre zone di diversa impostazione strutturale, l'ultima delle quali attua una funzione di coda grazie al suo andamento ed al cluster finale in rapido rilascio. Tale coda è preceduta da un momento di culmine di tutti i parametri (sottosezione b2) che raggiungono fra 2'08" e 3'10" il loro punto di massimo.

L'inserimento di un elemento strutturale con funzione di chiusura finale è riscontrabile anche nel pezzo di Tenney: preceduti da una sottosezione relativamente culminante (a2), gli ultimi 40 secondi presentano caratteristiche conclusive. Caratteristiche di questo tipo si riscontrano anche negli ultimi istanti del pezzo di Schaeffer dove il risultato è comunque di minore rilevanza per la sua relativa brevità ed i caratteri che risultano appena accennati.

Questi due ultimi lavori, insieme a quello di Stockhausen, presentano un'organizzazione di carattere strutturale completa su tutta la forma, pur con modalità tra loro alquanto differenti.

Tre sono le parti sia in Stockhausen che in Tenney. Da un punto di vista percettivo il lavoro del compositore tedesco risulta caratterizzato da un andamento tripartito, dove la breve zona centrale a2 svolge la funzione di ponte tra le due estreme, diversificate tra loro dai parametri timbrico-dinamici e di densità. Il compositore americano struttura invece la composizione secondo la logica di una continua intensificazione dei parametri, con lo scopo di creare un senso continuo di crescendo: le sottosezioni a1 ed a2 si differenziano per il fatto che la seconda mantiene stabili il maggior numero di parametri sul punto di culmine. La coda, come detto, chiude la composizione.

Infine *Etude aux sons animés*, che presenta tra tutti l'organizzazione strutturale più articolata e dove le due sezioni formali hanno un carattere autonomo: la prima, composta da tre sottosezioni, risulta compiuta nella sua morfologia circolare (a1 -a2 - a1'); la seconda percorre un andamento diversificato con un punto di rilassamento b3, dopo un iniziale aumento dei parametri, seguito da una conclusione in ulteriore crescendo. Il susseguirsi delle sottosezioni procede per progressiva aggiunta di elementi sonori nuovi con la conseguente eliminazione parziale di altri già utilizzati.

Per uno schema complessivo delle suddivisioni strutturali si veda la figura 2.

4. Considerazioni analitiche generali

Tre delle composizioni provengono dallo stesso studio di produzione (*Studie II*, *Incontri di fasce sonore* e *Artikulation*), la WDR di Colonia. L'impressione percettiva che ne risulta è quella di una relativa omogeneità di materiale, sicuramente dovuta alle comuni sorgenti ed ai vincoli che esse imponevano per qualità e natura. Ciò nonostante le differenze non mancano: se Evangelisti amplia la tavolozza timbrica rispetto allo *Studie II* (il primo brano in ordine di tempo) senza discostarsi molto da quelle morfologie sonore (soprattutto quelle di dinamica/involuppo), Ligeti è quello che si differenzia maggiormente imprimendo una strutturazione fraseologico-vocale caratterizzata da sequenze di eventi sonori con morfologia "liquida". Tale caratteristica che informa non più singoli elementi, bensì si dimostra una vera e propria strategia di organizzazione del moto interno agli oggetti, sembra una delle caratteristiche più innovative che il compositore ungherese apporta al linguaggio elettroacustico di quegli anni. Peraltra anche nell'ambito di scuole musicali e studi di produzione differenti si notano eventi timbricamente simili, per esempio alcune fasce toniche in crescendo presenti sia in Evangelisti sia in Maderna.

Per quanto riguarda le curve generali di tensione si può notare nei primi due lavori della WDR (Stockhausen ed Evangelisti) un tipo di andamento che si sviluppa in tre fasi e che ha il suo punto di massima nella regione centrale: anche questa somiglianza può in qualche modo sottolineare una certa vicinanza di intenti tra i due compositori.

Seguendo l'ordine cronologico complessivo dei brani, dal 1953 al 1961, si assiste anche ad un progressivo aumento dell'elemento "narrativo" nelle modalità organizzative. Assente, o quasi, in Stockhausen (se si esclude la curva di tensione in tre fasi di cui si è detto) si arriva al brano di Tenney: qui l'elemento narrativo appare unitario e sviluppato integralmente lungo tutta la composizione, dimostrando la capacità da parte del compo-

tore di rapportarsi con nuove modalità al paesaggio sonoro. *Incontri di fasce sonore* ed *Etude aux sons animés* risultano all'ascolto più strutturati, con una conseguente maggior "leggibilità" musicale, grazie ad un utilizzo di precisi elementi - le sottosezioni - che supportano con il loro intercalarsi la narrazione. In *Artikulation* la discorsività è garantita dalla strutturazione intrinseca ed estrinseca dei sintagmi; tale strutturazione, ancorché costruita secondo criteri seriali che sul piano strutturale ostacolano l'individuazione di schemi precisi, conferisce comunque coerenza all'intera composizione. Maderna, infine, nel suo generoso fare musica sviluppa invece il piano narrativo attraverso un lavoro di orchestrazione quasi strumentale del materiale sintetico.

Se in Stockhausen e Ligeti appare meno forte la preoccupazione a livello generale di una strutturazione *inizio-fine* del pezzo, pur con una sottolineatura del suono grave finale nel primo caso e della frase "lontana" iniziale nel secondo, in tutti gli altri compositori questa tendenza è più marcata attraverso l'uso tanto di elementi introduttivi quanto di code o di sottosezioni conclusive specifiche.

Per concludere, possiamo precisare che ciò che qui ha maggiormente interessato è stato il lavoro di confronto sui piani formale e strutturale. Per tale motivo non si è proceduto alle successive fasi di dettaglio che il metodo estesico-cognitivo prevede: catene sintagmatiche, strategie composite e strategie di significazione. Un lavoro analitico completo permetterà in futuro di comparare i brani in oggetto anche sotto questi altri aspetti.

Bibliografia

- M. Baroni e R. Dalmonte (a cura di): *Bruno Maderna. Documenti*, Milano, Suvini Zerboni, 1985.
- F. Bayle (a cura di), *Pierre Schaeffer - l'oeuvre musicale*, Parigi, INA-GRM, 1990.
- L. Camilleri: Metodologie e concetti analitici nello studio di musiche elettroacustiche, *Rivista Italiana di Musicologia*, XXVIII, 1993, pp. 131-174.
- F. Evangelisti: *Incontri di fasce sonore*, partit., UE 12863.
- F. Giomi e M. Ligabue: Un approccio estesico-cognitivo alla determinazione dell'objet sonore, in *Atti del Secondo Convegno Europeo di Analisi Musicale*, a cura di R. Dalmonte e M. Baroni, Trento, Università di Trento, 1992.

F. Giomi e M. Ligabue: Metodo estesico-cognitivo e analisi della musica contemporanea, *Sonus*, in corso di stampa.

F. Giomi e M. Ligabue: Evangelisti's composition Incontri di fasce sonore at W.D.R. Aesthetic-cognitive Analysis in Theory and Practice, *Journal of New Music Research*, in corso di stampa.

S. Heikinheimo: *The Electronic Music of Karlheinz Stockhausen*, Helsinki, Acta Musicologica Fennica, 1972.

M. Mila: *Maderna musicista europeo*, Torino, Einaudi, 1976.

K. Stockhausen: *Studie II*, partit., UE 12466.

J. Tenney: Esperimenti di musica di elaboratore 1961-1964, in *La musica elettronica*, a cura di H. Pousseur, Milano, Feltrinelli, 1976, pp. 231-238.

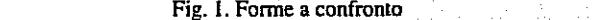
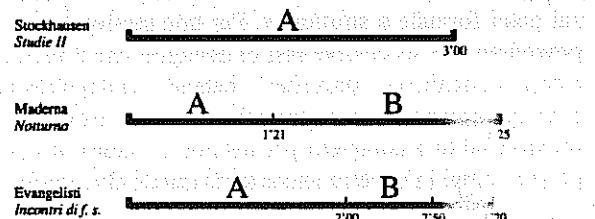


Fig. 1. Forme a confronto

Discografia

F. Evangelisti: *Incontri di fasce sonore*, CD "Cologne WDR - The earliest pieces", BVHAAS 9106, 1992.

G. Ligeti: *Artikulation*, CD "Gyorgy Ligeti", Wergo WER 60161-50, 1988.

B. Maderna: *Notturno*, CD "Bruno Maderna - Musica elettronica", Stradivarius STR 33349, 1994.

P. Schaeffer: *Etude aux sons animés*, CD "Pierre Schaeffer - l'oeuvre musicale intégrale 1948-1979", Vol. 3, INA C 1008.

K. Stockhausen: *Studie II*, CD "Elektronische Musik 1952-1960", Stockhausen 3, 1992.

J. Tenney: *Analog #1*, CD "James Tenney - Selected Works 1961-1969", Artifact Recordings ART 1007, 1992.

Fig. 2. Strutture a confronto

Stockhausen
Studie II

A a1 a2 a3

0'00 1'18 1'50

Maderna
Notturno

A b1 b2 coda

0'00 2'08 3'10

Evangelisti
Incontri di f.s.

a1 a2 a3 a4 B C

0'21 1'03 1'40 1'40

Schaeffer
Etude aux sons a

a1 a2 a1' b1 b2 b3 b4

0'42 1'18 1'37 1'37

3'14 3'34

Ligeti
Artikulation

A

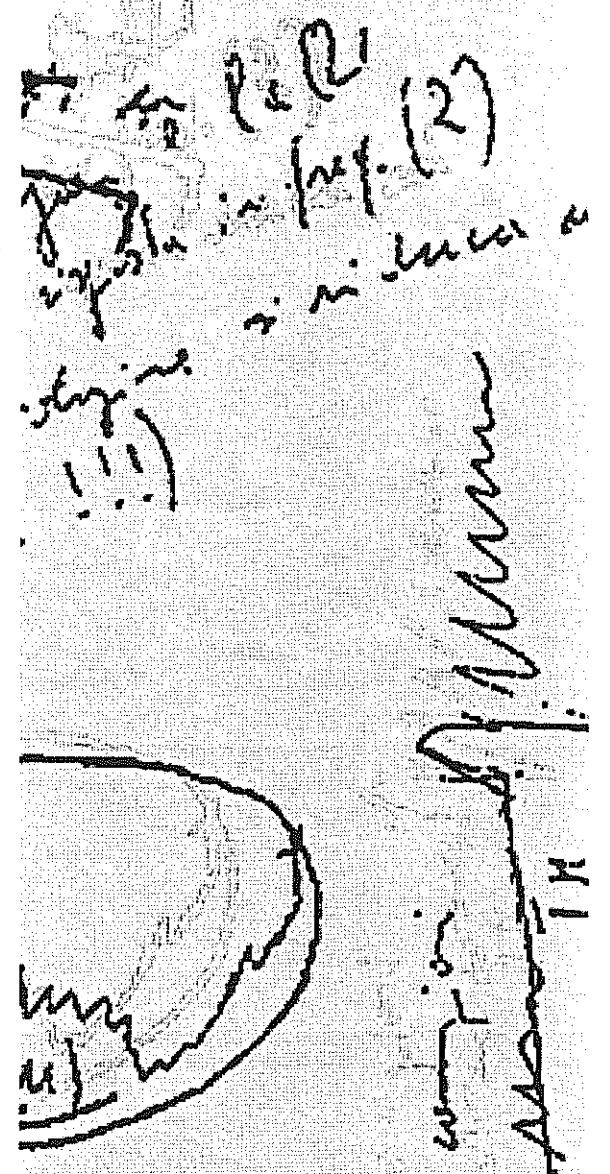
0'00

Tenney
Analog I

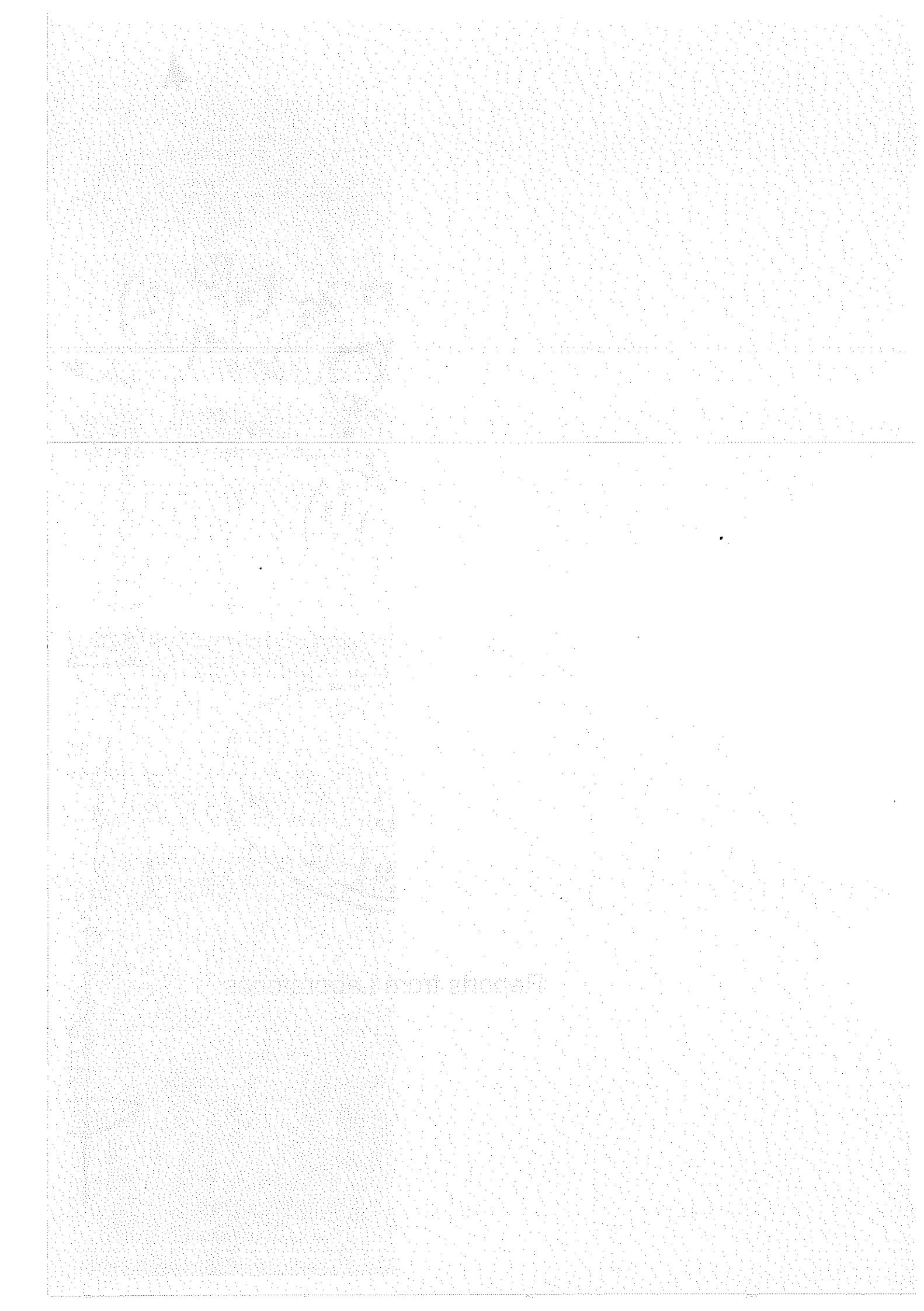
a1 a2 coda

0'00 3'18 3'41

350 million years old
Anatolian Taurus



Reports from Laboratories



Activity Report from Durham Music Technology

Takebumi ITAGAKI[†]♦ Douglas J.E. NUNN[†] Desmond K. PHILLIPS[†]
Dionissios BATJAKIS[†] Alan PURVIS[†] Peter D. MANNING[♦]

[†]School of Engineering, University of Durham,
South Road, DURHAM DH1 3LE, U.K. [♦]Department of Music, University of Durham,
Palace Green, DURHAM DH1 3RL, U.K.

Abstract

Durham Music Technology is an ongoing collaboration between the Concurrent Digital Signal Processing group and the Department of Music at Durham University. The group is interested in multi-processor architectures for real-time synthesis. The 160 T800 transputer network described below is an example of this work. A more recent project involves the construction of a multi-processor T9000 network. Each of the new transputers offers an order of magnitude improvement in performance, and the network is being designed with a memory capability of 8 Mbytes per node. Parallel algorithms which can exploit the concurrency of the network are also being developed. In particular the recognition of harmonically rich signals by logarithmic (constant Q) transformations shows considerable promise. An application of this work has been extended to capture visually the complex interaction of bell sounds during the ringdown process.

1. Multirate Optimisation of Additive Synthesis

Under investigation is the optimisation of the traditional sinusoidal oscillator-bank model of additive synthesis, well known for the diversity, richness and transformability of its sound. A key property is that analysis/resynthesis is supported. The algorithm forms a fundamental part in the foundations of a spectral modelling paradigm which enables manipulation of the raw sound. A disadvantage is that real-time computation is prohibitively expensive due to the bandwidth of control information, which has led in recent years to a concentration on software IFFT techniques. An alternative is the application of multirate DSP principles which integrate with traditional hardware solutions.

Note-based music has a stationarity property in that partial frequency trajectories are known *a priori*, provided margins are made for instrument physics and expressive dynamics. The observation is made that the computational cost of an oscillator is proportional to its update rate. A fixed sample rate of twice the peak frequency of the oscillator suffices during the lifetime of a note partial. This rate may be an order of magnitude less than the standard Nyquist rate. For instance, a 2-kHz partial computed at 4 kHz is ten times more efficient than at 40 kHz. Control bandwidth and the cost

of oscillator updates are reduced accordingly, but the optimisation is functionally transparent.

A synthesis filterbank, constructed from half-band filters, with octave-spaced bins provides a mechanism to allocate near-critical sample rates to oscillators [Phillips *et al.*, 1994]. One filterbank sums and interpolates a distribution of oscillators into a single stream. Additionally, a halving of update rate is achieved by representing sinusoids as complex exponentials and utilising negative frequency. This permits a flexible modular filterbank topology of an arbitrarily complete binary tree and a combination of octave/integer bin spacing. Also, a hierarchical spectral decomposition creates a framework for adapting the cost of computation to note frequency dynamics without artificial constraints on oscillator frequency range.

A single-bus VLSI coprocessor architecture is proposed. Complex exponentials are computed by a pipeline floating-point CORDIC vector rotation unit, efficiently replacing two multiplies and a sine and cosine computation. Control information is in the conventional form of piecewise linear envelope break-point queues from the host CPU. A hierarchy of burst scheduling and caching within a frame timing structure permits the CORDIC unit to operate at close to maximum throughput, providing a maximum of 6400 oscillators from a 20-MHz

clock. A DSP in the coprocessor computes filterbanks and communicates with the CORDIC unit and host CPU via a shared-memory model of interprocessor communication [Phillips *et al.*, 1995].

2. Real-time Synthesis on a Multi-processor Network

Since 1988, this research group has reported on issues concerning multi-transputer audio processors, such as [Purvis *et al.*, 1988]. A prototype architecture for a transputer network was described and demonstrated at the 1990 ICMC [Bailey *et al.*, 1990]. This has been developed into an audio processor, using 160 Inmos T800 floating-point transputers interconnected as a ternary tree. There are 16 transputers fitted onto a standard 3U printed circuit board, which represents a processing power of 140 MIPS. These transputers are permanently connected, but the software configuration of the network is flexible and reprogrammable. Ten such boards form a

network with a maximum processing power of 1.4 GIPS and a total of 640 kbytes on-chip memory.

This network has been used as a test-bed for network architectures for real-time synthesis, and has a custom-made MIDI-to-transputer interface. Using recursive sine oscillators, we have implemented an 88-note organ with fixed note allocation and a 27-note organ with dynamic note allocation [Itagaki *et al.*, 1994]. An 88-note organ with multiple sample rates has also been implemented as a partial realisation of the theory above [Phillips *et al.*, 1994]. Since the implementation is based on recursive sine oscillators using real numbers (complex numbers were used in the theory), its gain in the number of oscillators remains a factor of three. The network can run these programs in real time, at a maximum control rate of 150 keystrokes a second [Itagaki *et al.*, 1995a]. Meanwhile, we are implementing granular synthesis on the network [Itagaki *et al.*, 1995b] and are developing another multi-transputer network using Inmos T9000s.

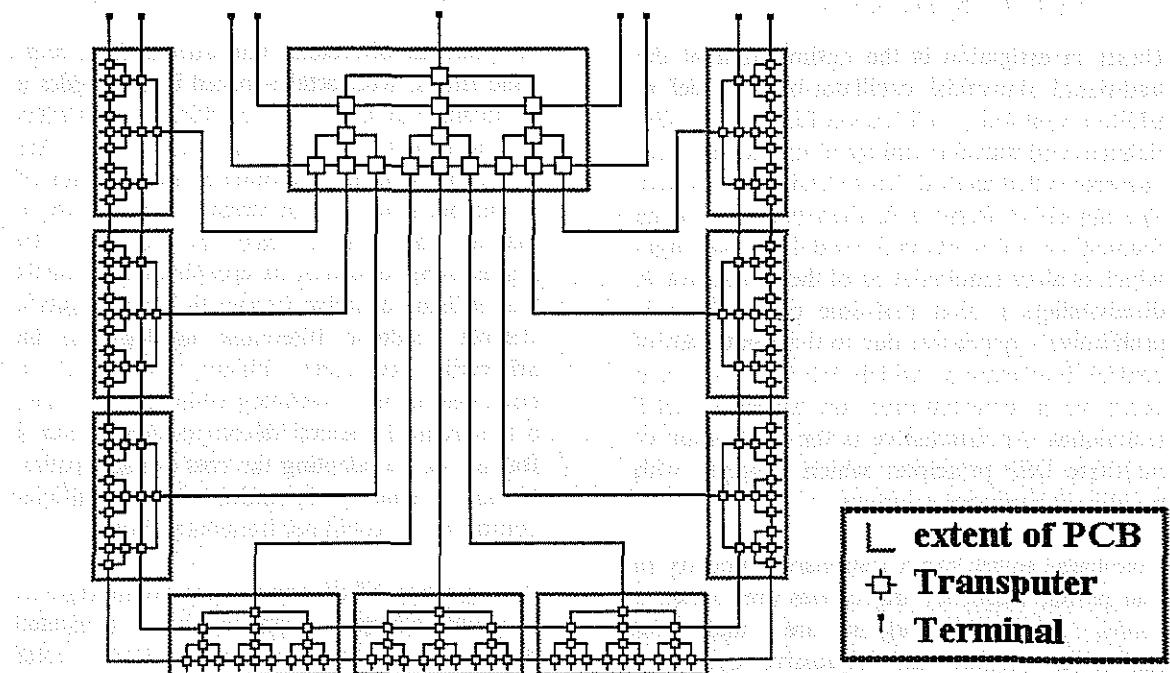


Figure 1: The 160-transputer network

3. Music Analysis

Another field of current research with many potential applications is music analysis. A prototype system has been developed for

extracting event parameters and timbral information from polyphonic music [Nunn *et al.*, 1994]. In order to satisfy the trade-off between time and frequency resolution, the system relies on a multirate analysis scheme.

This is implemented in 3L Parallel C on a Texas Instruments TMS320C40, hosted by a PC. The original data is decomposed into sinusoids, which are then recombined to form the control envelopes required for additive synthesis. The later processing stages are carried out using the PC alone, and this permits detailed graphical display of the data [Nunn *et al.*, 1995].

This system has immediate applications to the characterisation and automated transcription of music. Although the performance depends strongly on the precise nature of the input, the results have so far been encouraging. Such a system also offers the potential for more flexible composition tools that can manipulate and reinterpret existing music.

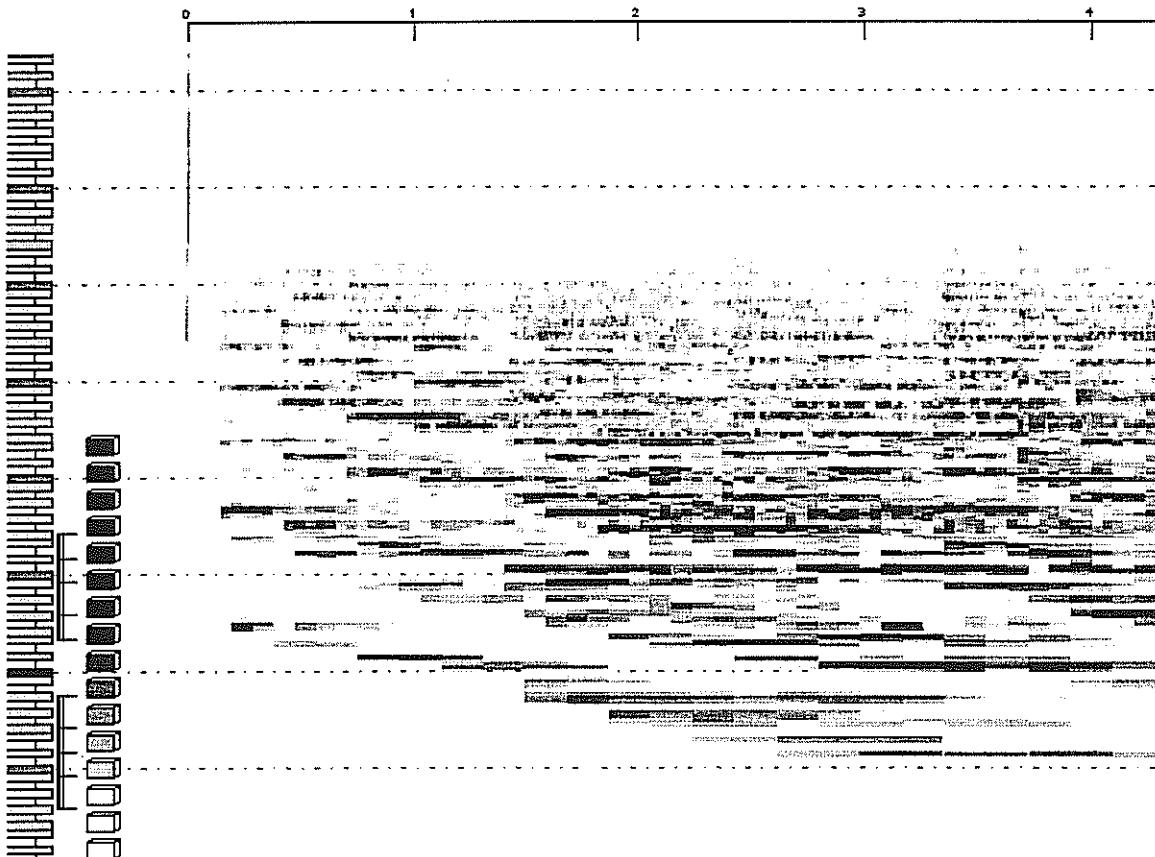


Figure 2: Logarithmic spectrum of a fragment of the ringdown of Durham cathedral bells

4. Musical Applications

As an interdisciplinary research group one of our objectives is the production of new tools for composers of electroacoustic music. This has involved extensive field-testing and custom modification of our more mature research prototypes to ensure error-free operation and an appropriately user-friendly environment for their creative use. Activity in this context has concentrated on the implementation of suitable synthesis algorithms on a network of three T800 floating-point transputers with full support facilities for studio composition. The most significant tools to have emerged to date in this

context are a transputer-based phase vocoder and a Parallel C version of the MIT software synthesis program Csound that incorporates dynamic scheduling routines to distribute the component synthesis tasks across the transputer network [Bailey *et al.*, 1993].

Acknowledgements

TI, DJEN, DKP, and DB are grateful to the Department of Engineering and the Department of Music for facilitating this research. DJEN and DKP are happy to acknowledge studentships from the Engineering and Physical Sciences Research Council.

References

- [Bailey *et al.*, 1990] N.J. Bailey, I. Bowler, A. Purvis, and P.D. Manning, "An Highly Parallel Architecture for Real-time Music Synthesis and Digital Signal Processing Application", in the Proceedings of the International Computer Music Conference, Glasgow, U.K.
- [Bailey *et al.*, 1993] N.J. Bailey, A. Purvis, I. Bowler, and P.D. Manning, "Applications of the Phase Vocoder in the Control of Real-time Electronic Musical Instruments", *Interface*, Vol. 22 (1993): pp 259-275
- [Itagaki *et al.*, 1994] T. Itagaki, A. Purvis, and P.D. Manning, "Real-time Synthesis on a Multi-processor Network", in the Proceedings of International Computer Music Conference, Aarhus, DENMARK
- [Itagaki *et al.*, 1995a] T. Itagaki, D.K. Phillips, P.D. Manning, and A. Purvis, "An Implementation of Optimised Methods for Real-time Sound Synthesis on a Multi-processor Network", (presented at) 5th International Conference on Parallel Computing, Gent, BELGIUM
- [Itagaki *et al.*, 1995b] T. Itagaki, P.D. Manning, and A. Purvis, "An Implementation of Real-time Granular Synthesis on a Multi-processor Network", in the Proceedings of International Computer Music Conference, Banff, CANADA
- [Nunn *et al.*, 1994] D.J.E. Nunn, A. Purvis, and P.D. Manning, "Source Separation and Transcription of Polyphonic Music", *Journal of New Music Research*, Gent, BELGIUM (in preparation)
- [Nunn *et al.*, 1995] D.J.E. Nunn, A. Purvis, and P.D. Manning, "Graphical Display of Musical Information", in the Proceedings of the Colloquium on Musical Informatics, Bologna, ITALY (in this issue)
- [Phillips *et al.*, 1994] D.K. Phillips, A. Purvis, and S. Johnson, "A Multirate Optimisation for Real-time Additive Synthesis", in the Proceedings of the International Computer Music Conference, Aarhus, DENMARK
- [Phillips *et al.*, 1995] D.K. Phillips, A. Purvis, and S. Johnson, "An Efficient Algorithm and Architecture for Real-Time Additive Synthesis of Musical Tones", (presented at) 21st Euromicro Conference, Como, ITALY
- [Purvis *et al.*, 1988] A. Purvis, R. Berry, and P.D. Manning, "A Multi-transputer Based Audio Computer with MIDI and Analogue Interfaces", presented at Euromicro 1988, Zurich, SWITZERLAND. published in *Microprocessing and Microcomputing* 25 (1989): pp. 271-276

Additional information on Durham Music Technology is available through our world wide web homepage; the URL is <http://capella.dur.ac.uk/doug/dmtg.html>.

Towards a Unified Representation of Sound and Analytical Structure in Music.

Igor Popovic,[‡] Ronald R. Coifman,[†] Jonathan Berger[‡]

[‡]Center for Studies in Music Technology, [†]Department of Mathematics
Yale University, 143 Elm Street, New Haven, CT 06520, USA

Abstract

We depart from our past experiences with applying modern techniques of digital audio signal processing to the problem of denoising historical music recordings. The strategy adopted in those projects utilized modern mathematical methods, such as wavelet transforms and wavelet-packet analysis, as well as adapted local trigonometric transforms. Our current work aims at developing refined techniques for the detection of higher-level musical structure in digital musical audio. We consider analytical transcriptions of audio signals which provide a synoptical view of local, midrange, and large-scale structural components in musical audio signals. We describe our work on preparing such signals for higher-level musico-analytical interpretation.

Introduction

Recently, we have reported our experiences with applying new techniques of digital audio signal processing to the problem of denoising historical music recordings [Berger, Coifman, Goldberg 1994]. In particular, we have achieved some success with extracting pitch and timing structure from an almost completely corrupted wax-cylinder recording of Johannes Brahms playing one of his Hungarian dances, as well as removing noise in a recording from 1903, of Enrico Caruso performing an aria. The strategy adopted in these projects utilized modern mathematical methods, such as wavelet transforms and wavelet-packet analysis, as well as adapted local trigonometric transforms [Auscher, Weiss, Wickerhauser 1992]. Our current work aims at developing techniques for the detection of higher-level musical structure in digital musical audio. We have reported preliminary results at the 1995 ICMC [Berger, Coifman, Popovic 1995]. In this paper we summarize our research and propose possible new directions.

Unified Representation of Musical Structure

Our efforts aim at the detection of various levels of structure in digitized musical sound. Current developments in the intersecting areas of signal processing and mathematical harmonic analysis have provided us with new and efficient ways of extracting time-frequency information from digitized audio signals. We believe that further development of our analytical methods will allow us to take meaningful steps towards a unified representation of musical structure. This representation is conceived as a model for a wide range of structural formations: on one hand, we wish to model micro-detail, contained, for ex-

ample, in the audio recording or real-time capture of a musical performance; on the other hand, we want to model larger structural units, such as those rendered in a conventionally notated score. These include individual notes, chords, polyphonic lines, etc. Applications of such a stratified, yet synchronous representation might range from symbolic-musical interfacing to digital audio in the context of sound editing, to real-time pitch- and timbre-tracking for interactive performance. For example, a sound editor driven by our envisioned representation would provide access to the audio material on the signal-level, that is, by sample frame; but also on a musically much higher level, namely, by musical segment such as measure, beat, phrase etc. (This has been proposed in [Chafe, Mont-Reynaud, Rush 1982] and [Foster, Schloss, Rockmore 1982].) In the performance domain, the capture of musical audio as signal and, simultaneously, as musical structure would expand possibilities for interaction in performance, in particular if we succeed to achieve at least partial satisfactory solutions to some classically-notorious problems, such as pitch- and timbre-tracking in polyphonic music.

Transcription and Generalized Musical Notes

An indispensable step towards the identification of higher-level musical structures in musical audio consists in preparing the raw signal by means of a detailed, yet compact *analytical transcription*. Currently, our primary technique consists in transcribing a digitized musical audio signal by means of decomposition into time-frequency atoms, i.e., quasi-notes which are sufficiently localized in both time and frequency. This is comparable to traditional manual music transcription as practiced, for instance,

while taking dictation for the purpose of training aural skills. Here, what is heard is to be transcribed using conventional notational symbols or notes. Unfortunately, the pitch-time concept provided by traditional music theory is idealized beyond being of any use in situations which require an efficient, yet detailed *automatic transcription* of musical structure from musical sound. The transcription of musical sound by means of standard musical notational devices presupposes an anachronistic Pythagorean model of musical structure, with each tone implemented by means of an ideal single-frequency oscillator, suspended in the grid of an illusory discrete time-frequency space. This paradigm allows for immediate transcription by means of noteheads on staff lines. The reduction of musical tones to notational notes may have pedagogical merit. It is a little "too efficient" for our purposes, however, in that it abandons any and all connections to the physical-acoustical realities of tones making up the sonorous reality of a musical piece.

Past research has revealed the fair amount of complex structure accompanying even the most elemental of musical events, such as single piano tones. We are currently searching for a transcription by means of *analytical notes* which themselves possess a fair amount of internal structure, yet allow for as efficient a transcription as conventional musical notes do. The basic idea here is to 'shrink-wrap' the typical 'spectral mess' making up a musical tone, without really hiding it. We have found an entry point to such a transcription in the employment of signal building blocks, which are satisfactorily localized both in time and in frequency. These are, specifically, wavelets, wavelet packets, and local trigonometric bases. We generalize the musical terms *transcription* and *note* to suit our intended purposes. Our transcription departs from mathematical harmonic analysis and includes recent advances in wavelet applications. The classical Fourier transform, performed on the time side of a signal expands that signal into sine and cosine basis functions. In audio, the result of the transform may be presented in the form of a sonogram, which in turn may be interpreted as a protocol of the spectral behavior of a signal over time. To achieve localization in time, however, a long signal has to be windowed, that is, segmented into chunks which are manageable by a short-time discrete Fourier transform. Strongly localized details such as fast transients, chirp-like frequency modulations, etc. are "smeared" across the full spectrum within each analyzed segment and thus become inaccessible to further analysis. Accordingly, the precision of localization in time and frequency represents a compromise. The wavelet transform expands a signal also in terms of simple basis functions. These functions possess an important additional parameter, namely scale. They are

a priori well-localized in time and frequency and a wide variety of such localized functions may be derived by simple scaling of a prototype or "mother wavelet." The wavelet transform uses small-scale basis functions to model fine localized detail of the analyzed signal, whereas dilated versions are matched to large-scale features. The basis functions are called wavelets because they have compact support over a finite, relatively short interval (as opposed to sines and cosines, which are infinite), all the while preserving orthogonality.

Transcription and the Best-basis Paradigm

Wavelet-packet analysis expands a signal into a redundant set of basic waveforms by applying recursively the wavelet transform to dyadic partitions of the signal [Coifman & Wickerhauser 1993]. The resulting space of scaled basis functions may be searched for one specific linear combination of basis functions, basis for short, which may be called optimal with respect to some user-defined criterion. Using this best-basis approach, a signal may be analyzed in terms of bases which minimize the entropy of the transcription relatively to the signal. Such bases have been useful in transcribing those components of a signal which are deemed structural or coherent. For example, signal reconstruction from such bases may result in significant noise reduction. We have found that the components transcribed as coherent via best-basis extraction from a wavelet-packet analysis expose local detail of interest (from the perspective of music analysis) such as the attack behavior of musical tones. In addition, features which are more continuous, such as the relatively slow spectral behavior of sustained tones, are transcribed efficiently as well. A synthetic example may illustrate this. In Figure 1.a-c, an artificial signal consists of a quadratic chirp up to the Nyquist frequency, 512 samples long, followed by 512 samples of a pure sine wave,

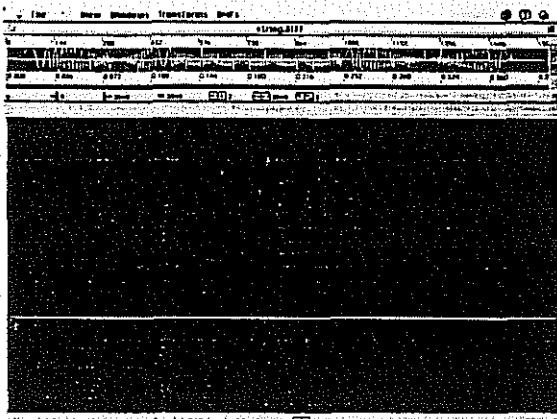


Figure 1.a

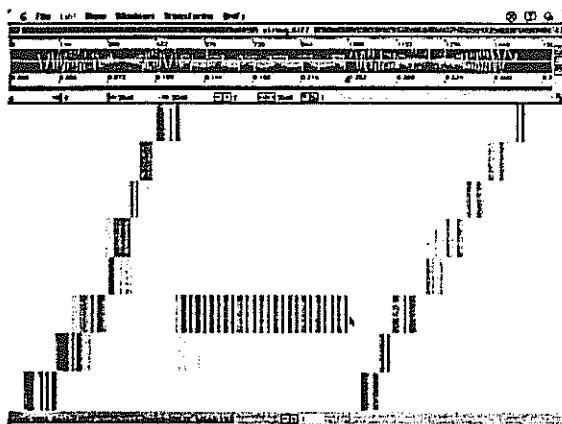


Figure 1.b

followed by a linear chirp. In Figure 1.a, a 2048 samples wide window is placed over the signal (zero-padded at end) and the DFT is taken. The pure sine wave component is localized well in frequency, but not in time. The starting and ending chirps smear over the entire spectrum and cannot be seen. In Figure 1.b, the same signal is transcribed by extracting the entropy-minimizing best-basis from a 3-level wavelet-packet decomposition. (Due to the small depth of the decomposition the frequency localization is not good here; this rough approximation, however, can be executed in many-times better than real time, even on a modest PC and may be useful for fast zooming-in onto interesting details.) Finally, Figure 1.c shows the same signal once again, transcribed via a 9-level adapted local trigonometric transform (ALTT, see below), performed directly on the time series.

As in denoising, we favor currently a wavelet-packet relative, namely the adapted local trigonometric transform, ALTT, performed on the signal spectrum, which yields libraries of time-frequency localized sines or cosines. As in classical sliding-window Fourier analysis, a long signal is segmented first. To combine good frequency localization with an adequate time localization, each segment is moved into the frequency domain via a discrete sine transform [Yip & Rao 1990]. A tree of smoothly cut and folded windows is constructed on the spectrum and the entropy-minimizing best basis is extracted. In the denoising procedure, a segment is then reconstructed from the head of a sorted queue of coefficients of each basis node, while separating the tail as noise. We do not reconstruct but simply interpret the coefficients as notes, in a generalized sense. Each box plotted in the time-frequency plane represents one such note. Forcing the analogy one step further, particular configurations might be called *generalized neumes*. Figure 2 depicts a moment from Debussy's piano prelude ... *La file aux cheveux de lin*, digitized from a commercial LP record and transcribed by an ALTT on the frequency side of the signal. Besides individual pitches and their timing, the transcription reveals interesting detail, in

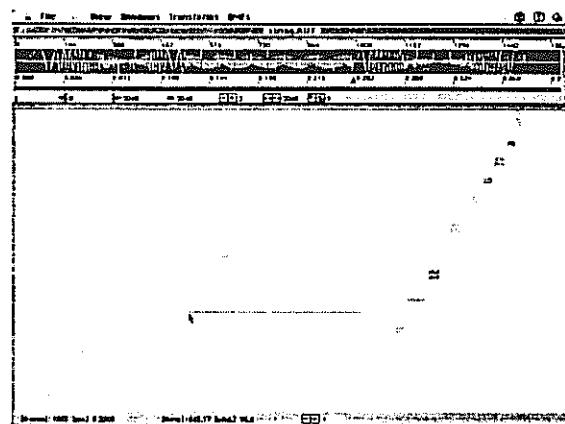


Figure 1.c

particular about the dynamic and temporal disposition of the chord on the downbeat of m.5 (the lowest note of the chord, marked by an arrow, is apparently played very softly; the isolated pseudo-notes at the bottom of the graph are pops and other noise from the recording).



Figure 2

Conclusion

The objects obtained from a wavelet-based signal decomposition, populating the time-frequency plane, are indeed musical notes, albeit in a generalized sense. They are parametrized building blocks which are localized in time and frequency, with the musical parameters frequency, onset time, duration, and amplitude (in the tf-plots above each box represents such a note, its parameters, and the region of uncertainty where it is presumed localized). Wavelet packets transcribe a signal simultaneously on many different structural levels. For the musician, this means to "have the cake and eat it too". Depending on the ana-

lytical intention, bases may be extracted which emphasize linear continuity of frequency tracks across time, allowing for note tracking and subsequent pitch inference. Other bases may be constructed from the same library tree to emphasize rapidly changing structures, even some specially adapted to chirp components (chirp transform). And, finally, bases may be constructed to maximize the power of discrimination between signals belonging to different classes, in terms of statistical linear discriminant analysis (see below). Our main concern at this point is to learn to interpret the multitude of perspectives provided by our tools and to unify them into a compact, yet detailed transcription, which is open for interpretation on a fairly low level, close to the signal that is, as well as on a level closer to that containing symbolic-syntactic musical information.

Future Steps

At this point we do not "clean up" our analyses, although the clarity of Figure 2 suggests that this could be done easily. Instead, we are seeking to preserve as much detail as possible, with future applications in mind. We believe that our transcription contains a lot of information which is structural and which waits to be further interpreted in terms of the tasks we are shooting for. It is important to keep in mind that the plots we are currently obtaining only resemble traditional sonograms. The scattering of coefficients around points of transient activity is characteristic, it is a pattern. We are currently testing the construction of bases which maximizes some statistically measurable distance between signals classified as belonging to distinct classes. This approach is proposed in [Coifman & Saito 1994] and [Saito 1994]. Although the technique is the same, the intention behind the transcription and basis construction is different: not the most efficient transcription of a signal is sought but, rather, one which is maximally different from signals declared as belonging to different classes. In other words, the transcription is performed so as to maximize those features which are presumed characteristic of one class but not another and best-basis selection is performed so as to maximize the distance from one class to another. Our first test results, obtained as of this writing, suggest that this technique might be developed further to tackle problems of the complexity posed by tasks such as polyphonic pitch-tracking and timbre discrimination.

References

- C. Chafe, B. Mont-Reynaud, L. Rush, "Toward an Intelligent Editor of Digital Audio: Recognition of Musical Constructs," *Computer Music Journal*, Vol. 6, No. 1, Spring 1982.
- S. Foster, W. Andrew Schloss, A. Joseph Rockmore, "Toward an Intelligent Editor of Digital Audio: Signal Processing Methods," *Computer Music Journal*, Vol. 6, No. 1, Spring 1982.
- K. Rao, P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic Press, Boston, 1990.
- R. Coifman, M. V. Wickerhauser: "Entropy-Based Algorithms for Best Basis Selection", *IEEE Trans. Inf. Theory*, Vol. 38, 1992 March, pp. 713-718.
- P. Auscher, G. Weiss, M. V. Wickerhauser: "Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets", in *Wavelets: A Tutorial in Theory and Applications*, C. K. Chui (ed.), Academic Press, San Diego, 1992, pp. 237-256.
- R. R. Coifman, V. M. Wickerhauser, "Wavelets and Adapted Waveform Analysis. A Toolkit for Signal Processing and Numerical Analysis", *Proceedings of Symposia in Applied Mathematics*, Vol. 47, 1993, pp. 119-153.
- J. Berger, R. Coifman, M. Goldberg: "A Method of Removing Noise from Old Recordings", *Proceedings of the 1994 International Computer Music Conference*.
- J. Berger, R. Coifman, M. Goldberg: "Removing Noise from Music Using Local Trigonometric Bases and Wavelet Packets", *J. Audio Eng. Soc.*, Vol. 42, No. 10, 1994 October, pp. 808-818.
- R. R. Coifman, N. Saito: "Constructions of local orthonormal bases for classification and regression", *C. R. Acad. Sci. Paris*, t. 319, Série I, 1994, pp. 191-196.

The Centro Tempo Reale

Nicola Bernardini
Alvise Vidolin

Centro Tempo Reale
Centro di Ricerca, Produzione e Didattica Musicale

Abstract

After a fairly long period of silence, since the end of 1993 the Centro Tempo Reale, a computer music center located in Firenze, Italy, and directed by composer Luciano Berio, has entered a new period of activity and production opening to many new works by prominent composers.

1. The Centro Tempo Reale: a presentation

The Centro Tempo Reale is a center dedicated to research, production and pedagogy in the field of computer music which was first created in 1988. In 1993 the Centro Tempo Reale has taken upon a new start upgrading most of its technology and producing several works of international importance.

Between 1993 and 1995, the following works will have been created and premiered:

- *Orfeo cantando...tolse* by Adriano Guarnieri, opera for two sopranos, electric guitars, flute, amplified choir, small ensemble and live-electronics (1994)
- *Noms des Airs*, boîte à musique by Salvatore Sciarrino (A. Vidolin et al.)
- *Quare Tristis* by Adriano Guarnieri, text by Giovanni Raboni, for solos, female choir, two instrumental groups, percussions and live-electronics
- ...1995...2995...3695..., *comædia harmonica* by Marco Stroppa for actress, a double-bass, computer music and live-electronics
- *Zaïde* by Wolfgang Amadeus Mozart/Luciano Berio, opera in two acts

Besides the production activity, the Centro Tempo Reale is also pursuing the restoring of old analog tapes of masterworks of electronic music, among which:

1. *Visage*

2. *THEMA (Omaggio a Joyce)*

3. *A-Ronne*

all by Luciano Berio.

Furthermore, the Centro Tempo Reale organizes every year a cycle of seminars devoted to both musical and technological topics (lecturers of the last two years have been: Andrew Gerszo, Gabrio Taglietti, Alvise Vidolin, Johan Sundberg, Stephen MacAdams, Angelo Orcalli and Jonathan Impett).

Other than touring abroad with the works produced in-house, the Centro Tempo Reale organizes also concerts and musical events in Firenze to create a wider public devoted to contemporary and electronic music in town.

The Centro Tempo Reale is funded by the Municipality of Firenze (*Comune di Firenze*) and the Region of Tuscany (*Regione Toscana*).

2 References

Vidolin, A. & Bernardini, N. *Noms des airs di Salvatore Sciarrino: una musica algoritmica di trasformazione sonora*, in Atti del XI Colloquio d'Informatica Musicale Bologna (forthcoming)

Centro Tempo Reale
Centro di Ricerca, Produzione e Didattica Musicale
Villa Strozzi
Via Pisana 77
50143 Firenze
Tel. ++39(0)55/717270
FAX ++39(0)55/717712
e-mail: nicb@nexttis.unipd.it, vidolin@dei.unipd.it

classical concept of the word ‘font’

‘Font’ is a term that has been used in the field of typography for centuries. It refers to a specific typeface or font family, which is a collection of characters with a common design. The term ‘font’ is derived from the Latin word ‘fons’, which means ‘source’ or ‘spring’. This is because the first fonts were created by hand, using a metal type foundry to cast individual characters from molten metal. The term ‘font’ is also used to refer to a specific style or character set within a font family, such as a bold or italicized version.

The concept of a font as a collection of characters with a common design is fundamental to the way we think about typography today. It is a key element of the design process, and it is used to create consistent and harmonious designs across different media.

However, the concept of a font has changed over time, and it is no longer limited to the physical world of metal type and ink.

In the early days of typography, the term ‘font’ was used to describe a specific typeface or font family, such as a serif or sans-serif font. Over time, the concept of a font has evolved to include not only the characters themselves, but also the overall design and layout of the text, such as the spacing and alignment.

Today, the term ‘font’ is often used to refer to a specific style or character set within a font family, such as a bold or italicized version. This is because the concept of a font has changed over time, and it is no longer limited to the physical world of metal type and ink.

The concept of a font as a collection of characters with a common design is fundamental to the way we think about typography today. It is a key element of the design process, and it is used to create consistent and harmonious designs across different media.

However, the concept of a font has changed over time, and it is no longer limited to the physical world of metal type and ink.

In the early days of typography, the term ‘font’ was used to describe a specific typeface or font family, such as a serif or sans-serif font. Over time, the concept of a font has evolved to include not only the characters themselves, but also the overall design and layout of the text, such as the spacing and alignment.

Today, the term ‘font’ is often used to refer to a specific style or character set within a font family, such as a bold or italicized version. This is because the concept of a font has changed over time, and it is no longer limited to the physical world of metal type and ink.

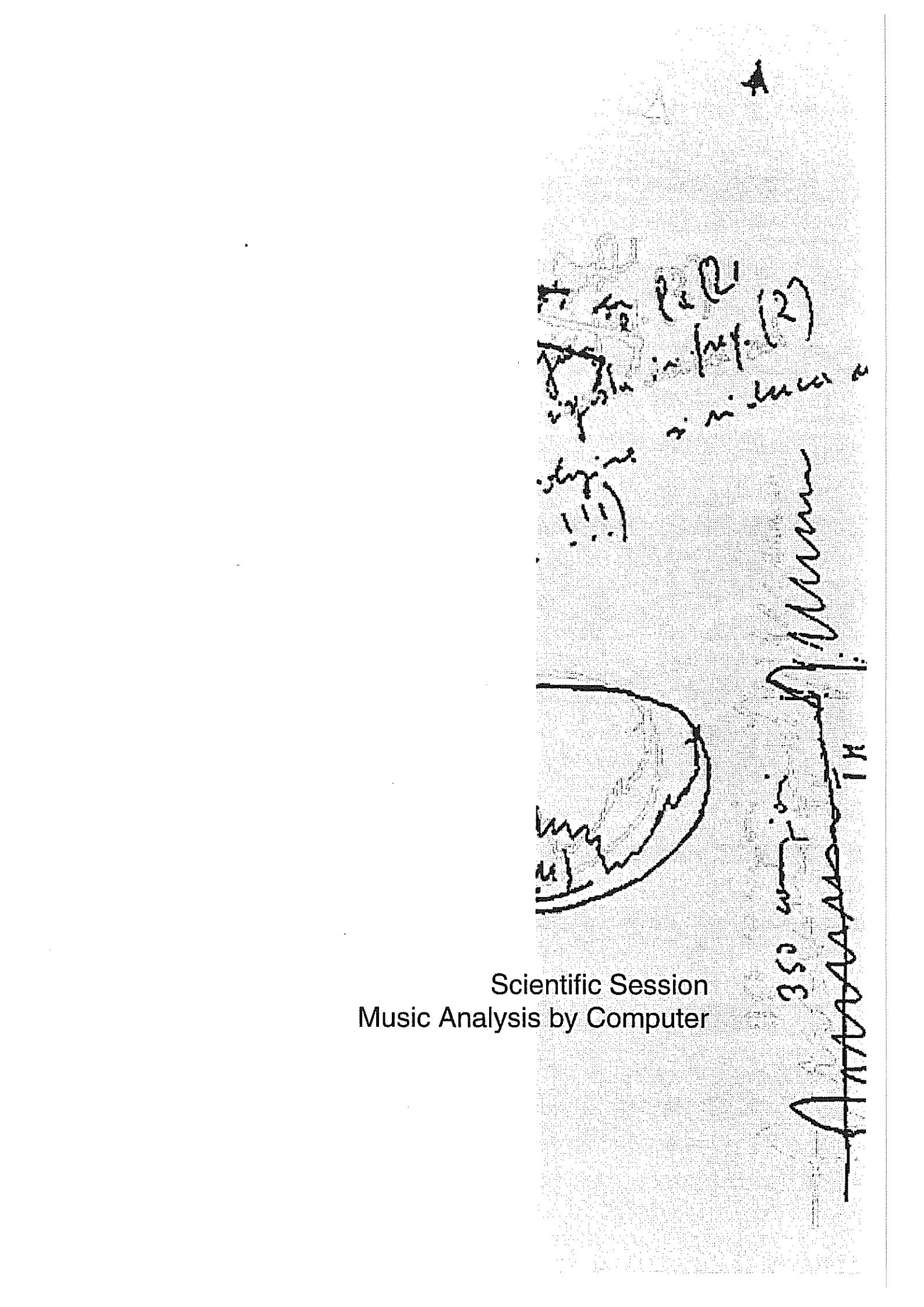
The concept of a font as a collection of characters with a common design is fundamental to the way we think about typography today. It is a key element of the design process, and it is used to create consistent and harmonious designs across different media.

However, the concept of a font has changed over time, and it is no longer limited to the physical world of metal type and ink.

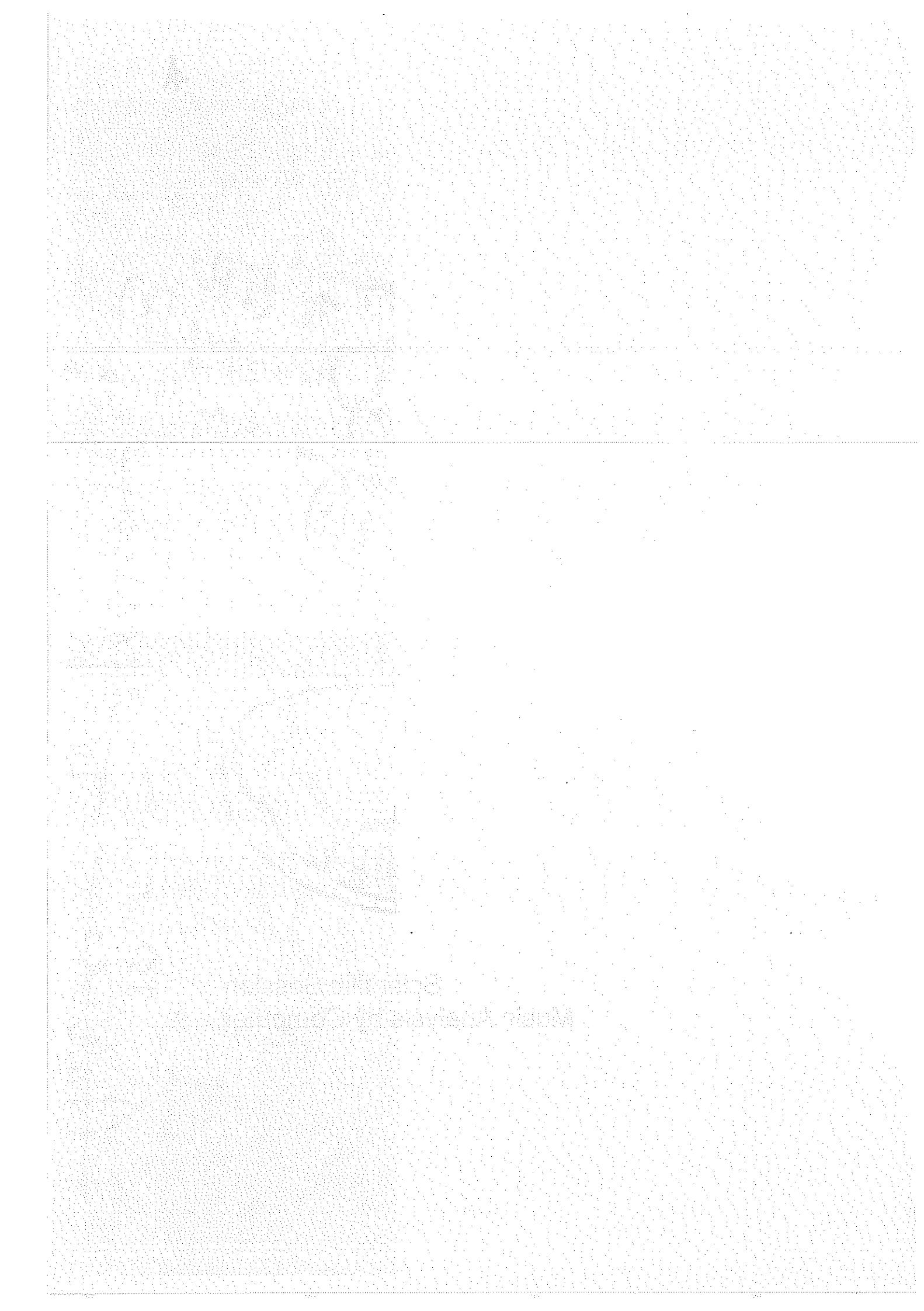
In the early days of typography, the term ‘font’ was used to describe a specific typeface or font family, such as a serif or sans-serif font. Over time, the concept of a font has evolved to include not only the characters themselves, but also the overall design and layout of the text, such as the spacing and alignment.

Today, the term ‘font’ is often used to refer to a specific style or character set within a font family, such as a bold or italicized version. This is because the concept of a font has changed over time, and it is no longer limited to the physical world of metal type and ink.

The concept of a font as a collection of characters with a common design is fundamental to the way we think about typography today. It is a key element of the design process, and it is used to create consistent and harmonious designs across different media.



Scientific Session
Music Analysis by Computer



Minimal Description-Length and Musical Analysis

Mauricio Annunziata, Pasquale Caianiello, Maria Cristina De Amicis

Laboratorio di Musica e Sonologia
Dipartimento di Matematica
Università dell'Aquila, Italy

e-mail: lms@vxscqaq.aquila.infn.it

Abstract

We describe the application of minimal description-length to musical analysis in the spirit of linguistic analysis. We present an experimentation on the analysis of rhythmic structure. We conclude that minimal description-length is cognitively relevant to music semiology.

1 Introduction

Semiology conceives of a music composition as a continuum of sound elements with distribution rules that determine their relative integration and exclusion relations. The ultimate result of an analysis is the reconstruction of the sound piece in term of sound units found with context dependent distribution features comparison and identifications. By and large, segmentation is the main analytical tool. A typical analysis is based on principles that depend on the nature of the piece, on the author's taste and dominant cultural categories.

This paper reports on a first attempt to provide an automatic tool to aid analysis with the application of the *minimal description-length* principle, whose relevance to cognition has already found many experimental proves in several scientific disciplines [1]. In agreement with musical semiology, the theory of inductive inference as well as other disciplines studying the natural development and acquisition of knowledge, recognizes analysis as the process of representing a phenomenon into a model. The minimal description-length principle, an information theoretic formalization of the *Occam razor* which is fundamental for epistemology, claims that the model should be as simple as possible. In this way, it is possible to capture difficult concepts as *learning* and *understanding* as processes of refinement of the

interpretation model, towards higher description efficiency.

The ingredients needed to turn this abstract ideas into a working computational system are three: a model representation, a complexity function for comparing different models, and a search strategy. We have obtained a further simplification by focusing on the rhythmic and melodic events separately. In the following section we provide a brief description of our system, its analytical postulating assumptions, its model representation, and symbol evolution strategies. As a first step towards verifying the functioning principles and the capabilities of the system in musical analysis, we have considered simple codified musical forms because they show an evident formal, melodic, and rhythmic structure: we report on the system's analytical behaviour on minuetto, marcia polacca, musetta, mazurka, valzer, tango. After the system's analysis, we can perform a double verification, cognitive and perceptive, on its results. We can either look carefully into the model constructed, a classical semiological analysis, and check for its cognitive relevance, or we can listen to a resynthesis that the system can perform with the model constructed, and check for its perceptive relevance.

Following linguistics, as music semiologists do [2], we consider a musical composition as a whole reality that needs to be cut in parts, parsed, to be understood. Computer science conceives of parsing as the processes of finding structural relations in a

sequence of terminal symbols, the whole reality, according to an a priori model, a grammar or any other equivalent computational model. This process calls for the necessity of a starting set of terminal symbols, *a priori observables* and a form model, in terms of which to represent the composition as a sequence. Our analysis procedure will select the form model that best describes the composition among possible models.

This general procedure can be rightly applied making different arbitrary choices on which observables to use in order to represent the composition and which evaluation function to evaluate the analysis. For the sake of this paper we concentrate on rhythmic observables and use minimal coding size as model evaluation function. Different choices of observables are suggested in the conclusion and will be adopted in future work.

2 The system AlephBeta

Our following analysis examples have been obtained with the *adaptive parser* embedded in the system AlephBeta [3].

Conventional computational linguistics uses static parsers that rely on a given grammar that identifies a general form with which to analyse a sequence. A composition that does not fit the given form would not be parsed and will never be parsed: new form patterns and ideas cannot be introduced on the run, they need to be embedded in the system in a session before the analysis.

AlephBeta tries to overcome this problem by evolving the form structure that it uses to parse, during the parse process itself. For representing form models, AlephBeta uses a particular implementation of augmented finite state automata. It manipulates and constructs them with a set of functions inspired by the theory of codes [4], and evaluates them, relatively to the phenomenon it is called to analyze, by means of information theoretical functions that measure their size and their coding efficiency. Finally, it uses conventional heuristic search strategies, appropriately improved as far as is allowed by the data structures used.

To be more precise, if A is its dictionary of prior observables, it first adopts the most general language, the theory of computation *universal language* A^* [5], and then it specializes it by evolving the prior observable symbols to a new set X which is a local minimum of the evaluating

function K defined as $K(X) = \frac{H(X)}{|X|} + |X|$

where H is the entropy, X is the average size of the set and $|X|$ is the overall size of the set computed in terms of the prior observables.

The nature of the sequences of observable events, which comes from compositions supposed to be experienced in a definite time orientation, recommends a choice for symbol evolution criteria that turns out convenient to limit the search space to manageable size: AlephBeta considers prefix coding sets only, to avoid parsing ambiguities and backtracking.

3 Experimentation

The experimentation that we propose aims at discovering whether the analysis postulates that AlephBeta uses are sufficient to discovering established composition forms with no prior knowledge of the forms themselves. As a first step towards verifying its functioning principles and capabilities in musical analysis, we have considered musical forms that show an evident formal rhythmic structure. A few compositions where given to the system in terms of notes durations only as in the following table.

Note durations

o	o.
d	d.
j	j.
f	f.
F	F.
ff	ff.

The set of rhythm patterns with their relative occurrence probability that AlephBeta evolved after its computation is given in the following tables.

Minuetto

j j j j	0.437
j j j	0.250
d	0.125
j d j	0.094
j d d d	0.063
d d d	0.031

Marcia

j d j	0.320
j d d j d j	0.272
j d j j j	0.137
d j	0.090
j d j j	0.046
d j d j j	0.045
j d j j j	0.045
j j j j j	0.045

Polacca	
♩♩♩♩	0.250
♩♩♩♩♩	0.200
♩♩♩♩♩♩	0.150
♩♩♩♩♩♩♩	0.100
♩♩♩♩♩♩♩♩	0.100
♩♩♩♩♩♩♩♩♩	0.050
♩♩♩♩♩♩♩♩♩♩	0.050
♩♩♩♩♩♩♩♩♩♩	0.050
♩♩♩♩♩♩♩♩♩♩	0.050

Musetta	
♩♩♩♩	0.285
♩♩♩♩♩	0.214
♩♩♩♩♩♩	0.178
♩♩♩♩♩♩♩	0.107
♩♩♩♩♩♩♩♩	0.071
♩♩♩♩♩♩♩♩♩	0.035
♩♩♩♩♩♩♩♩♩♩	0.035
♩♩♩♩♩♩♩♩♩♩	0.035
♩♩♩♩♩♩♩♩♩♩	0.025
♩♩♩♩♩♩♩♩♩♩	0.015

Mazurka	
♩♩♩	0.200
♩♩♩♩♩	0.150
♩♩♩♩♩♩	0.150
♩♩♩♩♩♩♩	0.150
♩♩♩♩♩♩♩♩	0.150
♩♩♩♩♩♩♩♩♩	0.100
♩♩♩♩♩♩♩♩♩♩	0.100

Valzer	
d.	0.385
♩♩♩	0.217
♩♩♩♩	0.114
♩♩♩♩♩	0.103
♩♩♩♩♩♩	0.068
♩♩♩♩♩♩♩	0.034
♩♩♩♩♩♩♩♩	0.029
♩♩♩♩♩♩♩♩♩	0.023
♩♩♩♩♩♩♩♩♩♩	0.016
♩♩♩♩♩♩♩♩♩♩	0.011

Tango	
♩♩♩♩♩♩♩♩	0.197
♩♩♩♩♩♩♩♩♩	0.166
♩♩♩♩♩♩♩♩♩♩	0.125
♩♩♩♩♩♩♩♩♩♩	0.114
♩♩♩♩♩♩♩♩♩♩	0.083
♩♩♩♩♩♩♩♩♩♩	0.083
♩♩♩♩♩♩♩♩♩♩	0.075
♩♩♩♩♩♩♩♩♩♩	0.062
♩♩♩♩♩♩♩♩♩♩	0.030
♩♩♩♩♩♩♩♩♩♩	0.025
♩♩♩♩♩♩♩♩♩♩	0.020
♩♩♩♩♩♩♩♩♩♩	0.010
♩♩♩♩♩♩♩♩♩♩	0.010

Similar results have been obtained with an experimentation on with a different choice of a priori observables coding melodic interval sequences. The patterns obtained could be used as a priori observable for higher level structure discovery.

4 Conclusion and future work

If listen to a resynthesis obtained as a random sequences with the distribution observed of the patterns obtained with the analysis, we obtain composition which show similarities with the form analysed. The similarity is increased if higher level structure is considered. We can conclude that minimal description-length is cognitively relevant to music semiology.

AlephBeta is a general purpose system, if appropriately interfaced it can deal with any sequence of symbols and can construct symbolic representations. However, what is symbolic is completely arbitrary, it can be perception as well as cognition: even though the examples that we present only concern the symbolic level of music, we expect this preliminary results to guide the application of Alephbeta to the level of sound.

References

- [1] R.J. Solomonoff: "A Formal Theory of Inductive Inference " Part 1 and 2, *Information and Control* 7 (1964), 1-22, 224-264.
- J. Rissanen, *Stochastic Complexity in Statistical Inquiry*, World Scientific, Teaneck NJ (1989).
- AIII Workshop on the theory and application of Minimal-Length Encoding*, Working Notes Stanford University 1990
- [2] K. Ruwet: *Langage, musique, poésie* 1972 (it.transl. Einaudi 1983)
- J.J. Nattiez: "De la sémiologie à la semantique musicale", in *Musique en Jeu* vol. 17, 1975, 3-10
- [3] P. Caianiello: *Learning as the evolution of representation* Ph.D. thesis, New York University. C.S. Tech. Rep. 477 (1989);
P. Caianiello: "ALPHABETA: a system for implementing and testing learning algorithms" in *Proceedings GAA91*, AIII (1991).
P. Caianiello: "Progress of the system Alphabeta" in *Proceedings GAA93*, AIII (1993)
- [4] J.Berstel,D.Perrin: *Theory of Codes*, Academic Press NY(1985)
P. Caianiello E. Loretì: "Generation of finite codes" *Tech. Rep. 47 Univ. dell'Aquila* (1993)
- [5] M. Davis, E. Weyuker: *Computability, Complexity, and Languages*, Academic Press (1983).

A Computational Model for the Discovery of Parallel Melodic Passages

Emilios Cambouropoulos, Alan Smaill

University of Edinburgh

Faculty of Music and Department of Artificial Intelligence

emilios@music.ed.ac.uk, smail@ai.ed.ac.uk

Abstract

In this paper an attempt is made to formalise musical similarity. A computational model is devised that discovers all possible parallel melodic passages (according to some musical parameter) by-passing traditional 'well-formedness' demands and selects the most prominent patterns according to their length, frequency of occurrence and degree of overlapping. The theory and results presented here are part of a more general research project on musical similarity.

1 Introduction

Being able to understand and reason about an object of our world means being able to see it in relation to other similar objects and in opposition to other objects of the world. This way an object may be classified within a class represented abstractly in the mind.

In music such classes are established primarily through internal self-reference (music rarely has direct external denotational associations). Structural repetition and similarity are crucial devices in the organisation of a musical piece. Similarity - for different parameters and levels of abstraction - allows musical objects to be defined and interconnected in an intelligible way. Similarity enables a musical surface to be segmented into meaningful units (segmentation is actually one of the weakest points of most of the existing musical theories) and relationships to be established between such units.

Despite this, even the most elaborate contemporary musical theories neglect to tackle the problem of parallelism in a systematic way (e.g. no detail is given in the GTTM, Lerdhal & Jackendoff 1983). Theories that attempt to formalise musical similarity either restrict themselves to a very well circumscribed and rather limited area of musical knowledge - e.g. Ruwet's machine (Ruwet 1987), similarity relations in pitch-class set theory (Forte 1973) - or allow a fair amount of musical intuition to the analyst - e.g. traditional thematic analysis, Reti's thematic processes (Reti 1951), paradigmatic analysis (Nattiez 1982, 1987).

In this paper we will briefly explore the concept of similarity and will attempt to create a formal description of it. The proposed theoretical framework will form the basis of a computational model that can discover and label similar (parallel) melodic passages occurring in a given melodic surface. The material presented here is only part of on-going research.

2 Formalising Similarity

We will attempt to systematise musical similarity by the use of the logical principle of identity. Similarity is an obscure and much dreaded notion. What are the conditions and limits under which two entities may be considered similar? Especially the definition of limits is a difficult, if not impossible, task. (It is like trying to define how many hairs should be missing from a man's head before he may be considered bald!)

We will consider two musical objects as being *similar* if they have identical values for at least one property pertinent to their description. The more properties they have in common the stronger the similarity is. If the two objects have identical values for all the properties under consideration they will be referred to as being *identical*. The similarity relation is a. reflexive (A is similar to itself) and b. symmetric (if A is similar to B, then B is similar to A). It is *non-transitive* (if A is similar to B and B is similar to C, then A is not necessarily similar to C) (see Rahn, 1983:50).

Below is an attempt to formalise similarity into a hierarchy of kinds and degrees of similarity based solely on the principle of identity (figure 1).

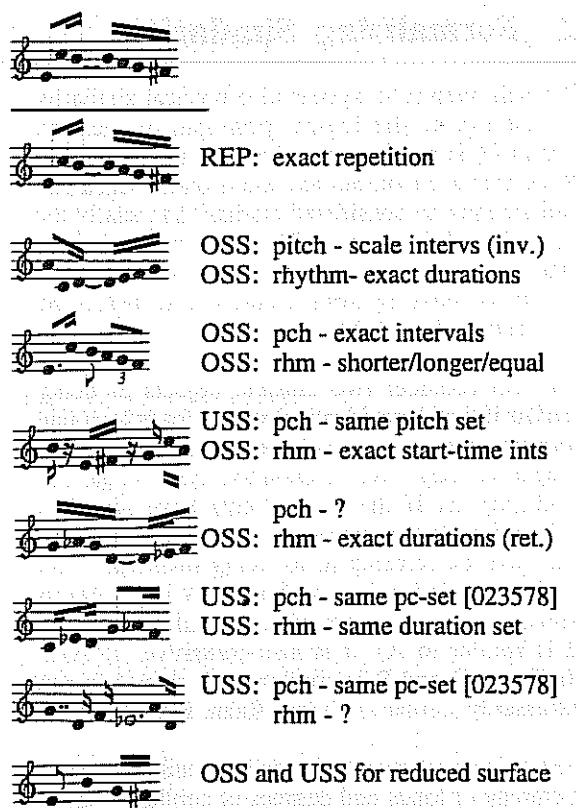
REPETITION This is the strongest form of similarity (actually identity, as defined above) where two musical groups are identical in *all* parametric levels (not start-times).

ORDERED-SET SIMILARITY (OSS) Two groups are similar when they are identical in at least one parametric level. The more parametric profiles the two groups are identical in, the stronger the similarity is. For example, a transposed retrograde-inversion of a pitch formation that has identical rhythm pattern as well, is a stronger OSS than two groups that are identical only in respect to rhythm.

UNORDERED-SET SIMILARITY (USS) Two groups are similar if they contain identical parametric values in at least one parametric level where grouping ignores ordering of elements. One example of this kind of similarity is pc-set theory.

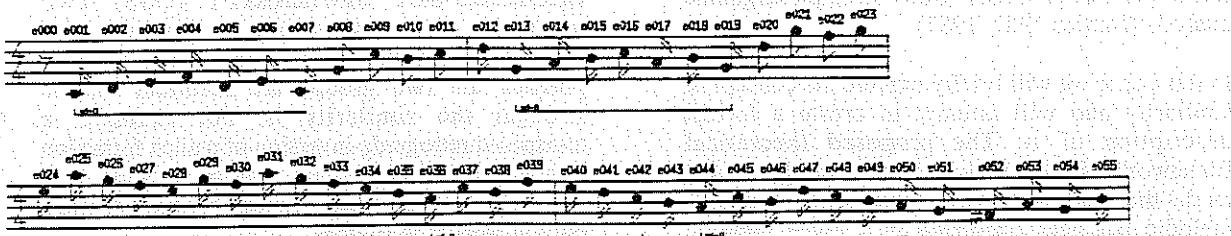
OSS is considered stronger than USS and within each kind of similarity there are many degrees. Two groups may be similar, making use of either or both kinds of similarities. For example, two groups may have an identical rhythmic pattern (OSS) and identical pc-set information (USS). In both OSS and USS we can also talk about weaker forms of similarity in terms of subsets and supersets.

figure 1: Some similarity relations between the first pattern to each of the following.



This kind of similarity analysis (parallelism) can be applied not only to foreground surfaces but to any reduced abstract level. Such reduced structures may be created by establishing preference rules for the

figure 2: One overlapping pattern-motive in the beginning of J.S.Bach's *Inventions à 2 voix*, No. 1



events that may be considered most prominent (statistical prominence, importance of position relating e.g. to strong-beats or melodic peaks or boundaries of groups etc.). This way acknowledgement is made of the fact that some events are structurally more significant than others and parallelism may thus penetrate deeper into musical structure.

A *multiple-viewpoint* approach to analysis (separate analyses for each structural and parametric level) is essential. Separate analyses are performed for the different parametric profiles of a melody (pitch, rhythm, dynamics etc) and for different levels of abstraction within each profile (e.g. for pitch intervals: exact intervals, scale-steps, contour etc.). Apart from the actual melody similarity analyses may be performed on reduced versions of it. The difficulty of such an approach is the interleaving of these different and often conflicting viewpoints into a single overall analysis. This issue will not be addressed in the current paper.

3 Overlapping of Patterns

Many contemporary theories - especially theories that have been influenced by linguistic theory - make hypotheses about the way a musical surface should be segmented that are too restricting and limiting. For example, the GTTM (Lerdahl&Jackendoff, 1983) assumes two kinds of rules the first of which are referred to as *well-formedness rules*. These rules allow grouping interpretations of a piece that comply with a strict tree-like hierachic non-overlapping structure (limited overlaps and elisions are considered to be exceptions to these rules).

In our view, such well-formedness rules should be considered simply as preference rules in a theory where the overlapping of patterns is the norm. Even in the classical tonal system it seems that the cases where such rules apply precisely is rather limited. Most music has a fair amount of ambiguous passages where not only the different parametric profiles conflict with each other making it impossible to find a well-formed description but even within a single profile a *non* well-formed description may be the most appropriate (figure 2).

Our cognitive skills attempt to impose a *well-formed* interpretation on a musical surface which is the *preferred* interpretation mainly for reasons of cognitive parsimony. However this process often fails leaving an unresolved ambiguity and uncertainty which is central to musical meaning and emotions. In music there are no absolute 'parsing' rules to which an analysis should comply. There are better or worse descriptions, more or less economic, closer or more remote to cognitive models, preferred or avoided within a certain context. In this sense, we consider closer to musical understanding theories that are *non-exclusive* i.e. 'theories which do not view new pieces as being *true* or *false*, but rather regard all representable musical surfaces as possible' (Conklin, 1991:2).

4 A Method for Exhaustive and Efficient Pattern-Matching

The similarity mechanism is employed in a bottom-up fashion i.e. starting from the smallest patterns and extending them to the maximum length. The well-formedness demands posed by a hierarchical structure of discrete levels with approximately equal length non-overlapping groups is bypassed. Overlapping of patterns is allowed.

For a given viewpoint of a melody (e.g. scale-step pitch intervals), the matching process starts with the smallest pattern length (2 elements) and then shifts stepwise through the entire array finding all identical patterns (original, retrograde, inversion and retrograde inversion) that are repeated at least once. Any interval that is not included in any matched pattern is removed from the initial array and a marker is placed in its position that doesn't allow matching to override it. This way the initial array is fragmented and shortened. As the matched patterns grow in size, the search space is reduced. When the last matching is found for the largest possible pattern, the matching process ends.

The matching mechanism is exhaustive i.e. it discovers all possible matchings and although it is computationally expensive, it is as efficient as possible (through the reduction of the initial search space). This procedure can become significantly faster if the user inserts *break* markers in the initial array for positions that are thought to be major division points in the melody (e.g. marked in the score by breath marks, large rests, slurs, fermatas and so on). It is also possible to define externally a range of pattern lengths for which the matching mechanism should be employed.

For hierarchically ordered viewpoints, i.e. for viewpoints that can receive structural descriptions that are necessarily a subset (or superset) of the descriptions produced by the same process on a

different viewpoint (e.g. exact interval - scale step interval viewpoints), the pattern matching process can be applied first to a more general viewpoint and the search may proceed within the patterns previously discovered. There is no reason to employ an exhaustive search to the more specific viewpoint. This again reduces significantly the search space and the computational time involved.

This whole similarity procedure is pursued for at least the three most significant ordered parametric properties of musical events i.e. pitch, duration (and start-time intervals) and intensities (timbre is currently not incorporated).

It is apparent that such a procedure for the discovery of parallel melodic segments will produce a very large number of possible patterns most of which would be considered by a human musician-analyst counter-intuitive and non-pertinent. How can the most prominent patterns be selected and the unimportant ones be filtered out? The next paragraph addresses this issue and proposes a possible solution.

5 Selection Mechanism

In an attempt to devise a procedure that can attach a prominence value on each of the previously discovered patterns we made a hypothesis in which the importance of a given pattern relies on the following three general principles:

- Prefer longer patterns
- Prefer most frequent patterns
- Avoid overlapping

Below is a function that calculates a numerical value for a single pattern according to the above principles:

$$f(\text{PL}, \text{F}, \text{DOL}) = \text{F}^a \cdot \text{PL}^b / 10^c \cdot \text{DOL}$$

where PL: pattern length i.e. number of elements in pattern

F: frequency of occurrence for one pattern

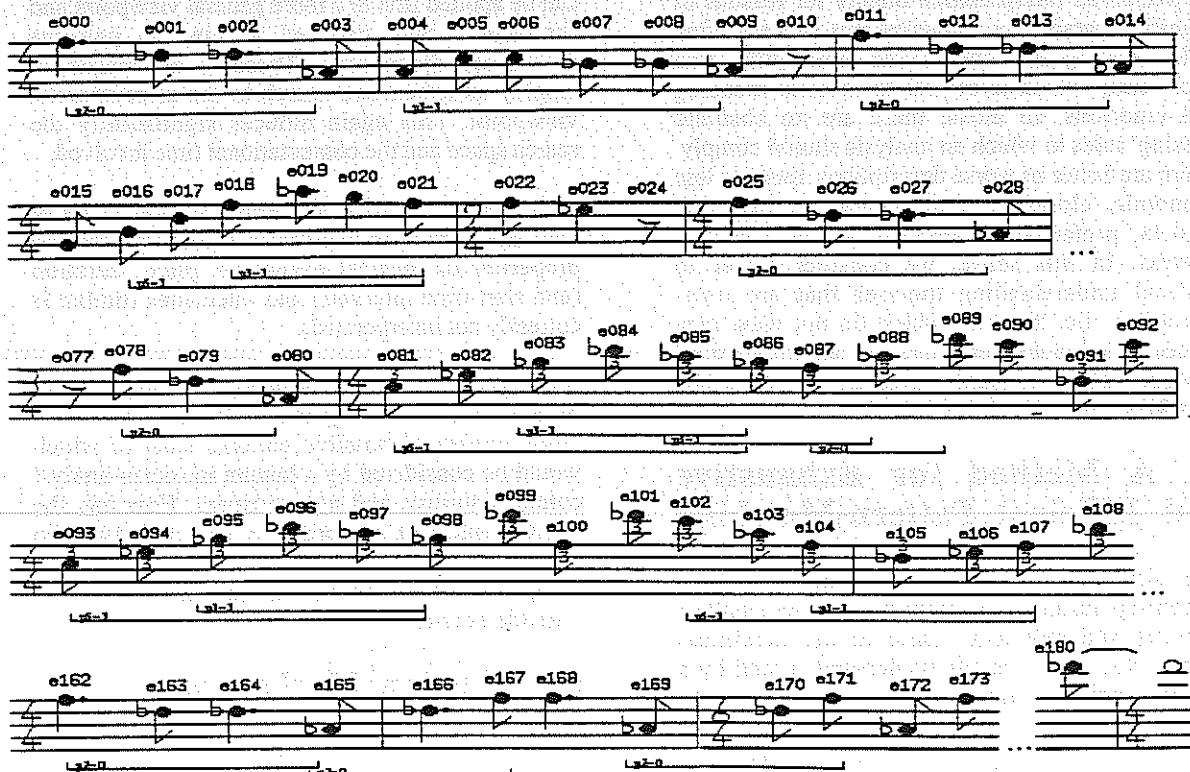
DOL: degree of overlapping

a, b, c: constants that give different weights to the importance of the above principles

For every pattern discovered by the matching process a value is calculated by the use of this function (the same constants should be used for all the patterns). The patterns that score the highest should be the most significant ones.

The system was tested, among other pieces, on *Niobe* - a solo piece for oboe - by B. Britten (from *Six Metamorphoses after Ovid*). The matching procedure was applied on a scale-step pitch interval profile of the monody and a total of 121 different

figure 3: Some of the patterns singled out by the selection mechanism for a scale-step interval profile of B. Britten's Niobe.



patterns were discovered. From these the selection module singled out only a few, some of which are depicted in figure 3. These pattern-motives may be

considered a rather plausible segmentation of *Niobe* which is not very far from a segmentation a music analyst would perform. This result is quite significant when one bears in mind that the system has produced such an analysis from a single profile without any additional knowledge such as bars, slurs, breath marks and so on.

The system analysis will be significantly improved when many similarity analyses are performed for multiple profiles and then combined to give an 'optimal' single overall multi-facet description (this interleaving stage is currently under investigation).

6 Conclusion

An analysis of a given melodic passage involves establishing a way of discovering and labelling similar musical segments. In this paper we have introduced a computational model that discovers such patterns for a given parametric profile of a melody. The matching process allows overlapping of patterns and then a selection method singles out the most prominent ones taking in account their length, frequency of occurrence and degree of overlapping. This model produces good results - although being at such a preliminary stage - which

will be significantly enhanced when multiple viewpoint analysis is pursued further.

References

- D. Conklin & I. H. Witten: *Prediction and Entropy of music*. Technical Report 91/457/41, University of Calgary, Dept. of Computer Science, Canada, 1991.
- A. Forte: *The structure of Atonal Music*. Yale University Press, New Haven, 1973.
- F. Lerdahl & R. Jackendoff: *A generative Theory of Tonal Music*, MIT Press, Cambridge (Ma), 1983.
- J. J. Nattiez: *Fondements d'une Sémiologie de la Musique*, Union Générale d' Editions, Paris, 1975.
- J. J. Nattiez: "Varese's 'Density 21.5': a study in semiological analysis", *Musical Analysis*, 1:3, 1982 pp. 243-340.
- J. Rahn: *A Theory for All Music*, University of Toronto Press, Toronto, 1983.
- R. Reti: *The thematic Processes in Music*, The Macmillan Company, New York, 1951.
- N. Ruwet: "Methods of Analysis in Musicology" *Music Analysis*, 6:1-2, 1987.

An Artificial Neural Net for Harmonizing Melodies

Dan Gang, Daniel Lehmann

Institute of Computer Science, Hebrew University
Jerusalem, Israel

Abstract

Whenever a musician needs to harmonize or analyze harmonization of a given melody, after establishing the key, he/she tries to extract, for each measure, some harmonic hint from the melody in the form of key notes. These key notes provide a sparse harmonic outline that will be completed to a full harmony using constraints derived from: the harmonic and the melodic contexts, the musical style and culture and the individual musical taste. Based on this analysis, we propose a Jordan's sequential neural net the structure of which reflects the way the human musician proceeds: it contains a sub-net that identifies key notes. The net is capable to learn simple harmonized melodies and generalizing what it has learned by harmonizing melodies it has never seen.

1 Introduction

Suppose you have heard today a hit on the radio, you can sing it with the singer and you almost know all the words.. You would like to play it to your friends at a party in the evening. You take the guitar and try to find the *right* chords, but the result is not convincing. It is really important to you, so you meet with your guitar teacher who can probably help you to find the chords. You sing the song and it takes only two minutes to your teacher to find the guitar accompaniment! "Tell me how you do it", you ask your teacher. But the teacher says: "I think I can not give you the algorithm or the exact rules for doing it, and moreover it depends on my mood, and it took me some years before I learned to do it well, so keep trying!". Harmonizing a melody is one of many musical activities that is difficult to explain or teach. A lot of people get frustrated when they are unable to find *sensible* chords for a melody. This paper will try to explain how musicians solve the problem of harmonizing a simple melody and will present a hierarchical neural network that effectively learns this task. We have tested the ability of an artificial neural network to learn the task of harmonizing very simple songs. The class of the songs is Western European, and the melodies and harmonies are at the level of a child after two years of playing the organ. There are two reasons for this choice of melodies. First, we want to understand the basic chord functions and how they are learned by beginners. Second, the aesthetic judgment on harmonizing simple melodies is probably less controversial.

2 The Problem

What gives an expert the ability to harmonize a melody so easily and so well? What makes a child after two years of playing the organ, able to cope with the harmonization problem? We do not claim we understand these processes fully, but we shall describe the steps a musician may use, and based on these steps, the architecture of the system that is capable of learning how to perform the task.

2.1 The Human Musician's Approach

Whenever a musician needs to analyze the harmonization of a given melody or to create an harmonization for a melody, he/she examines the notes of the melody in order to establish the key or the tonal centers. Then, using the knowledge that the harmonic rhythm is generally of the order of magnitude of a measure, he/she considers the notes in every measure, in order to extract some harmonic hint: key notes that comprise a sparse harmonic outline which will be filled and completed using constraints derived from: the harmonic and the melodic contexts, the regularities of the chord progressions in the musical style and culture, and the individual musical taste. Not all experts agree on how one finds the key notes, or on their exact relation with the chord finally chosen.

2.2 The Computational Approach

A neural net for harmonizing simple songs has been built and tested. It is capable of learning harmonized melodies, using a back propagation algorithm, and generalizing what it has learned by

harmonizing melodies it has never seen. We avoid the need to locate the key of the song by considering only songs written in Do major. In section 2.1 we described how a musician tries to extract a harmonic hint for a measure, by notes in every measure. In our system, the task of finding key notes is delegated to a sub-net that is trained separately. These key notes are a function of the notes that appear in the measure only: no melodic context is taken into account. The full harmonization is produced by a Jordan's sequential net [Jor86], described in Section 3. This net learns the regularity of the chord progressions in the musical style from examples. The harmonic context is used directly in this net, together with the key notes, to create a full harmonization, i.e., the chords. The melodic context is used only indirectly by the net, through the key notes produced by the sub-net.

3 Description of the network

The architecture of the network is described in figure 1. The sub-net for key notes learns to extract key notes out of the melody. It learns from examples: harmonized melodies. To define the key notes for a measure, in our examples, we used the intersection between the notes appearing in the melody and the notes that build the chords for the measure. The sub-net is a 3-layer standard feed forward network trained by back propagation. The input layer represents the notes of one measure of the melody and the output layer represents the key notes with twelve units for the chromatic scale.

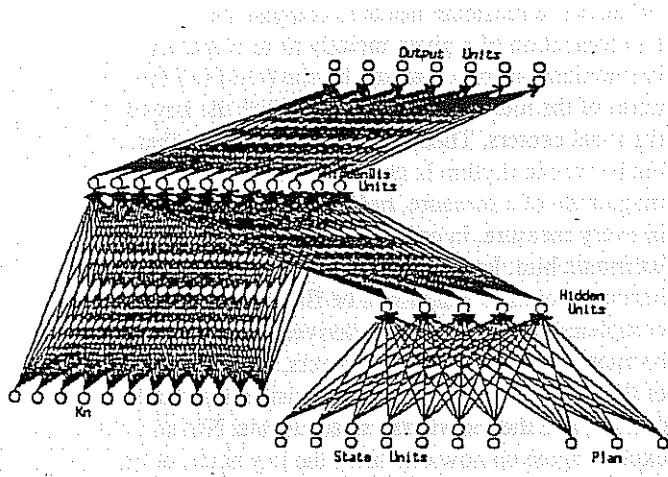


Figure 1: The net architecture

The 4-layer sequential net learns the sequence of chords, with consideration to the key notes that are propagated from the sub-net. The output units represent 14 chords - 7 for the Do Major scale's degrees, and 7 more for their Dominant 7 chords. The output layer is fed back in the state units of the input layer for the computation of the next chord: the value of a state unit at time t is the sum of its value at time $t-1$ multiplied by some decay parameter and the value of the corresponding output unit at time $t-1$.¹ The state units represent the context of the current chord sequence, and the output layer represents the prediction of the net for the next chord. The net also includes two internal hidden layers: Hidden and HiddenDis. The HiddenDis layer represents the chromatic scale: each of the twelve units represents a pitch. This layer is only partially connected to the output layer (for example: units Do, Mi, Sol are connected to the C unit in the output layer,² but unit Do is not connected to the G unit). The output layer of the key notes sub-net is fully connected to the HiddenDis layer, but some of the connections are fixed and some are learnable. For example: Do in the key notes sub-net output layer excites Do in HiddenDis layer, and therefore, indirectly, excites the C unit of the output layer; it also inhibits Si in HiddenDis and, indirectly, inhibits G in the output layer. In this way, we impose soft constraints on the way information is represented in the HiddenDis layer. The plan units in the input layer, are used to label different sets of notes' sequences. They have been used to harmonize in different styles, but this experiment will not be described here.

¹ These feedback connection do not appear in the figure of the net architecture.

² The bottom left side of the fig., that is denoted by kn, is describing only the output layer of the sub-net for key notes.

To implement the system architecture we used the PlaNet software [Miy91]. A full description of the network and the results obtained may be found in [Gan94].

4 Running the Net

4.1 The Learning Phase

For the learning phase, the net was given eighteen simple melodies with chords, and learned to reproduce these examples. We have tested the performance of the network with several different learning parameters (e.g., decay, number of hidden units). The net was able to learn the task relatively fast.³

4.2 The Generalization Phase

The network's generalization capability has been tested by giving it new melodies to harmonize. The results were compared to the original harmonization of the source. A trained musician has evaluated the resulting chords from the aesthetic and functional aspects. In the generalization process the net was tested by giving it four new melodies, and evaluating the resulting chords. These four new melodies were gradually growing in complexity, with regard to the density of the harmonic rhythm and the use of chord substitution. The resulting harmonization was found to be functionally appropriate by trained musicians. For some songs, the result obtained was almost identical with the harmonization proposed by the source, a music study book for children.⁴ For the other songs, the chords obtained were surprisingly aesthetic, flexible and different from the original. The chords resulting from the harmonization for the song "Supercalifragilisticexpialidocious" have been functionally appropriate,⁵ mainly because, the major functions in measures 4, 8, 12 and 16 are identical with the harmonization proposed by the source. The net, nevertheless, lacks consideration of the traditional musical hierarchy. This is revealed by the fact that the rhythm of harmonic changes is not symmetric. In measure 15 the G7 chord that gives us the sense of ending is missing, because the net lacks sense of the melody structure. This is to be expected, due to the rules of the game we have imposed on the net - it has information about the melody, only one measure ahead. The

chords resulting from the harmonization for the song "Soleado"⁶ are functionally appropriate, surprisingly aesthetic, flexible and different from the original. This song is the most complex in the set of the generalization learning examples, because of the density of the harmonic rhythm and the use of some chords substitutions.

5 Summary and Acknowledgments

We achieved our goal of building a neural net for harmonizing melodies, by dividing the task into two sub-tasks. After achieving the first sub-task (key notes), we try to achieve the second sub-task (harmonization), under the influence of the first. This neural network approach uses a simple representation and returns impressive results. It, nevertheless, lacks consideration of the traditional musical hierarchy. There is evidence that the harmonization may be extended to include individual and different musical tastes and styles. In [Todd89], Todd suggested the use of hierarchical sequential networks. This work is evidence that such networks, the structure of which reflects musical cognitive experience, are very promising. At the methodological level, probably the most important lesson of this work is, that one may use musical knowledge to direct the choice of the structure of the system.

Acknowledgments: the help of Naftali Wagner in preparing this paper is gratefully acknowledged. This work was partially supported by the Jean and Helene Alfassa fund for research in AI.

References

[Gan94] Dan Gang. A neural network that learns to harmonize melodies. Master's thesis, Institute of computer science, Hebrew University, 91904 Jerusalem, Israel, September 1994.

[Jor86] M.I. Jordan. Attractor dynamics and parallelism in a connectionist sequential machine. In Proceedings of The Eighth Annual Conference of the Cognitive Science Society, Hillsdale, N.J., 1986.

[Miy91] Yoshiro Miyata. A User's Guide o PlaNet Version 5.6 Computer Science Dept., University of Colorado, Boulder, 1991.

[Tod89] P.M. Todd. A connectionist approach to algorithmic composition. Computer Music Journal, 13:27-43, 1989.

³ For a decay parameter = 0.3, 5 hidden units and 18 songs it took 50 epochs to learn to reproduce the songs with a very small error.

⁴ See the song: "Lane in the Forest" in the Appendix. The upper line of the chords describes the chords resulting from the net.

⁵ See this song in the Appendix.

⁶ See the song: "Soleado" in the Appendix.

Appendix - The Net's Harmonization for New Melodies.

Lane in the Forest

C G⁷ C
C G⁷ C
F C C
F C C
F C Am
C C/G G⁷
C G⁷ C

Soleado

E⁷(Dm) C G C(Am) C Dmin G⁷
F G⁷ Em Am G⁷ C.
Dmin(G7) C G C Dmin G⁷
F G⁷ Em Am G⁷ C.

Supercalifragilisticexpialidocious

C G
C G
C G⁷ C
C G⁷ C
F F
F C C
C G⁷ C

Addresses: **Institute of Computer Science,**
Hebrew university - Givat Rame
Jerusalem 91904, Israel
email:{dang, lehmann}@cs.huji.ac.il
URL:<http://www.cs.huji.ac.il/~dang>
fax: 972-2-658-5439

FFT Analysis and Emotional Expression in Opera and Lied Singing

Eliezer Rapoport

Guest Scientist, Computer Science Institute, The Hebrew University
Givat - Ram, Jerusalem 91904, ISRAEL

September 11, 1995

1 Introduction

The Fast Fourier Transform (FFT) method is a very powerful tool in the analysis of acoustic and musical signals, leading to the decomposition of the signal into its component frequencies and intensities. This task can be routinely performed by means of modern and efficient algorithms on powerful computers or workstations, with very good resolution in time and frequency. An FFT spectrogram in the frequency - time representation gives, at every moment in time, the full spectral information (frequencies and intensities) contained in the signal, and effectively constitutes a full description of a musical event, from the full melodic phrase down to a single musical tone, whose development and evolution in time can be studied, deciphered, and interpreted. (limited only by the finite resolution). Musical tones, regardless of their human origin and musical purpose, are acoustic signals, and thus are physical entities. After being recorded by a microphone, stored on magnetic tape, compact disc, or computer memory, they can be subjected to physical measurements or mathematical analysis - in our case digital signal processing by FFT. The results: acoustic signals analysed by the FFT, are again, experimentally observable physical entities. The FFT spectrogram is a picture obtained by "freezing" a temporal event (an event streaming or flowing in time), and thus is a visual presentation of an acoustic audio signal and the spectral and temporal information contained in it. The human ear is also a frequency analyser of high sensitivity, operating on different principles. So we are concerned here with two independent acoustic signal processing systems, each with its merits and limitations. The human system is fully capable of extracting the emotional messages from the signals, whereas in the FFT system there is a code yet to be deciphered. In the present study, vocal spectrograms of a large number of opera arias and lieder, sung by famous artists, were studied by FFT analysis, with special attention to the evolution in time of single, individual vocal tones. Passage from the physical-scientific realm of acoustic signals and their decomposition into structures and components, to the realm of human feelings and emotions

expressed by these acoustic signals as musical tones in musical (melodic) phrases, is done in several steps: (1) First, still on the physical level: examination of the results of the FFT analysis, and identification of characteristic features, shapes, and structures occurring in the vocal tones, and the physical and other information that can be extracted from them. Since musical tones are time-evolving entities the structures observed are, obviously, temporal structures, the term to be used throughout this article. A tone with a definite characteristic structure, of musical and expressive significance, will be called a singing mode, or (simply) mode. (2) Seeking systematic trends and correlation with the emotional contents of the music and the text, by examining in what emotional context the various modes appear. (3) Seeking interpretation by use of general principles and music theory.

Results of our FFT analysis indeed revealed many characteristic temporal structures (modes) in individual vocal tones, having the unique characteristics that they consist of several initial stages, from the very onset (attack) until the sustained stage. Multi-stage structures are also observed in the decay stage. The initial stages are very significant, and indeed, form the basis for the classification and interpretation of these modes and their role in emotional expression. To date about forty such characteristic modes were identified and characterized, and others are still awaiting. These temporal structures were investigated in detail by examining which of them appeared in an aria or lied expressing a well-defined emotion or mental mood: excitement, tension, anxiety, rage, triumphant joy, sorrow, sadness, prayer, mocking, frivolity, happiness, love, etc., and a close correlation was found between the type of mode and the emotion expressed in the music and the text. Mutual exclusivity was strictly observed: modes of contrary type do not occur in the same musical phrase expressing a certain definite emotion, unless the two modes are intended to express contrary emotions, such as a sudden anxiety in a calm passage, or a release of tension in a passage full of excitement. Four mechanisms, superposed on the basic phonation (voice production) mech-

anism, were identified as a straightforward deduction from the mode shapes and structures. They are: (1) excitation of higher harmonic partials, especially those in the singing formant, (2) vibrato, (3) transition and pitch changes within the vocal tone, particularly a gradual pitch increase from the tone onset to the sustained stage. (4) Sforzando - a sharp and abrupt increase in pitch occurring right at the onset of the tone, lasting typically 50ms or less, then merging with vibrato. It appears in modes of highly excited nature and used as a special emphasis or accentuation of the tone. Artists, intentionally but unawaresly, use these mechanisms, and the interplay among them, in creating and shaping a large variety of nuances of temporal structures. The types of such nuances and the particular way in which they are used by the artist constitute the artist's style and personal interpretation in expressing the emotional contents of the music and the text. The various modes were classified into seven categories: (1) Neutral-Soft (NS), (2) Calm (C) with a subcategory Calm-Soft (CS), (3) Transitional-Multistage (TM) with subcategory Glide (G), (4) Intermediate (I), (5) Short (Sh), (6) Virtuoso (Vr), and (7) Excited (Ex), according to their structure and the corresponding type of emotion that they express. As a very close correspondence was indeed found between the expressed emotion and the singing mode, a hierarchical scheme of classification of the various modes was devised corresponding to a hierarchical classification of emotions, from the calm and subdued to the highly excited. This hierarchical scheme, and the seven mode categories, were originally deduced on the phenomenological level, by grouping together modes that have similar structures, and carry similar emotional messages, and classifying them accordingly, and according to the mechanisms that enter into operation, their timing, and the sequence order. The results, the categories of singing modes, and the correlation with emotional expression were recently described elsewhere (submitted to Journal of New Music Research). Here, a brief survey is presented. The mode types and categories, and the hierarchical scheme of classification are presented here as derived theoretically from postulates and general principles, and having experimental verification in singing modes that actually exist. Thus a very interesting link between emotional expression and artistic interpretation - which are esthetic, subjective values, and the physical-acoustical characteristics of the individual vocal tone on the micro level - scientific measurable entities, is revealed and established in the present study.

2 Examples of Temporal Structures (Singing Modes)

Only representative examples of the most important singing modes will be presented and explained here. These examples will help to illustrate the structural elements composing the modes and the operation of the

various mechanisms. In the figures: x-axis is time, full scale 1.9s, y-axis is frequency, full scale 6000Hz, each line is a harmonic partial. The acoustic waveform is shown on the top of each figure. Fig.1 shows four tones in Schubert "Ave Maria": the end of A- with the vibrato of the sustained stage (C mode, it's initial stages are shown in Fig.2), the ve, Ma (NS modes), whose beginnings and ends are marked by the vertical markers, and the beginning of -ri (C mode - in a legato connection to the Ma tone). As can be seen, the NS modes hardly contain any outstanding features, except for some jitter, and therefore not much expressiveness (to be further discussed in section 4.) Fig.2 shows a typ-

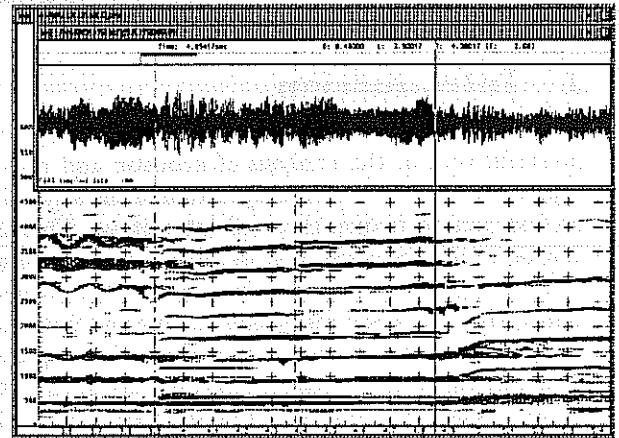


Figure 1: Spectrogram of NS modes from Schubert Ave Maria, sung by Felicity Lott, soprano.

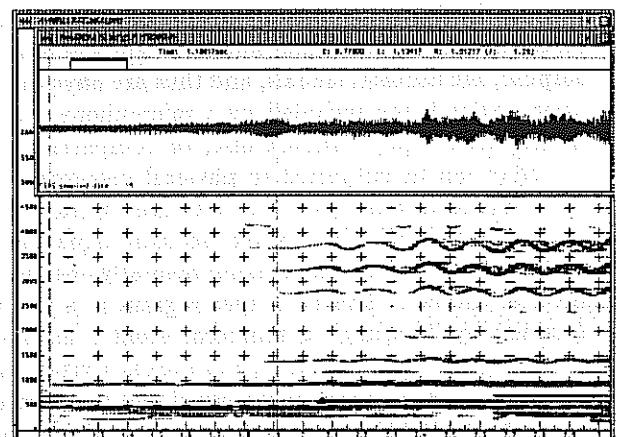


Figure 2: Spectrogram of a Calm mode, opening of Schubert Ave Maria, sung by Felicity Lott, soprano.

ical example of a Calm mode, the A-, opening of Ave Maria. The singer is Felicity Lott (IMP Classics PCD 898, rec. 1988). In Fig.2 the tone onset (marked by the left marker) consists of only the fundamental and the second harmonic (initial stage, lasting 0.6s), followed by a second stage (lasting 0.2s) in which higher harmonics are excited. In the third stage, low amplitude vibrato appears, lasting two vibrato periods, (about 0.35s), and finally in the fourth stage full vibrato is developed. The gradual evolution of a C mode is demonstrated. Fig.3. shows a typical example of a

mode belonging to the Transitional Multistage category, from *Un Ballo in Maschera* by Verdi, sung by Luciano Pavarotti. (Decca 410210-2, rec. 1985).

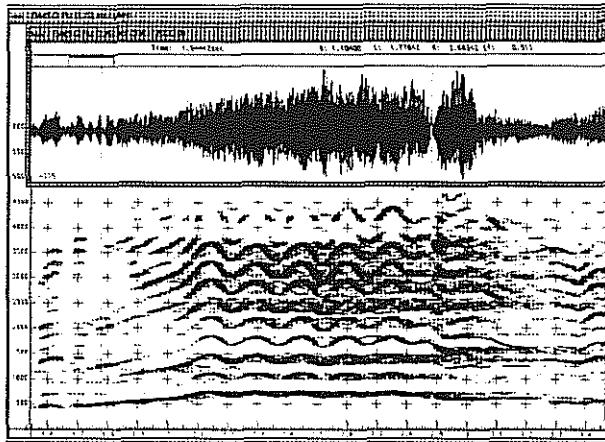


Figure 3: Spectrogram of typical example of a mode belonging to the Transitional Multistage category, from *Un Ballo in Maschera*, sung by Pavarotti.

It is seen that the initial pitch is lower than the pitch of the vibrato at the sustained stage, and a gradual pitch transition takes place during the initial stage (lasting 0.3s). Fig.4 shows a spectrogram of four tones of the excited mode sung by Pavarotti ("Figlio del Cielo..." from *Turandot* by Puccini - Decca 414 274-2, rec. 1972)

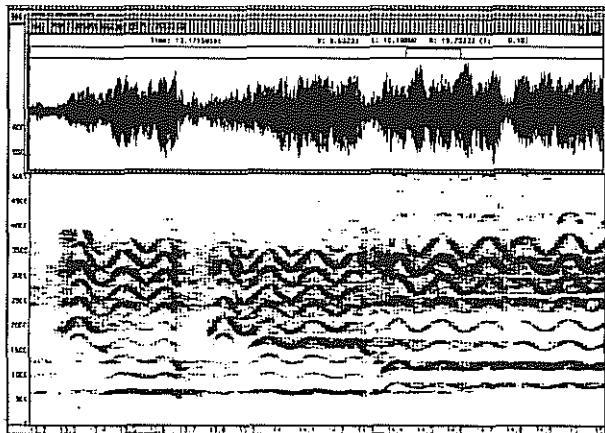


Figure 4: Spectrogram of four tones of the Excited mode, Pavarotti in Puccini's *Turandot*.

3 Notation

The following notation is introduced: p,P - denote onset of phonation at low intensity, and full intensity, respectively. (with respect to the intensity at the sustained stage).

e,E - excitation of the higher harmonics (singing formant), at low and full intensity, respectively.

v - onset of vibrato - low vibrato extent (amplitude), and lower frequency (than the sustained stage's values). V - onset of vibrato - the sustained stage value. s - onset of "small" vibrato (CS modes). t,T - onset of transition (small and large extent, respectively). The

transition is always in the "upward" direction, towards increasing frequency or pitch. Z, z sforzando, emphasis, strong and weak, respectively.(duration 50ms). j - jitter (random fluctuations in pitch). J - excessive jitter. (jitter occurs in any vocal tone). ; - This symbol, following one of the above denoted symbols denotes that the corresponding mechanism operated for sometime and then stopped. A symbol appearing to the right of the ";" symbol means that the corresponding mechanism starts after the previous one stopped. Processes that take place simultaneously are denoted by their respective symbols appearing successively. Thus, simultaneous onset of phonation (p or P), and excitation of the singing formant (e or E) are denoted: pe, or PE (depending on the intensity), whereas if they occur successively after a certain time lag, as a two-stage process, they will be separated by a hyphen: p-e or P-E. The symbol p-e-E denotes a three-step process: onset of phonation - singing formant excitation at low intensity - and finally excitation of the singing formant at full intensity. Similarly pev, p-e-v, PEV, pe-v-EV, denote the three mechanisms entering into operation in different ways, leading to different temporal structures of the vocal tone. Arrival at the state E automatically converts p into P. Once a mechanism starts operating, it continues to the end of the tone. Cessation of a mechanism is denoted by the symbol ";" following its symbol. Cessation of two mechanisms simultaneously is denoted by enclosing their two symbols in brackets, followed by the symbol ";".

4 Postulates and General Principles, and the Systematic Derivation of Hierarchical Scheme

With the help of the notation introduced in the previous section, one can construct the various singing modes, as the singer does, by using the five mechanisms (phonation + the other four). To this end, the following postulates and principles are employed:

1. The zero of the hierarchy: a tone devoid of any expression, or human emotion, is a mechanically-produced tone - a violin played by a motor-driven bow, a clarinet blown by a mechanical pump - leading to a non-varying monotonous tone. 2. Expression is built into the tone by introducing changes and complexity.

3. Musical tones are acoustic signals, and the emotional expressions are encoded in them as frequency and amplitude changes and modulations, periodic or non-periodic, thus shaping the tone's particular temporal and spectral structure. Every feature is meaningful, and nothing is haphazard.

4. All the acoustic information is contained in the FFT spectrogram down to the most subtle detail.

5. The multistage temporal structures of vocal tones is experimentally established. (Results of the present study).

6. The various temporal structures and shapes of the vocal tones are the result of the various ways and sequential order that the five mechanisms: phonation, vibrato, excitation of higher harmonics (singing formant), transition, and sforzando (elucidated in the present study) come into operation.

7. The universal rules of excitement and calmness in the simple form, relating abruptness to excitement, and gradual change to calmness or relaxation, form the basis for the classification of the singing modes, associating to them a degree of emotional excitement, and determination of the hierarchical scheme, in such a way that a tone structure of a more gradual character is associated with an emotional state of a more calm and subdued nature, whereas a tone of a more abrupt structure is associated with an emotion of a more excited nature.

8. Mutual Exclusivity - signals of contrary type should not occur in the same musical phrase expressing a certain definite emotion, unless, intentionally, the two signals are meant to express contrary emotions. Indeed this was confirmed experimentally.

9. The smallest number of mechanisms operating simultaneously will correspond to the most calm, relaxed or mild mode, whereas simultaneous operation of all the mechanisms will result in very excited modes.

10. Correspondence of the acoustic signals with the musical structure of the melodic line.

We generate the various singing modes, and classify them in hierarchical order, based on the number of mechanisms that operate, as follows:

(1) Operation of only one mechanism: phonation.

The modes p or P are simply the uninteresting, emotionless zero mode. The human voice always contains jitter, thus, modes pj or Pj appear as the initial stage in more complex modes (NS, CS, C).

(2) two mechanisms: phonation and vibrato.

Modes: pv or PV: These are mezzo-voce, sotto-voce and falsetto modes, characterized by their softness and mildness, in the absence of high harmonics, or with the latter suppressed to extremely low intensity.

(3) two mechanisms: phonation and excitation.

Modes: pe, PE, pe-E, and with jitter: pje, pje-E. The NS modes shown in Fig.1 are of Pje type.

We collect the modes from the aforementioned groups into the first, and softest category, and lowest in the hierarchy: Neutral-Soft (NS or N). Singers sing in these modes in arias and lieder of very subdued nature, expressing mostly sadness, such as Berlioz song "Au Cimetiere" from Les Nuits D'ete, sung by Anne Sofie von Otter, (DG 445823-2, rec. 1990-94) or Barbarina's aria "L'ho perduta" from Le Nozze di Figaro by Mozart, (sung by Yvonne Kenny, Decca 410150-2, rec. 1981).

(4) Three mechanisms: phonation, excitation, vibrato ("small"). The "small" vibrato is a very weak vibrato of amplitude (extent) of 20-30 cent, as compared with 60-70 cent for the Calm modes, and 100 cent for the Transitional Multistage and Excited modes.

Modes: p-e-s, pe-s, and pes appear alongside the NS modes, but are structurally different, and are classified as a subcategory Calm-Soft (CS), second in the hierarchy, and slightly more expressive than the NS category.

(5) Three mechanisms: phonation, excitation, vibrato (in succession).

Modes: p-e-v-EV, pe-v-EV, PE-v-V, P-E-V, Pv-e-E, pev-EV: these modes are grouped together in the second, the Calm (C) category. They are characterised by a gradual evolution in several stages from low phonation to a full operatic singing tone. The example of Fig.2 is a p-e-v-VE mode. Schubert lieder: Ave Maria, Nacht und Traume, Ganymed, and many others are sung in the C and the NS modes. These modes appear also in operatic arias.

(6) Four mechanisms: phonation, excitation, transition, and vibrato, operating in succession.

Modes: pet;EV, peT-E(T;)V, or peT;EV, and peT1;T2;EV, labelled T1, appear extensively in arias sung by tenors, and render timber embellishment, vividness, openness, and brilliance to the tones, used in expressing love in Verdi Un Ballo in Maschera (Pavarotti), and happiness in Verdi La Traviata (Pavarotti, Decca 410154, rec.1979, Alfredo Kraus, EMI CDS7 47059 8, rec. 1982). Fig.3 shows a peT-E(T;)-V mode. In a mode PE-T;V, the transition part is usually very steep. this is a stronger mode, labelled T2.

(7) The five mechanisms: phonation, sforzando, excitation, transition, and vibrato, operating in succession. mode T3: PEZ;-T-V, appears in phrases of greater excitement, and mostly in the beginning of a phrase of great excitement. The modes described in (6) and (7) are grouped together in the third, the Transition-Multistage (TM) category - higher in the hierarchy than the previous ones.

Glide modes (portamento) G1: PEVT;, and G2, PEVT;V are used mostly by soprano singers as a means of timber embellishment and as an artistic means of dramatic leap to climaxes, and are included as a subcategory of the TM category. Strong G modes sometimes start with sforzando: PEZ;VT, and appear in singing expressing great excitement.

(8) Four mechanisms operating simultaneously: phonation, sforzando, excitation, and vibrato.

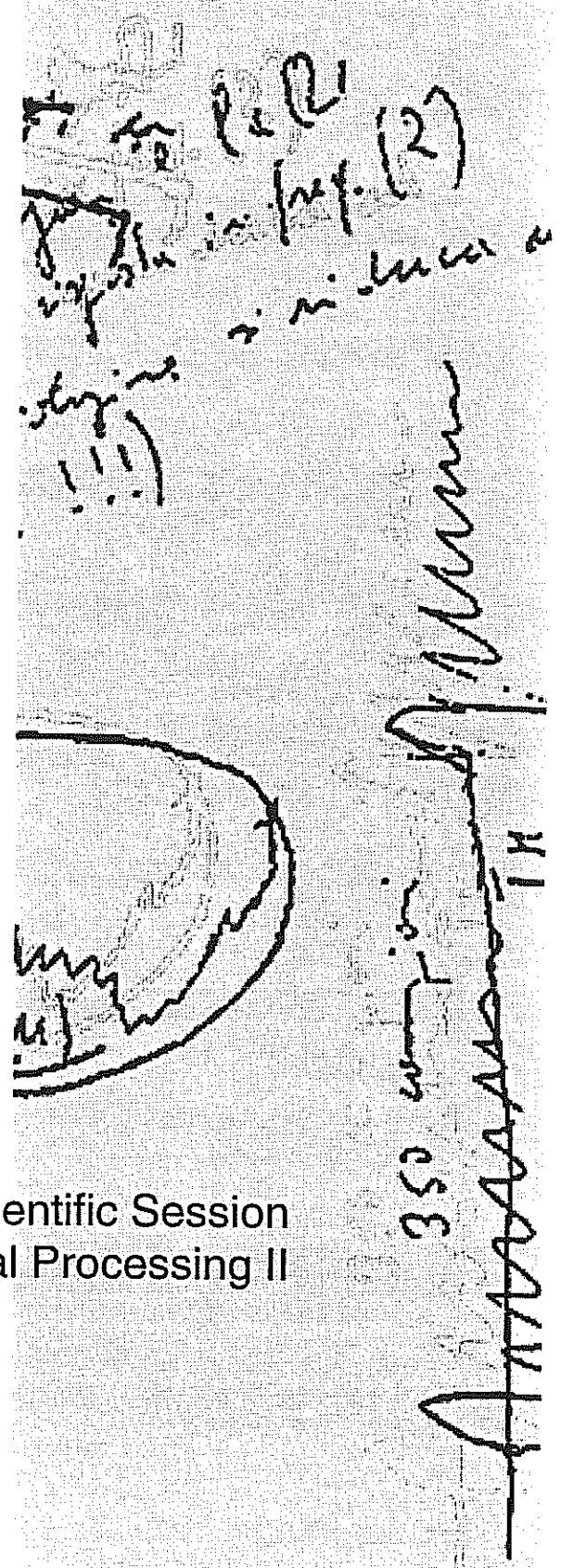
This yields the strongest excitement as expressed in modes of the Excited (Ex) category. To this category belong the PZEV; (Ex1 mode), and PzEV; (Ex2 mode) that appear in singing music and text of great excitement.

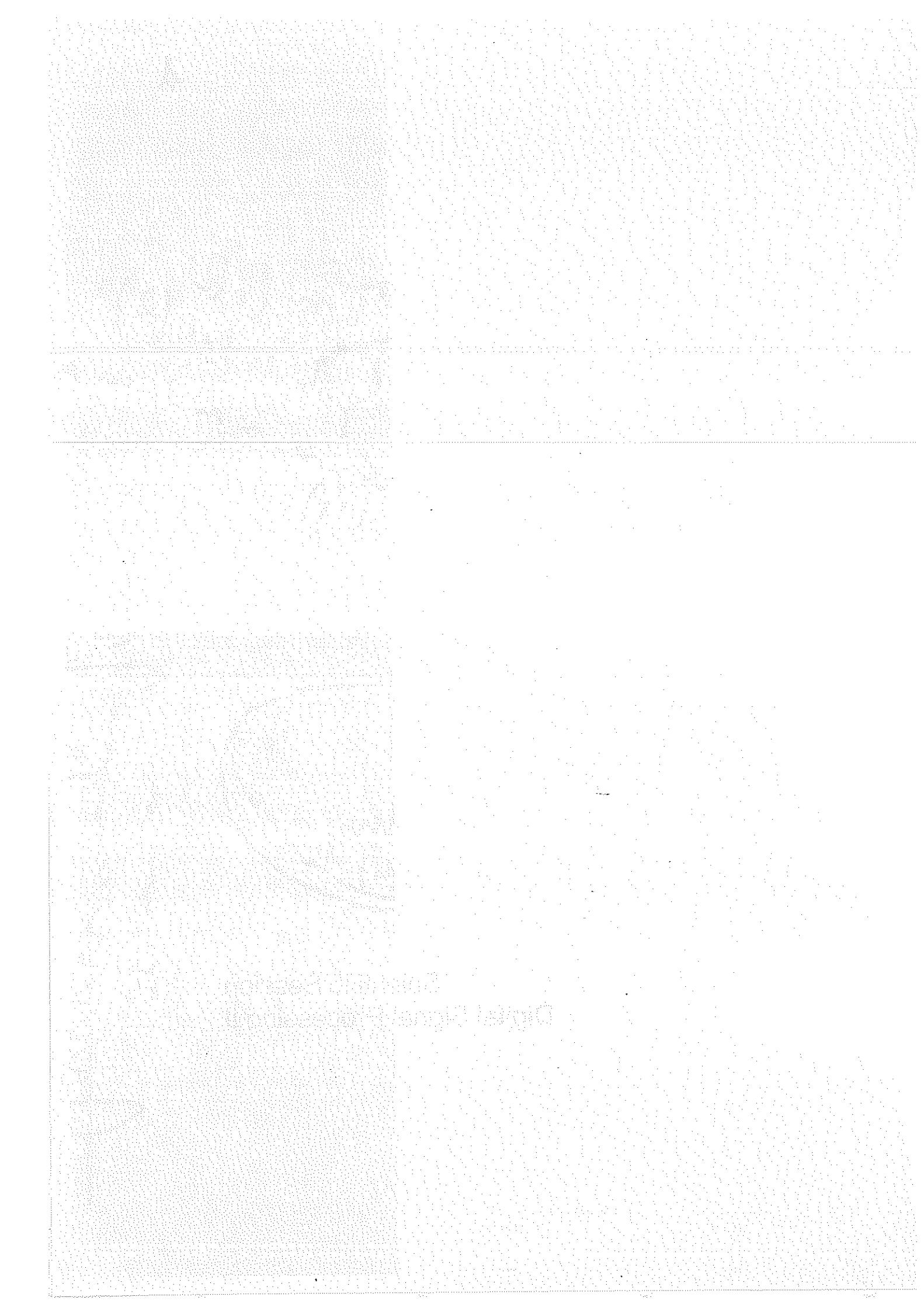
(9) The three categories: Short, Intermediate, and Virtuoso, will not be described for lack of space.

5 Acknowledgement

The author wishes to thank Professor Naftali Tishby, whose kindness and generosity made this work possible.

Scientific Session
Digital Signal Processing II





Fractional ARIMA Modeling of Microvariations in Additive Synthesis

Paolo Vettori

D.E.I. – DIPARTIMENTO DI ELETTRONICA E INFORMATICA
UNIVERSITÀ DEGLI STUDI DI PADOVA
via Gradenigo 6/a, 35131 PADOVA, Italy
e-mail: luis@dei.unipd.it, disel@dei.unipd.it

Abstract

In additive synthesis sounds of pitched musical instruments are modeled as a sum of slowly time varying partials. Besides these macroscopic variations of the spectral component there are microscopic deviations of both frequency and amplitude that contribute to create the timbre of sound; they are here called microvariations.

Fractional ARIMA models, suited to stationary and non-stationary signals and presenting both short and long time correlation, will be proposed in this work for modeling microvariations. They fit particularly to sounds related to chaotic phenomena; their relations with the classical fractional Brownian motion, therefore with fractals, are shown and discussed. New algorithms are proposed: the analysis gets a parameter related to the fractal dimension; the synthesis is obtained using approximating lattice filters.

1 Microvariations

Additive synthesis models sound signals as

$$s(t) = \sum_i a_i(t) \sin \left(\int_0^t 2\pi f_i(\tau) d\tau + \phi_i \right),$$

$a_i(t)$ and $f_i(t)$ representing instantaneous amplitude and frequency of the i -th partial (bold characters denote stochastic signals); they are the sum of a slowly time varying (in case constant) part and a microscopic deviation from this behaviour that here is called microvariation.

The question that gave rise to this work was: "is there a way to reproduce the *character* of microvariations in a pipe organ sound with little effort?" This paper is only trying to answer; anyway I hope it could show some interesting way to investigate further.

2 Fractals ... why?

Mandelbrot [4] considers the fractal geometry as the main way of studying nature: a typical case is represented by chaotic phenomena strictly bound to *fractal objects*. Sounds generated by wind instruments could therefore present some kind of fractal aspect due to air turbulence; we tried to find it in microvariations.

Fractals may be deterministic or stochastic: classical fractal examples (Koch-curve, Sierpin-

ski triangle) are deterministic; the one used to simulate landscapes [8] belong to the second type. Being interested in stochastic models, only the latter will be considered.

The starting point is BROWNIAN MOTION: $B(0) = 0$, $E[B(t_1) - B(t_0)][B(t_3) - B(t_2)] = 0$ $\forall 0 < t_0 < t_1 < t_2 < t_3$ (E is mathematical expectation: increments are statistically uncorrelated) and $B(t_1) - B(t_0)$ is a zero-mean gaussian variable with variance $t_1 - t_0$.

2.1 Continuous time fractal signals

The main feature of a continuous time stochastic fractal signal is statistical self-affinity: a signal is statistically similar to a piece of itself.

The principal fractal signal is the FRACTIONAL BROWNIAN MOTION $B_d(t)$ so defined:

$$\begin{aligned} B_d(t) \triangleq \frac{1}{d!} & \left[\int_{-\infty}^0 (|t - \tau|^d - |\tau|^d) dB(\tau) \right. \\ (1) \quad & \left. + \int_0^t |t - \tau|^d dB(\tau) \right], \\ B_d(0) \triangleq 0, \quad |d| < \frac{1}{2}, \quad d! & = \Gamma(d+1) \end{aligned}$$

thus FBM is the d^{th} stochastic fractional integral of BM (it is easy to verify that $B_0(t)$ is BM).¹

¹While 'stochastic' refers to the stochastic measure

Using $\stackrel{D}{=}$ to denote equal probability distributions, FBM has stationary increments

$$B_d(at) \stackrel{D}{=} B_d(s + at) - B_d(s)$$

and is statistically self-affine

$$B_d(at) \stackrel{D}{=} a^{d+1/2} B_d(t).$$

The derivative² of BM is the well-known signal called WHITE NOISE: $W(t)$; similarly FRACTIONAL GAUSSIAN NOISE, $W_d(t)$, is the derivative of FBM or the d^{th} fractional integral of WN.

It can be shown that FGN is a stationary signal with spectrum proportional to f^{-2d} . FBM, being its integral, should³ have spectrum proportional to f^{-2-2d} (this is the so-called $1/f^\alpha$ noise with $1 < \alpha < 3$). Its fractal dimension (cfr. [8]) is $D = (5 - \alpha)/2 = 1.5 - d$.

2.2 Discrete time fractal signals

Obviously self-affinity is inadmissible for a discrete time signal: however it is possible to find a signal somehow equivalent to FBM.

A first way [3] is sampling FBM and considering its increments:

$$w_d(nT) \triangleq B_d((n+1)T) - B_d(nT).$$

This leads to a stationary signal, called DISCRETE FRACTIONAL GAUSSIAN NOISE, with correlation $r_{w_d}(kT) \triangleq E[w_d((n+k)T)w_d(kT)]$ such that

$$r_{w_d}(kT) \propto k^{2d+1} \text{ as } k \rightarrow \infty.$$

The spectrum, with an analytical form hard to calculate and write, is still proportional to f^{-2d} for low frequencies.

3 Fractional ARIMA models

The second way to obtain discrete time fractal signals consists in *translating* the idea of fractional integral or derivative: in a *very* heuristic manner it is not so difficult to notice that putting

$$\widehat{B}_d(t) \triangleq \frac{1}{d!} \int_{-\infty}^t (t-\tau)^d W(\tau) d\tau$$

equation (1) may be rewritten as

$$(2) \quad B_d(t) = \widehat{B}_d(t) - \widehat{B}_d(0).$$

used, 'fractional' integral means that this is only a *fraction* of integral: it will be explained in section 3 that it is *equivalent* to the $-d^{\text{th}}$ fractional derivative.

²BM does not admit ordinary derivatives: only a generalized derivative is possible; so we obtain a generalized (in the sense of Schwartz's distributions) stochastic process.

³Unfortunately it is not stationary, so its spectrum is not defined; a generalization, the Wigner-Ville spectrum, confirms this intuitive fact.

$\widehat{B}_d(t)$ is WN filtered by $t^d/d!$ (whose Laplace-transform is $1/s^{d+1}$); thus its derivative, equal to $W_d(t)$ for (2), is $W(t)$ filtered by $1/s^d$ in the frequency domain.

To sum up: the d^{th} fractional integral *corresponds* to the d^{th} power of the integrative operator $1/s$ or to the $-d^{\text{th}}$ power of the derivative operator s .

To change from continuous to discrete time signals we must replace Laplace-transforms with Z-transforms and the derivative operator with the difference operator $\nabla \triangleq 1 - z^{-1}$. Therefore the fractional integrative operator for discrete time signals will be ∇^{-d} in the frequency domain.

3.1 Notation for ARIMA models

Considering 'z' as a shift operator such that $z^n x(k) \triangleq x(k+n)$ it is possible to write a standard difference equation like

$$(3) \quad \sum_{h=0}^p a_h x(k-h) = \sum_{h=0}^q b_h \varepsilon(k-h)$$

in a more compact way: if $A(z) = \sum_0^p a_h z^h$ and $B(z) = \sum_0^q b_h z^h$ then (3) is equivalent to

$$(4) \quad A(z^{-1})x(k) = B(z^{-1})\varepsilon(k).$$

A signal that may be well modeled choosing convenient coefficients a_h and b_h is called AUTO REGRESSIVE MOVING AVERAGE.

When $\varepsilon(k)$ is DISCRETE WHITE NOISE, i.e. a stationary signal whose correlation is $r_\varepsilon(0) = 1$ and $r_\varepsilon(n) = 0$ otherwise, then (4) defines a signal $x(k)$ having spectrum⁴

$$S_x(z) = \frac{B(z^{-1})B(z)}{A(z^{-1})A(z)}.$$

Given a (rational) spectrum it is possible to determine polynomials $A(z)$ and $B(z)$ having no zeroes inside the unit circle $\{|z| < 1, z \in \mathbb{C}\}$. This allows, from equation (4) and *inverting* $B(z^{-1})$ (or $A(z^{-1})$), to express $x(k)$ (or $\varepsilon(k)$) as a linear combination of past values of $\varepsilon(k)$ (or $x(k)$).

A zero on $|z| = 1$ would still permit this *inversion*, but would introduce another problem: it is incompatible with the hypothesis, tacitly assumed, that we work with finite variance signals.

In particular the difference operator ∇ has a zero in $z = 1$ and if it is a factor of the polynomial $A(z)$ then the signal $x(k)$ may have an infinite variance. This model, called ARIMA,⁵ was introduced by Box and Jenkins to treat non-stationary signals.

⁴The word 'spectrum' indicates here, depending on the context, both the Z-transform of the correlation and its Fourier-transform which is the previous evaluated at $z = e^{j2\pi f}$.

⁵'I' means INTEGRATED: 'differencing' the output is the same as 'integrating' the input.

3.2 Fractionally Differenced WN

However, if ∇^d appears on the left side of equation (4), $x(k)$ still has finite variance if $|d| < 1/2$. The simplest case is

$$\nabla^d \varepsilon_d(k) = \varepsilon(k);$$

the signal $\varepsilon_d(k)$ is called FRACTIONALLY DIFFERENCED WHITE NOISE. The input/output relation in the frequency domain is ∇^{-d} , the ‘translation’ of s^{-d} that should characterize a fractal signal.

In order to strengthen the motivation to say that FDWN is a fractal signal, it is easy to prove that its correlation is

$$r_{\varepsilon_d}(k) \propto k^{2d+1} \text{ as } k \rightarrow \infty$$

and the spectrum of FDWN is

$$(5) \quad S_{\varepsilon_d}(e^{j2\pi f}) = \frac{1}{(\sin 2\pi f)^{2d}} \propto f^{-2d} \text{ as } f \rightarrow 0$$

as for DFGN; furthermore DFGN and FDWN are connected each other by a precise relation [2].

The way that leads to FDWN goes beyond: a very simple but more flexible signal is the FRACTIONAL ARIMA

$$A(z^{-1}) \nabla^d x(k) = B(z^{-1}) \varepsilon(k)$$

or equivalently

$$(6) \quad A(z^{-1}) x(k) = B(z^{-1}) \varepsilon_d(k).$$

It has still a long time correlation dependent on the parameter d (or, it is the same, the particular behaviour (5) of the spectrum for low frequencies), but its short time correlation (or its high frequencies behaviour) may be controlled choosing appropriate coefficients of the polynomials $A(z)$ and $B(z)$.

Moreover d may assume any real value: e.g. $d > 1/2$ can model the (non-stationary) discrete time equivalent of FBM.

4 Algorithms

Looking at (6), the key for generating or identifying fractional ARIMA signals is FDWN: once we got a FDWN generator or we are able to find out the parameter d , then standard methods for ARMA models will do the rest.

4.1 Synthesis

An algorithm (explained in [3] for DFGN) whose heart is Cholesky decomposition of a square matrix, permits to generate a piece of signal knowing its correlation; it results computationally heavy.

I used Levinson algorithm: it allows to build a filtering lattice structure [6] able to generate a

signal whose spectrum approximates the ideal (5) while its correlation is equal to the requested one upon a value that must be chosen as higher [7] as d approaches 1/2.

4.2 Analysis

There is a great variety of algorithms to get the fractal dimension of a supposed fractal object. One of these (based on morphological filtering [5]) I worked with earlier, will be used to compare results.

The new algorithm is based on a statistical property of fractional ARIMA models [7]: if $\bar{x}(k)$ denotes the sample mean of n values of $x(k)$:

$$\bar{x}_n(k) = \frac{1}{n} \sum_{h=0}^{n-1} x(k+h)$$

then, if $x(k)$ is ergodic, $\bar{x}_n(k) \rightarrow \bar{x} \triangleq E[x(\cdot)]$ as $n \rightarrow \infty$; therefore the variance of its sample mean goes to zero. More precisely:

$$n^{1-2d} E[\bar{x}_n(\cdot) - \bar{x}]^2 \rightarrow C \text{ constant.}$$

From the algorithmic point of view let $x[k]$ be the vector containing values for $k = 0, \dots, N-1$ of the (supposed zero-mean) signal. Then

$$(7) \quad \bar{x}_n(k) = \frac{1}{n} \sum_{h=k}^{k+n-1} x[h] \text{ for } 0 \leq k < N-n$$

and the approximation of the variance of its sample mean is

$$\tilde{\sigma}_n^2 = \frac{1}{N-n+1} \sum_{k=0}^{N-n} \tilde{x}_n^2(k) \simeq C n^{2d-1}.$$

This equation, together with (7), gives

$$\tilde{\sigma}_n^2 = \frac{1}{n^2(N-n+1)} \sum_{k=0}^{N-n} \left\{ \sum_{h=k}^{k+n-1} x[h] \right\}^2;$$

once we have calculated the sequence $\tilde{\sigma}_n^2$, from the relation

$$(8) \quad \log \tilde{\sigma}_n^2 \simeq (2d-1) \log n + \log C$$

we can easily deduce the searched value of d by linear regression.

Another way, useful when N is too small, consists in considering the signal periodic, i.e. $x[k+iN]=x[k]$; so we obtain another approximation

$$\hat{\sigma}_n^2 = \frac{1}{n^2 N} \sum_{k=0}^{N-1} \left\{ \sum_{h=k}^{k+n-1} x[h] \right\}^2$$

that can be rearranged in a simpler form: putting $\zeta_n = n^2 N \hat{\sigma}_n^2$ then:

$$\zeta_{n+1} = 2\zeta_n - \zeta_{n-1} + 2 \sum_{h=0}^{N-1} x[h] x[h+n].$$

5 Experimental results

I synthesized and then analyzed FDWN signals, d varying from -0.45 to 0.45 with step 0.05 , to test the algorithms. Values found ($\pm 2\sigma$) are plotted in fig. 1 versus requested ones; the dashed line represents a similar try reported in [5].

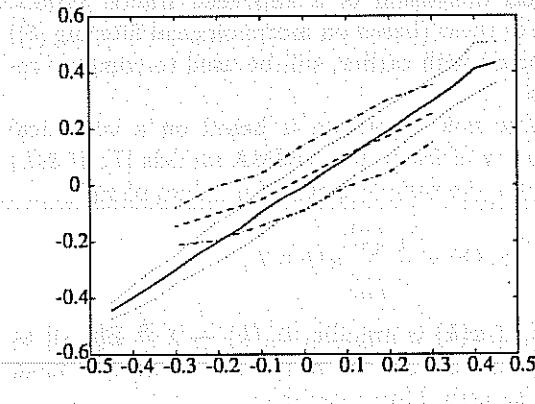


Figure 1: Analysis of synthesized FDWN signals

Applied to real microvariations the analysis algorithm gave good results: in general microvariations of different harmonics have the same behaviour; frequency microvariations are stationary while amplitude microvariations are non-stationary.

Fig. 2 reports $\log \tilde{\sigma}_n^2$ plotted versus $\log n$ (see equation (8)) for frequency microvariations of an 8 feet flute register, a very particular case.

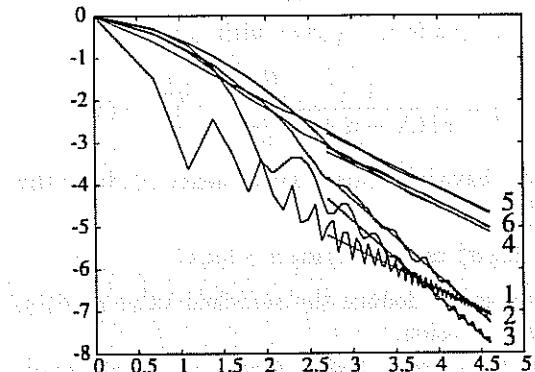


Figure 2: Frequency microvariations

Here, after a first transitory part, the second and third harmonic differ from the others: they seem to be ARIMA with $d \approx -0.41$ while the four remaining are 'simple' ARMA i.e. with $d \approx 0$. Also reed registers can be well modeled by ARMA confirming results of previous works [1] while other registers show values of d less or equal than zero belonging to the set of values for which the pro-

posed analysis and synthesis algorithms work at their best.

6 Conclusion

This paper shows a new way for modeling microvariations: traditional ARMA models [1] may be insufficient or lead to an expensive parameterization; a single parameter, d , representing long-term correlation, permits to get a better modeling with a smaller number of parameters allowing to handle even non-stationary signals.

However fractional ARIMA models and the new algorithms proposed, although conceived for the peculiar case of sound microvariations, could be useful in many other fields.

References

- [1] L. Bazzanella and G. B. Debiasi, "Microvariazioni dei suoni dell'organo a canne e loro modellizzazione per l'impiego nella loro sintesi numerica," in *Proceedings CIARM '95*, (Ferrara, Italy), pp. 67-72, May 1995.
- [2] J. Geweke and S. Porter-Hudak, "The estimation and application of long memory time series models," *J. Time Series Analysys*, vol. 4, no. 4, pp. 221-238, 1983.
- [3] T. Lundahl, W. J. Ohley, S. M. Kay, and R. Sifert, "Fractional brownian motion: A maximum likelihood estimator and its application to image texture," *IEEE Trans. on Medical Imaging*, vol. MI-5, pp. 152-161, Sept. 1986.
- [4] B. B. Mandelbrot and J. W. V. Ness, "Fractional brownian motions, fractional noises and applications," *SIAM Rev.*, vol. 10, no. 4, pp. 422-437, 1968.
- [5] P. Maragos and F.-K. Sun, "Measuring the fractal dimension of signals: Morphological covers and iterative optimization," tech. rep., Harvard University, Oct. 1991.
- [6] R. A. Roberts and C. T. Mullis, *Digital Signal Processing*. Reading, Massachusetts: Addison-Wesley Publishing Company, 1987.
- [7] P. Vettori, "Un approccio frattale nello studio delle microvariazioni di suoni d'organo," internal report, CSC - Università degli Studi di Padova, July 1994.
- [8] R. F. Voss, "Random fractal forgeries," in *Fundamental Algorithms for Computer Graphics* (R. A. Earnshaw, ed.), pp. 805-835, Berlin Heidelberg: Springer-Verlag, 1985.

Karplus-Strong Parameter Estimation

Gianpaolo Evangelista, Sergio Cavaliere

ACEL, Dipartimento di Scienze Fisiche,
Università "Federico II" di Napoli,
Mostra d'Oltremare, pad. 20, 80125 Napoli
e-mail: acel@na.infn.it
fax: +39-81-2394508

Abstract

The Karplus and Strong (KS) algorithm plays a central role in the synthesis of string instruments and percussions, making it possible to obtain excellent results at a reasonable computational cost. Often it is desirable to extract the KS synthesis parameters from natural sounds. However, the accurate estimate of the synthesis parameters is still an open problem. In this paper we try to provide new solutions by first introducing a closely related synthesis algorithm, the ICC technique, together with an analysis procedure, and then generalizing the ICC analysis method to the KS algorithm.

1 Introduction

The interest in physical models for the synthesis of musical signals has recently increased due to both the availability of powerful hardware and the naturalness of the sounds that can be obtained. The Karplus and Strong (1983) algorithm, together with its numerous variants and generalizations (Jaffe and Smith 1983, Borin et al. 1992, Paladin and Rocchesso 1992, Smith 1992, Van Duyne and Smith 1994, etc.), plays a central role in the synthesis of string instruments and percussions, making it possible to obtain excellent results at a reasonable computational cost. In several realizations, the KS algorithm constitutes a building block on which more complex synthesis schemes are based in order to take into account a number of factors such as the string-bridge coupling, the sympathetic vibrations of the other strings and so on.

In the KS algorithm, a buffer containing initial data is cyclically updated with filtered versions of itself, simulating a vibrating string and its interactions with the boundaries and the environment. The algorithm is efficiently implemented by means of a delay line and a suitable loop filter. The sound evolution is completely determined by specifying the filter impulse response and the initial data.

The accurate estimation of both these parameters is still an open problem and, in this pa-

per, we try to provide a contribution in this direction.

2 Available Estimation Techniques

In principle, the KS synthesis parameters may be directly estimated by means of costly and time-consuming measurements of physical quantities associated with the instrument. Alternately, it may be possible to extract these parameters from sampled sounds recorded from the instrument and several techniques have been suggested in the literature (Smith 1983, Karjalainen et al. 1993). In some of the estimation methods the KS algorithm is regarded as the realization of a comb-like autoregressive filter whose parameters may be evaluated by means of modified linear prediction (Smith 1982). It turns out that this technique does not always lead to stable filters and that high frequencies are not faithfully represented in the synthesis. Other methods rely upon the STFT, either extracting the frequency response of the loop filter as the arithmetic mean of the ratios of consecutive spectral segments, or performing linear regression of the partial amplitudes in a dB scale (Karjalainen et al. 1993). The filter is designed by means of standard techniques with the objective of matching the desired frequency response. The loop filter should also contain an adjustable fractional delay element, usually implemented in an all-pass filter, i.e. a

Blaschke product (Van Duyne and Smith 1994). This filter is required in order to compensate the phase delay and tune the tone. Once the delay and the loop filter become known, the excitation — or the initial data contained in the buffer — may be obtained by means of inverse filtering. However, one is interested in making the excitation signal of short duration.

3 A Related Algorithm: The Iterated Circular Convolution Method

An intuitive but crude approximation to the KS synthesis algorithm is given by the circular convolution method. In this technique, a length P buffer is periodically updated by substituting its content with the circular convolution of itself with a cyclic impulse response $g(p)$. Therefore, the sound is generated by means of iterated circular convolution, as shown in fig. 1.

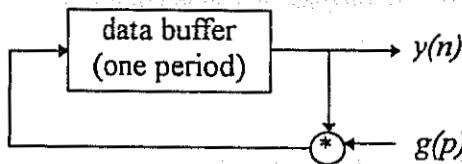


fig. 1: The Iterated Circular Convolution algorithm.

The frequency domain counterpart of this algorithm is given in terms of DFTs. In fact, if

$$X(m) = \sum_{p=0}^{P-1} x(p) e^{-j\frac{2\pi mp}{P}}$$

is the DFT of the initial buffer content and

$$Y_n(m) = \sum_{p=0}^{P-1} y(p + nP) e^{-j\frac{2\pi mp}{P}}$$

is the DFT of the n -th output period, we have:

$$Y_n(m) = G(m)Y_{n-1}(m), \quad (1)$$

where $G(m)$ is the cyclic frequency response and

$$Y_0(m) = X(m).$$

It is easy to see that

$$Y_n(m) = G^n(m)X(m).$$

In spite of the numerical differences existing between the original KS algorithm and the Iterated Circular Convolution (ICC) technique (see fig. 2), the acoustical results are very similar. This can be explained by the fact that the frequency spectra differ little (see fig. 3). The ICC method contains the intuitive principle that the frequency spectrum of the signal is obtained from an initially rich spectrum by successive filtering, thus resulting in a bright to dark transition that is typical of vibrating strings. The KS algorithm approximately works in the same way and the STFT based parameter estimation exploits this property in order to extract the loop filter frequency response.

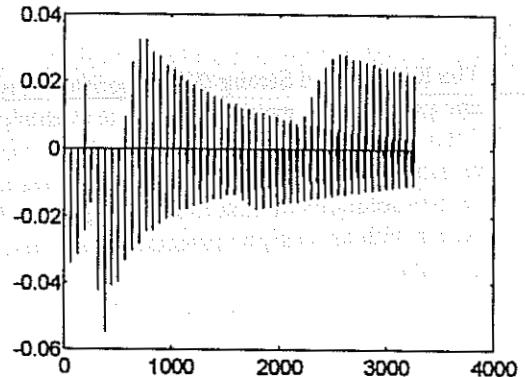


fig. 2: ICC and KS : time domain difference. Signals were normalized to their maximum amplitude.

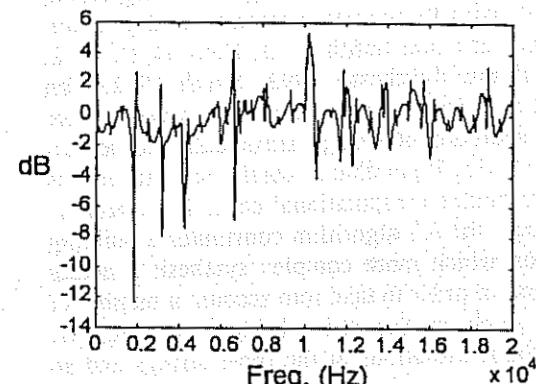


fig. 3: ICC and KS : the difference between the two dB spectra

Estimation of the ICC frequency response from sampled data may be carried out by inverting eq. (1), obtaining the sequences

$$\hat{G}_n(m) = \frac{Y_n(m)}{Y_{n-1}(m)},$$

which represent several possible estimates of $G(m)$. In order to obtain a single estimate of the ICC frequency response, one can perform a weighted geometric mean of K estimates, according to the following formula:

$$\hat{G}(m) = \prod_{n=2}^{K+1} (\hat{G}_n(m))^{W_n/K}$$

Weights W_n control the confidence, i.e., the contribution of the n -th estimate to the global estimate. If the weights are all set to be unitary, our estimates revert to the k -th root of the ratio of the DFTs of the $K+1$ -st output period to the DFT of the first period. However, in order to mask transient effects, it is often desirable and more accurate to weight the estimates according to their reliability. Weights may also be frequency dependent in order to limit problems with high-frequency fast decay. Typically, low frequencies are best estimated far from the initial transient while high frequencies are best estimated near the transient. A suitable weight normalization constraint is the following:

$$\sum_{n=2}^{K+1} W_n(m) = K$$

Once the ICC frequency response has been determined, one could invert eq. (1) again, for $n=1$, in order to estimate the initial buffer. However, we found it more convincing from an acoustical standpoint to fill the initial buffer with the first period of the original signal.

In our experiments, the estimated ICC frequency response has a tendency to be all-pass. This is explained by the fact that each period differs from its neighbors only for a small amount. Thus, the impulse response is roughly a Kronecker delta pulse.

There are several drawbacks in using the ICC method instead of the KS algorithm. A fundamental one is that the frequency response is rarely portable to a different pitch. In fact it depends on the period length P . A second drawback is that one can only generate sounds whose period is an integer multiple of the sampling interval. In the KS algorithm, non-integer periods are generated by adding a suitable phase factor to the loop filter.

A useful generalization of the ICC algorithm illustrated is to let the frequency response be time-varying. We experimented that, by using just four different frequency responses per tone, one can dramatically improve the acoustical result.

4 Proposed Estimation Techniques and Experiments

The KS algorithm is only capable of generating sounds whose partials decay exponentially. Therefore it is sometimes difficult to estimate the synthesis parameters from real-life sounds since the KS model may be not sufficient by itself and further processing is required. A good analysis algorithm is required to at least provide exact estimates when running on KS-generated synthetic data.

It is well-known that the signal $y(n)$ generated by means of the KS algorithm is given by the following loop-filtering equation:

$$y(n) = x(n) + \sum_{k=0}^L h(k)y(n-k-P), \quad (2)$$

with transfer function:

$$T(z) = \frac{1}{1 - z^{-P} H(z)},$$

where $h(k)$ is the loop-filter impulse response of degree L , P is the (integer) pitch parameter -- not necessarily the actual pitch, depending on the group delay of $H(z)$ -- and $x(n)$ is the input sequence consisting of the initial buffer, padded with zeros in order to achieve the required duration of the output. If we let $n=p+rP$ in (2), where $p=0, 1, \dots, P-1$, and r is the integer period-count index, we obtain the following pitch-synchronous form of the KS:

$$y_r^{(0)}(p) = x_r(p) + \sum_{k=0}^L h(k)y_{r-1}^{(k)}(p), \quad (3)$$

where

$$y_r^{(k)}(p) = y(p-k+rP),$$

and

$$x_r(p) = x(p+rP).$$

By taking the DFT, with respect to the index p , of both sides of eq. (3), we obtain:

$$Y_r^{(0)}(m) = X_r(m) + \sum_{k=0}^L h(k)Y_{r-1}^{(k)}(m). \quad (4)$$

Far from the initial transient we have $X_r(m) \approx 0$ and, for any fixed r , (4) becomes a system of algebraic linear equations in the $L+1$ unknowns $h(k)$, with known term $Y_r^{(0)}(m)$ and the P by $L+1$ coefficient matrix $Y_{r-1}^{(k)}(m)$, $m=0, 1, \dots, P-1$; $k=0, 1, \dots, L$. If, as in most practical cases, $L < P$, the

may be found for the loop filter impulse response $h(k)$. Further redundancy may be added by considering eq. (4) for several values of the period-count index r and finding an overall least squares solution of a large system of equations. Alternately, one may divide both sides of eq.(4) by the left-hand side $Y_r^{(0)}(m)$ and define the ratios

$$R_r^{(k)}(m) = \frac{Y_r^{(0)}(m)}{Y_{r-1}^{(k)}(m)}, \quad (5)$$

obtaining the following system:

$$\sum_{k=0}^L h(k) / R_r^{(k)}(m) = 1. \quad (6)$$

In order to reduce the size of the system while still taking into account several periods of the signal, the ratios (5) may be substituted in (6) by an estimate obtained by a method similar to the weighted geometric means illustrated in the previous section. Incidentally note, by comparing eq. (4) with eq. (1), that the KS algorithms reverts to the ICC in the very particular case that the output y satisfies the following condition:

$$Y_r^{(k)}(m) = Y_r^{(0)}(m) e^{-j\frac{2\pi km}{P}}$$

In order to assess the validity of the proposed KS analysis algorithm, we run some comparative tests on both synthetic and natural signals. We compared our results with those obtained using Modified Linear Prediction (MLP, system identification method) and linear regression of the log short-time spectrum (STFT method). On most real-life signals all these algorithms have a tendency to fail. This may be ascribed to the fact that the pure KS synthesis algorithm is not *per se* sufficient to accurately model sounds. Several physical phenomena such as dispersion, string stiffness and sympathetic vibrations may occur (Smith 1990). On synthetic sounds, generated by means of the KS technique using various loop filter responses, which we pretend to ignore during analysis, we reported some differences. Both STFT and MLP do not allow for phase retrieval: the estimated loop filter may, at best, match only the magnitude frequency response of the synthesis loop filter. On the contrary, in our method, the estimated loop filter matches both the desired magnitude and phase, i.e., it matches the impulse response of the synthesis loop-filter with a striking precision (max error less than 10^{-5}). Furthermore, in both STFT and MLP spectral matching is poor at high frequencies while this does not seem to happen with our technique. We would like to point out that in our method, even if the degree of the loop-filter is unknown to the analysis procedure, the estimated im-

pulse response is correct if a larger degree is assumed during analysis. The resulting impulse response is simply a zero-padded version of the original. This does not seem to happen with MLP in that the obtained frequency response does not necessarily improve if a larger degree is assumed in the analysis.

The proposed analysis method may outperform the available KS parameter estimation techniques. We are currently conducting more experiments on natural sounds in order to refine our method in those cases where the KS represents a good model.

Perceptual evaluation of the various analysis/synthesis procedures largely benefit from the real time implementation on the MARS workstation by IRIS.

References

- K. Karplus, A. Strong. "Digital Synthesis of Plucked-String and Drum Timbres", *Computer Music Journal*, Vol. 7, no. 2, pp.43-55, 1983.
- Julius.O.Smith. "Synthesis of Bowed Strings", *Proceedings of the ICMC'82*, Venice 1982.
- D. Jaffe and J. O. Smith. "Extensions of the Karplus-Strong Plucked String Algorythm", *Computer Music Journal*, vol.7,no.2, pp 56-69, 1983.
- Julius. O. Smith. "Efficient Yet Accurate Models for Strings and Air Columns Using Sparse Lumping of Distributed Losses and Dispersion", in *Colloquium on Physical Modeling*, Grenoble 1990.
- Julius. O. Smith. "Physical Modeling Using Digital Waveguides", *Computer Music Journal*, vol.16, no.4, pp 74-91, Winter 1992
- S. A. Van Duyne and J. O. Smith. "A Simplified Approach to Modeling Dispersion Caused by Stiffness in Strings and Plates", *ICMC Proceedings 1994*, pp.407-410.
- G. Borin, G.De Poli, A. Sarti. "Algorithms and Structures for Synthesis Using Physical Models", *Computer Music Journal*, Vol. 16, N. 4, 1992 pp.30-42.
- A. Paladin., D. Rocchesso. "A Dispersive Resonator in Real Time on Mars Workstation", *ICMC Proceedings 1992*, pp.146-149.
- M. Karjalainen, V. Valimaki, Z. Janosy. "Towards High-Quality Sound Synthesis of the Guitar and String Instruments", *ICMC Proceedings 1993*, pp.56-62.

A Hammer-String Interaction Model for Physical Model Syntesis

G. Borin, G. De Poli

C.S.C., D.E.I., Università degli studi di Padova via Gradenigo 6/a, 30131
Padova, Italia. Tel. 049/8277631-Fax 049/8277699- E-Mail depoli@dei.unipd.it.

A fast and accurate method for hammer-string interaction simulation in physical model synthesis is proposed. This method is based on the solution of the nonlinear system of a linear ideal string and a polynomial nonlinear mass-spring system constituting the hammer. The method allows to tabulate values for nonlinear characteristic of the felt without forcing to recalculate the table for system parameters variations.

1. Introduction

Physical model synthesis of piano tones requires a good identification of the parameters when realistic sound for musical purposes is required. As far as piano is concerned, classical models of hammer, based on a mass and a nonlinear spring which gives account of felt force/deformation characteristics, have been proposed, for example in [1], [2], [3], [7], [8] and they are often based on the relation:

$$f = k(\Delta y)^\alpha l(\Delta y)$$

where f is compression force [N], Δy is the felt compression [m], $l(\cdot)$ is the Heaviside function which implements the so called "contact condition" and k [N/m^α] and α [adim] are suitable constants.

This class of models has been proved efficient and satisfactory in a first order approximation by many authors; however, in real time synthesis applications, it suffers of some problems.

Main problem is that in a real piano k and α varies in a rather unpredictable manner along the keyboard. Hence, k and α estimation is possible only if enough experimental data are available; if not, a parametrizing trimming by trials must be done. Furthermore, experimental evidence [3] shows that α varies continuously between 1.5 and 5 from bass to treble: this makes difficult to implement the felt characteristic in a real time, fixed-point DSP architecture.

We propose a model of piano hammer which offers flexible parametrical control over the static characteristic of the felt. The model is accurate and efficient and it gives the possibility of varying the characteristic of the felt through two parameters, which control, almost independently, the hardness and the overall shape of the characteristic.

Tests on real time DSP implementing a classical hammer-string interaction model have shown a remarkable influence over spectral properties of sound produced in mid and high octaves, letting the player control *pp* and *ff* colors in a rather independent manner. These tests, however, have highlighted the instable behavior of the approximate model described in [4]. Hence, a closed form solution has been derived for felt in II and III order nonlinearity and a general solution for p -th order nonlinearity has been found. These solutions show an important property: the parameters of the model multiply dependent and independent variables of the resulting solution curves; this allows to tabulate one single curve of interaction in an adimensional form for every possible choice of model parameters.

2. Narrow and elastic hammer

In a perfectly elastic hammer, compression characteristic of the felt can be obtained in a "quasi stationary" way applying a known force and measuring the compression caused by the former. In this way, the experimental data obtained can be fitted by a polynomial approximation [1]. However, it has been shown that a simpler power law is often adequate in characterizing felt property [2], [3].

Since we need a general form, capable of simulating a continuous variation of α between 2 and 4 (at least) but we need a polynomial for ease in calculus and, if possible, only one control parameter, we propose a polynomial model based on linear interpolation of the II and IV order curves through an adimensional coefficient varying from 0 to 1.

$$f(\Delta y, \eta, k) = k\eta \left(\frac{\Delta y}{Y} \right)^4 + k(1-\eta) \left(\frac{\Delta y}{Y} \right)^2$$

In this model k is a force measured at the Y compression. η is the "shape coefficient"; when $\eta=0$ the felt exhibits a II order nonlinearity, when $\eta=1$ it exhibits a IV order nonlinearity. In the intermediate cases we get an intermediate behavior.

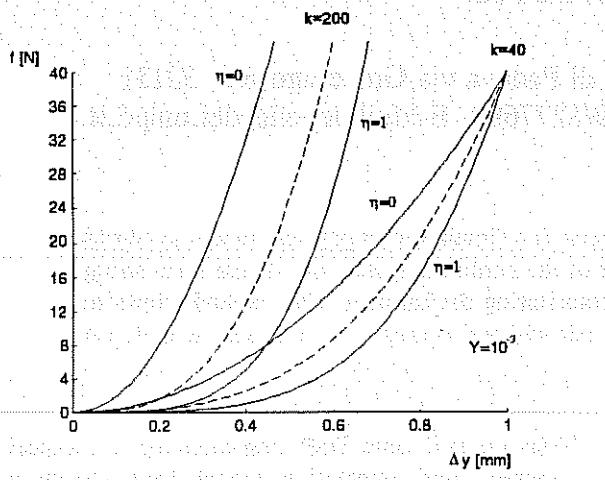


Fig. 1. Static force-compression characteristic of the felt. Dashed lines refer to an exact III order characteristic.

3. Approximated hammer-string model

In a classical hammer-string interaction model, the equations are:

$$\begin{cases} \frac{\partial^2 y}{\partial t^2} = \left(\frac{Z}{\sigma}\right)^2 \frac{\partial^2 y}{\partial z^2} \\ f(t) = k(y_h(t) - y(t))^p \\ f(t) = -m_h \frac{d^2 y_h}{dt^2} \end{cases}$$

with suitable boundary conditions. The discretized form, based on the backward and central difference approximation of derivatives, is:

$$\begin{cases} f(n) = 2Z(v(n) - v_i(n)) \\ f(n) = k \left(y_h(n) - T \sum_{j=0}^n v(j) \right)^p \\ y_h(n) = 2y_h(n-1) - y_h(n-2) - \frac{T^2}{m_h} f(n-1) \end{cases}$$

(see for example [5], [6] for resonator equations). This system is an implicit form in the variables $f(n)$ and $v(n)$. In the approximated solution method, described in [5], net velocity at the contact point is estimated by the following formula:

$$\begin{aligned} v(n) &= \frac{1}{2Z} f(n) + v_i(n) \\ &\approx \frac{1}{2Z} f(n-1) + v_i(n) \end{aligned}$$

Of course, the higher the sampling frequency, the more accurate the approximation of actual force with past force.

Approximated hammer-string model offers two main advantages: efficiency and flexibility. Efficiency comes from direct implementation of the equations, with no additional overhead in solving the nonlinear system. Flexibility comes from the fact that almost "every" form of compression characteristic can be directly implemented in the model; for example, hysteretic models described in [8] can be directly tested in approximated framework.

However, approximated model shows serious drawbacks: being based on an estimation of instantaneous force, it suffers of instability problems: if sampling frequency is not enough high, if string is short, if impact velocity of hammer is high or if hammer mass is too small, model behavior exhibits a strong departure from correct values.

4. Solution for nonlinearity of p-th degree.

In order to avoid instability problems, we now derive the digital solution of hammer-string interaction system in case of arbitrary p -th order nonlinearity, where p is an integer equal or greater than 2. The method is based on the separation of known terms, both instantaneous and "historical", from instantaneous and unknown terms. The latter are then rewritten as functions of force variable; in this way we get a p -th degree polynomial in $f(n)$ and one of its zeroes represents the solution of the system.

Equation for a p -th order nonlinear hammer-string interaction system are:

$$\begin{cases} v(n) = \frac{1}{2Z} f(n) + v_i(n) \\ f(n) = k(y_h(n) - y(n))^p \end{cases}$$

where $y(.)$ is string position at the contact point, $y_h(.)$ is hammer position and $v_i(.)$ is the incoming string velocity at the contact point.

Approximating continuous time integral with discrete sums using the trapezoids method, we get:

$$\begin{aligned} v_h(n) &= v_h(n-1) - \frac{T}{2m} (f(n) + f(n-1)) \\ y_h(n) &= y_h(n-1) + \frac{T}{2} (v_h(n) + v_h(n-1)) \end{aligned}$$

and for the expression of $v(n)$ as a function of $f(n)$:

$$y(n) = y(n-1) - \frac{T}{2} \left(v_i(n) + \frac{1}{2Z} f(n) \right) - \\ - \frac{T}{2} \left(+v_i(n-1) + \frac{1}{2Z} f(n-1) \right)$$

If we define:

$$b \equiv \frac{1}{2} \left(\frac{T}{2m_h} + \frac{1}{2Z} \right)$$

$$x(n) \equiv \frac{\Delta y(n-1)}{T} + v_h(n-1) - \\ - \frac{v_i(n) + v_i(n-1)}{2} - bf(n-1)$$

the felt compression becomes:

$$\Delta y(n) \equiv y_h(n) - y(n) = T(x(n) - bf(n))$$

and so the force expression becomes:

$$f(n) = a(x(n) - bf(n))^p$$

where $a \equiv kT^p$

We observe that in f expression, $x(n)$ definition collects all terms known at time n and the only term which is unknown is $f(n)$ itself. We can now rewrite the system:

$$\begin{cases} v(n) = \frac{1}{2Z} f(n) + v_i(n) \\ f(n) = a(x(n) - bf(n))^p \end{cases}$$

It is possible to obtain closed form solutions for nonlinearity of II, III and IV order analitically solving the equations above. The complexity of the resulting expressions and the need of tabulating the values of f as functions of x , however, make more interesting the use of iterative methods in calculating the zeroes of the resulting polynomials.

The tabular method, however, would be very inefficient if re-calculations of the solutions of the system were required for each variation of the parameters. Fortunately, this can be avoided: in the next paragraph we will derive a general multiplicative parametric form, which, for sake of brevity, will be called *quasi-nonparametric form*.

4.1 Quasi-nonparametric form

Let us consider the implicit expression of f as a function of x . If we pose:

$$C \equiv (pab)^{\frac{1}{p-1}}$$

$$X(n) \equiv Cx(n)$$

$$F(n) \equiv Cbf(n)$$

substituting, we find:

$$pF(n) = (X(n) - F(n))^p$$

One of the zeroes of the polinomial gives the values of $F(n)$ as functions of $X(n)$, and so allows the calculus of $f(n)$ as a function of $x(n)$. The relevant zero is in 0 when $x(n)=0$.

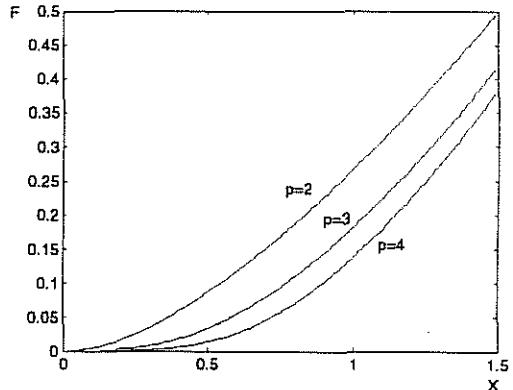


Fig 2. $F=Cbf$ [adim] as a function of $X=Cx$ [adim] for nonlinearities of 2nd, 3rd and 4th degree.

If we ipotize a model in which T , m_h , Z or k varies in a continuous way at roughly the same time scale of the audio signal, the multiplicative dependency of the (X,F) curves partially loses its utility, since we are forced to calculate C as $p-1$ -th root of pab . However, if p is left fixed, we can calculate C by means of a lookup table and so the value of $f(n)$ with a *double* lookup. The algorithm is:

1. find $C = (pab)^{\frac{1}{p-1}}$ in the first lookup table.
2. calculate $X=Cx$
3. find $F=Cbf$ in the second lookup table using index X .
4. calculate $f = F/Cb$.

Instead, if we ipotize a very slow variation of the parameters or a "stepwise" variation amongst interactions, since tuners usually change a parameter and then listen to the results, we can recalculate C and Cb offline and the calculus can be done by the host processor of the DSP.

4.2 Contact condition

We can also give the hammer-string contact condition as a function of the sign of $x(n)$ or $X(n)$.

As told in II order nonlinearity case, expression of compression of the felt during contact is:

$$\frac{\Delta y(n)}{T} = x(n) - bf(n)$$

Release condition can be then evaluated comparing $x(n) - bf(n)$ (or $X(n) - F(n)$) with zero. However, we note that for x near zero we have:

$$\lim_{x \rightarrow 0^+} \frac{f(n)}{x(n)} = 0$$

Hence, the release condition for Δy near zero can be given only as a function of $x(n)$ or $X(n)$. In other words, if $x(n)$ is greater than zero, we are in contact; when $x(n)$ becomes zero, contact ceases.

When there is no contact, $f=0$ for definition. Δy expression becomes:

$$\frac{\Delta y(n)}{T} = x(n) \quad \forall n \geq 0$$

Hence when there is no contact $x(n)$ is proportional to $\Delta y(n)$. It follows that also in this case the contact condition can be given as a function of $x(n)$ (or $X(n)$). In other words, if $x(n)$ is less than zero there is no contact; when $x(n)$ is zero, contact starts.

The union of the two cases studied allows us to define a contact condition based on the sign of $x(n)$. Hence, $x(n)$ can be regarded as a "pseudo-compression" variable. In summary, the recursive expression which updates $x(n)$ is sufficient both to calculate the value of $f(n)$ and to evaluate the contact condition.

4.3 Interpolating solutions

It is still possible to interpolate 2nd and 4th degree expressions in order to simulate a smooth variation of α between 2 and 4. This can be done with the following formula:

$$f(n) = \eta g_2(x(n)) + (1 - \eta) g_4(x(n))$$

where $g_2(\cdot)$ and $g_4(\cdot)$ are the second and fourth degree solutions curves. Hence, varying η from 0 to 1 we obtain a linear interpolation of the curves in the (x, f) plane, which corresponds to an interpolation of the curves in the $(\Delta y, f)$ plane.

5. Conclusions

A fast and accurate method for hammer-string interaction simulation in physical model synthesis has been proposed. This method was based on the solution of the nonlinear system of a linear ideal string and a polynomial nonlinear mass-spring system constituting the hammer. The technique employed allows to precalculate the solutions of hammer-string interaction system and to put them in a lookup table.

An arrangement of these solutions allows to use one single table of adimensional and nonparametric value for each degree of nonlinearity required. Furthermore, contact condition can be given in a very unexpensive form, which prevents the calculation of the actual value of felt compression.

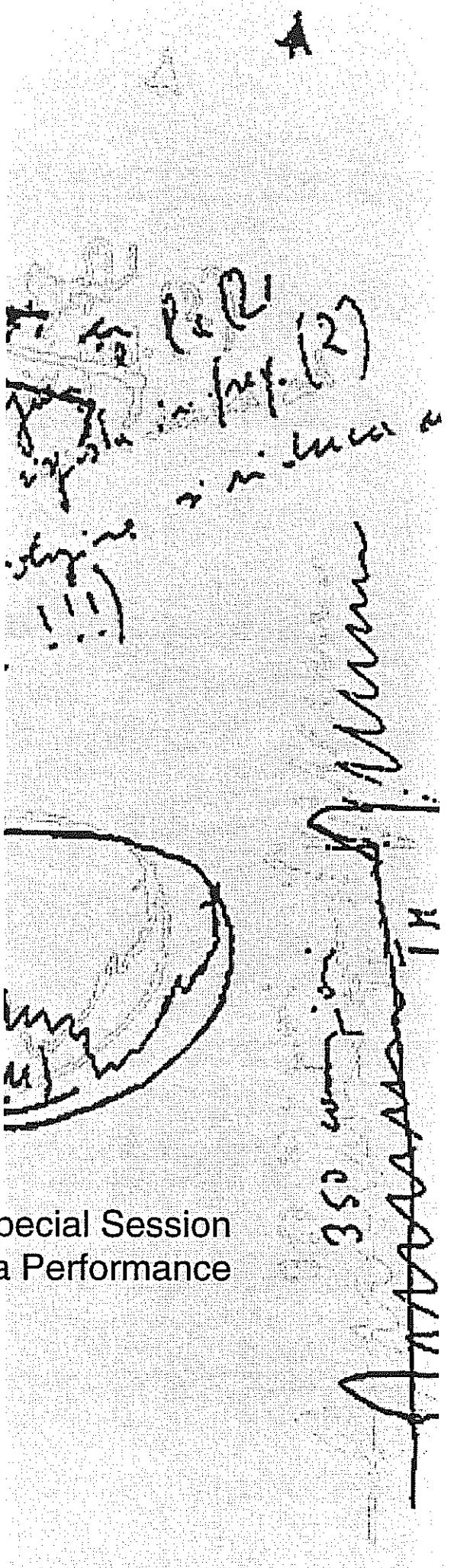
Linear interpolation between different degree of nonlinearity allows to simulate in a rough but very efficient way the continuous variation of the nonlinearity exponent required by an accurate model of the felt.

Acknowledgments

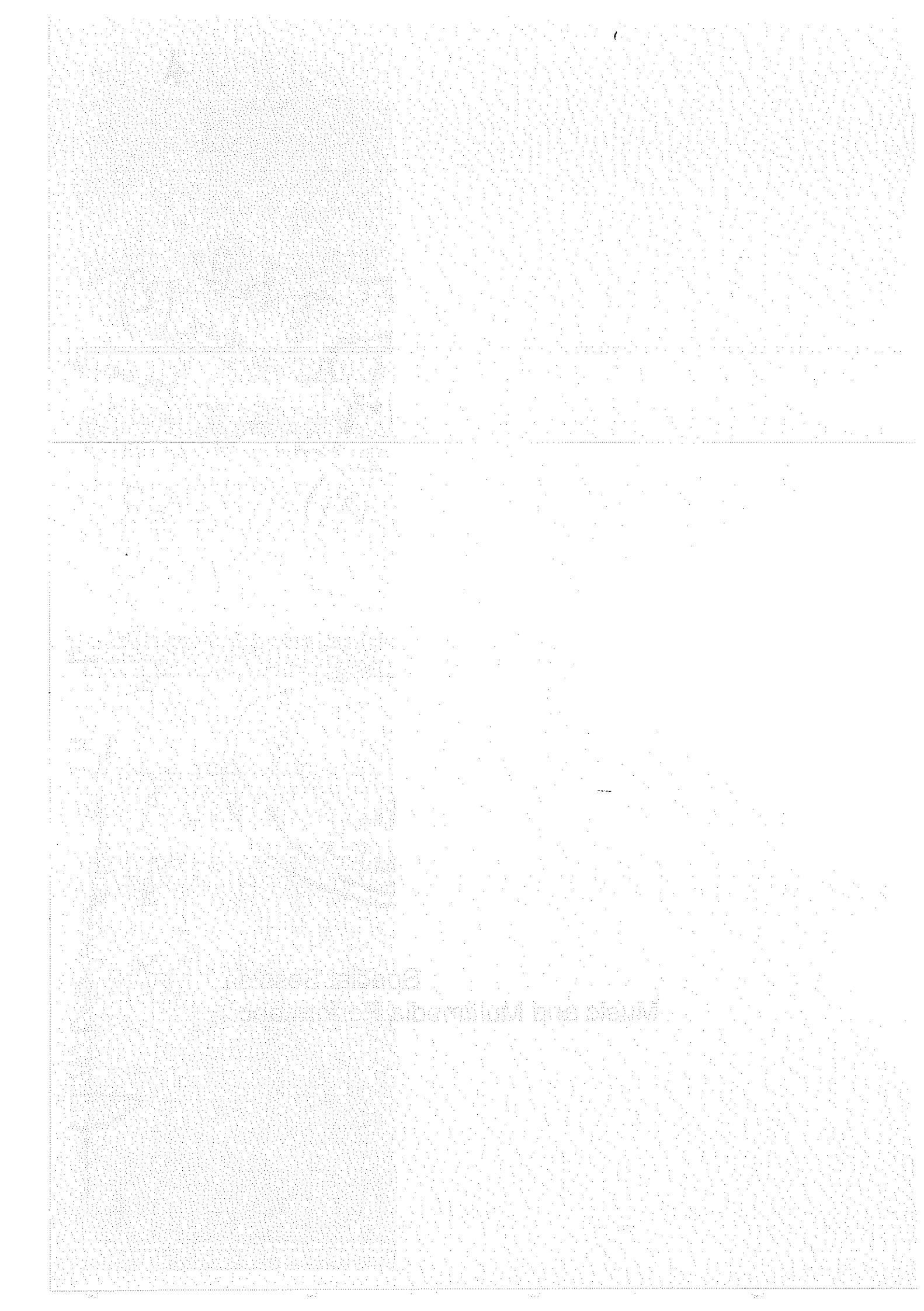
This work has been developed at C.S.C. D.E.I. in University of Padova during 1995 under a Research Contract with Generalmusic S.p.A.

References:

- [1] H.Suzuki: "Model analysis of a hammer-string interaction" *Journal of the Acoustical Society of America*, vol. 82 n. 4, pp. 1145-1151, Oct. 1987.
- [2] X.Boutillon: "Model for piano hammer: experimental determination and digital simulation" *Journal of the Acoustical Society of America*, vol. 83 n. 2, pp. 746-754, Feb. 1988.
- [3] D.E.Hall: "Piano string excitation VI: nonlinear modelling" *Journal of the Acoustical Society of America*, vol. 92, n. 1, pp. 95-105, Jul. 1992.
- [4] G.Borin, G.De Poli, A. Sarti: "Sound synthesis by dynamic interaction", in *Readings in Computer-Generated Music*, Los Alamitos (CA), IEEE Computer Society Press, 1992 pp. 139-160.
- [5] J.O.Smith III: "Physical modelling using digital waveguide" *Computer Music Journal*, vol. 16, n. 4, pp. 74-91, Winter 1992.
- [6] D.Rocchesso: Modelli generalizzati di strumenti musicali per la sintesi del suono" *Rivista Italiana di Acustica*, vol. 17, n. 4, pp. 61-71, 1993.
- [7] W.D.Zhu, C.D.Mote Jr.: "Dynamics of the pianoforte string and narrow hammers" *Journal of the Acoustical Society of America*, vol. 94 n. 4, pp. 1999-2007, Oct. 1994.
- [8] A.Stulov: "Hysteretic model of the grand piano hammer felt" *Journal of the Acoustical Society of America*, vol. 97 n. 4, pp. 2577-2585, Apr. 1995.
- [9] S.A.Van Duyne, J.R.Pierce, J.O.Smith: "Travelling wave implementation of a lossless mode-coupling filter and the wave digital hammer" *Proc. ICMC*, Arhus 1994, pp. 411-418.



Special Session Music and Multimedia Performance



Noms des airs di Salvatore Sciarrino

Una musica algoritmica di trasformazione sonora

Nicola Bernardini, Alvise Vidolin

CENTRO TEMPO REALE

Villa Strozzi - Via Pisana 77, I- 50143 Firenze

Tel. ++49 55 717270, Fax: ++49 55 717712

Email: nibc@nexttis.unipd.it, vidolin@dei.unipd.it

Abstract

Noms des airs, a musical composition by Salvatore Sciarrino, can exist only with the sounds produced by another musical work. In the first performance, held at the Cantine del Redi in Montepulciano the 31st of July 1994, the sounds processed by *Noms des airs* where those of *La Favola di Orfeo* (1932), an opera by Alfredo Casella which lasted 40 minutes and was simultaneously performed in the Teatro Poliziano, located not far away from the Cantine. The sounds of the *Orfeo* by Casella were picked up by microphones and carried over to the Cantine on separate channels to provide: a global stereo channel pick-up, the voice of *Orfeo*, the other singers voices, the orchestra and some stage noises. An algorithm-based digital processing system was placed in the Cantine. Such a system was transforming in real time the sounds of Casella in the music of Sciarrino. The public could listen to several sound transformations following a musical path which started from the exact reproduction of Casella's music proceeding gradually towards the original sounds of the opera by Sciarrino. The purpose of this presentation is to describe the main algorithms used providing sound examples, to analyze the design techniques of the performance environment and to discuss implementation and spatial sound distribution issues.

Noms des airs di Salvatore Sciarrino

La musica di *Noms des airs* di Salvatore Sciarrino esiste solo grazie ai suoni di un'altra opera musicale. Nella prima esecuzione che si è tenuta il 31 luglio 1994 alle Cantine del Redi di Montepulciano [1] i suoni elaborati da *Noms des airs* erano quelli de *La Favola di Orfeo* (1932) [2], un'opera lirica di Alfredo Casella della durata di circa 40 m che veniva eseguita in simultanea nel vicino Teatro Poliziano. I suoni dell'*Orfeo* di Casella erano captati da microfoni e trasportati nelle Cantine del Redi in modo da avere, su canali separati, la ripresa globale stereofonica, la voce di *Orfeo*, le voci di altri cantanti, l'orchestra, e i rumori di scena (fig. 1).

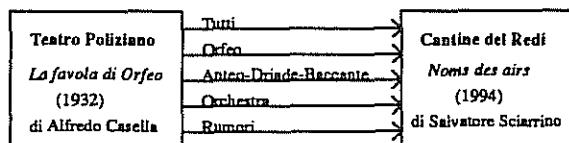


Fig. 1

Nelle cantine era installato un sistema di elaborazione digitale su base algoritmica che trasformava in tempo reale i suoni dell'opera di Casella nella musica di Sciarrino. Le varie

trasformazioni venivano ascoltate dal pubblico lungo un percorso che partiva dalla riproduzione integrale dell'opera di Casella, diffusa nel cortile soprastante le cantine del Redi, per arrivare in modo graduale, scendendo una lunga scalinata, ai suoni originali dell'opera di Sciarrino diffusi da un sistema multiplo di altoparlanti (fig. 2).

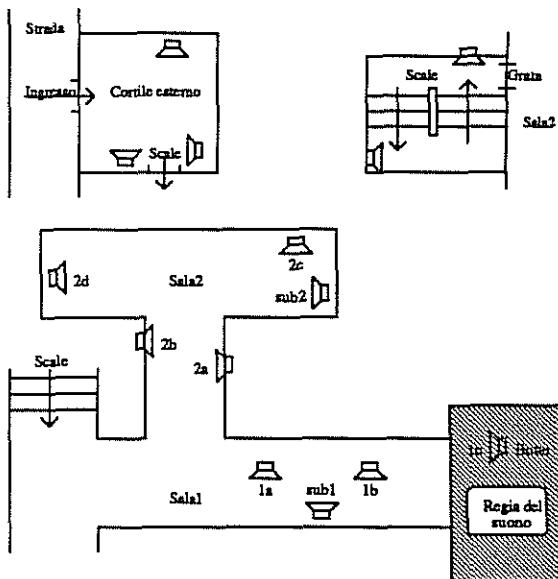


Fig. 2

La discesa nella musica di Sciarrino aveva come cornice il fascino architettonico e storico delle

suoni dell'orchestra. Oltre alla possibilità di attivare globalmente e separatamente i vari sistemi di trattamento, con questo programma si definiscono le caratteristiche delle curve gaussiane e si visualizzano le variazioni d'ampiezza mediante il movimento di tre gruppi di cinque potenziometri a slitta.

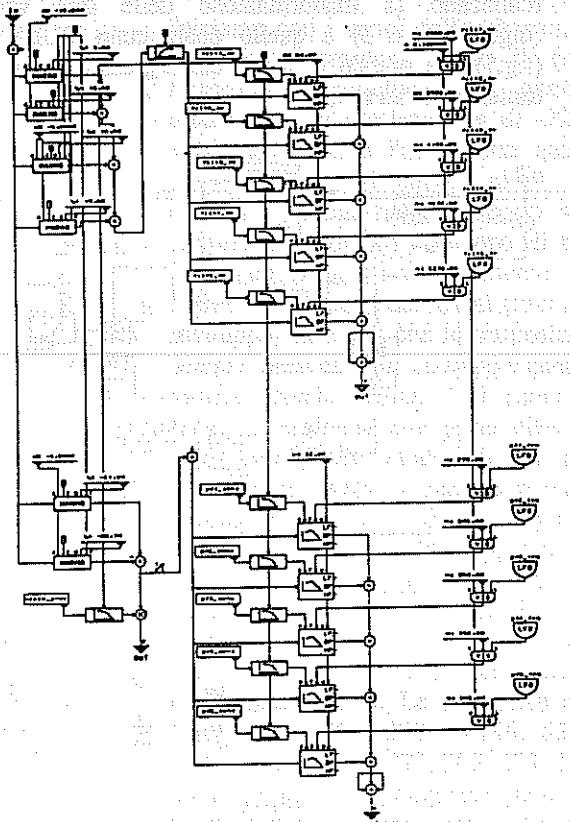


Fig. 6

operatore che si muoveva nelle Cantine e quello che operava al controllo delle apparecchiature.

Riferimenti bibliografici

1 Pirani Federico (a cura di), Catalogo del 19° Cantiere Internazionale d'Arte, Editori del Grifo, Montepulciano, 1994.

2 Alfredo Casella, *La favola di Orfeo*, Edizioni Carish, 1932.

3 Andrenacci, P., E. Favreau, N. Larosa, A. Prestigiacomo, C. Rosati, and S. Sapir. 1992. "MARS: RT20M/EDIT20 - Development Tools and Graphical User Interface for a Sound Generation Board." *Proceedings of the 1992 International Computer Music Conference*. San Francisco: Computer Music Association, pp. 344-347.

4 Puckette M. 1988. "The Pacher." In C. Lischka and J. Fritsch, eds. *Proceedings of the 1988 International Computer Music Conference*. San Francisco: Computer Music Association, pp. 420-425.

5 Opcode, Inc. 1990. *Max Documentation*. Palo Alto, California: Opcode, Inc.

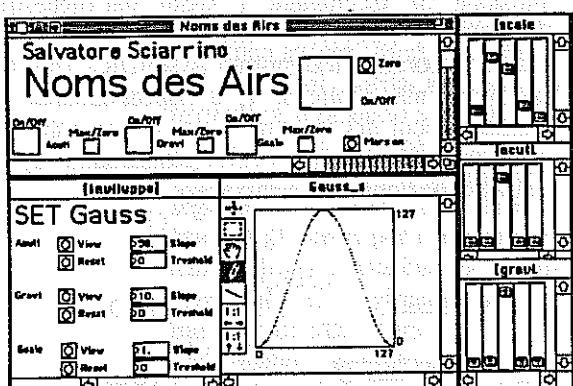


Fig. 7

Durante la rappresentazione gli esecutori al *live electronics* svolgono una funzione di controllo degli algoritmi in azione e di aggiustamento dinamico dei livelli precedentemente fissati in fase di prova. La dislocazione su più piani e in sale diverse delle diverse sorgenti elettroacustiche ha reso necessaria la comunicazione via radio fra un

Il Live Electronics in *Canti d'inquietudine* di Andrea Molino

Alvise Vidolin

CSC - Centro di Sonologia Computazionale dell'Università di Padova
Via S. Francesco, 11 - 35121 Padova
Tel. ++39 49 8273757 - Fax ++39 49 659975
Email: vidolin@dei.unipd.it

Abstract

Canti d'inquietudine è una composizione musicale per ensemble e live electronics di Andrea Molino. La elaborazione digitale del suono acustico è fatta dalla Workstation MARS (fig. 2). Il sistema di elaborazione è composto da sei linee indipendenti. Ciascuna linea è una catena composta da un modulatore di altezza, un generatore di ritardo con retroazione e un controllo stereofonico (fig. 1). I programmi Max sono utilizzati per controllare i parametri sia individualmente che globalmente (fig. 3). La partitura elettronica è composta da 23 parti. Durante la performance il musicista elettronico seleziona la parte da Max e suona un insieme di potenziometri MIDI.

Lo scopo di questa presentazione è di descrivere gli algoritmi principali fornendo esempi sonori, di analizzare le tecniche di progettazione del contesto esecutivo e di discutere di implementazione e di distribuzione spaziale dei suoni.

Introduzione

La caratteristica principale del *Live Electronics* è data dal lavoro diretto che il compositore fa sul suono anziché sul testo. Questo si può spiegare ricordando che tradizionalmente il compositore scrive la musica nella partitura ed utilizza diverse tecniche per lo sviluppo del discorso musicale come ad esempio ripetizioni, variazioni, sviluppo, contrappunto, armonia, figure, sfondi, ecc. Tali tecniche sono appunto applicate nella fase di scrittura della partitura. Con gli strumenti elettronici, invece, è possibile organizzare il discorso musicale agendo direttamente sul suono, come ad esempio ripetendo lo stesso evento sonoro, variandolo mediante una o più trasformazioni, moltiplicandolo o creando eventi ibridi che nascono dall'interazione elettronica fra due o più esecutori tradizionali. La musica che scaturisce dal *Live Electronics* non è meno deterministica dell'altra: è infatti possibile scrivere una "partitura" altrettanto rigorosa con la differenza che viene notato il *processo* di elaborazione che si deve attuare sui suoni anziché il risultato acustico che si deve ottenere [1].

Canti d'inquietudine

Questa premessa era necessaria per introdurre la parte di *Live Electronics* che completa la partitura di *Canti d'inquietudine* [2] e che riguarda i suoni che non si trovano esplicitati nello spartito ma che nascono dal trattamento di alcune sezioni musicali suonate dagli strumenti tradizionali. In questo caso la parte elettronica non viene utilizzata per generare suoni di sintesi, bensì per manipolare i

suoni dell'*ensemble* e utilizza prevalentemente due tecniche di elaborazione: la trasposizione di altezza e la traslazione nel tempo.

Il sistema di *Live Electronics* progettato per quest'opera è costituito da sei linee di trattamento indipendenti ciascuna delle quali è composta da un traspositore, da una linea di ritardo con retroazione e dal controllo stereofonico di uscita (fig. 1).

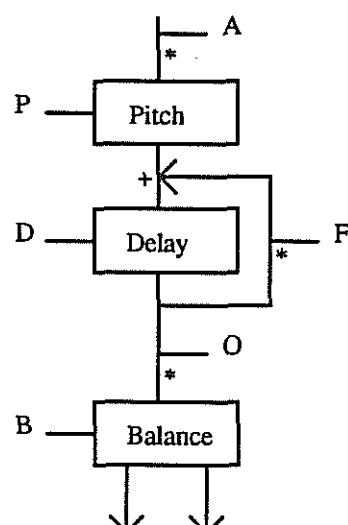


Fig. 1

La linea di ritardo non è altro che un generatore di echi con tempi di ritardo anche molto lunghi: nel nostro caso la durata massima è di 13". Inoltre, è possibile graduare il livello e la quantità di ripetizioni dell'eco che al limite può essere iterato all'infinito. Pertanto se i tempi di ritardo sono brevi

le sei linee di trattamento simulano i primi echi che in natura si verificano nelle sale dovuti alle riflessioni dei suoni sulle pareti: il risultato acustico, quindi, è una sorta di riverberazione. Se invece i tempi di ritardo sono superiori al decimo di secondo gli echi sono distinti e il risultato varia molto in dipendenza dal tipo di materiale sonoro impiegato: se i suoni sono tenuti si ha un prolungamento che può ricordare per analogia l'effetto del pedale di pianoforte; se i suoni sono percussivi si ottiene una moltiplicazione degli eventi. La presenza delle trasposizioni in altezza, poi, accentua maggiormente l'effetto moltiplicativo dilatando su vari registri i suoni originari.

Ambiente esecutivo

Per poter eseguire dal vivo i diversi tipi di trattamento previsti dal compositore è stato sviluppato presso il Centro di Sonologia Computazionale (CSC) dell'Università di Padova un ambiente esecutivo basato sulla Workstation MARS [3] dell'IRIS (Gruppo Bontempi-Farfisa) (fig. 2) controllata in tempo reale da un software originale sviluppato in ambiente MAX [4, 5] su Macintosh.

In fig. 3 è riportato l'algoritmo di una linea di elaborazione dei suoni che è stato poi replicato sei volte per ottenere il sistema di trattamento voluto. Le trasformazioni previste nel *Live Electronics* di *Canti d'inquietudine*, non sono vincolate all'utilizzazione della Workstation MARS, ma si possono realizzare anche utilizzando un sistema di apparecchiature discrete facilmente reperibili in commercio. Tali apparecchiature sono un *Pitch*

Shifter (detto anche *Harmonizer*) un *Delay* con *feedback* e un regolatore automatico di livello. Tali dispositivi devono essere controllabili via MIDI per utilizzare il suddetto *software* realizzato con MAX.

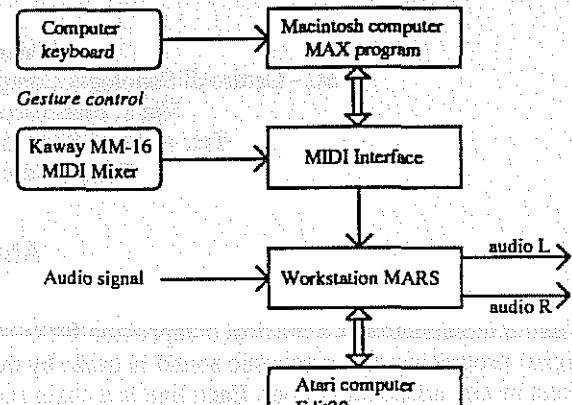


Fig. 2

La videata principale dell'ambiente esecutivo per il *Live Electronics* di *Canti d'inquietudine* è illustrata in fig. 4. Nella finestra *Score* vengono fissati i valori dei singoli parametri che possono essere memorizzati su file. Ciascun file può contenere otto configurazioni diverse. La finestra *Results* mostra i tempi di ritardo effettivi dedotti in base alla azione della deviazione aleatoria. La finestra *Monitor* mostra il valore dei parametri che viene inviato via MIDI ai processori di trattamento in seguito alla azione sui potenziometri.

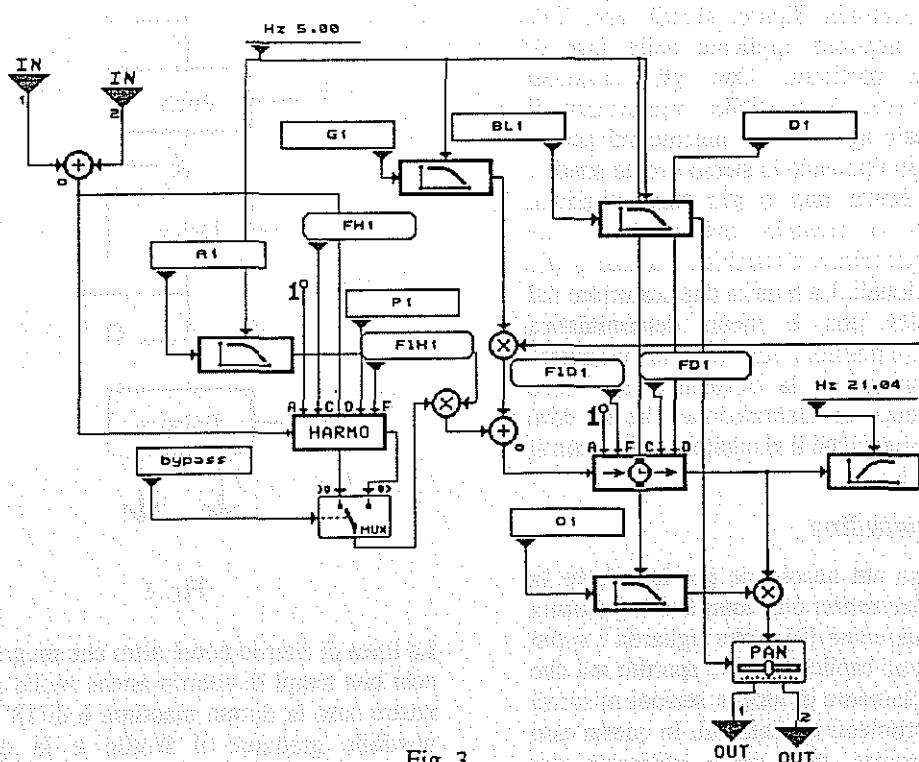


Fig. 3

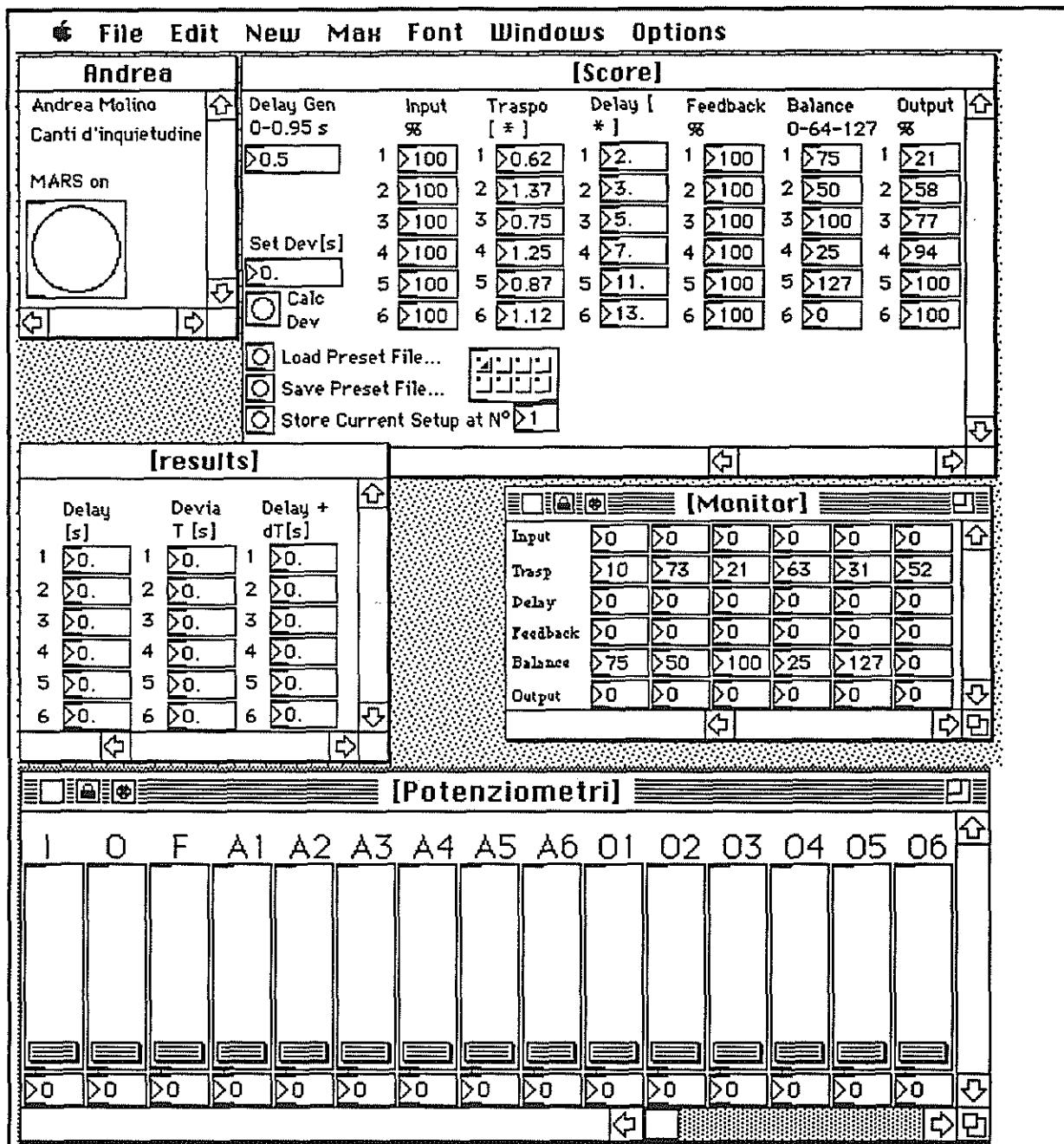


Fig. 4

Significato dei potenziometri. I: Input generale; O: Output generale; F: Feedback generale; A1-A6: input singolo; O1-O6: output singolo.

Tale ambiente consente di definire e variare sia alcuni parametri globali di tutto il sistema sia il dettaglio di ciascuna linea di trasformazione. In particolare i parametri di ciascuna linea sono: ampiezza di ingresso e di uscita, fattore di trasposizione, fattore di ritardo, guadagno di retroazione per fissare l'ampiezza e il numero delle ripetizioni, bilanciamento stereofonico. I parametri globali sono l'ampiezza di ingresso e di uscita, il tempo di ritardo e il fattore di deviazione aleatoria

dei tempi di ritardo. Pertanto il tempo di ritardo di ciascuna linea dipende da tre fattori: il valore globale, il fattore moltiplicativo e il tasso di deviazione aleatoria.

Partitura del live electronics

La partitura del *Live Electronics* di *Canti d'inquietudine* è composta da ventitré configurazioni diverse dei parametri sopra elencati, che corrispondono ad altrettante manipolazioni dei suoni originali. L'ambiente esecutivo è stato progettato e realizzato per consentire il passaggio istantaneo da un

configurazione ad un'altra in modo da poter soddisfare a qualsiasi esigenza di cambiamento sia durante le prove che nel corso della esecuzione. Inoltre alcuni parametri sono stati associati a dei controlli gestuali per consentire un'immediatezza esecutiva e una taratura più fine dei valori in base alla verifica d'ascolto. Tali parametri sono l'ampiezza globale di ingresso e di uscita, il guadagno globale di retroazione e i singoli livelli di ampiezza di ingresso e di uscita per ciascuna linea.

Le diverse configurazioni hanno degli elementi comuni che danno unitarietà ai differenti processi di trasformazione e di fatto sono delle precise scelte composite. Gli elementi più rilevanti riguardano la struttura dei tempi di ritardo, che segue la serie dei numeri primi (2, 3, 5, 7, 11, 13), e la struttura della collocazione spaziale degli echi nel fronte stereofonico (75, 50, 100, 25, 127, 0; essendo Left = 0 e Right=127). La struttura delle trasposizioni, pur avendo precisi legami armonici con le altre parti dell'opera, varia continuamente da una configurazione all'altra, come pure la presenza o il valore di retroazione.

La sperimentazione e la taratura dei parametri del *Live Electronics* di *Canti d'inquietudine* sono state realizzate al Centro di Sonologia Computazionale dell'Università di Padova da Andra Molino, Alessandro Pianu e Alvise Vidolin utilizzando una registrazione delle parti strumentali soggette al trattamento effettuata dal Gruppo Musica Insieme di Cremona diretto dall'autore. Per ciascuna sezione si è operato in tempo reale come se si stesse suonando dal vivo integrando lo schema con alcune note verbali sulla prassi esecutiva.

Francisco: Computer Music Association, pp. 420-425.

5 Opcode, Inc. 1990. Max Documentation. Palo Alto, California: Opcode, Inc.

Bibliografia

- 1 Vidolin A., "Problematiche e prospettive dell'esecuzione musicale con il mezzo elettronico", in *Suono e Cultura. CERM - Materiali di ricerca 1990-92*, a cura di Roberto Favaro, Quaderni di M/R 31, pp. 145-166, Mucchi ed., Modena 1994.
- 2 Molino A., *Canti d'inquietudine*, Ed. Ricordi, Milano, 1994.
- 3 Andrenacci, P., E. Favreau, N. Larosa, A. Prestigiacomo, C. Rosati, and S. Sapir. 1992. "MARS: RT20M/EDIT20 - Development Tools and Graphical User Interface for a Sound Generation Board." Proceedings of the 1992 International Computer Music Conference. San Francisco: Computer Music Association, pp. 344-347.
- 4 Puckette M. 1988. "The Pacher." In C. Lischka and J. Fritsch, eds. Proceedings of the 1988 International Computer Music Conference. San

**Software per la definizione di strumenti nella musica
elettronica:analisi e realizzazione di *Mikrophonie I* e *Solo* di
Karlheinz Stockhausen**

Simonetta Sargentì

MM&T,Musica Musicisti e Tecnologie,via Aosta 2, Milano,tel
33602627,fax 33105679

Abstract

This article describes a new realisation of *Solo* and *Mikrophonie I* by Karlheinz Stockhausen , with the aid of the workstation M.A.R.S. An analysis of the two pieces shows at first their main structural and musical characters ,the points in common and the differences between them. Follows a reflection on the characteristics and the functions of the electroacoustics media in the two pieces joined together by practice of live electronics and by a selection of alternative runs to choice of the performers . The last point of this paper explains our realisation of these compositions with the aid of the workstation M.A.R.S. using algorithms instead analog filters and tapes and shows the capability of recreating historic electroacoustic compositions with new technological supports.

1 Struttura formale e procedimenti compositivi

Mikrophonie I e *Solo* furono composte da Stockhausen tra il 1964 e il 1966 , anni nei quali egli fu particolarmente coinvolto dalla ricerca sulle possibilità di ottenere trasformazioni sonore dal vivo,utilizzando strumenti elettroacustici. (1) I due lavori hanno in comune, oltre all' impiego delle apparecchiature elettroacustiche per ottenere trasformazioni timbriche ed effetti dal vivo, anche alcune importanti caratteristiche formali e strutturali .

1.1 Analisi di *Mikrophonie I*

L'analisi di *Mikrophonie I* è stata condotta sulla traccia delle dettagliate indicazioni fornite dall'autore nelle pagine introduttive della partitura e nei testi esplicativi da lui stesso raccolti e pubblicati (v.note 1 e 2), ed in parte integrando tali indicazioni con un metodo di tipo percettivo, verificando se all'ascolto, le relazioni tra i vari episodi di cui il brano si compone ed il carattere di questi ultimi corrispondessero realmente alle indicazioni date. Il pezzo si presenta infatti nello stesso tempo come assai definito ed articolato nella sua struttura formale e nei rapporti tra i vari

momenti di questa e come sistema "aperto" per quanto si riferisce alla scelta dei parametri di cui comunque il timbro appare essere l'obiettivo principale, insieme con la ricerca di unire la possibilità del rigore formale con la libertà delle scelte esecutive.

Mikrophonie I, prevede un organico di due gruppi di tre esecutori ciascuno: un percussionista, un esecutore al microfono ed un esecutore al filtro e potenziometro. I due gruppi suonano alternatamente ,tranne che in tre punti del brano, nei quali si trovano insieme. Il pezzo si articola in 33 episodi diversi , collegabili tra loro secondo una successione scelta dagli esecutori stessi, tenendo conto pero' delle relazioni tra i vari episodi indicate dall'autore che si riassumono in tre possibili rapporti: somiglianza, differenza e contrasto. Questi si intendono riferiti al carattere percettivo dell'intero frammento in relazione a quello immediatamente precedente e a quello successivo, in rapporto quindi a tutti i parametri musicali di altezza ,durata e timbro nel loro complesso. Tale aspetto del lavoro mette in evidenza il problema della diversa percezione di elementi isolati o inseriti in un contesto e della differente percezione degli stessi in relazione a contesti diversi. e si ricollega all'idea di originare eventi sonori-oggetto all'interno dei quali è difficile separare la percezione dei singoli parametri.

Se la possibilità di scegliere un percorso sempre diverso in ogni nuova esecuzione, conferisce carattere in parte aleatorio a questa composizione, d'altro canto, proprio la necessità di determinare delle scelte percettive da parte degli esecutori, crea l'esigenza di stabilire alcune regole di concatenazione che l'autore indica in modo da rendere comunque coerente la struttura formale del pezzo . Egli propone uno schema di collegamento tra i vari momenti della composizione da cui si desumono con precisione le relazioni percettive che devono derivare dall'esecuzione.(2) I segni posti inferiormente allo schema , definiscono le relazioni fra le varie strutture, l'ultima delle quali deve essere scelta come punto di partenza dell'intera esecuzione.Si comprende quindi come il risultato di quest'ultima possa variare in rapporto alla scelta dell'episodio prescelto per l'inizio e per la conclusione del brano. Le relazioni tra i frammenti indicate nello schema ,vanno poi

osservate obbligatoriamente. All'interno di tale "macrostruttura" rigidamente definita, gli esecutori operano liberamente alcune scelte: 1) i tre momenti nei quali i due gruppi di esecutori si troveranno a suonare insieme e poiché questi tre frammenti non hanno indicazioni di rapporti con i precedenti e i successivi, possono essere inseriti in qualsiasi punto, sempre in modo da conferire equilibrio all'insieme del brano.

2) la determinazione della successione delle varie entrate di ciascun gruppo; i possibili attacchi sono indicati in partitura da frecce: nei punti in cui si trovano le frecce è possibile per il gruppo che in quel momento era in silenzio di attaccare. Questo costituisce un ulteriore elemento di diversificazione, in quanto la percezione del contesto varierà a seconda del punto in cui il gruppo successivo interverrà sul precedente ed in base alla possibilità che, mentre un gruppo attacca, quello che suonava in precedenza abbia già terminato il suo frammento, decida di terminarlo comunque o continui a sua volta a suonare fino ad averlo terminato.

3) la definizione delle durate, la cui unità di misura una volta prescelta, rimane poi costante per tutta la composizione.

4) la scelta dei materiali per produrre il suono. L'unico strumento acustico previsto dall'organico del pezzo è il tamtam, messo in vibrazione con materiali diversi. Indicazioni generali sugli oggetti da utilizzare sono date dal compositore in partitura dove viene segnato anche il modo in cui tali oggetti vanno impiegati. Rimane tuttavia un'ambito di libertà interpretativa sugli oggetti da usare e quindi una ulteriore possibilità di intervento creativo da parte degli esecutori; le indicazioni parlano di risuonatori piccoli o grandi, di carta o di vetro, di corde o tubi etc. Nella nostra realizzazione per esempio, abbiamo proceduto stabilendo per ogni sonorità richiesta una serie di oggetti e di materiali per ciascun effetto da produrre ed abbiamo poi ristretto il campo in base al risultato percettivo fornito da ciascun oggetto, arrivando a scegliere quello che sembrava avvicinarsi maggiormente al risultato richiesto. Sono stati così utilizzati come risuonatori recipienti di cocci, insalatiere di plastica, barattoli di varie dimensioni, tubi di carta, forchette, barattoli del caffè e così via. *Mikrophonie I* è quindi anche, indipendentemente dalla parte elettronica comunque integrante della composizione, un lavoro che si avvicina per alcuni aspetti anche all'idea della musica concreta. L'uso dei diversi materiali infatti, il cui effetto viene poi esaltato dall'intervento dei microfoni, dà luogo a veri e propri "oggetti sonori". Tale affermazione è anche confermata dalle indicazioni che troviamo nella partitura, dove parole quali "gracidare", "crepitare", "stridere", "strombettare" etc. sono immagini verbali di timbri-oggetto da produrre e sui quali gli esecutori devono trovare un accordo

percettivo per poter scegliere e definire i materiali e le modalità di esecuzione. Questa osservazione ci rimanda alle diverse modalità di scrittura utilizzate dal compositore per questa partitura: a) una scrittura tradizionale per le durate, le dinamiche e la parte di percussioni, b) indicazioni esecutive di movimenti, posizioni ed orientamenti per i microfonisti, c) una scrittura per la gestione dei filtri. Ognuna delle tre parti della percussione, del microfono e del filtro, è suddivisa in tre ambiti di altezze: grave, media e acuta, all'interno dei quali si agisce nei modi indicati per produrre i diversi effetti.

1.2 Analisi di Solo

Anche per *Solo*, l'analisi è stata svolta in gran parte sulle indicazioni del compositore e si è potuto rilevare che l'obiettivo principale anche di questo lavoro è dato dai rapporti complessivi fra i parametri scelti liberamente all'interno di una struttura rigorosamente definita.

Solo ha in comune con *Mikrophonie I*, una struttura formale in parte determinata ed in parte lasciata alla discrezione dell'unico esecutore strumentale, affiancato da tre assistenti per la realizzazione elettronica e la regia del suono. L'esecutore ha a sua disposizione sei pagine di musica da eseguire secondo un suo criterio, senza l'obbligo di una successione predeterminata. Egli può persino ritagliarsi alcuni frammenti della parte e ridisporli a suo piacimento, rispettando poi le indicazioni per le correlazioni dei vari momenti così ottenuti. Dopo aver scelto ed eventualmente "ricomposto" la parte strumentale, egli deve fare riferimento ad uno dei sei schemi interpretativi proposti per l'esecuzione, che prevedono i parametri da rispettare, oltre ai criteri esecutivi per gli interpreti della parte elettronica.

Anche in *Solo*, come in *Mikrophonie I*, devono essere considerate le relazioni di somiglianza, differenza, crescendo o decrescendo di tensione esecutiva, con riferimento per ogni frammento a quello immediatamente precedente e al successivo. Queste relazioni sono appunto indicate in ciascuno schema al di sotto della parte esecutiva degli assistenti alla regia del suono.(3) In *Solo* il problema della scrittura strumentale è risolto in modo tradizionale, con l'uso del pentagramma e si abbina allo schema operativo prescelto che contiene le indicazioni del compositore riguardo alle relazioni fra i momenti del pezzo ed un grafico che rappresenta la realizzazione della parte elettronica.

2 Funzione dell'elettronica nelle composizioni analizzate

Abbiamo rilevato nell'analisi delle partiture alcune caratteristiche strutturali di *Mikrophonie I* e di *Solo*

in comune ad entrambe le composizioni. Queste non vanno ovviamente disgiunte dalla presenza degli strumenti elettroacustici.. Si è già detto che l'autore era interessato negli anni che videro nascere questi pezzi, all'attuazione di trasformazioni timbriche dal vivo. Con *Mikrophonie I* egli attua un primo esperimento utilizzando una delle apparecchiature elettroacustiche più semplici e di uso più comune: i microfoni, che sono i protagonisti di questo brano, come si può dedurre dal titolo stesso. In questa composizione,i microfoni non vengono utilizzati per la sola amplificazione del segnale ,come solitamente avviene , ma per modificare sostanzialmente il carattere degli eventi prodotti ,creando gli *oggetti sonori* precedentemente citati nell'analisi strutturale. Essendo usati in modo particolare i microfoni non sono posizionati in modo definitivo e in un punto preciso dello spazio , ma sono usati dagli esecutori orientandoli dinamicamente secondo le indicazioni della partitura ed in relazione alle scelte timbriche concordate. Per elaborare ulteriormente questo risultato, Stockhausen utilizza dei filtri analogici regolati da potenziometri che producono bande di larghezza diversa. In tal modo gli esecutori ai microfoni e gli operatori ai filtri ,non si limitano ad agire meccanicamente sui potenziometri,ma diventano veri e propri interpreti del brano interagendo tra loro e con i percussionisti .L'elettronica dal vivo è dunque intesa in questo brano come trasformazione degli eventi sonori e propone quindi una nuova figura di tecnico-interprete-esecutore.

Anche *Solo* si inquadra in una ricerca nel campo dell'elettronica dal vivo che si rivolge ,piuttosto che alla modificaione timbrica e alla creazione di *oggetti sonori*, ad una stratificazione ottenuta sovrapponendo al suono prodotto dallo strumento melodico effetti di ritardo e di feedback. Ogni schema esecutivo del pezzo, si compone di sei sezioni ,a loro volta suddivise in più parti di durata diversa ,in ognuna delle quali è prevista una precisa elaborazione dei suoni eseguiti ,rapportata alle scelte del solista. Nella concezione originaria dell'autore , il suono viene registrato da un assistente addetto al controllo dei microfoni , inviato in un feedback da un secondo assistente ed infine agli altoparlanti da un terzo assistente che controlla le uscite. Le sei versioni prevedono un diverso modo di gestire i tre controlli e conducono quindi a differenti risultati musicali. La combinazione delle scelte del solista con le diverse versioni della elaborazione elettronica danno così luogo a una serie infinita di possibilità esecutive assai diverse fra loro. Anche in questo caso,gli assistenti sono veri e propri interpreti della parte elettroacustica ,dovendo accordarsi con il solista e seguire ,insieme con le loro indicazioni , le scelte musicali operate da quest'ultimo.

3 Realizzazione dei brani con la stazione di lavoro M.A.R.S.

3.1 Mikrophonie I

La realizzazione elettronica originale di *Mikrophonie I* ,prevede l'uso di due microfoni ,uno per ciascun gruppo di tre esecutori e similmente di due filtri analogici regolati da potenziometri . I filtri sono fissati ad un valore minimo di 30 e ad un massimo di 10000 Hz.

La banda così ottenuta è suddivisa dal compositore in tre sezioni: grave ,media e acuta , seguendo la tripartizione dell'intero lavoro *MikrophonieI* e la sua ampiezza è regolata via via dal potenziometro anche in relazione all'ambiente esecutivo.

Nella realizzazione proposta da MM&T con l'ausilio della stazione di lavoro M.A.R.S. si è operata una scelta formale analoga a quella voluta dall'autore, mantenendo lo schema di collegamento da questi utilizzato per la prima esecuzione del lavoro (Bruxelles ,dicembre 1964), si è poi operato via software sulla gestione dei filtri sostituendo i due filtri analogici con altrettanti filtri digitali .I due algoritmi identici tra loro,uno per ciascun gruppo di esecutori, sono stati realizzati con il software EDIT20 impiegato dalla stazione di lavoro M.A.R.S.: il segnale esterno entra nel filtro passabanda ed una volta filtrato, viene inviato alle casse . Durante l'esecuzione i due algoritmi nella modalità *orchestra* sono controllati via MIDI come segue:controllo 1 , segnale d'ingresso, controlli 2 e 3 , regolazione dell'ampiezza di banda, controlli 4 e 5 , uscita sulle casse.(v. fig.1)(4)

3.2 Solo

Per la realizzazione di *Solo* è stato studiato un algoritmo che riproduce un' apparecchiatura relativamente più complessa di quella necessaria all'esecuzione di *MikrophonieI*.

Questa consiste in due microfoni ,uno per ciascun canale e in un registratore a due canali e sei testine stereo .Poichè ogni schema esecutivo prevede sei sezioni di durata diversa, ognuna delle sei testine di riproduzione viene utilizzata per ciascuna sezione; l'intervallo di tempo che intercorre tra la testina di registrazione e quella di riproduzione deve essere quello previsto dallo schema esecutivo prescelto ed è direttamente proporzionale alla lunghezza del nastro La registrazione del segnale è finalizzata all'idea di poter riascoltare dopo un certo intervallo di tempo, quanto è stato precedentemente già eseguito, è dunque pensata per ottenere effetti di ritardo e di feedback.

L'assistente che controlla tale effetto, fa passare il segnale , dove richiesto ,utilizzando dei potenziometri ,in modo che questo possa essere

riascoltato dopo l'intervallo di tempo previsto. La necessità di registrare il segnale e di calcolare la lunghezza del nastro in funzione del tempo di ritardo desiderato è stata risolta con le potenzialità offerte dal software EDIT20. La realizzazione di MM&T è basata sulla traccia del VI schema esecutivo e sostituisce all'apparecchiatura descritta più sopra, un algoritmo di ritardo con feedback. Le durate delle sezioni del VI schema sono rispettivamente di 14,2, di 19, di 45,6, di 34,2, di 25,3 e di 10,6 secondi. L'algoritmo base capta il segnale esterno del microfono, inviandolo su due canali, controllati dal primo assistente; il segnale entra nel ritardo con feedback, controllato dal secondo assistente e viene infine fatto ascoltare sugli altoparlanti, qualora lo schema lo preveda, dal terzo assistente. Nei punti in cui le casse dell'uno o dell'altro canale sono chiuse, si ascolterà solo l'esecutore dal vivo, mentre nei punti in cui lo schema prevede la presenza sia del ritardo che del feedback, o di uno solo di questi effetti, essi si sovrapporranno in uscita, dando luogo ad una maggiore o minore densità sonora. (v. fig. 2)

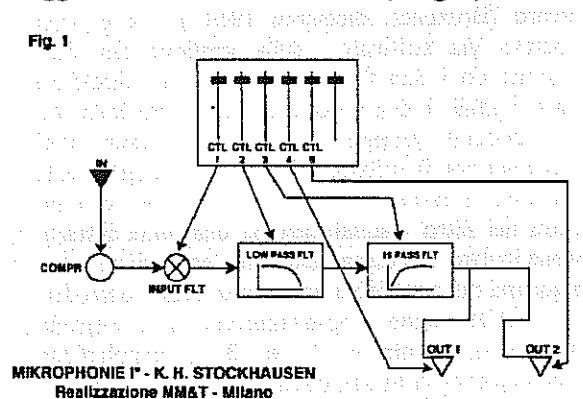
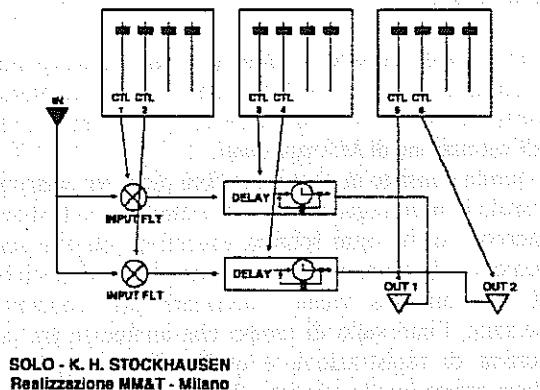


Fig. 1



Poiché l'algoritmo è strutturalmente identico per tutte le sei sezioni dell'esecuzione, è sufficiente modificarne la modalità *tone*, corrispondente al valore dei parametri di durata del ritardo. Nella nostra realizzazione di *Solo* sono stati costruiti quattro algoritmi di uguale struttura, due dei quali possono essere utilizzati rispettivamente per

la sezione A ed E con due *tones* differenti. Il terzo ed il quarto algoritmo, che hanno ritardi della durata di 46 e di 36 secondi, richiedono una quantità di memoria per la quale non è stato possibile creare più *tones* in un singolo algoritmo e sono stati dunque usati due distinti algoritmi. Questi sono stati poi strutturati nella modalità *orchestra* e controllati durante l'esecuzione via MIDI, come per *Mikrophonie I*. Il primo assistente ha a disposizione i controlli per l'apertura dei microfoni, il secondo quelli per la gestione del ritardo con feedback ed il terzo assistente infine, i controlli per la gestione delle uscite sugli altoparlanti.

5 Conclusioni

Dall'analisi del lavoro presentato emergono alcune osservazioni riguardo al senso di una riproposta di opere storiche con l'utilizzo di nuove tecnologie nell'ambito della musica elettronica. L'interesse di tali opere consiste nell'avere dato le prime soluzioni a problemi compositivi ancora oggi oggetto di ricerca. Il riproporle si caratterizza poi in modo nuovo, per l'utilizzo di mezzi diversi e assai più agili al fine di ottenere gli stessi risultati. Si tratta perciò di un lavoro che rimanda ad una sintesi tra recente passato e presente che da un lato ci consente di approfondire il passato e il modo in cui vennero per la prima volta ideate soluzioni allora del tutto nuove nel settore della musica elettronica; dall'altro ci consentono di studiare l'uso di nuovi mezzi che oltre a riprodurre opere già trascorse, si presentano come utili strumenti creativi per la musica di oggi.

References

- 1) K. Stockhausen: "Mikrophonie I" e "Solo", in *Texte fuer Musik* 1963-1970, vol. 3, Koenig, Du Mont, 1971 pp. 57-65 e 85-91.
- 2) K. Stockhausen, nella introduzione alla partitura di *Mikrophonie I*, Wien, Universal Ed., 1974, sono contenute le indicazioni dettagliate per l'esecuzione e lo schema di collegamento dei vari momenti proposto dall'autore..
- 3) K. Stockhausen, nella partitura di *Solo*, Wien, Universal Ed., 1969, come per *Mikrophonie I*, si trovano le indicazioni per l'esecuzione e i possibili schemi da scegliere.
- 4) Nella figura 1, il filtro appare schematicizzato; in realtà, ciascun filtro passabasso e ciascun filtro passaalto sono costituiti non da un solo modulo ma da due moduli in cascata, uno per ciascun filtro, per poter controllare meglio la pendenza del filtro stesso.

MediaSynth

a System for the Synthesis of Multimedia Scores based on Petri Nets

T. Belletti, G. Galizia, G. Haus

LIM - Laboratorio di Informatica Musicale
Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
via Comelico 39
I-20139 Milano (Italia)
fax +39 2 55006373
e-mail: music@imiucca.csi.unimi.it

Abstract

MediaSynth is a software system which allows the user to synthesize MultiMedia Scores (MMS) by means of the implementation of a Petri Net extension, called MultiMedia Petri Nets (MMPN). MediaSynth is an evolution of the ScoreSynth program (MIDI Score Synthesizer) which could process and structure elementary multimedia objects such as video sequences, digital audio, and MIDI sequences, according to the behaviour of user defined MMPN models.

Three user environments are available: an Editor of MMPN models, a Debugger of MMPN models (that executes the PN model and synthesizes the final MMS), and a MediaPlayer for MMS playing.

MediaSynth has the following features: wide functionality and simplicity of the iconic user interface (hierarchical decomposition, parallel processes structuration, synchronism, non-determinism); synthesis of multimedia processes (not only editing); generation of new MMS by means of minimum structural changes; composition of complex transformations by means of atomic instructions; possibility to export the MMS to other platforms.

1. Introduzione

Questo articolo mostra come sia possibile descrivere, elaborare e sintetizzare strutture multimediali tramite un tipo di rappresentazione più flessibile rispetto ai modelli forniti dai principali tool che combinano video, audio e altri media.

L'obiettivo è quello di integrare le conoscenze acquisite nel campo delle reti di Petri Musicali (MPN) con le nuove 'esigenze' multimediali: il tipo di rappresentazione da noi proposto eredita il modello simbolico-strutturale già studiato per le MPN e permette di descrivere ed elaborare esplicitamente ciò che noi chiamiamo oggetti multimediali (siano essi sequenze audio, video, MIDI o combinazione di questi) tramite algoritmi basati su un linguaggio appositamente definito. L'esecuzione del modello permette la sintesi di una *partitura multimediale* (MMS), cioè dello 'spartito' che integra i diversi media.

La nostra ricerca mira a confermare la validità del modello delle reti di Petri ampliandone il campo di applicazione.

2. Le reti di Petri Multimediali

Le reti di Petri Multimediali (MMPN) ereditano *in toto* la teoria delle MPN per la rappresentazione e la sintesi di partiture MIDI [Haus & Sametti, 1992].

La metafora usata nella definizione di una MMPN è ancora quella che assegna ai posti della rete il ruolo di *risorse*, i dati da elaborare, e alle transizioni il ruolo di *gestori delle risorse*. Alle MMPN si giunge, infatti, associando ai posti gli *oggetti multimediali* e alle transizioni *algoritmi di trasformazione* delle sequenze video, audio e MIDI. La marcatura e gli attributi della rete definiscono lo stato del modello che, evolvendosi secondo le regole note dalla teoria delle reti di Petri, genera una *partitura multimediale* (MMS), cioè uno strumento che descrive e sincronizza processi multimediali.

Dall'applicazione delle MMPN emergono i seguenti vantaggi derivanti dalla teoria delle MPN:

- *decomposizione gerarchica*;
- *strutturazione in processi paralleli*;
- *natura asincrona*;
- *non-determinismo*.

3. Gli oggetti multimediali: i MediaObject

La definizione di oggetto multimediale si basa sulla struttura di *filmato digitale* (*movie*) implementato da Quicktime, un'estensione di sistema che rende possibile l'elaborazione e l'editing di sequenze digitali dipendenti dal *tempo*.

Il *movie* è una struttura privata che contiene informazioni riguardanti l'archiviazione dei dati e i riferimenti alle sequenze temporizzate. Come nei filmati cinematografici, un *movie* può contenere più tracce connotate da un tipo di dato ben preciso: video, audio, MIDI ed altri tipi. Tali media presentano caratteristiche diverse (realismo cromatico, compressione, campionamento, ...) che, però, non condizionano il processo di sintesi nel quale gli oggetti multimediali sono riconosciuti secondo una 'interfaccia comune' data dal *nome del movie*, la *durata* e il *numero delle tracce*.

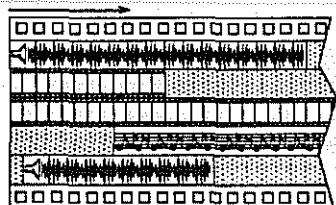


Fig. 1 - Struttura di un movie

La MMPN fa riferimento alla *struttura logica* del *movie*, dove ciascuna traccia ha associato un unico identificatore (*tipo, indice*) che ne indica la posizione nel filmato stesso.

4. MediaSynth

MediaSynth è un ambiente integrato per la progettazione di una rete di Petri Multimediale, la sua messa a punto, la sua esecuzione, la generazione di una partitura multimediale e la sua riproduzione. La filosofia di partenza è quella di ScoreSynth, un modulo della Intelligent Music Workstation, che rappresenta l'ambiente di programmazione delle reti di Petri Musicali.

Gli ambienti di *MediaSynth* sono l'*Editor*, il *Debugger* e il *MediaPlayer*.

Nell'*Editor* viene disegnata la rete in modalità grafica ed interattiva mediante i menu e una paletta di strumenti per le operazioni più frequenti. E' in questo ambiente che si inizializza il modello definendo uno *stato iniziale* dato dalla marcatura, dalla capacità dei posti e dalla molteplicità degli archi. In seguito si associano gli oggetti multimediali ai posti e gli algoritmi alle transizioni.

Nel *Debugger* avviene l'esecuzione della rete secondo la modalità prescelta: passo a passo (*Step*), senza interruzioni (*Execute*), fino alla creazione del prossimo oggetto (*MOS step*) oppure fino ai breakpoint inseriti dall'utente (*Break*). E' possibile seguire l'evoluzione della rete (marcatura) mediante finestre informative che vengono aggiornate in base allo stato di avanzamento.

Nel caso dell'esecuzione di un modello particolarmente lungo e complesso è possibile salvare lo stato attuale per poterne riprendere l'esecuzione in un secondo tempo dal punto in cui si era interrotto, senza dover ricominciare da capo (*Save Model state / Load Model state*).

Il *MediaPlayer* è l'ambiente dove vengono visionate le partiture multimediali (i filmati) generati nel *Debugger*. E' possibile procedere, attraverso i comandi tipici di un video-registratore standard, alla *'proiezione'* del filmato oppure alla *analisi* dello spartito generato per controllarne la sincronizzazione delle tracce o procedere alla attivazione/disattivazione di alcune.

L'utente può reperire in questo ambiente le informazioni riguardanti le sequenze digitali, come la dimensione in byte, il campionamento, il tipo di compressione, la durata, la qualità, etc. e può scegliere fra più soluzioni per una riproduzione 'ad hoc' delle tracce video o migliorare l'equalizzazione delle tracce audio.

5. L'algebra multimediale

L'idea principale è quella di poter trasformare, attraverso algoritmi, i diversi media, variandone le caratteristiche fisiche (*campioni*) o i parametri (*velocità, etc.*). Sarà possibile, per esempio, amplificare una traccia audio, cambiare il colore ad un insieme di fotogrammi, invertire le note di una sequenza MIDI o tagliare tutte le tracce di un *movie*.

Le entità elementari coinvolte nel processo di modifica sono: il *frame* per il video, il *campione* per il suono e la *nota* per il MIDI.

Gli intervalli temporali che specificano il dominio di applicazione degli algoritmi possono essere espressi in una delle seguenti notazioni, comunemente usate nell'editing digitale:

- a) frame (standard SMPTE): *ore:min:sec:frame*
- b) millisecondi: *ore:min:sec.msec*
- c) numero di campioni (audio): *intero positivo*

Tali notazioni possono essere usate indifferentemente all'interno degli operatori. Nel caso del MIDI esiste la possibilità di indicare anche l'*indice* della nota nella sequenza.

Gli *algoritmi* di *MediaSynth* rappresentano i meccanismi di trasformazione degli oggetti della rete. Sono costituiti da una o più funzioni, ognuna delle quali presenta una serie di parametri tra cui:

- il *tipo di traccia*: video, audio o MIDI;
- l'*indice della traccia*: seleziona le tracce interessate;
- l'*operatore*: specifica il tipo di operazione corredata dai suoi parametri;
- il *nome del posto*: che specifica il posto da cui bisogna estrarre l'oggetto, se presente, a cui applicare l'algoritmo.

Un algoritmo corretto viene eseguito con successo istanziando un oggetto multimediale ed evolvendo lo stato della rete, nel caso di errori sintattici o semanticci, l'algoritmo fallisce, senza, però bloccare l'esecuzione della rete.

Distinguiamo due classi di funzioni: *funzioni generiche*, che agiscono sul movie e *funzioni specifiche*, che agiscono su un tipo particolare di traccia e solo su quello.

5.1. Funzioni generiche

Hanno la seguente forma sintattica:

`:: < Operator_G >`

L'operatore definito nel simbolo non terminale ha effetto su tutto il movie, cioè su ciascun tipo di traccia.

Sono operatori di questo tipo Kill (K), Save (S), Multiply (M), Identity (I) e Stretching (T) che facilmente eliminano sottointervalli temporali, moltiplicano, ricopiano e allungano/contraggono sequenze di dati. Sono gli operatori più semplici e quindi hanno tempi di elaborazione brevi.

Un esempio di funzione generica è:

`::S; posto_1 +0:00:02.000`

che produce un oggetto composto dai primi 2 secondi del movie associato a `posto_1`.

5.2. Funzioni video, audio, MIDI

Seguono la forma sintattica:

`< tipo > : [< tracce >] : < operatore >`

Il simbolo non terminale `tipo` conterrà la lettera che identifica il tipo di traccia da elaborare: V (video), A (audio) o M (MIDI).

Il simbolo `tracce` specifica l'indice della traccia del tipo specificato, mentre `operatore` definisce, oltre all'operazione, l'insieme dei parametri necessari.

Facciamo una breve panoramica sulle categorie di operatori.

Per quanto riguarda gli operatori sul video sono stati implementati filtri sul colore (*changeColour*) e sul campionamento dei pixel (*Mosaic*) che modificano le caratteristiche cromatiche; funzioni di ridimensionamento dell'immagine (*Resizing*); funzioni di sovrapposizione di due filmati in varie modalità o di titolazioni (*Overlay*); assemblaggio di due (*Picture In Picture*) o più tracce (*Multiscreen*) con la possibilità di stabilire le regioni di display di ciascuna traccia. Particolare attenzione è stata riservata allo studio della giunzione (*videoJunction*) con cross-fade, cioè la possibilità di creare un effetto di transizione tra un video-clip e un altro. Gli effetti implementati copiano, in gran parte, quelli tipici del montaggio televisivo: *dissolvenza lineare, sipario, veneziana, zoom rettangolare e ovale, random e linear block, tendina orizzontale, verticale o diagonale*.

La classe degli operatori sull'audio è costituita da operatori che permettono di normalizzare un suono (*normalization*), amplificarlo secondo un coefficiente positivo oppure mediante un inviluppo definito dall'utente (*amplification*), produrre un effetto di eco (*echo*), fondere due o più tracce in una sola (*mix*), invertire l'ordine dei campioni di una traccia (*reverse*), distorcere un segnale mediante un altro in modo ciclico o non ciclico (*distortion*).

Anche nel caso dell'audio è stato definito un operatore di giunzione (*audioJunction*) con effetti di fading; durante la costruzione del cross-fade, si può scegliere una *funzione di fade lineare, quadratica, cubica o logaritmica*, definire gli *intervalli di sovrapposizione* o un *intervallo pausa* tra i segnali.

Le operazioni possibili sul MIDI sono quelle ereditate da *ScoreSynth* che permettono di *retrogradare* o *trasporre* sequenze di note, di *invertirne l'ordine*, di *modificare l'altezza, l'intensità e la durata*. Durante tali operazioni la traccia di comandi MIDI viene vista come il pentagramma musicale.

6. L'esempio del 'canone' applicato ai MediaObject

Forniamo un esempio per chiarire i concetti riguardanti la temporizzazione degli oggetti, il non-determinismo, l'uso delle macro e l'applicazione di operatori agli oggetti video, audio e MIDI.

Si parta dal seguente problema pratico.

Durante la fase di montaggio di un filmato si deve decidere con quale modalità assemblare una traccia video, una audio e una MIDI.

Si disegna un modello di rete in cui gli oggetti vengono istanziati con un ritardo, cioè 'a canone'. In Fig. 2a abbiamo la rete *Montaggio*, che ricalca il modello del canone già studiato per le strutture musicali e la rete *Tipo_Oggetto* (Fig 2b).

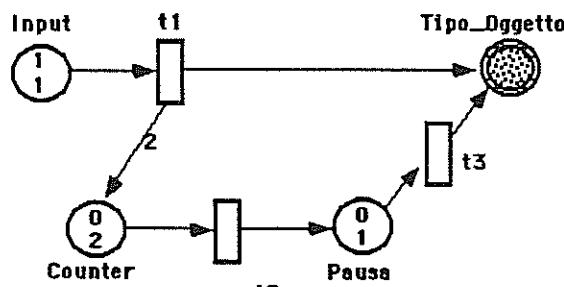


Fig. 2a - Rete Montaggio

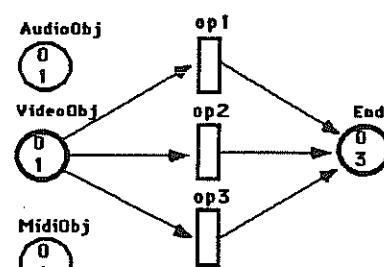


Fig. 2b - Sottorete Tipo_Oggetto

Il posto `Tipo_Oggetto` è dettagliato dalla sottorete omonima, mentre `Input` e `Counter` sono posti che non hanno alcun oggetto associato. Il posto `Pausa` ha associato un oggetto pausa di 2 secondi con l'attributo `PLAY` ed ha lo scopo di generare il canone in partitura.

La rete `Tipo_Oggetto` è la sottorete che viene richiamata dalla rete *Montaggio*: in essa i posti `AudioObj`, `VideoObj` e `MidiObj` hanno associato

un filmato monotraccia del tipo indicato dal movie e della durata rispettivamente di 6, 3 e 8 secondi.

Per parametrizzare la sottorete, abbiamo costruito una *macro*, che definisce il tipo di operatore per ciascuna delle transizioni (*op1*, *op2*, *op3*):

```
(macro associata al posto Tipo_Oggetto  
nella rete Montaggio)  
VideoObj: notplay  
AudioObj: notplay  
MidiObj: notplay  
End:  
op1: play  
      {V:1:CV; VideoObj 0:00:00:00,  
       0:00:06:00, yellow, blue}  
op2: (A:1:T, AudioObj *2)  
op3: (M:1:K; MidiObj -0:00:02.000)
```

Analizziamo i passi dell'esecuzione. Al tempo 0, dopo che sono scattate t_1 e t_2 , una marca è entrata in Pausa e un'altra in Tipo_Oggetto: tale evento ha attivato i processi associati, cioè di scrittura in partitura di una pausa e di un oggetto (Fig. 3).

Quando il processo associato al posto Pausa è terminato, la marca di questo posto è disponibile: viene abilitata la transizione t_3 che istanzia, per la seconda volta, l'oggetto associato a Tipo_Oggetto. Al tempo 2, la marca rimasta in Counter ha fatto scattare t_2 , che inizia il processo associato a Pausa.

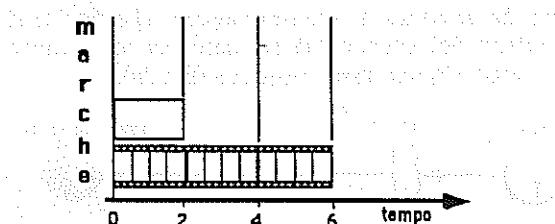


Fig. 3 - Media Object al tempo 0

La sequenza degli scatti si conclude con l'abilitazione, per la terza volta del processo associato a Tipo_Oggetto e il risultato completo è quello di un filmato di 3 tracce disposte 'a canone'.

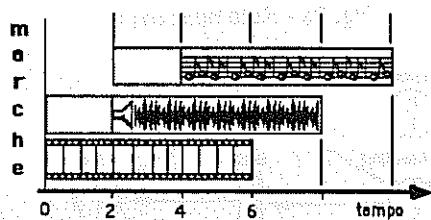


Fig. 4 - Media Object istanziati da Montaggio

Si noti che, nella sottorete *Tipo_Oggetto*, le transizioni *op1*, *op2*, *op3* sono in alternativa e, quindi, l'esecuzione del modello è *non deterministica*; ciò significa che non possiamo fissare la sequenza degli scatti, ma l'esecuzione sceglie una transizione secondo una distribuzione pseudo-casuale e uniforme dell'ordine.

Se si vuole eseguire deterministicamente il modello, bisogna introdurre nuovi posti nella rete (Fig. 5) che eliminano la caratteristica del *non-determinismo*.

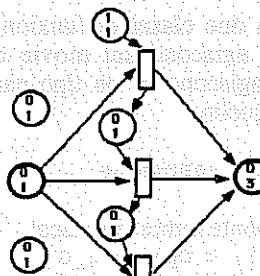


Fig. 5 - Modello di rete deterministico

La flessibilità di modelli basati su reti di Petri risiede, oltre che nei costrutti di base, nella possibilità di generare nuove partiture multimediali a partire da quelle precedentemente disegnate mediante minimi cambiamenti delle strutture o degli attributi.

Ringraziamenti

Un ringraziamento speciale ad Alberto Sametti per il suo ruolo "storico" nel progetto ScoreSynth e per lo sviluppo del sottosistema MIDI in Media Synth.

7. Conclusioni

Appare evidente che un campo di applicazione di MediaSynth è quello della *produzione di filmati*, dove è possibile verificare il modello delle reti di Petri rispetto al classico modo di lavorare dei sequencer multimediali.

Dal confronto di software commerciale per lo sviluppo di opere multimediali con MediaSynth emergono i seguenti vantaggi: maggiore *potenza di calcolo* nell'esecuzione di molte delle operazioni di modifica degli oggetti; estesa funzionalità e semplicità dell'*approccio grafico delle PN*; *sintesi di processi multimediali* (non solo *editing*); generazione di nuove partiture mediante *minimi cambiamenti strutturali*; composizione di trasformazioni complesse mediante *istruzioni elementari*; possibilità di *esportazione* di MMS su altre piattaforme.

Riferimenti bibliografici

[Belletti, 94] T. Belletti: *Un sistema per la sintesi di partiture multimediali mediante modelli basati su Reti di Petri: il sottosistema per i processi audio*, Tesi di Laurea in Scienze dell'Informazione, Università di Milano, 1994.

[Galizia, 94] G. Galizia: *Un sistema per la sintesi di partiture multimediali mediante modelli basati su Reti di Petri: il sottosistema per i processi video*, Tesi di Laurea in Scienze dell'Informazione, Università di Milano, 1994.

[Haus & Sametti, 92] G. Haus, A. Sametti: "ScoreSynth: a System for the Synthesis of Music Scores based on Petri Nets and a Music Algebra", in *Readings in Computer Generated Music*, D. Baggio Ed., pp.53-78, IEEE Computer Society Press, 1992.

[Petri, 76] C. A. Petri: "General Net Theory", *Proceedings of the Joint IBM & Newcastle upon Tyne Seminar on Computer System Design*, 1976.

Le problematiche della creazione di opere multimediali: composizione e performance nelle esperienze realizzate presso il L.I.M. dell'Università degli studi di Milano

Giovanni Cospito, Rosanna Guida

Laboratorio di Informatica Musicale Dipartimento di Scienze dell'Informazione
Università degli studi di Milano
Via Comelico 39, I-20135 Milano (Italia)
fax +39-2-55006373

Abstract

This article is related to the experiences of multimedia performances' design realized at LIM , the laboratory of musical informatics at the University of Milano. It discusses about the characteristics of some works realized and about some aesthetical and historical aspects . Some considerations are made about how to organize and about how to project a multimedia product.

1. Considerazioni introduttive

Il seguente articolo riferisce delle esperienze di produzione di opere multimediali presso il Laboratorio di Informatica Musicale dell' Università degli studi di Milano sotto la direzione scientifica del Prof. Goffredo Haus e la direzione artistica del M° Angelo Paccagnini, in collaborazione con i compositori Giovanni Cospito e Dante Tanzi e, per la computer grafica e il trattamento digitale di immagini video, Rosanna Guida.

Le produzioni hanno permesso di affrontare da punti di vista diversi alcune problematiche legate alla creazione di opere multimediali sia per quanto riguarda le tecniche e le procedure di produzione, sia per quel che riguarda la delineazione di campi di indagine estetica. Il prodotto multimediale non si presenta come un genere rigidamente codificato, ma assume forme molteplici in relazione ai media stessi che vanno a combinarsi ed al modello di fruizione a cui si riferisce.

Parlare di un prodotto multimediale tout court è una generalizzazione riduttiva di un mondo di prodotti che si differenziano in modo sostanziale nella forma e nei contenuti. Questa prima fase di prodotti sperimentali al LIM ha proprio messo in evidenza un insieme ricco di tipologie di lavori multimediali così come ha evidenziato un possibile percorso di sistematizzazione delle conoscenze e delle varie problematiche in questo ambito . Problematiche che vanno dalla ricostruzione critica delle cose che storicamente sono state fatte, alle mutazioni del quadro tecnologico di riferimento, ai problemi della progettazione interdisciplinare, ai luoghi ed ai modelli di fruizione ed, infine, ai vari tentativi di definire teorie estetiche che colgano il contributo specifico di questi prodotti.

2. Le produzioni

I lavori prodotti e la loro possibile classificazione per tipologie sono:

1. Video-Pittura e Musica:

" l'Armonia invisibile è superiore alla visibile"
di R. Guida, D. Tanzi;

2. Video-Concerto: " Impercettibile il Tempo "
di R. Guida, D. Tanzi;

3. Performance Multimediale: " Codici Sincretici "
di G. Cospito, F. Durante, R. Guida,
M. Pennese;

4. Suoni ed immagini sintetiche: " Limite "
di G. Cospito, R. Guida, M. Viel;

**5. Immagini live video trasposte:
" Ombre e riflessi "**
di G. Cospito, R. Guida.

Una Video-Pittura é una manipolazione video digitale di un'immagine pittorica (in questo caso si trattava di un quadro informale). Ciò che ne caratterizza la specificità rispetto ad altre forme di manipolazione dell'immagine é l'aderenza ai temi espressivi della pittura originaria trasposti in chiave dinamica ed inseriti in uno sviluppo temporale attraverso varie tecniche dell'image processing.

Nel Video-Concerto ciò che si osserva sullo schermo viene considerata parte integrante del concerto stesso: così come gli strumentalisti eseguono le parti musicali con il proprio bagaglio di gesti e significati, il video esegue una o più parti di immagini volte a sottolineare ed indicare un senso integrato con l'esecuzione musicale.

Ciò che caratterizza la performance é, in qualche modo, la presenza nell'evento, di una azione teatrale e quindi di uno spazio scenico opportuno. Nel caso di " Codici Sincretici ", si tratta di uno spazio scenico definito da una pittura-partitura, da varie postazioni in cui si muove una cantante-mimo e dall'uso di effetti luci. La cantante, gli strumentalisti e la musica elettronica spazializzata eseguono blocchi della pittura-partitura il cui ordine può essere deciso di volta in volta.

Sull'evento viene costruito un video che rimane l'atto mediale ed artistico ultimo dell'evento.

L'immagine sintetica appartiene al puro mondo dell'immaginario. In "Limite", il mondo immaginato non è direttamente visibile si trova al di là dello schermo e cerca, in vari modi, di sfondare questa sottile pellicola di confine per approdare al reale.

La video trasposizione di immagini e situazioni visive che accadono dal vivo permette, attraverso varie tecniche, di dilatare la percezione di eventi sia in senso spaziale che temporale. Nel caso di "Ombre e riflessi" è la microgestualità del flautista ad essere dilatata nel tempo e nello spazio.

I lavori descritti sono risultati molto efficaci dal punto di vista dell'impatto spettacolare e sono una testimonianza di quanto sia ricco il numero delle possibilità di coniare nuove forme di spettacolo. Restano, per essere discussi, tutta una serie di problemi. Per esempio: riusciranno queste forme ad evolversi distaccandosi dal puro effetto tecnologico oppure si sta creando una sorta di estetica del tecnologico avanzato fatta passare tanto necessaria quanto necessario ed inevitabile viene fatto passare il progresso tecnologico? Quali sono i luoghi, le modalità e le condizioni più adatte per la loro fruizione? Quanto l'idea di interattività inciderà sull'idea di artista e sul modo stesso di concepire e di fare arte?

3. Riferimenti storici e problematiche estetiche

Una breve analisi storica delle produzioni e delle performances multimediali porta ad individuare alcune categorie tecniche nel cui ambito si sono sviluppati i vari lavori.

La Performance multimediale, che, si può dire, esiste da sempre (già nel teatro greco si faceva uso contemporaneamente di macchine e risorse umane) si è oggi arricchita di tutte le tecnologie offerte dall'uso dei computer, dando luogo anche a generi di produzioni diversi a seconda del tipo di strumentazione adottato.

Le produzioni multimediali su CD-ROM inglobano le tecniche di trattamento delle immagini e di elaborazione dei filmati, le tecniche di animazione, le tecniche di strutturazione di percorsi fra testi, immagini e suoni, ma rimangono comunque molto diverse dalle opere che consentono, oltre all'utilizzo di queste tecniche, anche interazioni spaziali in loco o distribuite per il mondo attraverso le reti telematiche.

Un esempio di lavoro che usa soltanto le tecnologie multimediali necessarie alla produzione di CD-ROM è "City Project", un'opera di Matt Mullican, registrata su sei CD-ROM e che consente la visita interattiva delle

varie zone di una città, con l'attivazione di mutamenti di colore opportuni, in modo da simbolizzare l'abolizione delle frontiere geografiche a favore di un impegno sociale globale della città. Un esempio di tutt'altra natura e che estremizza l'uso di tecnologia avanzata coinvolgendo la rete telematica mondiale, è quello di Jean-Marc Philippe, la Roue Celeste, che consiste nel circondare la Terra con una corona di satelliti esistenti e orbitanti e nel modificare la luminosità dei satelliti stessi in modo da fornire un punto di vista metafisico della visione dello spazio.

Stiamo parlando di prodotti artistici per i quali esistono specificità proprie rispetto ai prodotti multimediali commerciali; oltre ai contenuti, risultano chiaramente distinti anche i percorsi costruttivi. Ciò che è ritenuto un criterio di qualità fondamentale dal punto di vista commerciale, può non essere affatto interessante dal punto di vista artistico oppure può essere stravolto nelle sue caratteristiche fondamentali. Ciò che guida l'orientamento delle produzioni artistiche sono le tendenze estetiche e poetiche in cui è immersa l'arte contemporanea. Purtuttavia, nel panorama delle teorie estetiche contemporanee esiste una certa separazione tra le idee che riguardano l'Arte che circola nelle gallerie e nei luoghi tradizionalmente deputati ad ospitare manifestazioni artistiche e le idee che supportano e legittimano le forme espressive legate all'uso di nuovi strumenti.

Al di là del puro esibizionismo tecnologico, fra le teorie estetiche che vengono in aiuto delle nuove realtà artistiche, citiamo l'Estetica Informazionale, introdotta da Abraham Moles, il Proceduralismo ed il Dataismo.

Nell'**Estetica Informazionale** il valore dell'opera d'arte risiede nel contenuto informativo dell'opera stessa. La quantità di informazione fornita deve contenere una parte innovativa ed una parte ridondante affinché essa sia comprensibile e risulti interessante. L'abilità di un artista consiste appunto nel dosare in modo interessante le quantità ridondanti e le quantità innovative.

L'**Estetica Informazionale** è la prima teoria che lascia spazio all'idea dell'informazione anche se per certi versi può risultare limitante.

Con il **Proceduralismo** l'aspetto importante del processo artistico-creativo è nel modo di procedere. L'opera ha valore se è prodotta tramite comandi e strutture di controllo. L'intervento sulle immagini, ad esempio, non è diretto bensì avviene tramite la formulazione di comandi che descrivono il comportamento di un modello concettuale.

La teoria **Dataista**, comunque assorbibile nel Proceduralismo si fonda soprattutto sulla possibilità di riprodurre le opere in copie perfettamente uguali. Ciò porta ad attribuire il valore dell'opera ai dati che la descrivono.

In un'altra direzione si muove invece l'estetica Cyberpunk, erede della Psichedelia degli anni '60, di cui uno dei più interessanti esempi è costituito dal Gruppo "La Fura dels Baus". Durante le loro performances (multimediali appunto) si mescolano primitivismo e cibernetica, riti tribali e mondi ipermediali in un connubio stridente ma pieno di tensione e di pathos.

Ed infine citiamo l'Estetica della New Age, alla quale è possibile assimilare molte delle produzioni video musicali di questi ultimi anni, il cui aspetto più evidente è costituito dall'immersione in ambienti psichici rilassanti, ripetitivi, lenti, con immagini (video e computergrafica) e suoni che invitano, attraverso tecniche che rasentano l'ipnosi, al distacco: una sorta di estetica dell'evasione.

Relativamente alla forma della performance multimediale possiamo brevemente rilevare alcune differenziazioni fondamentali:

- forme autonome non preesistenti,
- forme invasive di tipologie di performance preesistenti,
- forme chiuse,
- forme aperte basate sull'interazione.

4. Organizzazione e problemi della produzione multimediale

Una produzione multimediale richiede due ordini di capacità progettuali; la prima è legata alle specifiche competenze relative ad un dato medium (ad esempio la musica, la ripresa video, etc..) la seconda, di ordine più generale, è legata alla costruzione specifica del prodotto o evento multimediale che implica la definizione di una logica di rapporto efficace fra i vari media. L'organizzazione diventa particolarmente articolata perché oltre a richiedere la partecipazione di numerose competenze fra loro diverse, richiede anche un lavoro di interazione da cui risulti una coerenza intrinseca dell'opera nel suo complesso; ad esempio, il design dell'interfaccia grafica, i ritmi di scansione visiva e le sequenze sonore non possono essere indifferenti fra loro. In questo ambito sono richieste nuove figure professionali capaci di progettare e di articolare con efficacia l'organizzazione della produzione di opere multimediali. Anche in ambito tecnologico la scelta è fra sistemi integrati o sistemi specializzati. E' ovvio che gli apparati tecnologici necessari si moltiplicano in relazione al numero dei media usati per cui una tendenza alla ottimizzazione ha portato sul mercato Workstation dedicate al multimediale e che possono esaurire in se stesse tutto il processo di produzione. Resta comunque evidente il fatto che la qualità del risultato è proporzionale al livello di competenze umane nei singoli settori, per cui, ad esempio, è preferibile affidare il lavoro sulle immagini ad un video maker così

come quello sull'audio a fonici e musicisti che lavorano in propri ambiti tecnologici e, di conseguenza, ciò pone il problema di come strutturare una progettazione collettiva del lavoro.

Una classificazione delle tecnologie hardware e software usate può essere :

- tecniche dell'acquisizione e del trattamento,
- tecniche della sintesi,
- tecniche del montaggio e della sincronizzazione,
- tecniche dell'interazione,
- tecniche di controllo live per performance multimediali.

A proposito di ambienti dedicati al controllo di performance multimediali, presso il LIM è stato prodotto "Multimedia Performer" che è un sistema per la performance multimediale basato su protocollo MIDI.

Bibliografia

M. NADIN, "Emergent Aesthetics - Aesthetic Issues in Computer Arts", in "Leonardo, Computer Art in Context", Supplemental Issue, pg. 43-48, 1989.

J. ROSEBUSH, "The Proceduralist Manifesto", in "Leonardo, Computer Art in Context", Supplemental Issue, pg. 55-56, 1989.

T. DEWITT, "Dataism", in "Leonardo, Computer Art in Context", Supplemental Issue, pg. 57-61, 1989.

A. MOLES, "Art et Ordinateur", 1990, Ed. Blusson, Paris.

F. POPPER, "L'art à l'age électronique", 1993, Ed. Hazan, Paris.

G. CASATI, Un sistema per la interpretazione di partiture multimediali basato su protocollo MIDI, Tesi di laurea, Università degli studi di Milano, Corso di Laurea in Scienze della Informazione, 1992-93.

L. CONCA, Un sistema per la performance multimediale basato su protocollo MIDI, Tesi di laurea, Università degli studi di Milano, Corso di Laurea in Scienze della Informazione, 1992-93.

在這段時間，我會把所有關於我的問題都寫在紙上，然後把紙撕掉，再把紙撕掉，直到我覺得自己已經沒有問題了。

我會想盡辦法去尋找答案，但這些答案永遠不會滿足我。

我會不斷地尋找，直到我發現自己已經沒有問題了。

The HARP/Vscope Interactive Dance Music System

Antonio Camurri, Roberto Chiarvetto, Alberto Massari, Riccardo Rossi

DIST - University of Genova, Laboratorio di Informatica Musicale

Via Opera Pia 13 - 16145 Genova

{music, dumbo, alby, ricky}@dist.unige.it http://MusArt.dist.unige.it

Abstract

This paper describes an interactive system for the real time acquisition of human movement (e.g., dance), for the "intelligent", high-level control of sound, music, visual media, and actuators in general (e.g., mobile robots for on-stage applications like theatre, dance, museums, and entertainment). The sensor system utilized is an evolution of the V-scope system, developed by Lip Ltd. The supervision software is based on the HARP hybrid agent architecture.

1 Introduction

As a part of a general investigation on multimodal interaction in multimedia systems in the framework of the Esprit 8579 MIAMI Basic Research Project, our research is mainly concerned in the study and development of autonomous, multimodal interactive systems (MISs), i.e., systems characterized by multimodal, real-time interaction with user(s), possibly in unstructured and evolving environments. The goal of the research is to develop an integrated system driven/tuned by the movements of the actors or dancers, using specific metaphors for reaching, grasping, turning, pushing, navigating, playing, communicating states or "emotions", etc. The outputs of such a systems can be sound, music, light, image, speech, as well as movements of autonomous systems capable of navigation on stage. Axel Mulder (1994) gives an interesting survey and a classification of a subset of such systems, that he calls *virtual musical instruments*. Well-known examples in this tradition are *hyper-instruments* (Machover and Chung 1989), which, similarly to physical musical instruments, allow performers to map gestures and continuous movement into sounds (e.g., Sawada et al 1995; Vertegaal and Ungvary 1995). Conducting is another research field in which interesting results have been reached (Morita and Otheru 1991). See (Winkler 1995) for another interesting survey and discussion. In another metaphor, *multimodal orchestras* can be applied to choreography, dance, music.

2 Approach

The approach we adopted in this project differs from existing systems mainly for the fact that most of them basically involve simple cause-effects mechanisms between modalities: the metaphor usually adopted is the *musical instrument*. In this case, a fixed, simple mapping between human movement and sound output is defined. In our approach, a greater emphasis is given to the integration of both *hyperinstrument* and *orchestra* behaviours in a system able to detect and to track atomic, local movement features as well as to observe general body

movement features, in a sort of Gestalt approach. In other words, the metaphor is "a human observer" of the dance, where the focus of attention changes dynamically according to the evolution of the dance itself and of the music produced. In our approach, the HARP system (Camurri et al 1991, 1995) is delegated to model these more complex aspects. Simple examples of general movement measures and tracking are "the amount of volume the dancer occupies in a time window", "the 'smoothness' of the movement", the detection of repetitive patterns which characterize a given context, in general qualitative information integrated over time. Correspondingly, the music needs to be represented and manipulated at several levels of abstraction, from the atomic sound parameters to the overall musical structure, the orchestration.

3 System Architecture

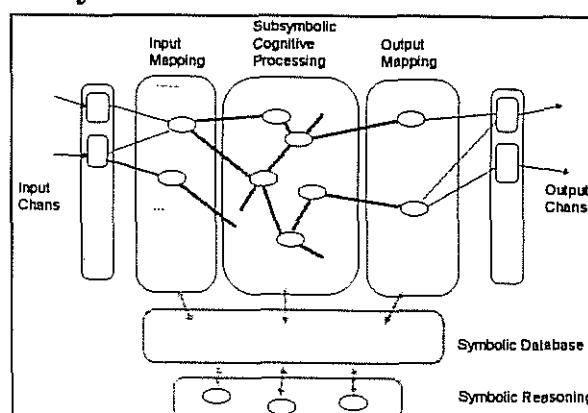


Figure 1. HARP Hybrid Agent Architecture

Following the scheme depicted in figure 1, we can individuate the following basic building blocks that constitute the HARP architecture:

- *Input Mapping*: A group of agents able to receive signals from sensors and map them into perceptual spaces (e.g., a self-organizing map able to classify input trajectories).
- *Output Mapping*: A group of agents that manages the results coming from the cognitive processing

agents, typically high-level parameters (e.g., the space of the “emotional” parameters for controlling the expression of an artificial face or the high-level timbral spaces of a section of a composition).

- *Cognitive Processing*: Two kinds of activity cooperate: (i) *Subsymbolic agents* perform subsymbolic planning and reasoning; (ii) *Symbolic reasoning*, able to perform symbolic inference and planning in the symbolic KB.
- *Symbolic KB*: high-level, symbolic representation of the domain space (e.g., music composition and performance spaces, movement, gesture spaces).

The particular *HARP/Vscope* application is composed of four main subsystems:

- a) human movement acquisition agents, based on the V-scope sensor system, including data pre-processing and filtering;
- b) human movement and gesture recognition agents, including agents based on the force field metaphor;
- c) (cognitive) symbolic agents, for the representation and management of both music and movement spaces;
- d) system output agents: sound, music, computer animation.

An excerpt of the symbolic KB is shown in figure 2, and a corresponding agents software architecture is shown in figure 3.

In figure 2, double arrows represent inheritance (IS-A) links, single arrows are relations including *aggregation (part-of)* and temporal relations; relations can be defined between concepts (classes) represented by ellipses. This language is derived by KL-ONE (Woods and Schmolze 1992). On the left part of figure 2 the situations and the significant parts of the body tracked in the experiment (hands and torso) are represented. The symbolic KB is completed by the agents and icons linked to situations, actions, and their derived subclasses. In the next sections we analyse the network of agents.

3.1 The V-scope interface

The *VScope* agent is designed to acquire information on the position of a number of V-scope markers typically placed on the body of a user. It manages both low-level serial communication and the link with client modules. V-scope is a wireless infrared/ultrasound sensing device developed by Lip Ltd. for the real-time acquisition of the position of up to eight markers placed on the human body (e.g. on the articulatory joints) or in general on moving objects (e.g. a video camera). The hardware is composed of the markers, three tx/rx towers for real-time detection of marker position, and a main processing unit connected via a serial link to a computer. The sampling rate can vary from 5 to several hundreds of milliseconds per marker. As for

the limits imposed by the V-scope acquisition hardware, V-scope works on a stage that can vary from 2 to 5 meters in depth: faster sampling corresponds to a smaller area, due to the limitations of the ultrasound technology. Our experimental results show that 12-15ms per marker is a good tradeoff between speed (a satisfactory value for human movement acquisition) and stage size. We experimented a measure accuracy of the V-scope hardware in the range of ± 0.5 cm, an acceptable magnitude for our applications.

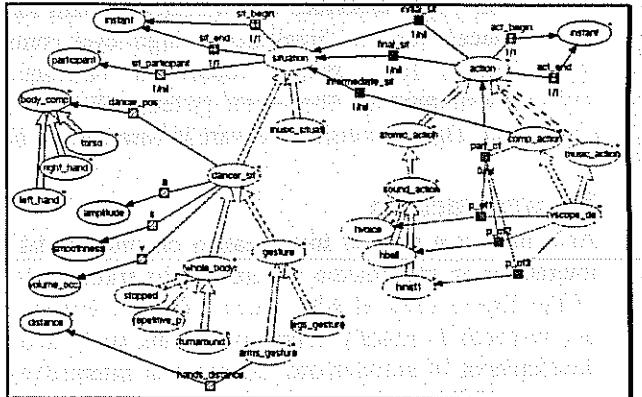


Figure 2

The symbolic knowledge related to the management of the V-scope data is the left part of the KB in figure 2. The icon attached to *vscope_situation* (the *Shared DLL* dynamic link library in figure 3) makes the filtered sensor system data available to agents. Both low-level methods for V-scope hardware control and high-level communication methods are encapsulated in the *Shared DLL* icon. The movement data preprocessing agent filters the raw data from the V-scope icon to ensure no spurious data is present and that values are within a meaningful range.

3.2 Movement and Feature Extraction: Neural Agents and Force Field Metaphor

Gesture recognition is delegated to several concurrent agents, each dedicated to a different kind of recognition task. The Gesture/Movement agents implemented in the experiment here described belong to two different categories: neural agents and force field agents. An interesting category of feature extraction agents is based on self-organising neural networks: the idea is that the movements of (parts of) the body (typically arms and legs) in the sensorized space can be observed and classified in the same manner as it is done in handwriting recognition processes. Analogously to “grapheemes”, i.e., handwriting primitives, we trained nets with simple movement gesture primitives (*gestlets*). In the current implementation, two neural agents have been developed for hands trajectories classification. Several different arm gestures can be recognized. The concurrent raising and lowering of both hands, as well as symmetries and rhythmic patterns are observed by means of a cooperation of both neural agents and symbolic agents, thus integrating local

sensors data (e.g., the distance between hands, relative gesture speed), the output of the neural agents, and the local context information in the symbolic database. Other body gestures (e.g., raising and lowering the body) can be detected, including leg movements. Current experiments regard the classification of full body movements with several degrees of freedom in n-dimensional spaces.

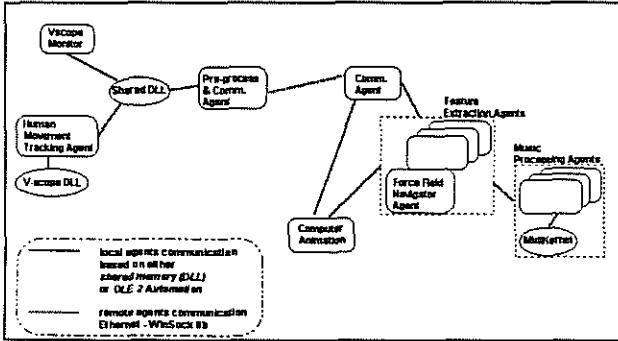


Figure 3

Another interesting category of agents for the interpretation of movement data is based on the force field metaphor: for example, we investigated the mapping of (x,y) coordinates of the dancer into force fields. The force field with three peaks corresponding to areas in the sensorized stage can be seen in figure 4 in the video monitor in foreground. Each area is characterized by a different behavior, i.e. different mappings of movement/sound.

Another group of agents is delegated to extract higher-level whole-body gestures, to model more complex music/movement correlations: examples are "how fast the movement is", "how *in tempo* the dancer is moving", "how he/she occupies the volume", "the *smoothness* of the movement", etc. This kind of information is obtained from integration over time of sensor data. Two different time slice ranges, approximately 0,5-1s and 3-5s, are used. Time slices on which agents operate can vary dynamically, e.g. on the basis of the amount of "quality" of the recognition: for example, a decreased quality in the movement recognition (e.g., different agents return conflicting data) can cause the agents to vary their time granularity and/or their time slice on which they operate.

3.3 Symbolic Agents

In the simplest case, the recognized features can activate agents able to generate music events: often it is useful to "reason" on the input flow, to keep into account what is going on from the recent past in the dance and the music structure, before considering which is the most suitable system output. Symbolic agents are delegated to this "intermediator" role: they are internal reasoning modules, i.e., they operate on the symbolic representations of the movement and music spaces. They cannot directly access the external world. Symbolic agents can be activated by the instantiation of particular assertions of situations. Situations are instanced in the symbolic knowledge

base when something "significant" happens, e.g., a particular gesture has occurred, a movement "style" context is changing, etc.

Therefore, the flow of new instances in the knowledge base can trigger symbolic agents which can cause further modification in the knowledge base, e.g., to force a change of the situation context by changing the music compositional rules to be applied, to modify the roles and the agents parameters corresponding to changes in the sensitivity to movement patterns (a sort of guide mechanism of the focus of attention). In our sample application, symbolic agents implement composition rules for determining the music structure produced by the dancer (See the second demo application described below).

3.4 Output agents

The system is presently designed to control sound and music in real time and particular computer animation. Sound management agents have been implemented. Further output agents regard the control of computer animated human models: most of them reside on a SGI Indigo workstation.

The MIDI sound output agent is a low-level icon which manages the incoming messages (MIDI sequences and other parameters) from other agents through OLE links, and schedules the out queue.

4. System Applications

In the first application, we use three different hyper-instruments, corresponding to different areas of the sensorized stage: These areas/hyper-instruments are mapped in the force field visible in the video monitor in figure 4. Figure 4 shows HARP/V-scope at work with a user moving in such three hyper-instrument areas (the computer screen with the force field window indicating the position of the user can be seen). In this experiment we used three markers - one for either hand and a third for generic body location.

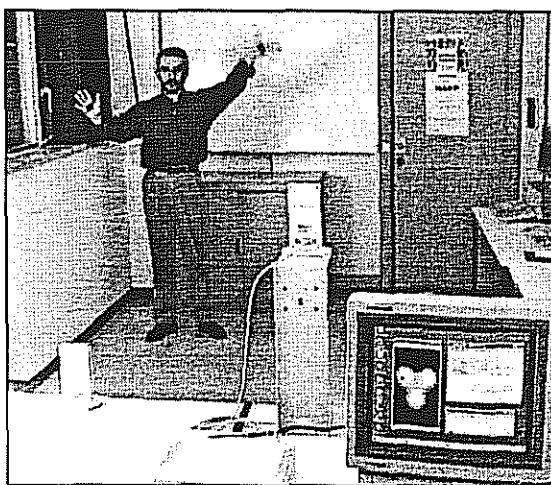


Figure 4.

This last one is used to capture (i) body position in the force field map (x and y coords), and (ii) body height position (z coord), e.g. it establishes whether the dancer is standing or crouched. It is possible to

recognize different hand gestures for each different hyper-instrument. The markers on the hands can be tracked only if the user keeps the hands open. This is used to control the sound output: starting and stopping sound outputs is performed respectively by opening and closing a hand. When the performer is in the center of an area, we obtain maximum presence of that particular instrument, while the other two are absent. As the dancer moves from one peak to another, a cross-fading effect from one instrument to the other is attained: in general, the output of the three instruments is mixed according to the shape of the force field. Thanks to simple symbolic agents, the style of the dancer changes dynamically the reactivity and the behavior of the hyper-instruments, allowing interesting dynamic integration of movement and sound.

The three hyper-instruments (sound synthesis techniques) controlled in real-time in the starting situation in the demo are the following: (i) in the upper left area, a bell toll synthesis is used (EMU Proteus); (ii) in the upper right area, a formants vowel synthesis is used (IRIS MARS); (iii) in the central area, an orchestral string sound synthesis is performed (EMU Proteus).

A second application regards the control and navigation in composition/performance spaces. In this case, force field maps can assume completely new meanings: they represent "composition / performance spaces" which can be navigated. In the application we developed, each area in the map is associated to compositional and performance rules able to control the timbral, harmonic, and melodic output. According to the movement "style" of the dancer, such compositional, cognitive space is navigated. In simple words, the situation evolves as follows: the three areas correspond to (i) a starting area, characterized by simple movement/sound association in a hyper-instrument style, which allows the control of amplitude, pitch, and timbral density of a "naïf", crude sound; (ii) the second area might be roughly defined as a "minuetto"-style composition area: smooth, symmetric, and rhythmic movement patterns cause the navigation toward this area, in which the hyper-instrument gradually transforms itself into a "hyper-composer", where movement contributes to activate compositional and generative rules, reasoning algorithms, able to generate phrases and counterpoint in a sort of minuetto style; (iii) the third area is roughly defined as a "dodecaphony"-style composition area: the navigation to this area is driven by another, different set of movement patterns and gestures. The navigation from an area to another causes gradual, continuous changes in the degree of intervention of the rule contexts, based on the shape of the map: in short, particular navigation percourses covering intermediate positions between two areas can cause different "mixin" of composition / performance rules.

Recently, we used the system in two concerts in Ventimiglia (see figure 5) and Madrid (organized by CEMAT). The system controls live electronics situations and the parameters of a simulated string instrument. More markers have been used, including ankles tracking. The use of the movement to control simulated physical models demonstrated to be a promising area of investigation.



Figure 5. HARP/V-scope at work in the concert "En Voyage", First Intl. Theatre Festival "Al Confine" (July 1995, Ventimiglia), music by G.Palmieri, dancer N.Ragni, fagotto R.Vernizzi, sax F.Aroni Vigone, bass S.Ferrari. The system is here used in a live electronics situation: the dancer intervenes in real time on the output of the acoustic instruments. A V-scope marker is visible on the right hand of the dancer (photo M.M.Pasqualini).

Acknowledgements

This research is partially supported by Esprit MIAMI Project 8579 and by MURST special projects. We thank Giuseppe Gambardella, Carlo Innocenti, and Giuliano Palmieri for useful discussions, the SoundCage staff, and Nunzio Crisa of IBM SEMEA.

References

- A.Camurri, P.Morasso, V.Tagliasco, and R.Zaccaria, "Dance and movement notation", in P.Morasso and V.Tagliasco (Eds.), *Human Movement Understanding*, North Holland, 1986.
- A.Camurri, C.Canepa, M.Frixione, and R.Zaccaria, "HARP: A System for Intelligent Composer's Assistance", *IEEE COMPUTER*, 24(7), July 1991.
- A.Camurri, F.Giuffrida, G.Vercelli, "A system for the real-time control of human models on stage." Proc. X Colloquium on Musical Informatics, 1993.
- A.Camurri, Catorcini, A., Innocenti, C., Massari, A. "Music and Multimedia Knowledge Representation and Reasoning: the HARP System", *Computer Music Journal*, 19(2), MIT Press, 1995.
- H.Morita, S.Otheru, "Computer Music System which Follows a Human Conductor", Proc. Intl. Computer Music Conference ICMC-89, 1989.
- A.Mulder, "Virtual Musical Instruments: Accessing the Sound Synthesis Universe as a Performer", Proc. First Brazilian Symp. on Comp. Music, 14th Annual Congr. Brazilian Comp. Soc., Caxambu, Minas Gerais, 1994.
- H.Sawada, S.Ohkura, S.Hashimoto, "Gesture Analysis Using 3D Acceleration Sensor for Music Control", Proc. ICMC-95, Banff, ICMA, 1995.

Gli amori di Tersicore e Silicio

Elisa Vaccarino

critico e studiosa di danza
imagescript@znort.it
Italia

Abstract

Dance is now able to interact with computer in many different ways: by software aimed at creating and recording choreography, by dataspot to transfer movement from human body to screen, by interface with video and stage décor, by sensors reacting to light and human movement in order to generate sound, by interface with telephone lines to work on teledance, by CDrom, by Internet connection.

1 Introduction

E' indubbio che nel rapporto tra le arti teatrali e le nuove tecnologie si pone oggi una evidente ritrovata centralità del corpo, in senso fisico, psicologico, creativo, estetico, filosofico.

Tra il corpo con i suoi cinque sensi e le macchine, capaci di investirli quasi tutti - manca solo l'odorato, perché l'elettronica non odi - con parole scritte, immagini, parole dette, suoni, rumori, animazioni, c'è una relazione biunivoca strettissima; Paul Virilio, "dromologo", teorico della nuova civiltà superveloce della comunicazione, sostiene che sono le macchine a guardarsi dai loro schermi e che i nostri corpi sono ormai protesi delle macchine, siano esse computer o robot.

Vuoi in questa configurazione estrema, di rovesciamento in chiave cyborg della relazione uomo-mondo, tutta artificiale, tutta simulata, vuoi in termini meno apocalittici, si è rapidamente presa coscienza del rapporto nuovo e diverso che ormai sussiste tra movimenti, suoni, parole e immagini dal vivo e schermi multipli che abitano la nostra casa e le nostre giornate, non soltanto quelli ormai abituali della TV e del video, ma soprattutto quello del computer, che interagisce con il video, che interagisce con il telefono, collegandoci in rete con altri computer senza confini di spazio e di tempo, e che riassume quindi in sè tutti gli altri schermi.

Circa le conseguenze di tutto questo sulla concezione e sulle modalità di uso del corpo nella vita e nel lavoro quotidiano, e tanto più nella pratica performativa, il dibattito, come si suole dire, è più che aperto. Come si posizionerà, in questo quadro la danza? E in quali relazioni con la musica?

Già il cinema e il video hanno profondamente modificato la danza teatrale, dotandola di una fortissima attenzione all'immagine e all'inquadratura, e di metodiche prese a prestito con profitto, dal ralenti al montaggio, senza dire dell'uso frequentatissimo di vere e proprie, composite, colonne sonore.

E prima ancora, in questo secolo segnato fin dall'inizio dalla velocità delle comunicazioni e dalle macchine, i Futuristi intuirono come la prima rivoluzione tecnologica avrebbe ridisegnato il mondo e, in particolare, il mondo dell'arte, musica e danza compresa; basta pensare a "Feu d'artifice" di Igor Stravinsky, che Giacomo Balla rese come caleidoscopio di forme e colori danzanti, al "Ballo meccanico" di Ivo Pannaggi, ai progetti di danze con ballerini-gas, alle aerodanze di Giannina Censi, al balletto "Anihccam 3000" di Fortunato Depero, oggi soggetto di un significativo remake di Lucia Latour.

In quello stesso periodo cruciale, gli anni '20, Oskar Schlemmer, ideatore del notissimo "Balletto triadico" per la Bauhausbühne, crea "La danza del vetro", "La danza del metallo", "La danza dei fili", lavorando per "astrarre" la forma umana.

Astrazioni di forme e colori che anche Alwin Nikolais in America utilizzerà ampiamente in anni più vicini a noi come materiale dei suoi spettacoli, per cui sarà autore anche della musica.

Quali saranno adesso, a fine secolo, in piena rivoluzione elettronica, gli input sulla scena danzata, gestuale e sonora, della realtà virtuale e delle reti telematiche planetarie come Internet, che dislocano metaforicamente il corpo in tempo reale? Essendo quello sopra delineato un terreno mobile, in cui ogni giorno nuove esperienze e nuove applicazioni entrano in campo, l'unica cosa ragionevole che si può tentare di proporre oggi è un viaggio ragionato nel vasto territorio degli intensi rapporti tra performing arts e nuovi strumenti tecnologici, visti nelle diverse articolazioni che li caratterizzano, senza aspettative esoterico-millenaristiche - già altri avevano ipotizzato erroneamente la morte del teatro dopo l'avvento del cinema e la morte di quest'ultimo dopo la diffusione domestica della TV - e senza timori panici, di tipo orwelliano, di fronte alla "disumanizzazione" della cultura e dell'arte. Lasciando da parte, in questa sede, le cruciali questioni della democrazia elettronica, qui si cercherà qualche prima risposta alla domanda: in

quali direzioni gli "attrezzi" elettronici, di cui ci serviamo ormai comunemente, sono agenti di cambiamento a largo raggio rispetto alle arti vive e alla percezione che ne abbiamo?

E per parlare dell'oggi della danza in relazione al computer, ecco alcuni degli scenari in atto: possibilità di scrivere una coreografia al computer; possibilità di far interagire tramite computer corpo, luce e suono, dal vivo o per simulazione in ambiente virtuale; utilizzo di immagini elettroniche per la videodanza e per la scenografia di danza; riproduzione del movimento umano in un simulacro grafico tramite una tuta dotata di sensori, CD Rom con contenuti legati al teatro di danza; teledanza via modem, danza e danze in Internet.

2 Presentation

Cominceremo da "Life Forms", software elaborato da un gruppo di studiosi canadesi, dell'Università Simon Fraser di Vancouver, tra cui Thecla Schiphorst e Tom Calvert, di cui si sta servendo da tempo Merce Cunningham, maestro del postmodern USA, già pioniere del video con Nam June Paik.

"Life Forms" serve a creare il movimento, a memorizzarlo, a insegnarlo, visualizzandolo in 3D, con possibilità di giri, stop, accelerazioni e rallenti, sulla base di una time line di 40 frame per secondo, e non su base musicale. Il che corrisponde alla strategia di lavoro condivisa con John Cage, suo partner creativo d'elezione, di autonoma convivenza tra musica e danza.

Nicole e Norbert Corsino, notissimi autori marsigliesi di circumnavigazioni coreografico-geografiche, hanno girato "Totempol", che incorpora sequenze elaborate con questo stesso programma Life Forms, in parallelo ai movimenti umani ripresi dalla telecamera.

Sempre per disegnare il movimento umano, dalle produzioni presentate al Siggraph USA, arriva "Dance interactif" della New York University Media Lab, con l'impiego di una figura umana "realistica" di cui controllare la velocità e le direzioni di movimento.

Michel Redolfi del Cirm, Centro Internazionale di Ricerca Musicale di Nizza, ha ideato e realizzato con la collaborazione di Luc Martinez "In corpus", cioè un "concerto subacquatico", come "cyber-spazio da vivere"; i nuotatori, nella piscina olimpica di Tolosa, muniti di cuffie da bagno colorate, seguite da telecamere numeriche, comandano la partitura audiovisiva, sempre variabile.

"The Trace, Remote Insinuated Presence" di Rafael Lozano-Hemmer, artista elettronico madrileno, illustra invece, inserendolo in ambiente virtuale, attraverso due stazioni di lavoro interconnesse, il rapporto tra due agenti elettronici a distanza, la luce e il suono.

Una prospettiva che potrebbe somigliare a quella dei videogiochi.

Anche "Surface Tension", ancora di Rafael Lozano Hemmer, è un'applicazione dell'elettronica alla performance dal vivo; un occhio elettronico segue il

movimento di una danzatrice, reagendo ai suoi spostamenti nello spazio.

"Electro Clips" di Christian Möller mostra come può avvenire dal vivo un'interazione elettronica con uso della luce per manipolare il suono da parte del danzatore Stephen Galloway, uno degli interpreti più noti del Frankfurt Ballett di William Forsythe; Galloway incontra luci passive, fisse e mobili, che determinano il silenzio, oppure luci attive, che determinano il suono; il suono è generato da sensori che reagiscono alla luce, alla sua quantità e direzione, ed è trasmesso da altoparlanti.

Si tratta, potremmo dire, di un solo come nella storia ne abbiamo conosciuti altri, in cui il performer crea la propria musica, ma questa volta lo fa con l'intervento del computer. Forsythe stesso, che sta lavorando a un proprio CD Rom ha chiesto ai suoi danzatori, per una sezione di "Alie/NA(c)tion" del '93, di creare i propri movimenti su una griglia di punti nello spazio e di tempi in cui passare dal punto precedente al successivo, elaborata al computer.

Sul supporto CD Rom, che permette di immagazzinare suoni, scritti, immagini, video, e di percorrere i contenuti in modo attivo si è esercitato con un autentico prodotto d'autore Bob Wilson, mettendo in gioco le sue straordinarie attitudini pittoriche e le sue doti di scrittura per ideogrammi/fotogrammi; ne risulta un album personalissimo di appunti visuali, grafici e sonori. Un altro percorso per creare movimenti è possibile attraverso l'uso di tute equipaggiate con sensori che trasmettono gesti e passi da un agente umano a una figura creata al computer in 3D. E' il caso della marionetta Computoon, del MediaLab di Parigi, partner di Canal +; un'applicazione di una metodica simile è stata utilizzata per la pubblicità della Shell, presentata a Imagina di Montecarlo, che fa danzare auto e pompe di benzina, con esiti molto interessanti e divertenti.

Sul fronte della tele-danza, è interessante visionare qualche esempio delle sperimentazioni del gruppo canadese Very Vivid, che utilizza il Mandala Virtual Reality System.

Si tratta di danza in ambiente virtuale, attraverso l'uso di una telecamera che riprende una persona in movimento, Vincent John Vincent, performer, davanti allo sfondo blu tipico anche del chroma-key TV; il che permette di aggiungergli lo sfondo desiderato, spesso registrato da una seconda telecamera; il tutto prende l'aspetto di un concerto rock, con una parte dal vivo e una di strumenti e oggetti virtuali con cui interagire. Il computer, infatti, riconosce i movimenti del performer e fa in modo che "combacino" con gli oggetti che disegna, accordandoli all'immagine digitalizzata del ballerino stesso, che pare suonare gli strumenti mentre danza.

A questo primo performer se ne può aggiungere un secondo, presente dal vivo, magari dall'altra parte dell'oceano, con lo stesso sistema in uso per le teleconferenze, trasmettendone cioè l'immagine via telefono e rinviandola indietro, compressa, accoppiata con quella del partner telematico.

Esperimento effettuato all'ultima Biennale di Charleroi, "Corpo e tecnologie".

Proprio da Charleroi, dalla compagnia residente Plan K, provengono alcune immagini di "Ex machina" di Fabrizio Plessi, artista video, e di Fredric Flamand, coreografo; in questo caso è di scenografia elettronica che si deve parlare. Plessi, tra l'altro, è docente proprio di questa nuova materia alla Kunsthochschule für Medien di Colonia.

Infine, nello specifico danza, qualche esempio di quanto si trova attualmente in rete, la rete per eccellenza, cioè Internet, chiarisce la situazione: dai frammenti video dell'americana Molissa Fenley e di Amanda Miller, attiva a Francoforte, alle Dance Pages encyclopediche sul server Web della lionesse Estelle Suche all'accesso agli archivi della Dance Collection di New York al Dance Server inglese alla www-directory Jahoo ai dialoghi tra utenti su come organizzare una gara di ballo di sala.

References

P. L. Capucci, a cura di: *Il corpo tecnologico*, Baskerville, Bologna 1994.

AA. VV., *La scena immateriale*, a cura di A. Ferraro, G. Montagano, costa & Nolan, Genova 1992.

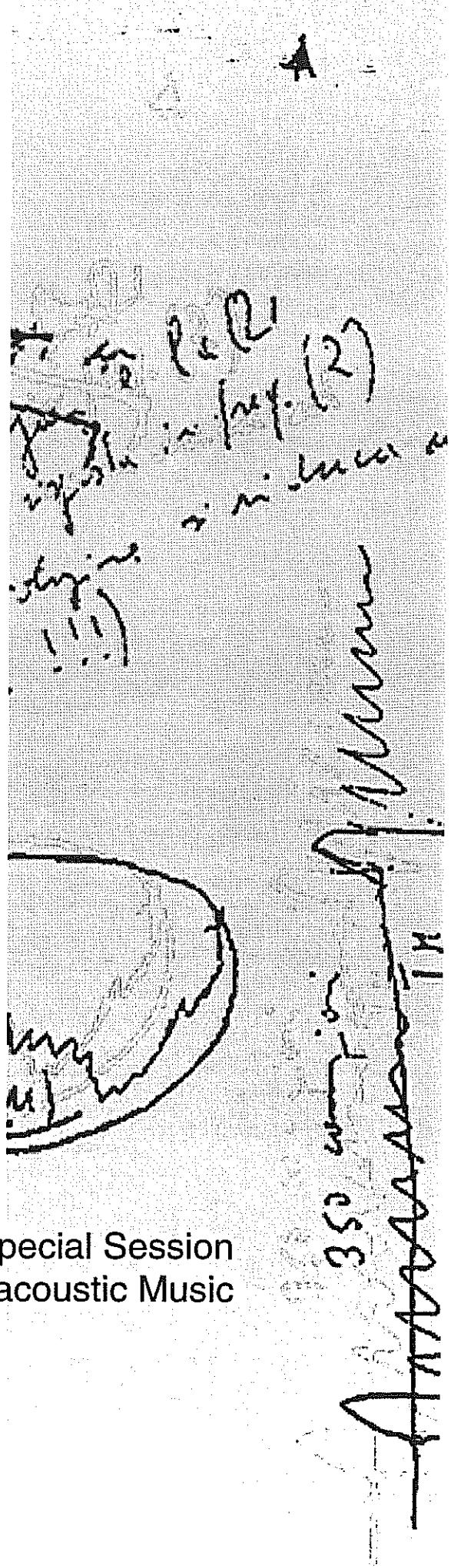
F. Berardi "Bifo", *Mutazione e cyberpunk*, costa e nolan, Genova 1994.

and the company's ability to grow its business through organic growth and acquisitions. The company's ability to execute its strategy will depend on its ability to identify and evaluate opportunities and manage its resources effectively. The company's success will also depend on its ability to maintain strong relationships with its customers, partners, and employees.

The company's management team has extensive experience in the industry and is well-positioned to lead the company through its growth trajectory. The company's board of directors is composed of experienced individuals who provide oversight and guidance to the management team.

Overall, the company appears to be well-positioned for future growth and success.

**Special Session
Radio and Electroacoustic Music**



no place like me
I am the boss of the world - this is older

Un percorso tra musica elettroacustica e radio italiana: 47 anni di *Prix Italia*

Francesco Giomi

Divisione Musicologica CNUCE-C.N.R.
Firenze - Italia

Abstract

The paper examines the relationships between the radio medium and electroacoustic music with a particular attention to musical works expressly composed for radio broadcasting. In 1948 the RAI Italian broadcasting company founded an international competition, called *Prix Italia*, to promote experimental pieces produced for radio communication. Some relevant historical and more recent winner pieces are described, together with the main techniques of the radio studio used by composers to realize them.

La radio est pour ceux qui
ont des images pleine la tête
(Luc Ferrari)

1. Introduzione

Il legame tra la radio e la musica elettroacustica è sempre stato una costante fin dai primi anni delle sperimentazioni musicali contemporanee. Basti ricordare come gli studi di musica elettronica di Colonia e di Parigi siano nati proprio nell'ambito delle rispettive istituzioni radiofoniche nazionali.

Karlheinz Stockhausen esprimeva nel 1959 sulla rivista *Die Reihe* le sue idee circa l'importanza della radio per la musica elettronica ma è interessante notare come già nel 1944 Pierre Schaeffer conduceva le prime sperimentazioni elettroacustico-radiofoniche, componendo il pionieristico programma *La Coquille à planètes* e avviando tutta una serie di ricerche in questo senso. In America poi, se pur con tagli estetici alquanto diversi, si assiste alle esperienze di John Cage: la sua composizione *Imaginary Landscape n. 1* (1939), primo lavoro da eseguire specificamente registrato o per radio e il suo testo del 1942 sugli auspici di una "musica radiofonica sperimentale" apparso sulla rivista *Modern Music*. Non solo dunque l'inserimento dello strumento radiofonico all'interno della composizione ma la radio come obiettivo finale del brano musicale.

Già prima degli anni cinquanta inizia a delinearsi una traccia linguistica specifica che vede, e vedrà sempre più negli anni a seguire, la radio sia come mezzo primario per la diffusione della musica elettroacustica sia come protagonista di un discorso compositivo autonomo: essa diviene oggetto di interesse da parte del compositore che ne sfrutta le tecniche e le specifiche possibilità espressive. Abbiamo oggi la possibilità di mettere in rilievo un patrimonio in molti casi poco conosciuto che trova proprio nella sua specificità radiofonica le ragioni per una sua qualificazione sia in chiave

storica sia in chiave estetica.

2. Il *Prix Italia*

In Italia l'attività si concentra inizialmente intorno agli studi della RAI-Radio Televisione Italiana: da un lato, sin dal 1955, lo Studio di Fonologia di Milano si caratterizza come uno dei punti di riferimento della sperimentazione musicale europea; dall'altro, nel 1948 prende vita il *Prix Italia*, un'iniziativa il cui intento originario risulta quello di premiare e promuovere opere per la comunicazione radiofonica di alto valore poetico ed estetico.

Il *Prix Italia* ha proseguito il suo percorso fino ad oggi arricchendosi via via di nuove categorie (tra cui, per esempio quella per programmi televisivi dal 1957), ponendo all'attenzione generale molte fra le migliori opere radiofoniche sperimentali, e premiando lavori originali di compositori come Berio, Henze, Maderna, Pousseur, Penderecki, Sciarrino, Stockhausen e tanti altri.

Attualmente, il *Prix Italia* annovera come membri più di 60 organizzazioni radio-televisive internazionali di circa 40 paesi.

L'attenzione per i programmi musicali è sempre stata una costante della manifestazione fin dagli inizi, tanto che dal 1953 la competizione è stata suddivisa in diverse categorie tra cui due specificamente musicali (*Prix Italia* e *Premio RAI*) nell'ambito delle quali, fino dalle primissime edizioni, sono state presentate opere musicali in cui era presente una sempre maggiore interazione del tradizionale bagaglio strumentale con l'elemento elettroacustico.

3. Tecniche elettroacustiche per la radio

Un ascolto ed un esame delle note illustrate relative ai

primi brani elettroacustici premiati al *Prix Italia* aiutano a mettere in luce tutta una serie di strumenti e di tecniche impiegate per la loro realizzazione, tecniche e mezzi che risultano prevalentemente quelli tipici dello studio radiofonico e che diventano obiettivo della progettazione compositiva dell'autore. Eccone alcuni esempi:

- giustapposizione di nastri e/o di parti vocali
- manipolazione di nastri (variazione della velocità, riproduzione al contrario, tagli, ecc.)
- uso attivo dei microfoni
- riverberazione
- filtraggio
- uso di oscillatori e generatori di rumore

Anche la stereofonia diventa agli inizi degli anni sessanta un parametro compositivo di cui tenere conto; uno dei primi esempi lo si trova nell'opera *Ondine* del giapponese Akira Miyoshi (premio nel 1960) in cui alle parti orchestrali e per coro femminile si aggiungono una serie di suoni elettronici registrati stereofonicamente. Un altro esempio illustre ci arriva da Stockhausen che, da sempre sensibile al parametro spaziale in musica, propone per il concorso del 1977 il suo *Sirius* (per musica elettronica, tromba, soprano, clarinetto basso e basso) nella cui descrizione il compositore afferma: "Il posizionamento delle fonti foniche nel panorama stereofonico ha una straordinaria importanza nella composizione. La 'geografia' della musica richiede, per una realizzazione veramente radiofonica, un nuovo e diverso impiego della stereofonia". Molto prima di queste affermazioni - nel 1962 - il *Prix Italia* aveva aggiunto una specifica categoria al suo elenco, categoria successivamente unificata con quella monofonica - nel 1971 - in conseguenza della grande diffusione di questa tecnica.

Le tecniche si sono ovviamente arricchite con il progredire del percorso tecnologico-musicale e si sono gradualmente affrancate dallo stretto legame con lo studio di produzione radiofonica. L'elenco dei lavori premiati del 1979 riporta, per esempio, *Pianophonie* di Kazimiers Serocki il quale, descrivendo il suo lavoro musicale aggiunge: "Per la prima volta nella musica radiofonica varie combinazioni di apparecchiature elettroniche sono state memorizzate grazie ad un programma inserito in un micro-elaboratore". Anche se i computer venivano impiegati in musica da più di dieci anni, si tratta comunque di un esempio interessante di applicazione informatica alla musica, il primo tra i vincitori del premio radiofonico.

4. Le opere

Per opere elettroacustiche si intendono, oltre a quelle totalmente realizzate con mezzi elettronici o con organico misto (parte strumentale e parte elettronica), an-

che quelle in cui i mezzi tecnici peculiari alla radio siano stati impiegati per manipolare e rielaborare in senso creativo parti strumentali o recitative.

Le composizioni che vengono qui citate non rappresentano una panoramica esaurente, costituiscono solo una campionatura significativa dei lavori premiati nel corso di quasi cinquant'anni e che, in molti casi, vedono come luogo di produzione studi radiofonici tra i più rappresentativi d'Europa. Un primo esempio di opera elettroacustica - secondo premio nel 1950 - è *D'un diable de briquet* del compositore belga Raymond Chevreville, realizzata negli studi dell'Institut National de Radiodiffusion (INR) di Bruxelles. Si può parlare in questo caso - così come per *Le joueur de flûte* di Marius Constant, premiata ex-aequo nel 1952 e prodotta a Parigi presso il Club d'Essai dell'RTF - di autentico pionierismo del genere. Nel primo caso, il racconto sinfonico da Andersen per voce recitante, coro maschile e orchestra con manipolazioni su disco e nastro magnetico, la musica sottolinea il carattere novellistico del racconto e l'autore gioca sia con la varietà e la novità dei timbri sia con processi di accelerazione e sovrapposizione di medesime parti orchestrali.

Ancora altri lavori elettroacustici vengono premiati nel corso degli anni cinquanta: *Orestes* di Henk Badings (Studi della NRU di Hilversum, 1954), *Ruisseau* di Maurice Jarre (Studio privato di Parigi, 1955), fino ad arrivare a Henri Pousseur che riceve il premio nel 1960 per *Electre*, dove egli integra la ripetizione di masse sonore vocali in un contesto elettroacustico, nell'ambito delle sue esperienze di correlazione tra produzione sintetica dei suoni e strutturazione seriale.

Se da un lato gli anni sessanta vedono una forte presenza di autori giapponesi dall'altro compositori europei di rilievo propongono lavori di grande respiro musicale: basti pensare a Luciano Berio ed al suo *Labourintus II*, Premio Rai nel 1966.

Nel 1972 Bruno Maderna (che aveva iniziato proprio con Berio l'esperienza elettroacustico-radiofonica nel 1954 con *Ritratto di città*) compone il suo quarto radiodramma, *Ages*; realizzato in collaborazione con Giorgio Pressburger presso lo Studio di Fonologia della RAI di Milano, l'opera comprende una forte componente elettronica dove canto, voci, strumenti, cluster ed eventi sintetici sono fusi in una generosa miscela multifonica. Sovrapposizione di voci, modulazione, variazione di velocità, eco e riverberazione sono le tecniche con cui i materiali originari sono stati elaborati su nastro magnetico. *Ages* vince il *Prix Italia* nello stesso anno.

In tempi più recenti si riscontrano programmi musicali di natura più squisitamente elettroacustica: nel 1985 viene presentata ed ottiene una menzione *Atlantide* di Michel-Georges Brégent, opera che riassume molti degli stilemi peculiari a tutta la fiorente scuola elettro-

custica canadese, da sempre attiva sul versante della più radicale sperimentazione radiofonica. Diviso in sette sezioni, il poema raccoglie da un lato la suggestione proposta dal titolo sul continente scomparso ma non mancano, dall'altro, riferimenti ecologici e politici: tra le varie sorgenti sonore sono inclusi eventi naturali e metropolitani.

L'uso di materiali di varia e diversa provenienza è una caratteristica peculiare anche al brano con cui la compositrice finlandese Kaija Saariaho ottiene il premio nel 1988: *Stilleben*. Provenienti dagli ambienti sonori più disparati, i suoni trasformati appaiono organizzati contemporaneamente ad un susseguirsi di interventi vocali parlati e cantati in tre lingue diverse, a sottolineare le grandi tematiche del lavoro: il viaggio, la distanza, la comunicazione. Completamente di impianto elettroacustico, il lavoro era stato commissionato dalla radio finlandese per l'esplicita partecipazione alla manifestazione italiana, ma non sono mancati in seguito altri importanti riconoscimenti internazionali.

Proseguiamo questa panoramica con un altro compositore francese di rilievo: *Luc Ferrari*, attivo da anni a Parigi, merita due ravvicinati riconoscimenti per *Et si toute entière maintenant* nel 1987 e *La escalera de los ciegos* nel 1991. Nel secondo caso si tratta di una produzione spagnola dove Ferrari impiega anche materiale di origine concreta (del resto egli è uno dei principali artefici della *musique concrète*) mentre il riferimento del titolo è ad un luogo di Madrid chiamato "la scala dei ciechi".

Infine, segnaliamo un recentissimo lavoro del compositore italiano Luca Francesconi: *Ballata del rovescio del mondo*, realizzato a Milano presso il centro AGON e vincitore nell'ultima edizione del premio. Il radiofilm fa parte di una serie di sedici "opere" di trenta minuti che la RAI ha commissionato ad altrettanti compositori e scrittori dell'ultima generazione. Questa operazione ha rappresentato un'eccellente occasione di incontro sia tra autori musicali e letterari che tra questi e alcuni importanti centri di produzione ed edizione musicali.

5. Conclusioni

In passato, il rapporto tra la radio e la musica elettroacustica è stato assai fertile; la separazione spesso avvenuta dopo gli anni sessanta tra studio radiofonico e studio di produzione musicale ha inevitabilmente contribuito a far perdere una certa consuetudine a comporre per la radio. Ciò nonostante molti compositori hanno dimostrato non solo interesse per questo tipo di mezzo di comunicazione ma anche grandi capacità nel saperlo piegare ad eccellenti risultati estetici.

La radio si candida quindi, durante questa fase di rilancio più volte annunciata, a divenire uno dei canali pri-

vilegiati non solo per la diffusione della musica elettroacustica ma soprattutto per la produzione di opere artistiche innovative.

Ringraziamenti

Il presente contributo ha preso spunto da una serie di ricerche effettuate in collaborazione con la segreteria del *Premio Italia* della RAI. Desidero ringraziare particolarmente Diana Palma e Renata Hinrichsen per il prezioso aiuto e la disponibilità che mi hanno offerto per la realizzazione di questo progetto.

Bibliografia

- AA.VV.: *Radio-Phonics and other Phonics*, in *Musicworks* 53, 1992.
- M. Baroni e R. Dalmonte (a cura di): *Bruno Maderna. Documenti*, Milano, Ed. Suvini Zerboni, 1985.
- H. Davis (a cura di): *International Electronic Music Catalog*, Paris-New York, GRM-ORTF/IEMC, 1967.
- D. Kahn e G. Whitehead (a cura di): *Wireless Imagination. Sound, Radio and the Avant-garde*, Cambridge Mass., The MIT Press, 1992.
- H. Pousseur (a cura di): *La musica elettronica*, Milano, Feltrinelli, 1976.
- F. K. Prieberg: *Musica ex machina*, Torino, Einaudi, 1963.
- E. Restagno: La musica alla radio, in *La Radio - storia di sessant'anni 1924/84*, Torino, ERI, 1984 pp. 112-114.
- D. Romeo (a cura di): *Prix Italia 1948/1993*, Roma, RAI, 1993.
- P. Schaeffer: *Pierre Schaeffer par Sophie Brunet suivie de réflexions de Pierre Schaeffer*, La Revue Musicale, Parigi, Ed. Richard-Masse, 1969.
- K. Stockhausen: *Electronic and Instrumental Music*, in *Die Reihe* 5, 1959.
- A. Vidolin: Avevamo nove oscillatori, in *I Quaderni della Civica Scuola di Musica* 21-22, 1992 pp. 13-16.

Elenco delle opere premiate citate

- D'un diable de briquet*, R. Chevreuil (Andersen)
Institut National Belge de Radiodiffusion (INR) - Bruxelles
1950 - 40' - monofonica
Racconto sinfonico per voce recitante, soprano, coro maschile e orchestra con manipolazioni su disco e nastro magnetico (premio di qualità, musica strumentale) (Secondo premio - categoria unica 1950)
- Orestes*, Henk Badings (J. Starink)
Nederlandse Radio Unie (NRU) - Hilversum
1954 - 54' - monofonica
Opera radiofonica
(Prix Italia per programmi musicali 1954)
- Ruisselle*, Maurice Jarre (R. Pillaudin)
Club d'Essai, Radiodiffusion-Télévision Française (RTF) - Parigi
1955 - 57'25" - monofonica
Poema radiofonico per voce, musica e microfoni
(Prix Italia per programmi musicali 1955)
- La grande tentation de Saint-Antoine*, Louis de Meester (M. de Ghelderode)
Institut National Belge de Radiodiffusion (INR) - Bruxelles
1957 - 55' - monofonica
Cantata satirica per solista, coro, 3 orchestre e nastro magnetico
(Prix Italia per programmi musicali 1957)
- Electre*, Henri Pousseur (P. Rhallys)
Studio de Musique Electronique de Bruxelles APELAC - Bruxelles
1960 - 48' - monofonica
Musica per il balletto omonimo sulla tragedia di Sofocle (Premio RAI per programmi musicali 1960)
- Laborintus II*, Luciano Berio (E. Sanguineti)
Office de Radiodiffusion-Television Francaise (ORTF) - Parigi
1965 - 33'30" - stereofonico
Per mimi, recitanti, insieme vocale, orchestra da camera e nastro magnetico (Premio RAI per composizione musicale con testo 1966)
- Ages*, Bruno Maderna/Giorgio Pressburger (Shakespeare)
Studio di Fonologia RAI - Milano
1972 - 29' - stereofonica

Invenzione radiofonica (Prix Italia per programmi dove la musica è la parte dominante) di cui nel suo discorso di premiazione si legge: «*Le levitationniste*»

Pianophonie, Kazimiers Serocki
Arbeitsgemeinschaft der öffentlich rechtlichen Rundfunkanstalten der Bundesrepublik Deutschland
1978 - 38' - stereofonica
Per pianoforte, suoni elettronici e orchestra
(Prix Italia per la musica 1979)
Atlantide, Michel-Georges Brégent
Cité du Havre studio of the Société Radio-Canada - Montréal
1985 - 25' - stereofonica
Poema radiofonico per voci, diversi ensembles, suoni elettroacustici e suoni ambientali
(menzione Prix Italia 1985)

Et si toute entière maintenant, Luc Ferrari (C. Fellous)
Radio France - Parigi
1987 - 34' - stereofonica
Racconto sinfonico
(Prix Italia per la musica 1987)

Stilleben, Kaija Saariaho (Kafka, ecc.)
YLE Experimental Studio - Helsinki
1987/88 - 22'09" - stereofonica
Composizione radiofonica per nastro magnetico
(Prix Italia per la musica 1988)

La escalera de los ciegos, Luc Ferrari
Radio Nacional de Espana (RNE 2) - Madrid
1991 - 34' - stereofonica
Serie di racconti musicali radiofonici
(Premio speciale per la musica 1991)

Ballata del rovescio del mondo, Luca Francesconi (U. Fiori)
AGON Acustica Informatica Musica - Milano
1994 - 30' - stereofonica
Radioscena musicale
(Prix Italia per la musica 1994)

La radio e la nascita della musica elettroacustica

Nicola Scaldaferrri

Laboratorio di Informatica del Dipartimento di Musica e Spettacolo
Università di Bologna

Abstract

Questo testo illustra l'importanza della radio per la nascita della musica elettroacustica. Vengono osservati i contributi di tipo estetico e tecnologico che essa apporta; vengono poi osservate le ragioni di un'autonomia della musica elettroacustica rispetto alla radio.

1 Introduzione

Il contributo della radio alla nascita della musica elettroacustica è di importanza fondamentale; esso interessa numerosi aspetti che vanno da quello tecnologico a quello estetico. Negli studi radiofonici i compositori iniziano a familiarizzare con apparecchiature elettriche e a manipolare i supporti; sarà la radio a fornire il sostegno concreto per l'istituzione studi di musica elettroacustica; è nell'ambito della ricerca della musica per la radio che verranno individuate le caratteristiche che saranno proprie della musica elettroacustica.

Nella prima metà del nostro secolo la radio svolge un ruolo assai importante nella promozione della nuova musica; in questo ambito nasce un forte interesse per la sperimentazione tecnico-musicale. Nei primi decenni del secolo è soprattutto l'ambiente tedesco a mostrarsi particolarmente attivo. Sorgono numerose riviste (come la rivista *Musik und Maschine* edita dalla Universal Edition) dedicate al rapporto tra musica e tecnologia. A Berlino vi è una cattedra di radiofonia tenuta da Hindemith, autore delle prime importanti composizioni per strumenti quali il Trautonium e l'organo meccanico. Tra Hindemith e Friedrich Trautwein, costruttore del Trautonium, si crea già quel rapporto di collaborazione tra compositore e tecnico che costituirà una costante degli studi di musica elettroacustica. Ma gli strumenti "protoelettronici" come le Onde Martenot e il Trautonium, pur attirando molto interesse, vengono impiegati quasi esclusivamente per le loro singolari soluzioni timbriche, all'interno di una concezione della composizione musicale di tipo tradizionale. L'esigenza di una musica fondata su principi completamente nuovi, sia dal punto di vista compositivo che recettivo, nascerà soprattutto grazie alle trasmissioni radiofoniche.

2 Musica radiogenica e musica radiofonica

Tra gli anni '20 e gli anni '30, da quando la radio diventa un apparecchio di uso comune, si sviluppa un forte dibattito sulla musica radiogenica che

trova largo spazio anche sulla stampa dell'epoca. La questione sorge soprattutto a causa del livello non ottimale dei mezzi di riproduzione del suono, che incidono in maniera negativa sulla qualità delle musiche trasmesse. Si cerca dunque di individuare la musica più adatta per la trasmissione per radio, sia tra quella del passato che tra quella dell'epoca presente; i criteri di scelta consistono nella leggerezza e linearità della scrittura, nella scelta di organici ridotti e di timbri strumentali puri. Compositori adatti alla trasmissione radiofonica sono Bach e Mozart, poco adatto invece Wagner. Vengono inoltre promosse manifestazioni e selezioni di nuova musica radiogenica. I compositori del tempo sono consapevoli del condizionamento esercitato dalla radio sul proprio lavoro; questo emerge anche da un'inchiesta del 1937 di Guido M. Gatti sulla Rassegna Musicale realizzata per stabilire l'influenza della radio sulla produzione musicale. Va segnalata in questo ambito la posizione di Casella che, per quanto si rivelò scettico sugli sviluppi autonomi di una musica per la radio, ammette la compatibilità tra le esigenze del mezzo radiofonico e l'estetica musicale neoclassica: la "semplificazione" richiesta dalla musica per la radio di fatto è coerente con la semplificazione musicale che si verifica in questi anni.

Si fa però strada la consapevolezza che la radio non costituisce un mezzo neutro di comunicazione ma possiede caratteristiche tecniche che si riflettono sul messaggio trasmesso. Si giungerà presto così a riconoscere alla radio una duplicità di funzione: se da un lato è un formidabile mezzo di diffusione, dall'altro può rappresentare un mezzo d'espressione in grado di aprire la strada a nuove forme artistiche. Arnheim, che riassume in maniera assai significativa il dibattito di quegli anni sulla radio, distingue nettamente la funzione comunicativa da quella espressiva del mezzo radiofonico e suggerisce le prospettive artistiche che possono derivare dal suo utilizzo come mezzo espressivo:

«La radio ha offerto un'esperienza nuova all'artista, al dilettante d'arte e al teorico in quanto sfrutta per la prima volta esclusivamente l'udibile senza far ricorso al visibile, all'immagine, a cui sempre è stato associato, tanto nella natura quanto nell'arte [...] Tutto ciò che accade davanti al microfono deve ri-

spettare le regole dell'arte uditiva» (Arnheim 1987, p. 8).

La radio riduce l'esperienza di recezione al solo ascolto: il punto di partenza dell'arte radiofonica deve consistere nello sfruttamento esclusivo dei valori uditivi; l'arte radiofonica deve dunque presentare quella caratteristica che negli anni '50 Schaeffer e Poignot, mutuando un termine della scuola pitagorica, definiranno acusmatica, cioè significativa a partire solo dalla sua dimensione acustica.

La forma artistica che per prima sfrutta questi principi è il radiodramma, che conosce dagli anni '30 un grande successo. Preoccupazione principale degli autori di radiodrammi è quella di riuscire, tramite il solo ascolto, a trasmettere la pienezza dell'azione, rendendo percettibili fenomeni quali la caratterizzazione del personaggio e la drammatizzazione.

Parallelamente al radiodramma, in parte proprio grazie ad esso, si verifica un ampliamento del paesaggio sonoro; l'immissione di suoni non musicali all'interno delle transmissioni radiofoniche va infatti acquistando un rilievo sempre maggiore, al punto da uscire dall'ambito effettistico e costituire un fenomeno a sé. Arnheim segnala il contributo che l'ampliamento del paesaggio sonoro, realizzato tramite la radio, può portare allo sviluppo del linguaggio musicale:

«La riscoperta del suono nel rumore e nel linguaggio, il collegamento tra musica, rumore e linguaggio in un materiale sonoro unitario è uno dei grandi compiti artistici della radio [...] Diventa sempre più importante la ricerca di effetti di rumore [...] Malfatti da parte della musica pura, trovano un ottimo impiego nella radio dove si tratta di elevare con una rappresentazione stilizzata ad un livello irreale non solo il suono del parlare umano, ma anche i rumori naturali. Quello che più conta nello sviluppo della musica dal punto di vista della radio è che la conquista di caratteri espressivi - sia essa stata favorevole o meno per la musica - ha creato dei buoni presupposti per una cultura sonora unitaria che potrebbe permettere un adeguato sviluppo della nostra sensibilità verso la musicalità del linguaggio e del sonoro in generale» (Arnheim 1987 p. 19-26).

La riflessione di Arnheim prelude ad una musica che considera la radio e la tecnologia come punto di partenza per la sperimentazione di nuovi orizzonti espressivi - ricordiamo come da un'ipotesi in linea con le intuizioni di Arnheim, quella del valore musicale del linguaggio, partiranno Eco e Berio nel 1958 per la realizzazione di Omaggio a Joyce.

Nella ricerca della musica per la radio è possibile dunque osservare due fasi distinte. La prima è quella della ricerca della musica radiogenica, cioè una musica di tipo tradizionale con caratteristiche che la rendono particolarmente adatta alla trasmissione per radio; la seconda è quella della musica radiofonica, musica del tutto nuova che si sappia basare sulle peculiarità tecniche della radio.

La musica radiogenica poteva ancora avere come quadro di riferimento una musica di tipo strumentale tradizionale; per la musica radiofonica questo non è più possibile: la musica tradizionale infatti, ascol-

tata alla radio, per quanto dotata dei requisiti radio-genici, presenta sempre un implicito riferimento all'esecuzione concertistica che ne è alla base. L'ostacolo maggiore per la creazione di una musica radiofonica mediante gli strumenti tradizionali è rappresentato dalla riconoscibilità, all'ascolto, di tutto ciò che rimanda all'esecuzione dal vivo. Affinché la musica radiofonica possa superare tale situazione sono necessarie perlomeno due cose: in primo luogo che ricorra a suoni diversi da quelli prodotti su strumenti tradizionali; in secondo luogo che si avvalga di un linguaggio autoreferenziato, in grado di esaurire le proprie pertinenze esclusivamente all'interno della dimensione sonora. La nascita della musica elettroacustica in Europa sembra essere la risposta appropriata alla ricerca della musica radiofonica. La recezione della nuova musica sembra riguardare solamente il fenomeno sonoro; il problema dell'interpretazione, il rituale concertistico e tutto quanto nella musica tradizionale costituiva un corollario inscindibile del dato sonoro e contribuiva alla creazione dell'aura, sembra essere scomparso. I complessi aspetti tecnici che stanno alla base non raggiungono l'ascoltatore; colpisce invece l'aspetto "inauditò" di questa musica, che si tratti dei suoni sinusoidali di Colonia o dei suoni concreti manipolati di Parigi.

3 Promozione e diffusione della musica elettroacustica

Le radio europee degli anni '50 si fanno promotrici della ricerca tecnologico-musicale. Oltre alla più generale attenzione della radio verso le nuove musiche e all'esigenza di una musica adatta ai propri mezzi tecnici, vi sono aspetti contingenti che giustificano questo interesse: da un lato la presenza ai vertici dirigenti delle radio di personaggi che promuovono questa ricerca (Wladimir Porché a Parigi, Alberto Mantelli e Gino Castelnuovo a Milano), dall'altro la politica di espansione culturale perseguita dalla radio. In quegli anni si giunge infatti all'istituzione del terzo programma radiofonico, esplicitamente creato per trasmissioni di tipo culturale e musicale. E' significativo ricordare che Alberto Mantelli e Luigi Rognoni, curatori del terzo programma istituito dalla RAI nel 1950, saranno nel 1955 tra i fondatori dello Studio di Fonologia Musicale della RAI.

Un importante incentivo in ambito radiofonico verso la ricerca sonora sperimentale era venuto dalla RAI nel 1948 con l'istituzione del Premio Italia per le opere musicali radiofoniche. Nelle composizioni presentate Premio Italia l'impiego della sonorizzazione si spinge già alle soglie della musica elettronica e concreta; scrive Mantelli a proposito di questi lavori:

«[...] la musica, similmente a quanto accade nel teatro musicale, è l'elemento costitutivo dominante. E allora si verifica questo fenomeno che merita di essere tenuto in conto: l'intervento registico gravita essenzialmente attorno alla musica e si configura,

nei casi più tipici e più indicativi, come una esperta e specializzata manipolazione dei suoni all'uscita del microfono. Fino a giungere alle soglie di quei tentativi e di quelle realizzazioni che cominciano a profilarsi nel settore della musica concreta e musica elettronica» (Mantelli 1956 p. 140).

Gli studi europei di musica elettronica costituiscono quasi sempre delle dirette emanazioni delle strutture radiofoniche, la cui attività di promozione della nuova musica si intensifica nell'ultimo dopoguerra. I tre importanti studi di Parigi, Colonia e Milano, dove opereranno Schaeffer e Henry, Eimert e Stockhausen, Berio e Maderna, verranno ufficialmente istituiti presso le sedi radiofoniche delle tre città (RTF, WDR e RAI). Il piccolo studio musicale di Darmstadt, diretto da Hermann Heiss, è sostenuto anch'esso dalla radio di Colonia. Lo studio di Baden-Baden, creato nel 1957 in occasione della visita in Europa di Wladimir Ussachewsky, è sostenuto dalla Sünwestfunk. Nel 1957 a Varsavia viene istituito lo Studio Eksperimentalne grazie al sostegno offerto dalla radio polacca.

La fondazione degli studi per la composizione musicale elettroacustica costituisce il traguardo di un'attività di sperimentazione avviata, con mezzi radiofonici spesso inadeguati, all'interno degli studi radiofonici tradizionali. A Colonia la fondazione dello studio avviene nel 1953, ma la sperimentazione sonora inizia nel 1951 grazie soprattutto ad Eimert che lavora presso la radio come curatore del programma notturno. Schaeffer inizia le sue ricerche sulla diffusione del suono presso la radio francese nel 1942; nel 1948 fonda il Club d'Essai e solo nel 1951 il Groupe de recherche de musique concrète (GRMC), che diventa un reparto stabile della RTF - dal 1958 il GRMC prenderà il nome di Groupe de Recherches Musicales (GRM), tuttora attivo. Anche a Milano la fondazione nel 1955 dello Studio di Fonologia Musicale corona un'attività di ricerca già in atto: Maderna infatti si interessa dal 1952 di composizione elettronica, mentre Berio collabora alla sonorizzazione dei programmi della radio dal 1953.

La radio, oltre ad istituire gli studi (che di solito hanno l'obbligo di produrre i materiali per la normale attività di sonorizzazione radiofonica), è anche il principale committente delle opere elettroacustiche, delle quali cura anche la diffusione. Questo si rivela di importanza vitale per i primi passi della musica elettroacustica, che non dispone di spazi propri per esecuzioni di tipo concertistico. Tra le numerose trasmissioni radiofoniche dedicate alla nuova musica, vanno ricordate quelle "storiche" del 20 giugno 1948 dalla RTF con la messa in onda degli *Etudes de bruits* di Schaeffer che costituisce l'atto di nascita della musica concreta, quella del 18 ottobre 1951 della WDR che vede la partecipazione di Eimert e Meyer-Eppler con l'ascolto dei primi esperimenti elettronici. Inoltre è nelle manifestazioni e nei festival di musica contemporanea, spesso promossi dalle radio, che trovano ospitalità le nuove opere elettroacustiche. Un caso significativo è rappresentato dalla stagione *Musik der Zeit* di

Colonia, dove trovano spazio, in un concerto del 26 maggio 1953, i lavori elettronici di Beyer e Eimert, e dove accanto ad opere come *Gruppen*, vengono presentate in prima esecuzione opere come *Gesang der Jünglinge* e *Continuo*. Anche la stagione degli Incontri Musicali di Milano, istituiti da Berio in seno all'attività musicale svolta presso la radio con Maderna, diffonde, oltre ai lavori strumentali, le opere dello Studio di Fonologia e degli altri studi.

4 Convergenze e diversità

Negli scritti dei compositori di musica elettroacustica degli anni '50 vengono ammessi i debiti della nuova musica nei confronti della radio, intesa come istituzione (Schaeffer dedica il libro *A la recherche d'une musique concrète* alla RTF e a Porché come segno di gratitudine per il sostegno ricevuto) ma anche come laboratorio di un nuovo modo di far musica. All'atto della fondazione dello Studio di Fonologia di Milano, Berio, nell'esprimere le intenzioni poetiche che lo animavano, sottolinea gli elementi di continuità con la ricerca della musica radiofonica:

«E' curioso notare come in questi ultimi trent'anni le idee su un'arte e un'estetica radiofonica non abbiano ancora potuto fissarsi in termini precisi e come l'esperienza passata non abbia fruttato che rare e generiche conclusioni che potessero costituire la base di un'eventuale grammatica radiofonica [...] Lo Studio di Fonologia Musicale, mettendo l'insieme di tali mezzi a disposizione di speciali produzioni che tentino una ricerca espressiva, può contribuire ad un incontro durevole tra gli strumenti e le possibili intuizioni di un attuale linguaggio radiofonico» (Berio 1956 p. 108).

E' però anche chiaro che il nuovo genere musicale segna un salto di qualità nell'utilizzo musicale dei mezzi tecnologici; la tecnologia non viene infatti impiegata solo per la riproduzione del suono ma soprattutto per la sua produzione ex novo.

Adorno, mettendo in evidenza la relazione di continuità tra esperienza radiofonica e musica elettroacustica, vede in quest'ultima anche un momento di risacca della musica d'arte dal rischio di mercificazione cui andava incontro a causa della riproducibilità tecnica del suono:

«Dietro la musica elettronica, e dietro l'attrattiva che se ne sprigiona, in definitiva sta oggettivamente l'idea di liberare i mezzi tecnici dalla schiavitù della mera riproduzione della musica, la quale di per sé si oppone a tale riproduzione. L'onta della musica meccanizzata, l'essere mero calco di un'altra cosa e dunque inautentica, esige di essere cancellata. Segretamente le speculazioni elettroniche mirano all'ideale latente di una musica radiofonica che non fosse più un semplice riempitivo della radio» (Adorno 1969 p. 267).

Stockhausen, nel ribadire il salto di qualità che si verifica con la musica elettronica nell'utilizzo dei mezzi tecnologici, condivide con Adorno l'idea che la musica elettronica costituisce un momento di ri-

scatto della musica d'arte dal rischio di mercificazione: «Che cosa hanno fatto finora i produttori discografici e radiofonici? Hanno riprodotto; hanno riprodotto una musica che in passato era stata scritta per le sale da concerto e i teatri d'opera, come se il film si fosse limitato a fotografare i vecchi pezzi teatrali. E la radio cerca di perfezionare tecnicamente questi reportage di concerti e di opere in modo che l'ascoltatore riesce sempre meno a distinguere la copia dall'originale: l'illusione deve essere totale [...] Ora, sebbene la radio fosse già diventata una tale fabbrica di conserve, accadde qualcosa di inaspettato: fece la propria apparizione la musica elettronica; una musica nata in maniera del tutto funzionale dalle particolari condizioni della radio. Essa non viene registrata con microfoni su un podio e conservata per essere poi in seguito riprodotta ma nasce invece con l'aiuto della valvola elettronica, esiste solo su nastro e può venire ascoltata solo con altoparlanti [...] Si può giudicare la musica elettronica come si vuole: la sua necessità risiede già solo nel fatto di indicare alla produzione radiofonica una via di sviluppo. La musica elettronica non utilizza nastro e altoparlante solo per riprodurre, ma per produrre. Chi ascolta per mezzo di altoparlanti capirà prima o poi che ha molto più senso se dall'altoparlante esce una musica che non si può ascoltare altrimenti che con un altoparlante» (Stockhausen 1976 p. 50).

5 Superamento dell'orizzonte radiofonico

Ma la musica elettronica è portatrice di molte novità che le impediscono di stare racchiusa a lungo negli ambiti della radiofonia. Essa introduce, mediante la possibilità di manipolazione del supporto, nuovi paradigmi compositivi che avranno una forte influenza anche sulla musica strumentale tradizionale. Presenta poi alcuni aspetti che presto si riveleranno incompatibili con la diffusione per radio. Il più importante di questi fenomeni riguarda la nuova funzione ricoperta dallo spazio. Da luogo di diffusione del suono quale era nella musica strumentale tradizionale, esso diventa, a tutti gli effetti, un parametro compositivo e come tale richiederà specifici canali di diffusione. Questa nuova utilizzazione dello spazio, favorita dal potenziamento dei mezzi di riproduzione del suono, fa il suo ingresso ufficiale con *Gesang der Jünglinge* di Stockhausen, esplicitamente composta per cinque gruppi di altoparlanti. Da questo momento in poi la musica elettronica richiederà dei criteri di diffusione del suono ben lontani da quelli che poteva offrire la radio; si renderà invece necessario il recupero delle sale da concerto, che saranno dotate quasi sempre di caratteristiche particolari - ricordiamo ad esempio la sala sferica proposta da Stockhausen e ripresa anche da Xenakis, o l'*Acousmonium* elaborato nel 1973 da François Bayle presso il GRM.

Un secondo fattore di incompatibilità tra radio e musica elettronica riguarda il recupero della dimensione concertistica. La riproducibilità tecnica del suono, fondamento stesso della musica elettronica (con la rinuncia al hic et nunc dell'esecuzione come momento di ri-creazione dell'opera) sembravano in un primo momento mettere in crisi il carattere auratico dell'opera musicale; in realtà con il "concerto" di musica elettronica si verifica un recupero dell'aura, se non addirittura un suo potentziamento. Il concerto tecnologico, con la presenza del compositore, degli altoparlanti e perfino dell'iluminazione appositamente creata, giungerà ad avere una valenza esoterica sconosciuta al concerto tradizionale - Ebbeke notava ad esempio che opere come *Hymnen*, oltre a non essere composizioni "da camera", richiedono un ascolto di tipo collettivo. Il terzo fattore di superamento dell'estetica radiofonica è rappresentato dall'avvento della musica mista, che reintroduce lo strumento tradizionale nella pienezza delle sue funzioni timbriche e rituali. La musica elettronica sembrava inizialmente voler segnare un punto di non ritorno nei confronti della musica strumentale, ma subito si assiste alla reintegrazione dell'elemento umano accanto alla macchina; i primi esempi di musica mista, quale *Musica su due dimensioni* (1952) di Maderna, sono contemporanei ai primi lavori di musica elettronica. Contrariamente alle previsioni della prima ora, la musica strumentale non verrà soppiantata da quella elettronica; proprio il rapporto dialettico tra le due dimensioni costituirà invece uno degli aspetti più fecondi della musica dei decenni successivi.

Non mancheranno opere elettronastiche dall'esplicita destinazione radiofonica - basti pensare già a *Visage* di Berio - ma saranno frutto di una precisa volontà poetica. Così come la radio, da parte sua, pur ricorrendo sempre più spesso alle risorse della musica elettronica, prosegue per una propria strada.

Radio e musica elettronica continuano ad intrecciarsi fino ai giorni nostri (pensiamo ad esempio alle diffusioni del *Cycle Acoustique* del GRM su Radio France o all'utilizzo di effetti elettronustici nei radiodrammi), ma nella consapevolezza delle rispettive autonomie.

References

- Adorno T. W., "Impiego musicale della radio", in Il fido maestro sostituto, Torino, Einaudi, 1969.
- Arnheim R., La radio, l'arte dell'ascolto, Roma, Editori Riuniti, 1987.
- Benjamin W., L'opera d'arte nell'epoca della sua riproducibilità tecnica, Torino, Einaudi, 1966.
- Berio L., "Prospettive nella Musica", Elettronica, Vol. 5, N. 3, 1956 pp. 108-115.

Dack J., "Pierre Schaeffer and the Significance of Radiophonic Art", *Contemporary Music Review*, Vol. 10, Part 2, 1994 pp. 3-11.

Ebbeke K., "La vue et l'ouïe - problémathique des partitions dans la musique électro-acoustique", *Contrechamps*, Vol. 10, 1990 pp.70-79.

Hindemith P., "Corso di musica per film nell'ambito dei corsi di radiofonia. Rapporto sull'attività dell'anno scolastico 1929-1930", Paul Hindemith nella cultura tedesca degli anni Venti, a cura di C. Piccardi, Milano, Unicopli, 1991 pp. 356-357.

Mantelli A. 1956, "Problemi di regia radiofonica", *Elettronica*, Vol. 5, N. 3, 1956 pp. 134-145.

Piccardi C., "L'eterogeneità allo specchio. La musica nel laboratorio della radiofonia", *Musica/Realtà*, Vol. 40, 1993 pp. 51-82.

Raven-Hart R., "Radio, and a New Theory of Tone-Quality", *The Musical Quartely*, Vol. 17, 1931 pp. 380-388.

Scaldaferri N., Lo studio di Fonologia Musicale della RAI di Milano negli anni cinquanta, tesi di laurea, DAMS, Università di Bologna, a.a. 1993-94.

Stockhausen, "Musica elettronica e musica strumentale", La musica elettronica, a cura di H. Pousseur, Milano, Feltrinelli, 1976 pp. 45-51.

Nicola Scaldaferri
I - 85030 San Costantino Albanese (PZ)
++ 39 973 91054
E-mail: regazzi@circfid.unibo.it

Il progetto CRM per una drammaturgia dell'ascolto radiofonico e del suo spazio virtuale

Michelangelo Lupone, Enrico Palandri, Laura Bianchini, Susanna Tamaro

CRM Centro Ricerche Musicali
Roma, Italia

Abstract

In this paper a description is given of the fundamental aspects of scientific research as developed at CRM by composers Laura Bianchini and Michelangelo Lupone and related to the production of two "Radiofilm" "Immobile e doppio" and "In un grattacielo", commissioned by Radiotore RAI (tails by Susanna Tamaro and Enrico Palandri).

In the two pieces the following compositional and analytical criteria have been employed: 1) Narrative Scheme and Acoustic Interlacing; 2) Sense and Sound of Speech; 3) Reference, Relations and Signal Processing of voice and instruments sounds; 4) Sound Spatialization; 5) Early Reflections and Waveguide Models.

1. Introduzione

Tra le recenti iniziative tese a sottolineare l'utilizzo della radio come mezzo privilegiato dell'ascolto musicale e più in generale come mezzo di comunicazione, quella dei Radiofilm rappresenta la più avanzata in senso estetico e tecnologico.

Il presupposto di fondere il testo letterario e la musica in un'opera dove questi interagiscono profondamente fino a tentare una diversa codifica linguistica, è stata vissuta dagli autori da una parte come sfida per la definizione di nuovi modelli espressivi, dall'altra come reale incentivo all'incontro con i centri di ricerca e produzione musicale elettronica italiani.

In particolare Radiotore RAI ha affidato ai Centri Agon di Milano, CRM ed Edison di Roma e alle Case editrici Edipan, Ricordi e Suvini-Zerboni il coordinamento e la gestione artistica, ponendo come vincolo o per meglio dire, incentivo, l'utilizzo di qualsiasi mezzo elettronico atto a favorire l'ascolto radiofonico dell'opera.

2. Il progetto drammaturgico

I testi scelti per le opere prodotte presso il CRM sono due racconti nati con intento letterario e, per essere introdotti nella diversa organizzazione narrativa proposta dal Radiofilm, hanno subito una vera e propria riscrittura.

Il lavoro di adattamento è avvenuto separatamente per le coppie di autori 1) Bianchini-Tamaro, 2) Lupone-Palandri e i testi definitivi hanno evidenziato le seguenti soluzioni drammaturgiche:

1. a) Narrazione corale multifonica e multilingue. Il coro descrive e sottolinea la drammaticità dell'azione o anticipa gli avvenimenti narrati dai protagonisti estraniando l'azione dal presente.
b) Scomposizione del testo in fonemi. I fonemi vengono trattati come unità minime di senso musicale.
c) Struttura a episodi. La narrazione di cinque storie è scandita da una cesura di riferimento.
2. a) Narrazione in forma di brevi segmenti di informazione.
b) Unità di senso verbale e musicale. Il ritmo strumentale è sincrono alla frase e l'intonazione della frase è riprodotto dalle altezze e dal timbro degli strumenti.
c) Struttura a tre livelli contemporanei. Sovrapposizione di testo comprensibile, testo trasformato e tessitura sonora.

3. Schema narrativo e intreccio sonoro

Un particolare approccio allo schema narrativo del testo è stato scelto in ambedue le opere: si è sviluppato un sistema di relazioni sonore basate sul timbro, il ritmo e le altezze della recitazione e dei suoni strumentali ed elettronici.

Emerge in questo modo una fitta tessitura tra i suoni verbali e non, che più volte maschera le parole per lasciare libero l'ascoltatore di ricostruire il senso attraverso la sola emergenza sonora.

Nel caso di Bianchini-Tamaro il coro, realizzato mediante l'elaborazione delle voci dell'attrice e della vocalist, congiunge l'evento narrativo a

quello musicale attraverso la trasformazione delle parole in suoni.

Nel caso di Lupone-Palandri l'immagine del "bancario" freddo, cinico, insensibile e astuto quanto ignorante e intelligente sono il perno di un intreccio che transita dal reale all'immaginario, dal passato al presente senza discontinuità. Nella scrittura musicale questo soggetto è stato trattato con elementi sonori che enfatizzano la dialettica grottesca tra la stupida formalità del "bancario" e lo scenario infernale, smodato del diavolo Barbariccia.

4. Senso e valenza sonora del testo

Trattare la parola affinché possa diventare elemento sonoro che supera o integra il suo stesso senso, è stato il lavoro più arduo e interessante affrontato dagli autori. In particolare, per rendere più evidente la trasformazione, sono stati utilizzati contemporaneamente accorgimenti di recitazione ed elaborazione elettronica della voce recitante.

Una sorta di geometria logica inscrive le parole in base al ritmo, le assonanze, le rime e le formanti timbriche che modellano artificialmente la voce. In fig. 2 è mostrato il programma *Polar Synthesis* basato su filtri passa banda entro cui sono state inserite le voci. I filtri, dinamicamente variati in frequenza centrale e larghezza di banda, hanno "deformato", "trasformato" e "congiunto" musicalmente anche le parole il cui suono e significato sono profondamente diversi.

5. Univocità, dipendenza, interazione e trasformazione

Le strategie sperimentate per integrare il linguaggio letterario a quello musicale, soprattutto da parte degli autori degli anni Cinquanta-Sessanta rappresentano ancora oggi un valido esempio di statuto funzionale entro cui stabilire le regole del complesso rapporto fra testo e musica.

La sfida posta dai Radiofilm doveva inevitabilmente ritrovare i processi formativi delle opere radiofoniche di quel periodo e tentare di approfondire e sviluppare, con il contributo di mezzi tecnici molto più evoluti, strategie e regole di un metalinguaggio frutto della fusione di testo-musica e soprattutto basato sul mezzo di diffusione radiofonico.

Il lavoro di ricerca e analisi svolto al CRM sulle opere di teatro musicale di quel periodo, ha permesso di raccogliere e sviluppare un semplice ed efficace sistema di regole funzionali alla percezione, poi complessivamente applicato in ambedue le Radiofilm, anche se con molte e

importanti differenze, in parte dovute ai materiali sonori in parte dovute alle strategie narrative.

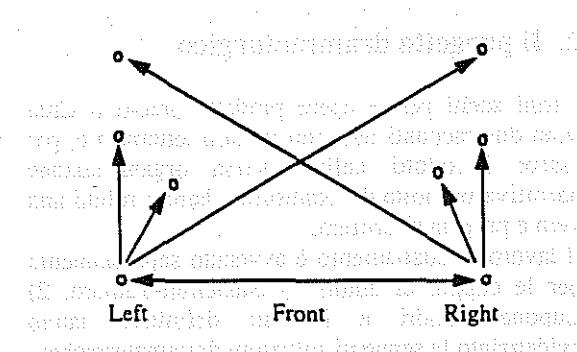
Il sistema si basa su quattro regole poste come primitive, ossia modelli di comportamento semplici sui cui sviluppare i processi combinatori che presiedono al rapporto fra testo e musica:

- **univocità:** regola che riconosce alla parola o al suono la completa indipendenza
 - **dipendenza:** sistema di regole che privilegia uno dei due elementi per intrinseca necessità di sviluppo narrativo e/o formale
 - **interazione:** sistemi di regole di dipendenza biunivoca
 - **trasformazione:** regola che presiede al processo di mutazione della parola in suono.
- #### 6. Posizione, movimento e velocità dei segnali sonori

Uno degli aspetti progettuali e compositivi che più ha interessato gli autori, soprattutto per le implicazioni di tipo percettivo, è stato il disegno di uno "spazio scenico virtuale": l'idea è stata di proporre attraverso il solo ascolto stereofonico, una immagine dell'ambiente in cui si sviluppa la narrazione o il dialogo rendendo poi tutti i suoni mobili in questo contesto.

L'approccio di molte composizioni radiofoniche tenta un tipo di simulazione dello spazio con l'uso di rumori ambientali che, attraverso una riconoscibile e semplificata semantica, costruisce nell'immaginazione dell'ascoltatore, il luogo dell'azione. Diversamente, abbiamo tentato nelle due opere di costruire uno spazio che si evolve in ragione degli elementi sonori che vi sono inseriti e dei contenuti narrativi. Si è resa evidente così una mutazione dell'ambiente che partecipa parimenti alle regole di trasformazione ed evoluzione dei materiali sonori.

Lo schema riassuntivo dello spazio entro cui sono stati collocati e mossi i suoni delle due opere è rappresentato qui di seguito:



La traiettoria descritta dalle frecce, individua anche l'ultima collocazione suono. Tale punto viene raggiunto con velocità costante in quanto le prove di accelerazione effettuate su diversi sistemi di ascolto, presi come riferimento per l'emissione radiofonica, non hanno dato risultati significativi. Per costruire lo "spazio virtuale" è stato realizzato sul sistema *Fly30* l'algoritmo mostrato in fig. 1. Il segnale sonoro è inserito in due bracci paralleli dove due funzioni modellano l'escursione di ampiezza relativamente ai canali sinistro e destro. Il ritardo inserito tra le funzioni di controllo dell'ampiezza assume, in base ai valori temporali assegnati, la duplice valenza di filtro passa basso ed elaboratore di fase del segnale.

7. Suoni alla parete, suoni in guida d'onda

Oltre al trattamento dell'energia, che ha permesso di organizzare gli spostamenti del segnale sonoro, per una più efficace definizione percettiva dello "spazio virtuale" le due opere hanno utilizzato anche elaborazioni che possono trasformare lo spettro del segnale in modo da rendere più sensibile percepitivamente la collocazione e il movimento del suono.

In particolare, l'opera di Lupone-Palandri utilizza due sistemi di elaborazione basati su filtri digitali le cui caratteristiche hanno permesso agli autori di produrre suoni più o meno vicini all'ascoltatore e suoni progressivamente direzionati da contenitori tubiformi.

Il primo sistema di elaborazione è basato su 11 filtri Comb e simula il comportamento delle prime riflessioni su una parete contro la quale vengono inviati i suoni. La modifica dei parametri dei filtri evidenzia gli spostamenti della sorgente lungo la parete.

La seconda elaborazione è effettuata con il programma "Pipe Simulator" (fig. 3) creato dal CRM per lo studio del modello fisico cilindrico.

Attraverso questo programma basato su una combinazione di filtri Allpass e FIR, il segnale sonoro introdotto, subisce profonde trasformazioni timbriche soprattutto quando le dimensioni del cilindro e il comportamento riflettente delle pareti assume valori estremi.

Bibliografia essenziale

L. Bianchini, M. Lupone, "The activities of Centro Ricerche Musicali" in Leonardo Music Journal, Pergamon Press, Oxford, 1992

L. Bianchini, M. Lupone, "L'Immaginario Radiofonico", Ed. CIDIM, Roma, 1994

G. Barbieri, Nota critica PAN CD 3055, Edizioni Musicali EDIPAN, Roma, 1995

S. Cappelletto, Nota critica PAN CD 3058, Edizioni Musicali EDIPAN, Roma, 1995

M. Lupone, "Civiltà del suono. Le questioni del cambiamento" in Il complesso di Elettra, CIDIM-Cemat, Roma, 1995.

E. Palandri, "Allegro Fantastico", Ed. Bompiani

A. Pellecchia, A. De Vitis, "SINTESI POLARE Applicazioni in campo musicale di filtri digitali operanti al limite della stabilità" in Atti X CIM, Milano, 1993.

A. Pellecchia, A. De Vitis "Fly30: un sistema programmabile per l'elaborazione numerica dei segnali musicali in tempo reale" in Atti del XX Convegno Nazionale di Acustica, Roma, 1992.

S. Tamaro, "Per voce sola", Ed. Baldini Castoldi.

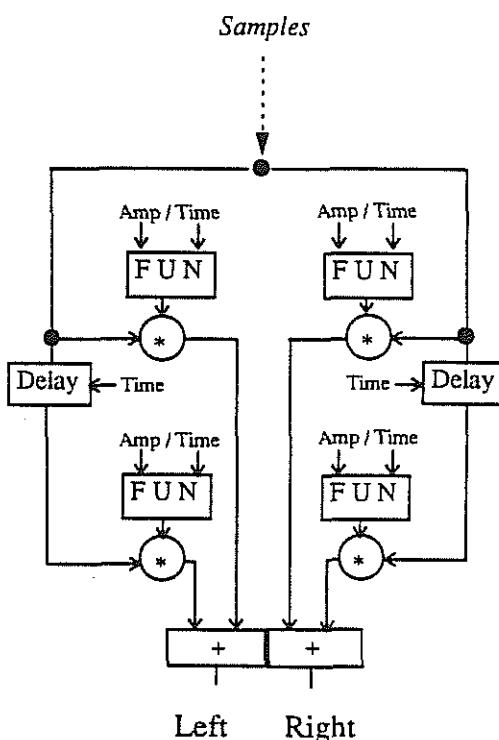


Fig. 1: Algoritmo che controlla la distribuzione dell'energia su due canali (Fly30 System)

CRM - Centro Ricerche Musicali
via Lamarmora 18 - I 00185 Roma, Italia
tel. xx39-6-4464161, xx39-6-4467911

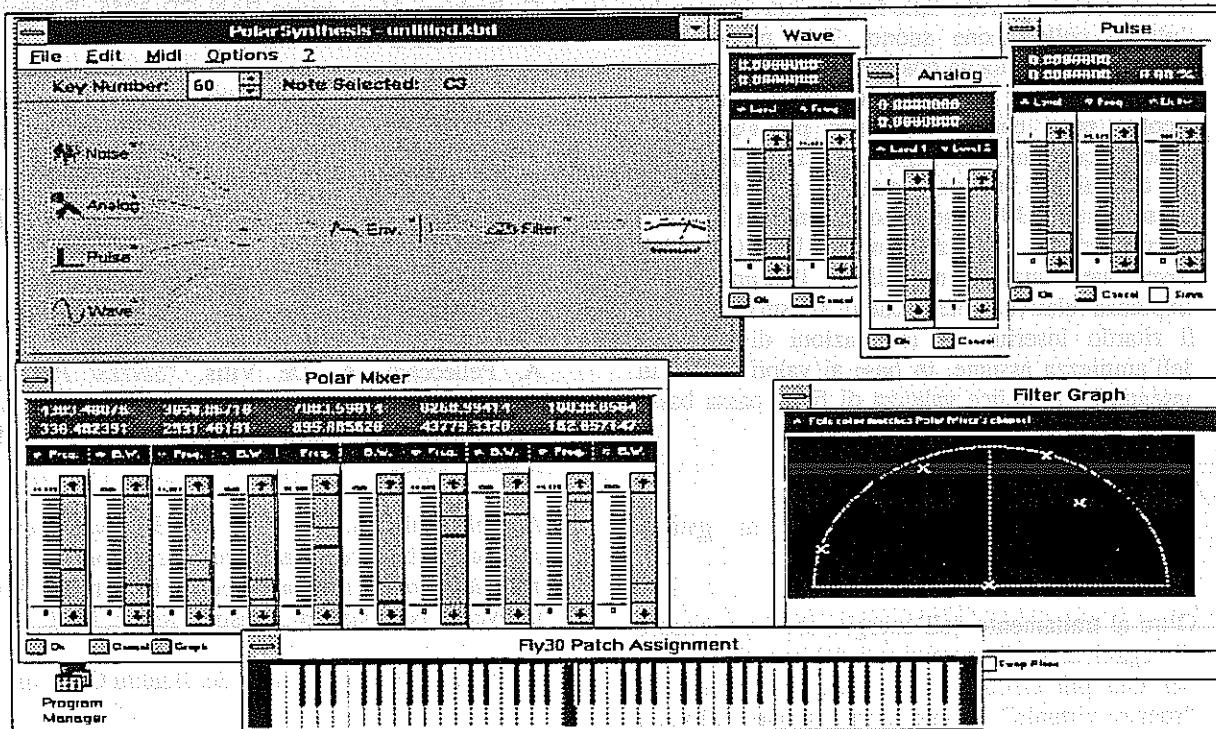


Fig. 2: POLAR SYNTHESIS - per Sistema Fly30 © 1991 CRM Centro Ricerche Musicali - Roma
Programma applicativo per la sintesi polare (© CRM 1991).

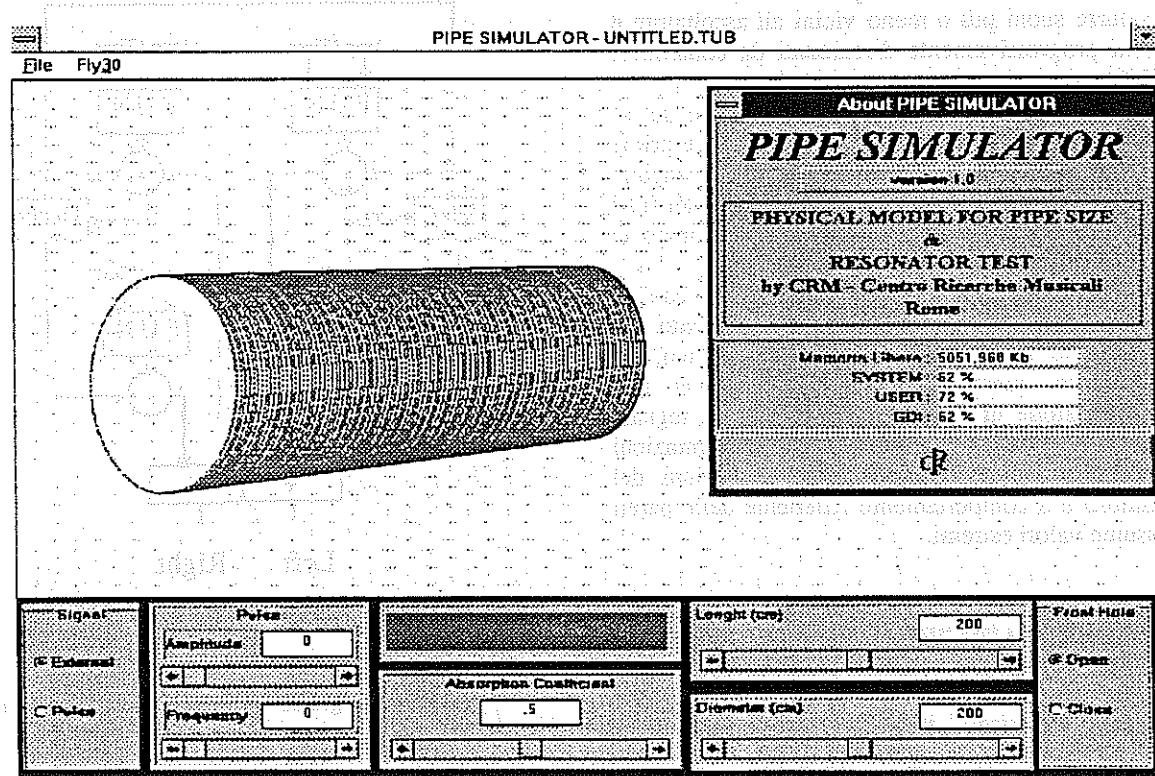
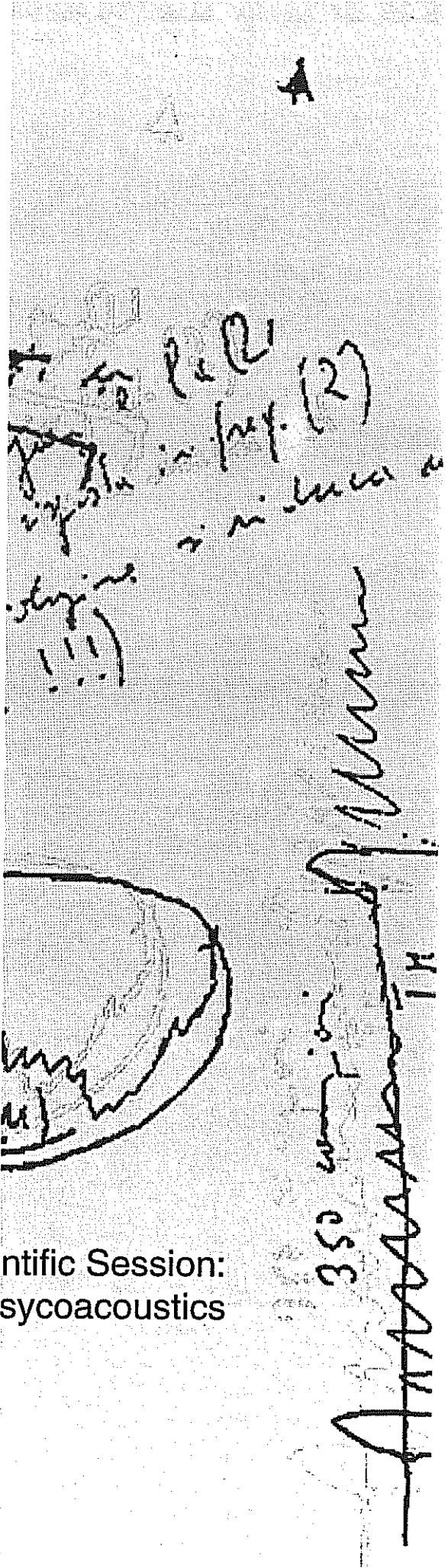


Fig. 3: PIPE SIMULATOR - per Sistema Fly30 © 1995 CRM Centro Ricerche Musicali - Roma
Programma applicativo che simula il modello fisico di un tubo, per lo studio delle riflessioni.



Scientific Session: Acoustics and Psycoacoustics

2012-01-12
containing 24 brs of wood

Optimizing Self-Organizing Timbre Maps: The Effect of Auditory Images and Distance Metrics

Petri Toivainen

Department of Musicology, University of Jyväskylä
Jyväskylä, Finland

Abstract

The effect of using different auditory images (AI) and distance metrics (DM) on the final configuration of a self-organized timbre map is examined by comparing distance matrices, obtained from simulations, with a similarity rating matrix, obtained using the same set of stimuli as in the simulations. Gradient images (GI), which are intended to represent idealizations of physiological gradient maps in the auditory pathway, are constructed. The optimal AI and DM, with respect to the similarity rating data, are searched using the gradient method.

1 Introduction

There have been a number of studies aiming at extracting the most salient acoustic attributes affecting the perception of timbre (see, e.g., Iverson & Krumhansl, 1993). A method widely used in these studies is similarity rating (SR): subjects are asked to rate, on a given scale, the similarity of all possible pairs in the set of stimuli. Multi-dimensional scaling (MDS) is then used to map the tones into a low-dimensional timbre space. By examining MDS maps it has been found that the brightness of a tone is one of the main contributors to the perception of timbre. Dynamic attributes, mostly in the onset portion of a tone, have also been shown to have a considerable influence on the perception of timbre.

A timbre space can also be constructed from a set of acoustical signals by means of connectionist models; for this purpose, the Kohonen self-organizing map (KSOM) (Kohonen, 1989) has been used by a number of researchers (De Poli, Prandoni, & Tonella, 1993; Cosi, De Poli, & Lauzzana, 1994; Feiten & Günzel, 1994; Toivainen, Kaipainen, & Louhivuori, in print).

There is evidence that the human auditory system encodes frequency transitions (Steiger & Bregman, 1981). In the cochlear nucleus, cells responsive to onsets and offsets have been found (Pickles, 1982). Shamma et al. (1992) found that the distribution of responses encodes the locally averaged gradient of the acoustic spectrum.

During the self-organizing process of the KSOM, the input vectors are compared with the synaptic vectors of the network using a DM. The prevalent choice for DM in the KSOM simulations is the Euclidean metric. It, however, does not necessarily correspond to how the auditory system compares stimuli.

The goals of the present study were (1) to examine the effect of different AIs and DMs on the final configuration of a self-organized timbre map; and (2) to explore whether the KSOM is able to project

the used AIs into two dimensions while maintaining the metrical relations between the images. The main method was that of calculating correlation coefficient values between distance matrices obtained from the simulations and a SR matrix obtained using the same set of tones. The overall aim was to find the AI and DM which would yield the highest correlation with the SR data. First, sets of AIs were constructed by varying the degree of emphasis on the onset of tones, and combining spectral images (SI) and GIs, i.e., images which were supposed to qualitatively represent responses of FM- and AM-sensitive neurons as well as neurons encoding the spectral gradient. Second, matrices of inter-image distances were calculated using various Minkowski metrics; the obtained distance matrices were compared with the SR matrix by calculating Pearson correlation coefficients. Third, a series of KSOM simulations was carried out with the obtained sets of AIs. Finally, matrices of response distances on the KSOM were calculated using two types of DMs; these were then compared with the SR matrix by means of the correlation coefficient.

2 Materials

The tone material consisted of 27 tones, produced by additive synthesis. The synthesis algorithm was controlled by six variable parameters: brightness, attack time, attack asynchrony, sustain time, decay time, and time scaling. The values of the parameters were chosen so that they produced a wide range of different tones.

3 SR Experiment

Nine subjects participated in the SR experiment. All of them have at least moderate skill in some musical instrument. During the experiment subjects heard each possible pair of tones, presented in random order. They were asked to rate the similarity

of each pair on a scale between zero and twelve. In this study, the SR matrix obtained by averaging the SR matrices of each subject was used.

4 Auditory Images

The timbre maps were constructed by a two-stage process. First, the AIs were computed using an auditory model. Second, the obtained AIs were fed into a KSOM which then projected them onto a two-dimensional map.

To obtain AIs, the stimuli were preprocessed, using the peripheral part of an auditory model by Van Immerseel and Martens (1992), modified by Leman (1994) for musical purposes. The output of the model is a 20-component vector, updated every 0.4 ms; each component is supposed to represent the probability of neural firing since the last update. The Minkowski distance of two real-valued vectors, $\mathbf{p} = (\xi_1, \dots, \xi_n)$ and $\mathbf{q} = (\eta_1, \dots, \eta_n)$, is defined by

$$\|\mathbf{p} - \mathbf{q}\|_\lambda = \left[\sum_{i=1}^n |\xi_i - \eta_i|^\lambda \right]^{1/\lambda} \quad (1)$$

By varying the value of the Minkowski exponent λ , a wide range of different metrics can be achieved. The SI of a tone, having a duration T , can be expressed as a scalar-valued function on the subset $[f_0, f_1] \times [0, T]$ of the frequency-time-plane:

$$S: s(f, t), f \in [f_0, f_1], t \in [0, T]. \quad (2)$$

Here f_0 and f_1 denote the minimum and maximum of the frequencies included in the image, respectively; $s(f, t)$ is obtained by temporally integrating the output of the auditory model:

$$s(f, t) = \int p(f, t') \frac{1}{\tau} e^{-t'/\tau} dt' \quad (3)$$

Here $p(f, t)$ denotes the neural firing probability of a neuron with a best frequency f , within a time interval $[t - dt, t]$; τ denotes the time constant.

The onset of the SI is emphasized by defining an exponential mapping of time (see Fig. 1):

$$\tau_k(t) = \begin{cases} t, & k = 0 \\ T \frac{e^k - 1}{e^k - 1}, & k > 0 \end{cases} \quad (4)$$

and a mapping on the frequency-time plane

$$\sigma_k: \sigma_k(f, t) = s(f, \tau_k(t)). \quad (5)$$

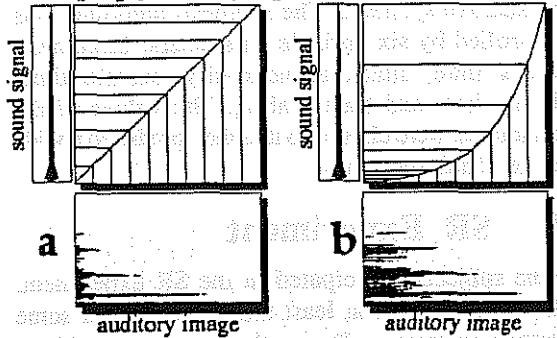


Fig.1. Emphasizing the onset in the AIs by means of an exponential mapping of time.

The exponential parameter k defines the degree of emphasis on the onset; subsequently, it is referred to as the emphasis parameter. The onset-emphasized SI, Σ_k , is now defined to be

$$\Sigma_k: \sigma_k(f, t), f \in [f_0, f_1], t \in [0, T]. \quad (6)$$

The physiological gradient maps are modelled by means of directional derivatives. For a given direction vector $\mathbf{e} = (e_f, e_t)$ on the frequency-time-plane, the rate of change of function σ_k along the direction of \mathbf{e} is expressed by its directional derivative, defined by

$$\gamma_k^{\mathbf{e}} \equiv \frac{\partial \sigma_k}{\partial \mathbf{e}} \equiv \mathbf{e} \cdot \nabla \sigma_k \equiv e_f \frac{\partial \sigma_k}{\partial f} + e_t \frac{\partial \sigma_k}{\partial t}. \quad (7)$$

When \mathbf{e} is parallel to the f -axis, the value of the corresponding directional derivative yields the frequency derivative of the spectrum at a given point of the frequency-time plane; $\gamma_k^{\mathbf{e}}$ can, thus, be regarded to be an idealization of the physiological spectral gradient map. When \mathbf{e} is parallel to the t -axis, $\gamma_k^{\mathbf{e}}$ yields the temporal change of the image and can, thus, be seen as an idealization of responses of AM-sensitive cochlear nucleus cells. When \mathbf{e} is parallel to neither of the axes, the corresponding directional derivative can be seen to represent the response of FM-sensitive neurons. For a given \mathbf{e} and k , the corresponding GI, $\Gamma_k^{\mathbf{e}}$, is now defined to be

$$\Gamma_k^{\mathbf{e}}: \gamma_k^{\mathbf{e}}(f, t), f \in [f_0, f_1], t \in [0, T]. \quad (8)$$

An illustration of GIs is presented in Figure 2.

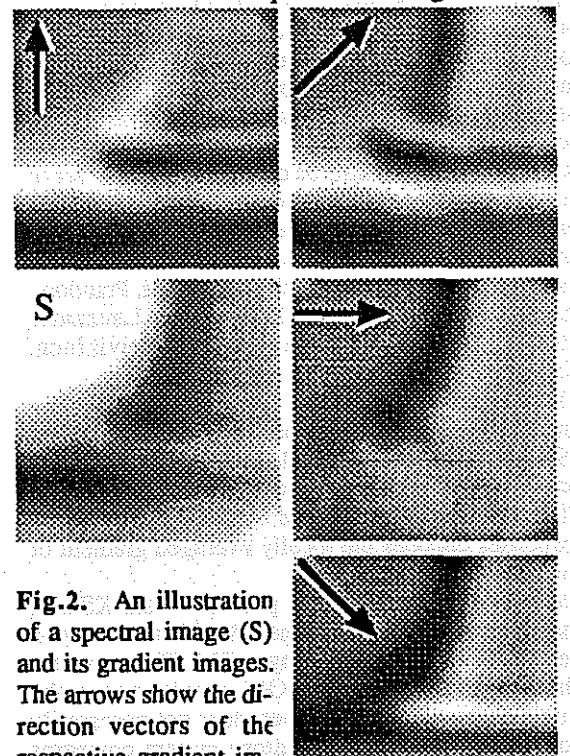


Fig.2. An illustration of a spectral image (S) and its gradient images. The arrows show the direction vectors of the respective gradient images.

For a set $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n\}$ of direction vectors on the frequency-time plane, a composite AI can be constructed as a vector-valued function, defined by

$$\Phi: \phi(f, t) = (\alpha_1 \sigma_k(f, t), \alpha_1 \gamma_k^{e_1}(f, t), \dots, \alpha_n \gamma_k^{e_n}(f, t)), \\ f \in [f_0, f_1], t \in [0, T]. \quad (9)$$

The α 's are the weighting factors of the corresponding images.

The Minkowski distance of two (vector-valued) AIs, Φ_1 and Φ_2 , is defined, analogously to Eq. (1), by

$$\|\Phi_1 - \Phi_2\|_\lambda = \left[\iint \|\phi_1(f, t) - \phi_2(f, t)\|_\lambda^{\lambda} df dt \right]^{1/\lambda} \quad (10)$$

where $\|\phi_1(f, t) - \phi_2(f, t)\|_\lambda$ is the Minkowski norm of vector $\phi_1(f, t) - \phi_2(f, t)$, and the integration is carried out over $[f_0, f_1] \times [0, T]$.

5 Metrics on the KSOM

To obtain distance matrices from the KSOM simulations, two different DMs were used. In the response focus metric (RFM), the locus of response to each input vector is defined to be the position of the neuron whose weight vector is closest to the input vector in question. In the centroid-of-activation metric (CAM), the locus of response is defined to be the centroid of the activation pattern. The distance between two responses is then defined as being the Euclidean distance between the loci of response. — The distance metric between the AIs will subsequently be referred to as the AIM.

6 Results

6.1. Auditory images

To begin with, the correlation between the SR and the AI distance matrix, denoted by $\text{corr}(D, S)$, was calculated using the SIs only, with parameter values $\lambda = 2$ (Euclidean metric), $k = 0$ (no emphasizing of onsets). This yielded the correlation value $\text{corr}(D, S) = 0.677$.

Next, λ and k were varied between $1 \leq \lambda \leq 10$ and $0 \leq k \leq 0.48$, with steps of 1 and 0.02, respectively. The value of $\text{corr}(D, S)$ as a function of λ and k is presented in Figure 3. The maximum correlation value, obtained with $\lambda = 1$ and $k = 0.20$, was $\text{corr}(D, S) = 0.868$.

The significance of GIs in timbre perception was studied by constructing combined AIs using the SI and the four GIs presented in figure 2 (referred to as the f-, c1-, t, and c2-gradients, from the top clockwise). The gradient ascent method was used to find the combination of parameter values $p^* = (\lambda^*, k^*, \alpha_f^*, \alpha_{c1}^*, \alpha_t^*, \alpha_{c2}^*)$ which would yield the maximum value of $\text{corr}(D, S)$. Since in the calculation of Pearson correlation coefficient the variances of the two variables are normalized, multiplying the α 's with a constant does not alter the correlation. Therefore, α , was fixed to value 1. The search for the maximum was started with zero values for the other α 's. In a series of calculations, the initial values for λ and k were chosen from the neighbourhood of the point $\lambda = 1$, $k = 0.20$, i.e., the point of maximum correlation in Figure 3.

The maximum value achieved was $\text{corr}(D, S) = 0.882$; this was obtained with the parameter values $\lambda = 0.690$, $k = 0.21$, $\alpha_f = 1.510$, $\alpha_{c1} = 0.010$, $\alpha_t = 0.262$, and $\alpha_{c2} = 0.000$. As a result of adding GIs and allowing the Minkowski exponent to change continuously, a slight increase in correlation was thus achieved, the main contributors to this being parameters λ and α_f .

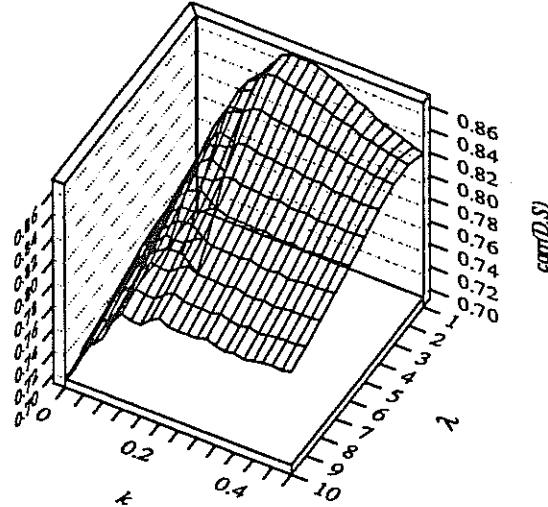


Fig.3. The value of $\text{corr}(D, S)$, using the SIs only, as a function of λ and k .

6.2. KSOM simulations

Using various sets of AIs as input, a series of KSOM simulations was carried out; the dimension of the input vectors was 400 (20x20), unless otherwise indicated. During the training procedure, the Minkowski metric was used. In each simulation, a 12 x 12 Kohonen network was trained for 50000 cycles, using a constant learning rate of 0.05. In the beginning, the radius of the topological neighbourhood was 6, and it was linearly decreased after every cycle so as to reach zero at the end of the training. From the obtained timbre maps, distance matrices were calculated using both the RFM and the CAM. These matrices were then compared with the SR data by calculating the Pearson correlation. First, the KSOM was trained with AIs which did not include any GIs. The correlation values obtained from simulations with two different sets of AIs are presented in Fig. 4, bars a-b. In both simulations, the AIM correlation was the highest, followed by the CAM and RFM correlations, in that order. That might indicate that it is not possible to reduce timbre to two dimensions without compromising the metrical relationships between the input stimuli. Second, the post-Kohonen correlation values are high when the AIM-correlation is high, and vice versa; this shows that the chosen preprocessing strategy is an essential factor in this kind of modelling. Last, the CAM correlations are regularly higher than the respective RFM correlations; taking into account the whole activation patterns of the KSOM seems, thus, to

yield a better metric than paying attention only to the unit with the highest response.

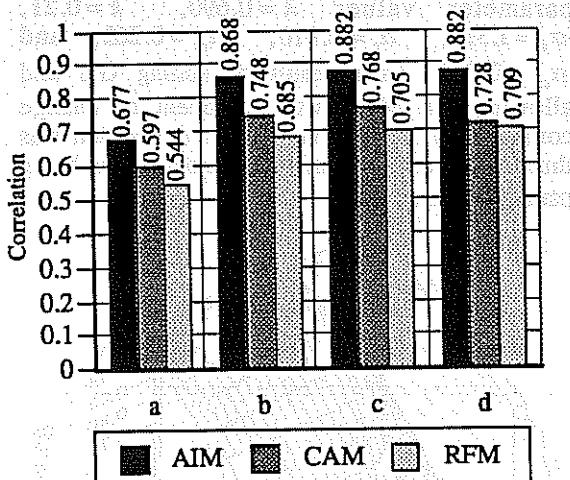


Fig.4. AIM, CAM, and RFM correlations obtained from four simulations. (a) $\lambda = 2$, $k = 0$, no GIs; (b) $\lambda = 1$, $k = 0.20$, no GIs; (c) SI and f-GI, one net; (d) SI and f-GI, three nets.

At the next stage, the KSOM was trained with sets of AIs which included the SI and the f-GI of each tone. The values of the parameters were $\lambda = 1$, $k = 0.21$, $\alpha_f = 1.510$, and $\alpha_i = \alpha_{i1} = \alpha_{i2} = 0$. The last three GIs were ignored to reduce computational load. Two different network architectures were used (see Fig.5). First, a single KSOM was trained with AIs consisting of a SI and an f-GI; the components of the latter were multiplied by 1.510. In the second approach, two KSOMs were first trained, one with the SIs and the other with the f-GIs. For each input stimulus, the coordinates of the centroids of activation on these maps, thus forming a 4-dimensional vector, were used as input to a third KSOM.

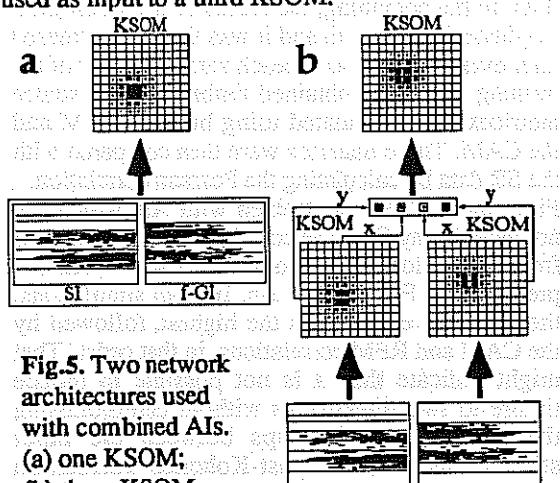


Fig.5. Two network architectures used with combined AIs.

(a) one KSOM;

(b) three KSOMs.

The CAM and RFM correlation values, obtained using these combined auditory images, are presented in Figure 4, bars c-d. Also in these simulations the CAM and RFM correlations were lower than the respective AIM correlations. High AIM correlations tended to imply high CAM and RFM correlations,

and vice versa. Using the one-network architecture (Fig.5.a) yielded the highest CAM correlation value, 0.768; the highest RFM correlation value, 0.709, was obtained using the three-network architecture (Fig.5.b).

References

- P. Cosi, G. De Poli, G. Lauzzana: "Auditory modelling and self-organizing neural networks for timbre classification", *JNMR*, Vol. 23, N. 1, 1994, pp. 71-98.
- G. De Poli, P. Prandoni, P. Tonella: "Timbre clustering by self-organizing neural networks", *Proc. of X CIM*. Milan: University of Milan, 1993, pp. 102-108.
- B. Feiten, S. Günzel: "Automatic Indexing of a Sound Database Using Self-Organizing Neural Nets", *Computer Music Journal*, Vol. 18, N. 3, 1994, pp. 53-65.
- P. Iverson, C. L. Krumhansl: "Isolating the Dynamic Attributes of Musical Timbre", *JASA*, Vol. 94, N. 5, 1993, pp. 2595-2603.
- M. Leman: "Schema-Based Tone Center Recognition of Musical Signals", *JNMR*, Vol. 23, N. 2, 1994, pp. 169-204.
- J. O. Pickles: *An Introduction to the Physiology of Hearing*. London, Academic Press, 1982.
- S. A. Shamma, S. Vranic, P. Wiser: "Spectral Gradient Columns in Primary Auditory Cortex: Physiological and Psychoacoustical Correlates", *Advances in the Biosciences*, Vol. 83, 1992, pp. 397-404.
- H. Steiger, A. S. Bregman: "Capturing Frequency Components of Glided Tones: Frequency separation, orientation, and alignment", *Perception and Psychophysics*, Vol. 30, N. 5, 1981, pp. 425-435.
- P. Toivainen, M. Kaipainen, J. Louhivuori: "Musical Timbre: Similarity Ratings Correlate With Computational Feature Space Distances", *JNMR* (in print).
- L. M. Van Immerseel, J.-P. Martens: "Pitch and Voiced/Unvoiced Determination with an Auditory Model", *JASA*, Vol. 91, N. 6, 1992, pp. 3511-3526.

Address

Petri Toivainen
Department of Musicology
University of Jyväskylä
PL 35
FIN-40351 Jyväskylä
Finland
e-mail: ptoivia@jyu.fi

Timbre Characterization with Mel-Cepstrum: a Multivariate Analysis

Nicola Boatin, Giovanni De Poli, Paolo Prandoni

CSC - Department of Electronics and Informatics, University of Padova, Italy

Abstract

In this paper, 21 musical sounds are coded as a series of Mel-Frequency Cepstral Coefficients. By regarding the coefficients as dimensions, a Principal Components Analysis of the data extracts three factors which together account for about 80% of the total variance. The interpretation of these factors contributes to a better understanding of the main variables which shape the quality of a sound.

1 Introduction

Timbre is a multidimensional feature which describes the quality of the sound when other factors such as pitch and loudness are kept constant. A classical method has been employed several times in the past to build a timbre space: similarity ratings provided by a group of listeners are organized by means of a Multidimensional Scaling analysis; the empirical interpretation of the dimensions of the resulting timbre space leads to insightful hypotheses on the major features of musical timbres. A correlation procedure usually follows, in which these features are tentatively paired with physical quantities underlying the sound. These studies provided a first general understanding of the major variables which define the musical timbre as a sound feature; on the other hand, a clear methodology has not yet been proposed to extract some "timbral coordinates" from a given, arbitrary sound.

More recent researches attempted to analyze musical timbre starting directly from the acoustic signal. More or less simplified ear models are the tools employed to extract perceptual parameters from the waveform. These parameters are subsequently mapped onto a timbre space, often exploiting the self-organizing capabilities of artificial neural networks.

The representational capabilities of perceptual qualities offered by the Mel-Frequency Cepstral Coefficients (MFCC) are widely employed in speech recognition systems. In [3] we have verified that these capabilities are extremely effective even when applied to the analysis of sound quality. Sounds which appear timbrically close to the listener lead to representations which are topologically close in the MFCC domain. Moreover, the overall topological organization in the coefficients' space of the MFCC representation of sounds ap-

pears to be in general agreement with the usual organization of timbre spaces defined by previous psychoacoustical researches.

In this work we try to extend these results by applying a multivariate analysis to several prototypical sets of MFCC-coded musical sounds. The interpretation of the principal components in the resulting space is the starting point for a better understanding of the main variables which shape the quality of a sound.

2 Parametrization

The *Mel Frequency Cepstral Coefficients* (MFCC) were first introduced by Davis e Mermelstein in a comparative study of different speech coding techniques [1]. In the method, filterbank analysis and cepstral analysis are combined: the short-term log-energy output of a mel-spaced filterbank is mapped by a DCT onto the set of the MFCC coefficients. The filterbank is constituted by 27 partially overlapping triangular filters, equally spaced on a mel-frequency scale. The mel scale itself is defined as

$$\text{mel}(f) = \begin{cases} f & f \leq 1 \text{ kHz} \\ 2595 \log_{10} \left(1 - \frac{f}{700} \right) & f > 1 \text{ kHz} \end{cases}$$

As opposed to the speech analysis case, the importance of higher frequency components in the perception of musical sounds leads to a version of the filterbank which extends up to 8 kHz. The coefficients are computed using a 23.2 ms Hamming window with a 4 ms time-shift.

An inverse DCT over the coefficients c_i leads back to the spectral envelope, $C(\mu)$, which is expressed in dB and warped along the frequency axis according to the Hz-mel mapping above. As is usual in cepstral techniques, dropping the higher order coefficients (*liftering*) prior to computing the

inverse transform leads to a smoothed version of the spectral envelope, $\tilde{C}(\mu)$, which is very often more useful in the subsequent analyses. The first coefficient, c_0 , is also dropped. This coefficient is indeed proportional to the energy of the input, so that, by its removal, an amplitude normalized representation of the signal is easily accomplished. This is going to be extremely useful when comparisons between different signals are going to be taken into account.

This cepstral analysis involves a cosine transform, which is an orthogonal transform; the "natural" choice for a distance metric in the MFCC space is thus the Euclidean distance. It can be shown that the Euclidean distance between MFCC sets is equivalent to the distance between (possibly smoothed) spectra in the mel-frequency domain; this seems to be a reasonable measure of the similarities between sounds.

While not a cepstral extraction in the usual sense, the effectiveness of the MFCC is mainly due to the mel-based filter spacing and to the dynamic range compression in the log filter outputs. Both these features mimic the physiological processes of the inner ear.

3 Experimental results

Several sets of different musical sounds have been used, all of which led to similar final results. In this paper we will report the analysis of the same set of musical sounds as used by Carol Krumhansl in [2], where musical timbre is studied by means of listeners' responses. The actual sound samples have been obtained by FM sound synthesis on a Yamaha Tx802 synthesizer (Tab. 3). Each of the 21 tones has the same pitch (E_b 4, freq. 311.1 Hz), and was sampled at 44.1 kHz.

Horn	Trumpet
Trombone	Harp
Trumpet	Oboe
Vibraphone	Striano
Sampled Piano	Harpsichord
Tenor Oboe	Oboe
Bassoon	Clarinet
Vibraphone	Obochord
Bowed Piano	Guitar
Stringed Piano	
Guitarnet	

Table 1: Musical Instruments (Yamaha Tx802 synthesizer).

In the present study the quasi steady-state portion of the tone is taken into account. The sound signal is sliced into overlapping frames;

each frame is converted to a MFCC representation, namely a vector of MFCC coefficients which can be regarded as a point in a multidimensional MFCC space. Experimental evidence shows that the points stemming from the analysis of a single instrument are tightly clustered together. As a consequence, the centroid of the cluster is chosen as the prototypical MFCC coordinate of the instrument. This does not lead to misrepresentations as the tight clusters originated by the different instruments map into well separated regions of the MFCC space.

The advantages of this representation appear clearly from the analysis of the variance distribution of the data along the different steps of the coding process. Fig. 1 displays the variance (individual and cumulative) of the instrumental data as appears at the output of the filterbank; this is almost uniformly distributed along the entire spectrum, that is, along the axes of the associated space. In other words, in this frequency space there are no predominating axes. In the MFCC space, on the other hand, the variance is primarily distributed across the first coefficients, as appears in fig. 2. By selecting these coefficients as the dominant geometrical coordinates, a MFCC subspace is obtained which adequately represents the data. The first seven coefficients prove to be sufficient to this aim.

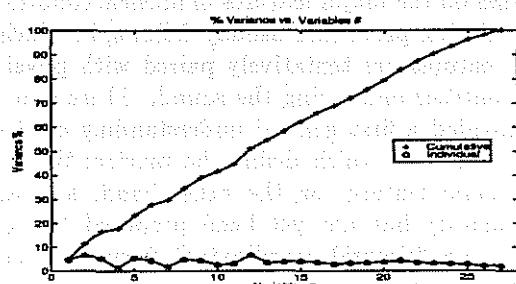


Figure 1: Percent of the total variance associated to the filterbank channels.

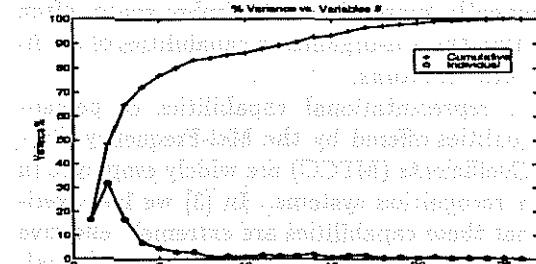


Figure 2: Percent of the total variance associated to the MFCC coefficients.

Given the previous results, the next goal is

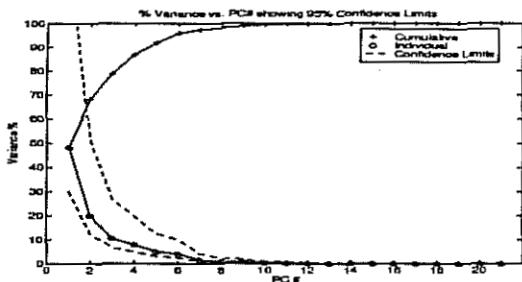


Figure 3: Percent of the total variance associated to the Principal Components

to find an optimal result in the definition of a minimal subspace in the MFCC space which preserves the relevant information in the signals. The Principal Component Analysis (PCA) is a multivariate statistical tool whose formulation relies on the properties of the *orthogonal linear transforms*. Given a p -dimensional data set, by means of scaling and rotations the principal information is moved onto a reduced set of variables. By carefully choosing the transformation matrix, the new set of variables proves to be uncorrelated: the information pertaining to each of the variables can thus be analyzed in an independent fashion. The two major objectives of a PCA analysis are then the *compression of the data*, and the *optimal interpretation of the data*. These goals are achieved if in the original data set the variance in the information is naturally associated to a few principal components, so that the representational loss associated with the dropping of some of the components is relatively small. Geometrically, the PCA defines a set rotated of coordinates in the space in which the new axes coincide with the directions of maximum variance.

A PC analysis performed on the MFCC-coded instrumental set reveals that the 80% of the variance is concentrated in the first three components [3]. A three-dimensional space is thus able to provide a "correct" topological organization within the limits of the retained information.

Several interpretations can be carried out on the space thus obtained. The origin represents the average spectral envelope of all the timbres in the data set. Fig. 4 clearly shows the typical low-pass shape which characterizes almost all musical instruments. In fig. 5, 6 and 7 the three direction cosines related to the PCA spatial transformation are shown. Each one represents the spectral contribution of the correspondent spatial dimension of the timbre space.

To better evaluate the global features of the timbre organization thus obtained, it is useful to represent the smoothed spectral envelopes at the extremes of the axes as the sum of the average

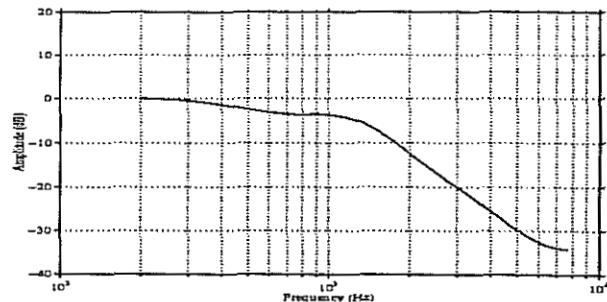


Figure 4: Overall average spectral shape

spectral envelope (the origin) with the eigenvalue-weighted contributions (positive on the right side, negative on the left one) of each of the direction cosines. In this new representation, the distribution of the spectral energy along the axes of the new PCA space appears more clearly (fig. 8, 9 and 10).

4 Discussion and results

The analysis of the plots shown so far leads to several conclusions. Clearly, the first axis is related to the spectral energy distribution, called *brightness*. The spectral envelope associated to the first principal component (fig. 5) shows a boosting of the low frequencies for positive values of the coordinate, whereas negative values are linked to frequency values above 1.5 kHz. These interpretations are confirmed by the spectral envelopes shown in fig. 8. Along this dimension, bright-sounding instruments such as the *oboe*, the *bassoon*, and the *horn* are at a maximum of the geometric distance and of the perceptual distance from the darker-sounding instruments such as the *vibraphone*, the *guitar*, and the *piano*.

The second dimension is correlated to the energy values in a broad frequency range which encompasses the whole band for musical sounds, 0.6–6 kHz. From the plots of fig. 6 it could be assumed that the fundamental characteristics be an apparent spectral irregularity. Fig. 9 shows how

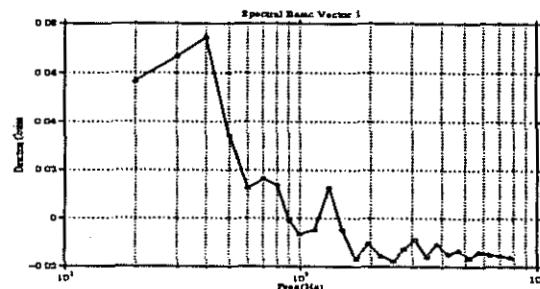


Figure 5: Spectral envelope (1st direction cosine)

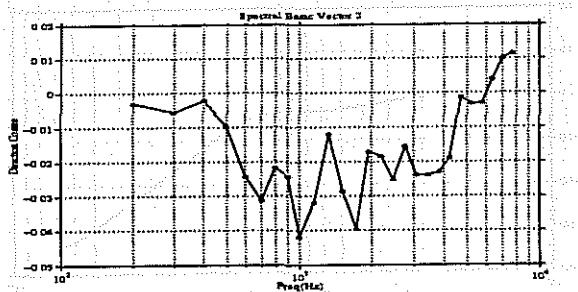


Figure 6: Spectral envelope (2nd direction cosine)

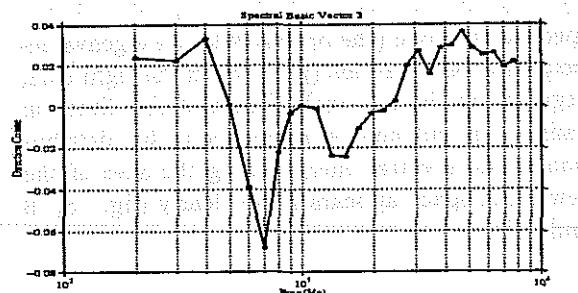


Figure 7: Spectral envelope (3rd direction cosine)

The third dimension seems to be associated to subtler aspects of the spectral characteristic; it is correlated to the energy content of a narrow region of the spectrum centered around 700 Hz. This set of frequencies has an extremely important role in acoustic perception, and is a commonly used parameter in audio equalization. A possible hypothesis would be that this frequency region underlies a differentiation criterion which is similar, albeit finer, to the general distinction between low-pass and high-pass instruments. At the positive end we have the *clarinet* and at the negative end the

trombone, horn, vibraphone; at one end of the axis there is an amplification in the region corresponding to the knee of the spectral envelope, whereas at the other extreme there is a smoothing in the slope discontinuity in the spectral envelope, which changes into an almost monotonic curve. At one end we have *trombone, horn, vibraphone*; at the other we have the *guitar, harpsichord*.

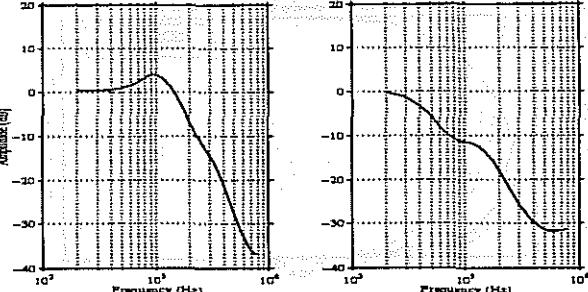


Figure 9: Spectral envelope (2nd coordinate)
chromatically insipidly soft

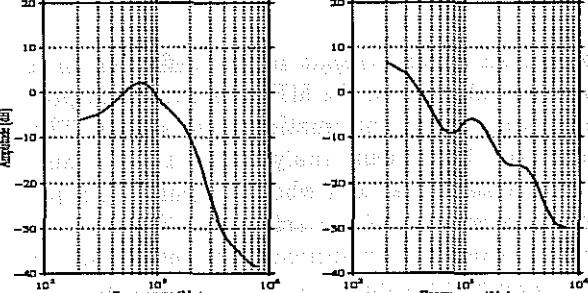


Figure 10: Spectral envelope (3rd coordinate)

5 Conclusions

Other experiments with different sound sets confirmed the fact that the first two dimensions extracted by the PCA control the overall spectral shape, the "cutoff" frequency, and the spectral slope; the third dimension is always related to the energy content of a spectral region bounded between 700 and 900 Hz. We are led to conclude that this feature is a differentiating factor in the quality of musical timbre which acts in an independent fashion from the quality called brightness.

References

- [1] Davis, S.B., and Mermelstein, P. *Comparison of Parametric Representations for Monosyllabic Word recognition in Continuously Spoken Sentences*, IEEE Trans. ASSP, vol. 28(4), pp. 357–366, 1980.
- [2] Krumhansl C.L. *Why is musical timbre so hard to understand?*, in S. Nielzen, O. Olsesson(ed.), *Structure and perception of electroacoustic sound and music*, Elsevier, p. 43–53, 1989.
- [3] Cosi P., G. De Poli, P. Prandoni, *Timbre characterization with Mel-Cepstrum and neural nets*, Proc. 1994 ICMC, pp. 42–45.

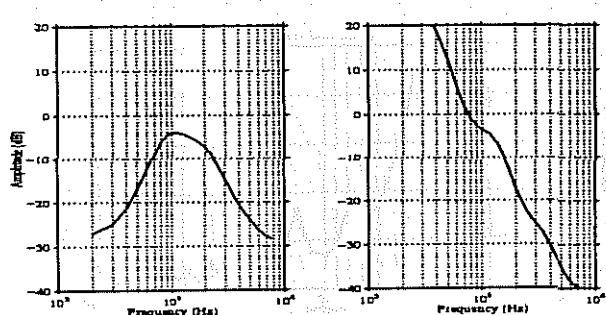


Figure 8: Spectral envelope (1st coordinate)

A Numerical Investigation of the Representation of Room Transfer Functions for Artificial Reverberation.

Davide Rocchesso

Centro di Sonologia Computazionale

Dipartimento di Elettronica e Informatica

Università degli Studi di Padova

Email: roc@csc.unipd.it

William Putnam

Center for Computer Research

in Music and Acoustics

Stanford University

Email: putnam@ccrma.stanford.edu

Abstract

In this paper we try to establish the system order for the correct representation of a set of room transfer functions (RTF), in order to partition memory between the common recursive part and the non-recursive part specialized for each RTF. To this end, we apply a few system theory concepts on a set of simulated rectangular rooms, whose impulse responses were generated using the image method [1]. A further validation of our results is provided by an analysis of the frequency density of a comb-filter modeling of the recursive part.

1 Introduction

Several models have been proposed in the literature for the representation of room transfer functions. FIR filters have been used for echo cancellation, especially using adaptive filtering techniques [2].

Recursive filters are often used for artificial reverberation. These usually take the form of comb and allpass filters, and are often cascaded with tapped delay lines [9, 6, 4, 8].

More recently, it has been proposed to model the RTFs of a given room by a set of common poles, provided by an IIR filter, and an FIR part which varies according to source and receiver positions within the room [3]. A similar approach is taken in [8] for building artificial reverberators, where the recursive part is implemented by means of a feedback delay network (FDN), and the FIR part is a tapped delay line. In this latter model, the FIR part can be associated with the early reflections of the room, while the IIR part can be interpreted as a representation of normal modes and diffusion. This physical correspondence allows one to control the filter parameters in a natural way, and physical consistency is automatically achieved.

An open question that we address in this paper is that of the memory requirements necessary for the correct representation of a set of RTFs of a room. Furthermore, we address the issue of partitioning this memory between the common recursive part and the non-recursive part, which is specialized for each RTF.

In our investigation, we apply system theoretic

concepts on a set of simulations we have conducted using the image method [1]. Since the image method can be shown to converge to a correct representation of normal modes [1], we believe that it constitutes a useful tool for studying the room response, even in the steady state. This is an approach which guarantees a large and controllable dynamic range. This allows for a straightforward and repeatable construction of reference rooms, as opposed to actual measurements in actual rooms. In another project, we used the same set of simulations for determining numerical conditions for invertibility of RTFs [7].

2 Methodology

In this section we briefly introduce the system theory background which is needed for our investigation. A thorough treatment of this material can be found in [5].

The Hankel matrix of a single-input-single-output system can be constructed from m samples of a simulated impulse response $h(\cdot)$:

$$\mathbf{H}_h = \begin{bmatrix} h(0) & h(1) & \dots & h\left(\frac{m}{2}-1\right) \\ h(1) & h(2) & \dots & h\left(\frac{m}{2}\right) \\ \vdots & \vdots & \dots & \vdots \\ h\left(\frac{m}{2}-1\right) & h\left(\frac{m}{2}\right) & \dots & h(m-1) \end{bmatrix} \quad (1)$$

Given a linear system in its matrix representation

$$\Sigma = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \quad (2)$$

H_h is the product of the Observability and Reachability matrices [5]. Thus, a singular value decomposition (SVD) of H gives information about the dimensionality of a minimal realization of Σ , which is equivalent to its reachable and observable part, i.e., it gives the memory requirement of the system. In particular, the matrix S of the singular values has a form

$$S = \begin{bmatrix} s_1 & 0 & \dots & 0 & 0 \\ 0 & s_2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & s_n & 0 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix} \quad (3)$$

where n is the order of the reachable and observable part, i.e. of the minimal realization.

In practice, using measured or simulated impulse responses, the singular values s_i will never go exactly to zero, corresponding to the fact that an infinite state space would be necessary to represent three-dimensional wave propagation in discrete time. Nevertheless, we can plot the singular values on a log scale (see fig. 1) and consider n to be the abscissa corresponding to a reduction of, say, $60dB$ in the singular values.

3 The Experiment

In figure 1 we have depicted the singular values of a simulated impulse response for a room with edges $8m \times 10m \times 6m$, with a sample rate of $1KHz$. A line has been drawn to interpolate the data from the sample 20 to 480. We cut the spike in very-low frequency, since we believe it is due to the fact that the simulated impulse response has a strong DC component. We see that the singular values go down with a slope of about $0.26dB/sample$, thus indicating that about 231 memory cells are needed to represent the RTF with an “accuracy of $60dB$ ”, and about 369 memory cells are needed for $96dB$ of accuracy.

It has to be noticed that, throughout our discussion, the accuracy of representation is measured directly in the space of singular values of the impulse response to be modeled, rather than actually modeling the room and measuring a signal-to-noise ratio. For instance, on the basis of the example of fig. 1, we can say that, in principle, there exists a discrete-time linear system having 369 state variables, which can reproduce the impulse response $h(\cdot)$ within the 16 bits of accuracy.

Now, let's consider more than one impulse response simultaneously, namely a system with r inputs and one output. The product of the controllability and reachability matrices now takes the

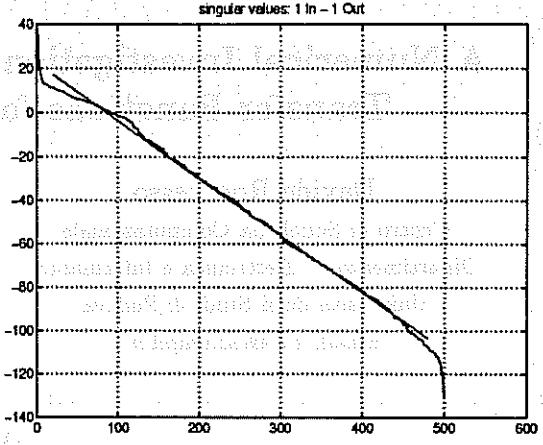


Figure 1: Singular values for 1 input - 1 output

$$\begin{aligned} h(0) &= [h_1(0) \ h_2(0) \ \dots \ h_r(0)] \\ h(1) &= [h_1(1) \ h_2(1) \ \dots \ h_r(1)] \\ h(2) &= [h_1(2) \ h_2(2) \ \dots \ h_r(2)] \\ &\vdots \\ h(\frac{m}{2}-1) &= [h_1(\frac{m}{2}-1) \ h_2(\frac{m}{2}-1) \ \dots \ h_r(\frac{m}{2}-1)] \\ h(\frac{m}{2}) &= [h_1(\frac{m}{2}) \ h_2(\frac{m}{2}) \ \dots \ h_r(\frac{m}{2})] \\ h(m-1) &= [h_1(m-1) \ h_2(m-1) \ \dots \ h_r(m-1)] \end{aligned} \quad (4)$$

where any $h(\cdot)$ is a $1 \times r$ vector. We can still perform a SVD on this matrix and compute the memory requirements for a correct simultaneous representation of the different RTFs.

When a second input is added to the previous one-input-one-output impulse response, we notice that the slope of the singular values reduces to about $0.2dB/sample$ (fig. 1), thus indicating that about 300 memory cells are needed to represent the RTFs with an “accuracy of $60dB$ ”, and about 480 memory cells are needed for $96dB$ of accuracy.

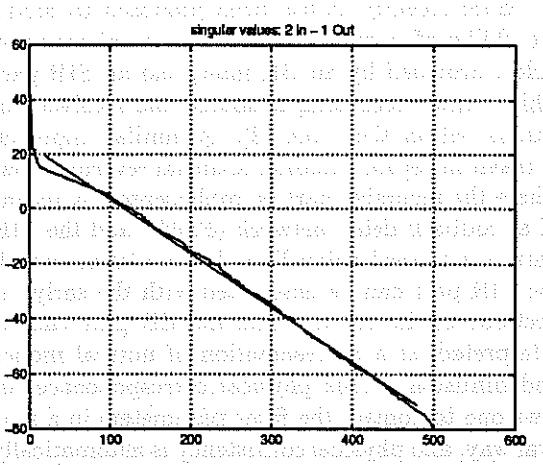


Figure 2: Singular values for 2 inputs - 1 output

Given an accuracy of $60dB$ we have, for one input and one output

$$M_1 = 231 \text{ memory cells} \quad (5)$$

and, for two inputs and one output

$$M_2 = 300 \text{ memory cells} \quad (6)$$

Let us assume that there is a common recursive part M_r . Let's also assume that the non-recursive parts are totally independent for different inputs or outputs. This means that, if in the one-input-one-output case we have a non-recursive memory requirement M_f , in the two-input-one-output case the non-recursive memory requirement has to be $2M_f$.

We come up with the equations

$$\begin{aligned} M_1 &= M_f + M_r = 231 \\ M_2 &= 2M_f + M_r = 300 \end{aligned} \quad (7)$$

The system (7) can be solved and it provides

$$\begin{aligned} M_f &= 69 \\ M_r &= 162 \end{aligned} \quad (8)$$

In other words, in a single RTF representation, about 70% of the memory should be put on the recursive part.

This simple memory allocation has been done on the basis of a somewhat arbitrary assumption. The same assumption has been taken in the past with no formal justification. We now try to justify the presence of a common IIR part and of independent FIR sections by considering a third RTF. When taking an additional input and computing the SVD on the respective Hankel matrix, we get a slope of 0.17dB/sample in the singular values (fig. 3), thus indicating that about 352 memory cells are needed to represent the RTFs with an "accuracy of 60dB ", and about 564 memory cells are needed for 96dB of accuracy.

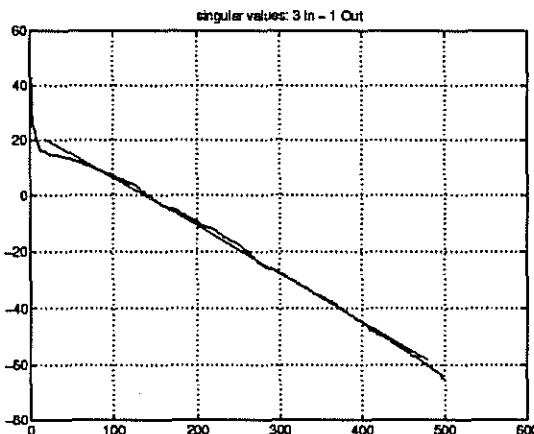


Figure 3: Singular values for 3 inputs and 1 output

If our model is correct, the system

$$\begin{aligned} M_1 &= M_f + M_r = 231 \\ M_2 &= 3M_f + M_r = 352 \end{aligned} \quad (9)$$

shouldn't give a solution very different from (8). In fact, we get

$$\begin{aligned} M_f &= 61 \\ M_r &= 170 \end{aligned} \quad (10)$$

so that about 73% of the memory should be put on the recursive part. This is different from the previous 70%, but the difference is compatible with the tolerance of the line fitting and with the somehow arbitrary choice of the lower and higher cutoff frequencies for the linear interpolation.

Our calculations are very sensitive to the actual placement of sources and receiver. In particular, symmetric positions should be avoided, since they break the assumption of independent non-recursive transfer functions.

4 Physical considerations

The experimental evidence fits well with the memory splitting as determined by strictly physical considerations of the model [8].

In the cited model, the digital representation of the recursive part of a RTF is essentially a superimposition of recursive comb filters. If g is the attenuation coefficient in the feedback loop of a comb filter, then the approximate bandwidth of one of the resonances is

$$\Delta f = \frac{(1 - g^{1/n}) F_s}{2\pi} \quad (11)$$

where F_s is the sampling rate. The coefficient g is related with the reflectivity of the walls. Each comb filter in the model is associated with a harmonic series of normal modes, and with a direction in space. Progressively shorter delay lines are used for representing higher order modes so that, assuming that the reflectivity is kept constant, the bandwidth (11) also increases.

The richness of the model increases as long as an increasing number of directions in space is considered. It is easy to compute the number of memory cells needed for the case at hand, for an increasing number of space directions (fig. 4). It makes sense to stop the increase in the number of directions when the bandwidth of a resonance exceeds the mean distance among two adjacent resonances. It is well known that in an actual three-dimensional enclosure, this distance decreases quadratically with frequency in a rectangular environment, hence at some point it will be less than Δf . From a simulation perspective, it makes sense to try to reproduce the resonances which are clearly separable, thus limiting the number of comb filters to be used. As stated in [8], this corresponds to finding a limit to the number of directions in space where we are considering plane-wave propagation.

If we approximate our room by sampling 8 directions in space, we find that the mean number of resonances per Hz (i.e. the frequency density) is about

$$f_d = 0.27 \quad (12)$$

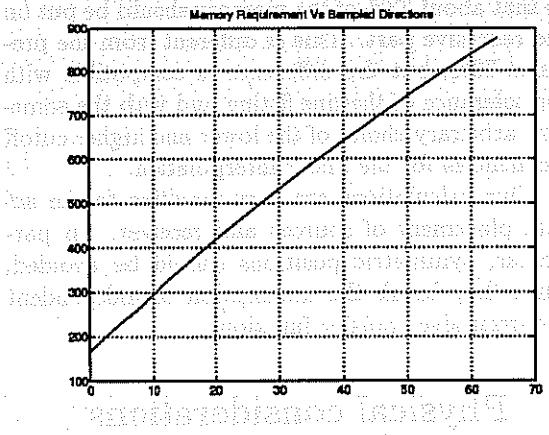


Figure 4: Memory requirement for an increasing number of sampled directions

The resonance bandwidth in low frequency ranges from 1.8 Hz to 4.3 Hz . These values are very close to the reciprocal of the frequency density (i.e. the mean peak-to-peak distance) which is 3.7 Hz .

The conclusion we can draw from these speculations is that 8 is a good choice for the number of sample directions, and that it is not worth going much higher, since the resonance peaks would not be resolvable. Now, from fig. 4 we can deduce the memory occupation of these 8 comb filters, which is 270 memory cells. Somewhat surprisingly, this number is close to the memory requirement for the recursive part, as computed by means of the singular value decomposition of the Hankel matrix.

5 Conclusion

We have outlined two criteria for estimating the memory requirements for a discrete-time modeling of room transfer functions. The first method is based on the singular value decomposition of Hankel and block Hankel matrices, while the second method relies on the discrimination of resonance peaks in the frequency domain. Even though these two criteria do not show an evident relationship, we found that they provide similar estimates in a practical case. This result is a clear evidence of the fact that any RTF can be split in an FIR part and an IIR part, and that this latter part is independent on the source or receiver position, being given by the common resonances of the room. While this split of RTFs was assumed in previous works, this investigation shows that it is justified by numerical results and it indicates a couple of ways of setting the system dimensions.

References

- [1] J.B. Allen and D.A. Berkley, Image Method for Efficiently Simulating

Small-Room Acoustics, *J. Acoust. Soc. Am.*,

vol. 65, 1979.

- [2] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 1991.

[3] Y. Haneda, S. Makino, and Y. Kaneda,

Common Acoustical Pole and Zero Modeling of Room Transfer Functions, *IEEE Trans. on Speech and Audio Proc.*, vol. 2, n. 2, apr. 1994.

- [4] J.M. Jot, An Analysis/Synthesis Approach to Real-Time Artificial Reverberation, proc. Int. Conf. Ac. Speech and Sig. Proc., San Francisco, 1992.

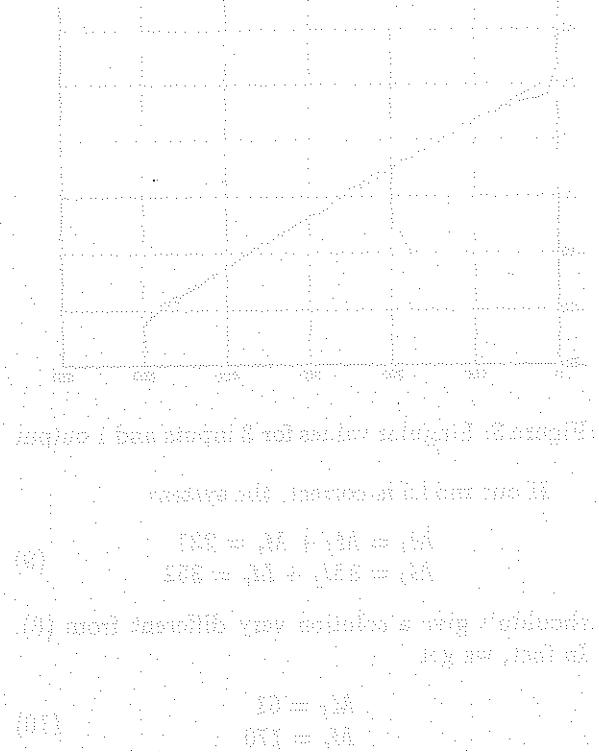
- [5] T. Kailath, *Linear Systems*, Prentice Hall, 1980.

- [6] J.A. Moorer, About this Reverberation Business, *Comp. Music J.*, vol. 3, n. 2, 1979.

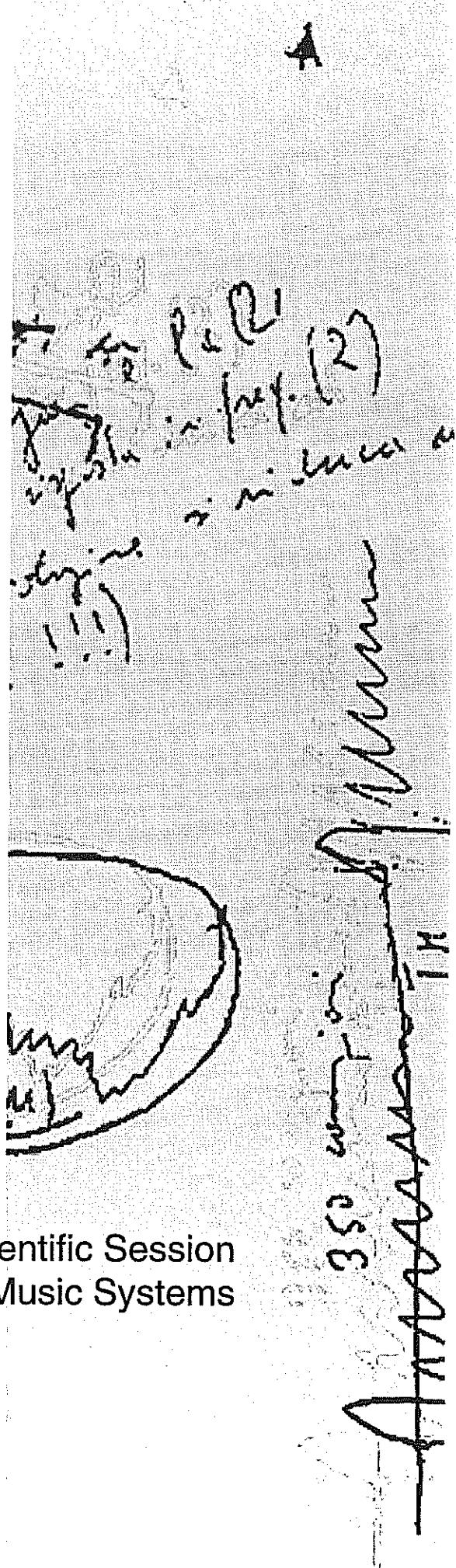
- [7] W. Putnam, D. Rocchesso, and J.O. Smith, A numerical Investigation of the Invertibility of Room Transfer Functions, proc. IEEE ASSP workshop on Applications of Signal Processing to Audio and Acoustics, Mohonk, NY 1995.

- [8] D. Rocchesso, The Ball within the Box: a Sound Processing Metaphor, *Comp. Music Journal*, vol. 19, n. 4, 1995.

- [9] M.R. Schroeder, Natural Sounding Artificial Reverberation, *J. Audio Eng. Soc.*, vol. 10, n. 3, 1962.



Scientific Session
Computer Music Systems



alarme alarme alarme

Newton MusicKit

Scrittura musicale con pen-computer

Paolo Fossati⁺, Goffredo Haus⁺, Alberto E. Minetti^{*}

⁺ Università degli Studi di Milano, Dept. of Computer Science,
via Comelico 39, Milano, Italy, tel.+39-2-55006338
e-mail music@imiucca.csi.unimi.it, fossap@pippo.sm.dsi.unimi.it

^{*} Istituto Tecnologie Biomediche Avanzate, C.N.R., via Ampère 56,
20131 Milano, Italy, tel. +39-2-70643362, fax 2663030,
e-mail minetti@itba.mi.cnr.it

Abstract

Current composing technology, by using a mouse pointer to select notation items on a palette, does not allow a fast and efficient input on the screen. Differently, Pen-based computers can recognize hand written musical score and position the notation items on the staves according to algebraic rules just after each single stroke (a stroke is a xy coordinate array, equally spaced in time, from the last pen-down to the last pen-up). Furthermore, the editing process of already written scores is greatly enhanced by the pen-screen interaction. Apple Newton has been used as a starting platform for this project. Despite of the reduced screen area, the high pixel resolution (320 x 240) and the associated RISC processor allow a rather natural handwriting and a fast recognition (future versions will include wider screens). The recognition algorithm and all the managing software has been programmed by using the object-oriented language for Newton named Newton ToolKit.

1. Caratteristiche generali

I sistemi per l'edizione elettronica di testi musicali sono generalmente basati sulla selezione mediante mouse di simboli precostituiti e organizzati in font e/o di primitive grafiche per il disegno di simboli di forma e dimensione variabile (legature, barrette, forcelle, parentesi, etc.).

La scrittura mediante penna consente potenzialmente tanto una maggiore flessibilità nel repertorio di simboli quanto un'introduzione più efficiente e intuitiva. Il pen-computer è lo strumento che può consentire di trasferire la scrittura tradizionale a mano su computer.

La tecnologia software necessaria è in parte affine a quella dei sistemi per la lettura automatica di partiture (OMR) [Selfridge-Field, 1994]. Si differenzia principalmente poiché in questo caso si tratta di riconoscere il movimento dei punti (la traccia della penna) con relativi parametri (pressione, tipo di pennino, velocità, etc.) piuttosto che i punti della bitmap di una partitura digitalizzata.

In questo ambito si collocano alcuni progetti di ricerca, come quello in corso presso l'università di

Berkeley [Garrett, 1994] e naturalmente anche il presente progetto.

Newton MusicKit è infatti stato concepito come una applicazione di utilità per il compositore che vuole scrivere mediante penna la sua partitura direttamente sullo schermo del computer. Sia perché il progetto è nella sua prima fase di attuazione, che per motivi di potenza di calcolo insufficiente del modello di pen-computer attualmente disponibile, il prototipo da noi realizzato costituisce un primo passo nella realizzazione della tecnologia software necessaria per la scrittura musicale al computer mediante penna. I limiti del prototipo attuale sono quindi da intendere come l'espressione dei vincoli qui citati.

2. Funzionalità e modalità operative dell'applicazione

L'inserimento degli oggetti è determinato dal movimento della penna sullo schermo di Newton. In particolare l'algoritmo di riconoscimento è attivato quando la penna è sollevata dallo schermo. Durante l'inserimento di un simbolo è quindi

necessario non distaccare la penna dallo schermo a pressione di Newton fino a che non sia stata ultimata la scrittura del simbolo in questione. Da un'inchiesta, precedente il lavoro, effettuata su musicisti compositori di professione, l'imposizione non è parsa di eccessivo rilievo, ed alcuni di loro utilizzano questa tecnica per scrivere velocemente. Nel manuale d'utilizzo verranno dati consigli su come tracciare i simboli più complessi, come il diesis ed il bequadro, nel modo sopraccitato.

Per il riconoscimento di alcuni oggetti è necessario tracciare solo una parte di essi affinché questi vengano riconosciuti correttamente. Questa convenzione è utilizzata per la maggior parte dei tratti che dovranno essere riconosciuti come note.

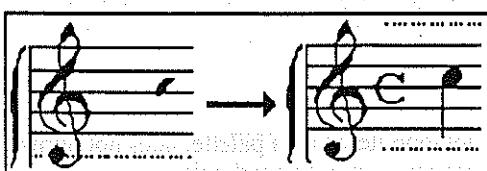


Fig. 1 - Tracciato di un simbolo riconosciuto come quarto.

Le note con la testa piena compaiono sempre con il gambo, rivolto verso il basso o verso l'alto a seconda dell'altezza in cui sono posizionati nel pentagramma. Per individuare una nota con queste caratteristiche è sufficiente tracciare una piccola spirale all'altezza desiderata sul pentagramma che compare sullo schermo di Newton.

La stessa convenzione è stata utilizzata per le note con la testa vuota. In questo caso si potrebbe verificare un equivoco tra la nota con valore di un intero che non richiede nessuna gamba verticale e la nota con valore di una metà. Il problema è stato risolto riconoscendo la nota di un intero solamente se all'interno della battuta non compaiono altri simboli e se il tracciato è posto verso il centro della battuta.

Questo è stato possibile perché la durata massima di una battuta non può superare il valore di un intero. Per ragioni di dimensioni dello schermo del dispositivo su cui si è lavorato, si è limitato l'intervallo dei tempi che possono assumere i valori tra l'ottavo e la nota di un intero.

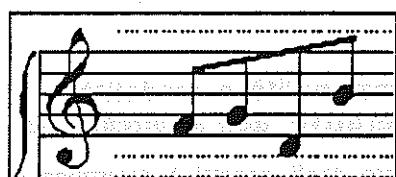


Fig. 2 - Gruppo di quattro note di un ottavo.

Le note di un ottavo vengono ottenute dopo il riconoscimento della nota di un quarto tracciando la curva che le caratterizza (o più semplicemente con un Tap) a fianco della stanghetta. Nel caso ci siano più note di un ottavo successive queste verranno legate automaticamente da una sbarra orizzontale che segue l'andamento dei simboli. La direzione della stanghetta dell'ultima nota inserita, verso il basso o verso l'alto, viene assegnata automaticamente a seconda delle note che la precedono. I gruppi sono formati al massimo da quattro note.

Per ottenere un accordo, possibile solamente con note dello stesso valore, si utilizzano le tecniche sopraccitate posizionando il tracciato sopra o sotto la nota riconosciuta. L'unica impostazione è quella di scrivere la nota che dovrà comporre l'accordo dalla parte opposta alla stanghetta della nota esistente. In caso di diversa necessità è comunque possibile ruotare la nota per variare la posizione della stanghetta.

E' possibile aumentare la durata della nota con il punto che dev'essere tracciato come una spirale di dimensione minore rispetto a quelle delle note, nelle vicinanze che seguono la nota riconosciuta. Altri simboli, come le alterazioni e le pause devono invece essere tracciati nella loro completezza.

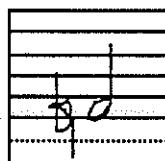


Fig. 3 - Alterazione.

Le alterazioni devono essere tracciate dopo che la nota a cui sono riferite è stata riconosciuta nelle vicinanze che la precedono. Dopo essere stata riconosciuta l'alterazione verrà posizionata automaticamente a fianco della nota e farà logicamente parte integrante di essa. I tipi di alterazione possibili sono il bequadro, il diesis ed il bemolle.

Per evitare l'utilizzo di menu per effettuare operazioni particolari, come la cancellazione o la selezione di uno o più simboli, si sono introdotte delle convenzioni che consistono nell'utilizzo di alcuni tracciati predefiniti. Per effettuare la cancellazione di un qualsiasi oggetto che compare sul pentagramma è necessario tracciare su di esso una X, mentre per selezionare uno o più simboli è necessario tracciare un cerchio intorno ad essi.

Sono state utilizzate voci di menu che riguardano la gestione dei file e la modifica di alcuni parametri di funzionamento della macchina, come lo spessore del tratto, laddove è necessario utilizzare chiamate di sistema.

Inoltre per agevolare l'utente e velocizzare il lavoro si è costruita una finestra di strumenti di lavoro dalla quale, selezionando una delle azioni indicate (che sono accuratamente descritte nel manuale d'utilizzo), si possono variare o modificare i parametri, di uno o più simboli selezionati, come la durata o l'altezza, o eseguire operazioni come la cancellazione di un gruppo di oggetti o il "Play" della selezione.

Per quanto riguarda il "Play" della selezione la qualità ottenuta è quella dello speaker di Newton e quindi di bassa qualità, ma utile per verificare la correttezza dei dati musicali inseriti. La possibilità di emissione di suono è fornita anche all'interno del menù "Option"; selezionando "Sound On" è possibile ascoltare il suono relativo all'altezza della nota riconosciuta.

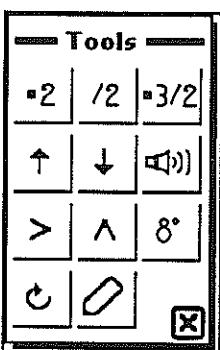


Fig. 4 - Finestra tool a disposizione nel programma.

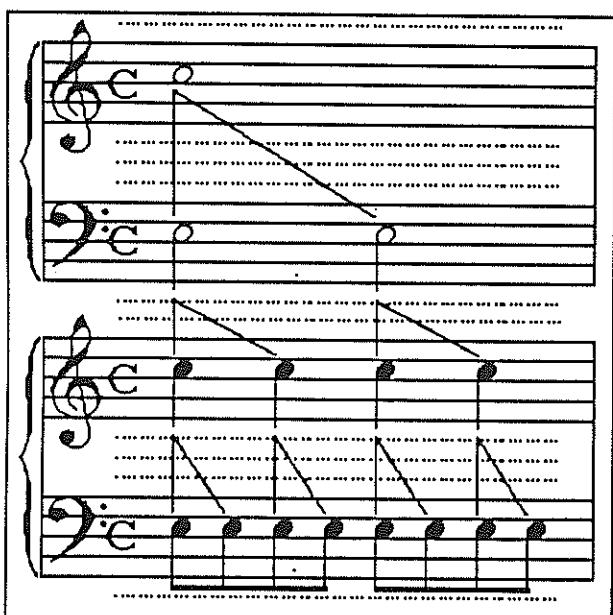


Fig. 5 - Schema di posizionamento delle note nel pentagramma

3 . Posizionamento e gestione delle note

Considerando tanto elementi di teoria musicale che le caratteristiche del software esistente per l'editing di partiture, il posizionamento dei simboli sul pentagramma avviene secondo lo schema indicato in figura.

Ogni simbolo occupa una posizione ben precisa, a seconda del tempo selezionato, ed è distanziato dagli altri simboli di una quantità di spazio proporzionale alla durata del simbolo che lo precede.

Durante l'inserimento dei simboli vengono effettuati controlli per verificare che non si ecceda il valore possibile all'interno della battuta. In caso si verificasse un errore di questo tipo, l'operazione di riconoscimento è annullata.

In corrispondenza di un pallino pieno tracciato in ultima posizione in una battuta dove manca solo un ottavo per essere completa, viene assegnato automaticamente questo valore e corrispondente figurazione alla nota riconosciuta.

Interessanti sono le funzioni di editing messe a disposizione dal programma sul set di simboli riconosciuti.

Innanzi tutto è da segnalare la possibilità di effettuare lo scroll dei pentagrammi quando quelli rappresentati sullo schermo di Newton all'atto dell'avvio del software MusicKit sono completi. Questa operazione è effettuabile utilizzando le frecce poste sul display standard di Newton. E' inoltre possibile, utilizzando il comando "Go To" nel menù "Option", raggiungere direttamente una battuta specificata evitando lo scroll delle battute intermedie.

La modifica dei parametri di un simbolo riconosciuto si può ottenere in alcuni casi agendo direttamente sull'oggetto in questione, oppure facendo riferimento alla finestra di tool precedentemente descritta.

Per quanto riguarda le note è possibile modificarne il nome andando a variare la posizione verticale nel pentagramma. Per effettuare questa operazione è necessario puntare la testa della nota con la penna per poi trascinarla in verticale sul pentagramma. L'operazione è possibile solo all'interno del pentagramma relativo ad una chiave musicale. È possibile variare l'altezza di una o più note di un semitonio utilizzando le frecce verso l'alto o verso il basso messe a disposizione nella finestra dei tool.

Sono possibili operazioni su parametri ritmici, come la possibilità di raddoppiare o dimezzare la durata di una nota. Questo tipo di operazione è effettuabile dalla finestra dei tool ed è applicabile ad un singolo oggetto. E' da notare che ogni volta che si compie una modifica ritmica della battuta, anche a seguito della cancellazione di uno o più oggetti, si verificherà un riordinamento della stessa con la

rigenerazione di tutti i simboli. Quindi la battuta risulterà sempre ordinata secondo lo schema precedente.

E' infine possibile l'inserimento di un oggetto in una posizione diversa da quella finale. Anche in questo caso il simbolo verrà collocato nella posizione corretta con il conseguente riordinamento degli altri simboli. Questa operazione risulta difficile da effettuare tra note con la durata di un ottavo per questioni di spazio.

Nelle funzioni di editing è da segnalare la possibilità di aggiungere segni dinamici quali legature, oltre ad accenti e simbolo di ottava.

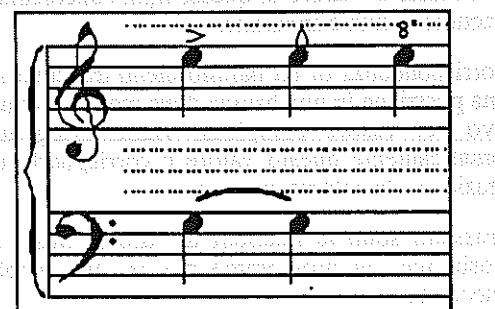


Fig. 6 - Accenti e simbolo di ottava.

Per i simboli di accento e ottava è necessario utilizzare la finestra dei tool a disposizione del programma. Questo tipo di operazione è applicabile anche ad una selezione multipla di simboli.

Le legature invece vanno tracciate direttamente sopra le note interessate cercando di tenere come punti di riferimento di partenza e di fine del tratto i centri delle note stesse.

Il lavoro può infine essere salvato all'interno di un file e recuperato successivamente per essere completato o rivisto. I files memorizzati possono anche essere eliminati da programma utilizzando la voce "Remove" dal menù file.

Durante le operazioni che richiedono un certo tempo, come la rigenerazione di una battuta complessa o il salvataggio e recupero di un file, compare o la figura di un cronometro nel centro dello schermo che indica l'attesa, oppure una finestra con la descrizione dell'operazione in corso.

4 . Considerazioni finali

Il primo prototipo di Newton MusicKit incoraggia ad ipotizzare sviluppi futuri e potenzialissimi applicazioni di questo mezzo di composizione musicale. Le limitazioni oggi imposte da uno schermo verticale di piccole dimensioni, ossia 24 battute per pagina e la massima risoluzione temporale confinata all'ottavo, saranno superate con

l'arrivo di nuovi modelli di Apple MessagePad. La versione finale, inoltre, potrà utilizzare la comunicazione seriale o all'infrarosso per trasmettere, con protocollo LocalTalk, gruppi di battute in formato Standard MIDI File (o General MIDI) a un desktop computer per il successivo inserimento in spartiti più complessi o per essere eseguite.

Quando i dispositivi di pen-computing includeranno un acquisitore/produttore sonoro di adeguata qualità, sarà possibile non solo eseguire la partitura in situ (accordi compresi), ma anche progettare un modulo di riconoscimento della nota da un input acustico.

L'attuale stato di avanzamento nello sviluppo di MusicKit preclude un uso professionale a breve termine, a meno che il compositore lo impieghi come block notes portatile per idee musicali. L'applicazione più immediata riguarda invece la didattica musicale. L'intuitività dell'interfaccia utente, la velocità di inserimento di partiture, e il poter stampare ed eseguire semplici sequenze monodiche consente allo studente di verificare in tempo reale il risultato di cambiamenti d'altezza e di durata delle note. Inoltre, sfruttando le capacità di interconnessione offerte dall'hardware (Apple MessagePad può essere connesso in rete LocalTalk con computer ed altri MessagePad) si può realizzare un'aula per l'educazione musicale dove l'insegnante si colleghi con ogni studente per verificare e supportare l'apprendimento.

La stessa struttura software, opportunamente modificata, potrebbe servire a creare una raccolta di esercizi basati sul meccanismo quiz/gioco volta a sviluppare l'attitudine musicale. Anche in questo esempio l'inserimento delle risposte tramite penna digitale (insieme al feedback acustico in tempo reale) rendono l'esercitazione più intuitiva.

Ringraziamenti

Si ringraziano I.T. Software s.r.l. di Milano per il supporto tecnico-logistico, ed Apple Computer Italia SpA per aver messo a disposizione un pen-computer Newton (MessagePad 110), un computer Macintosh 840 AV ed il software di sviluppo.

Riferimenti bibliografici

[Selfridge-Field, 1994] E. Selfridge-Field: "Optical Recognition of Musical Notation: a Survey of Current Work", in *Computing in Musicology* -- Vol. 9, CCARH - Stanford University, 1994, pp. 109-145.

[Garrett, 1994] G. E. Garrett: "The Activity of CNMAT at Berkeley", *Computer Generated Music Newsletter* - Vol. 2, N. 2, IEEE Computer Society Press, Washington, 1994, pp. 5-8.

A Gesture Interface Controlled by the Vocal Tract

Roberto Bresin, Nicola Orio
E.mail: rb@csc.unipd.it

C.S.C. - Centro di Sonologia Computazionale, Università di Padova
Padua, Italy

Abstract

The interaction between musicians and electronic instruments, by now, is not deep like with traditional acoustic instruments. We can recovery from this situation increasing the amount of information the musicians gives to the synthesis algorithm. In our work we realize a new gesture interface which can be used while playing a typical MIDI instrument: a keyboard or a guitar. Control is obtained with postures of the vocal tract, so the user have only to change the shape of his mouth. With two wave guides the interface puts and picks up a signal in the vocal tract, waveguides are very little troublesome for the musician: they reach the mouth just like a clarinet mouthpiece. Analysing the signal a software routine estimates the oral cavity filter parameters and extracts the needed information about the posture musician is assuming. We use the Linear Prediction Method to make that analysis. Then, using the Main Component Method, we map the user's vocal tract postures in three-four dimensional orthogonal space; this method needs a short training in which the musician assumes all the postures he/she can or he/she wants to use for controlling. The array describing the position of a particular posture in this space it's used for the control. It can be used to modify in real time, the interface gives a new output every 0.1 seconds, a synthesis algorithm or musical effects. It's also possibly use the interface in other field, for example a mouse for disabled, an help in logopedia or in phonetics.

1 Introduction

It's a common opinion that playing an electronic music instrument gives artistic results not comparable with those obtained with acoustic instruments.

We think that this gap is not due to a less potentiality of the new instruments but to the lower control the musician has on them: there is still a low interaction between musicians and electronic instruments and so there is a low amount of information the musician can give to the sound synthesis algorithm.

To recovery from this situation, we propose a new gesture interface. We want this new interface to have some peculiar features. First of all that it could be used while playing a common MIDI instrument (a keyboard or a guitar), so it should not collide with movements and postures necessary to play. Another important feature is the control dimensionality, which has to be at least 2; otherwise all the work will be unnecessary because there are many one-dimension controllers. Then it should be also fast in response, easy to use and, if possible, cheap to build.

2 The EK Controller

The gesture interface we projected is controlled by musician's mouth: particularly with postures of the vocal tract. We choose the mouth because, to play normally, the musician has to be free hand, and also because in general we have a greater control on our mouth shapes than other body parts like feet or neck and so on. We are interested in postures and not in movements because we think that, from one side, controlling will be easier and, on the other, the musical interpretation is more related with positions than with speed and direction (note that the musician is already playing with movements, on which speed and direction depends volume and pitch).

The musician also does not have to make something distracting like speaking, making sounds or simply blowing like with the breath controller: he/she has only to change the inside geometry of his vocal tract.

From an aesthetical point of view the controller remembers the kazoo: so we called it EK Controller (Electronic Kazoo).

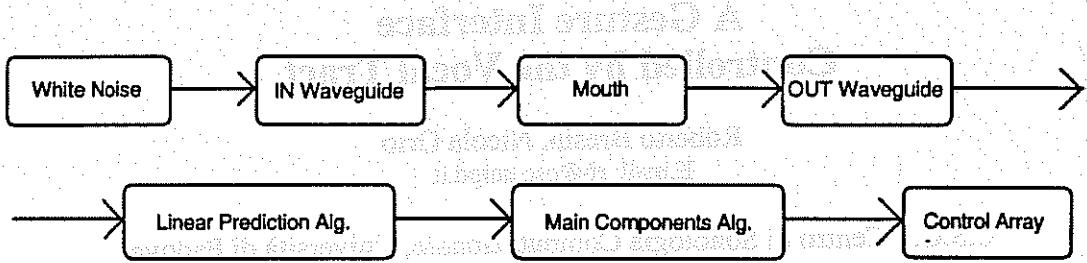


Fig. 1
The EK Controller Model.

3 Man-machine Interface

As we just say, we are interested on vocal tract postures. These are estimated considering the vocal tract like a filter; in particular it can be considered an all-pole filter. Estimating filter parameters means to have the necessary information on the shape of musician's mouth. To evaluate these parameters we need to analyse how a known signal is modified by the filter. To make this we use two waveguides; they are enough small to be very little troubling for the user: they reach the mouth like, for example, a clarinet reed. The first waveguide, which on its outside end has a loudspeaker, put a signal inside the mouth; the second one picks up the filtered signal and records it with a microphone. At the moment the signal is a white noise, generated by a software routine, sampled at 22 KHz. But we tried also with other kind of signals: square and triangle wave, ramps and pulse train. We found that best results in terms of control are obtained, apart from the white noise, with the pulse train; this is interesting because in this way it's possible to use a very simple hardware signal generator instead of the software routine, relieving computer work.

The signal recorded by the microphone is then sampled with an AD converter at a sampling rate of 8 KHz (the standard rate for speech-like signals). We are forced to choose this rate, and also the one for the noise, by the characteristic of the computer we used. In fact we develop the interface on a NeXT computer, which has, built-in, a microphonic input sampler at 8 KHz and two audio outputs at 22 KHz and 44 KHz.

4 Detecting Mouth Shape

Once we have the sampled signal we have to analyse it. There are many algorithms developed for sound, and especially speech, analysis. We imposed the following characteristics: stability, good results with low sampling rate and few samples, real time working.

We choose the well-known Linear Prediction Method (LPC) developed by Markel and Gray (Markel et al., 1976). We use a model simpler than LPC; in fact only vocal tract characteristics are involved, and so we do not need to consider lips and glottis. This leads to a model which is nearer to the ideal condition of an all-pole filter than LPC. This choice explains also why we use a white noise signal to excite the system, in LPC model the vocal cords signal is approximated with a noise; but a good approximation of that signal is also the pulse train, and this explains the results we explained before.

The LPC method can calculate two sets of filter parameters: the polynomial coefficients of the transfer function and the reflection coefficients. We use the second set; it gives the reflection coefficients of the vocal tract thought like a resonance tube, and it's an array which values goes from -1 to +1. It's possible to choose the array dimension in a small range of values that are related to the sampling rate: we decided to start with a dimension of 12; we tried also with bigger dimensions but results were similar.

5 Obtaining Effective Control

For having a real, effective control we can't use the reflection coefficients. In fact there is a deep correlation among the elements of that array, for example the second one has always opposite sign of the first one. And it's hard to imagine how a man succeed in moving the mouth in 12 uncorrelated directions.

So we need to extract the real information that the reflection coefficients array is carrying.

The problem can be subdivided as follows:

- What's the effective dimension of the information carried from an array of correlated elements?
- Is it possible, having in input an array of correlated elements, obtaining a second array, with a smaller dimension, and with independent elements, which carries the same information of the first one?

Finding the solution of this two points means obtaining a map of the vocal tract postures; and so we can use this map to have a control array the elements of which can be used to modify some parameters of a sound synthesis algorithm

We find the solution looking how were treated similar problems. In particular this kind of dimensionality problems are found in statistics. In that case the tag is how to describe a population who has some measurable characteristics with the minimum amount of data.

Usually they start with a number of sampling of the population, then they study the cross-correlation among elements trying to find what are the most important combinations.

Similarities with our problems are evident so we tried with an algorithm used in statistics, the Main Component Method (Johnston, 1987): after evaluating the dimension of the information carried by the sampled arrays, it calculates a matrix. This is the projection matrix from the high dimension space of correlated elements to the smaller one with independent ones.

To work correctly this algorithm needs, as we said, a number of sampling of the population. So we provide for a short training period: the user has to assume all the postures he/she can (or he/she wants to use for the control), and these are used to evaluate the control dimensionality and the transform matrix.

After this, all we have to do is to premultiply the matrix with the reflection coefficients array, and we obtain our control array.

This training is useful also because it permits that each new user customises the interface, so to reach all the combinations of the control array.

6 Results

We built a prototype of the gesture interface and we started to analyse the outputs.

Firstly we wanted to know if the system worked correctly. So we directed the reflection coefficients array to the computer screen, just to see if to a particular posture always corresponds the same output configuration. This was verified, apart from a small oscillation of the values around an average value. This oscillations may be explained considering small errors in numeric approximation, and involuntary micro movements of the mouth.

A simple way to avoid this is to make a weighted average with the current output value and some previous values. This also gives a small hysteresis to the EK Controller when vocal tract postures change fast but we do not consider it a problem.

Then we started to study the outputs of the Main Component Method. After some training of about 500 different samples, we saw that the control

dimensionality was from 3 to 4. Especially when the user becomes more familiar with the interface, control dimensionality is always 4.

In our opinion this is a very important result, because other gesture interface controller does not reach a dimensionality so high without using hands or, in general, hindering common movements done during play. Instead, with our controller, we can easily play a keyboard or a guitar to give to the synthesiser information about pitch, timing and volume and add more information about timbre or effects parameters (like reverber, tremolo, chorus and so on) with the EK Controller.

Finally we studied how the control output changes when the user varies gradually his/her mouth shape from one phonemon to another. In the graphics plotted (Fig. 2) the user is changing his vocal tract postures from phonemon "u" to phonemon "e".

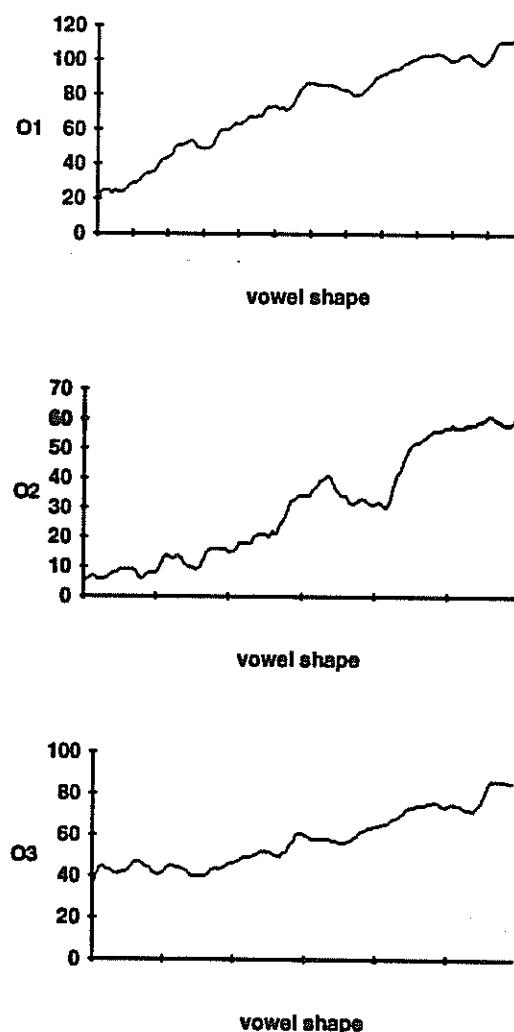


Fig. 2
In the graphs are plotted the 3 control outputs values when the user is changing mouth shape from phonemon "u" to "e".

We studied many of these trends to see how the outputs changes in correspondence with particular postures like the ones used for vowels. As it can be seen on this graphics, the output varies without many oscillations and we can recognise a simple path in the control space. By now the gesture interface EK Controller gives a new output every 0.1 seconds, but it's possible to increase the speed. Considering the use in musical live performance, we think that an higher speed was unnecessary.

7 Using the EK Controller

The next step was to see how long it takes to learn to correctly use the EK Controller. We have to note that the use of the interface is not immediate. This because the map of vocal tract postures is made algorithmically without considering particular postures that may be natural for the user.

So the musician has to learn to use the interface: just like a musician in general has to learn to use a new instrument. We measured approximately the time a new user spend to learn (times reported are not referred to a continuous use). To make this we tried to control only one output, then two and so on. Outputs were associated to sliders position on the screen.

The mono-dimensional control is quite immediate. After few minutes the user learns how to move the slider up and down, and after some times succeeds in moving it to absolute positions.

Also two-dimensional control is easy. In less than half an hour it's possibly to control the relative position of the sliders. It is harder to succeed in changing only one output value: it takes another hour.

Even more difficult is to control three and four dimensions, it will takes some days. A time however shorter than the time necessary to learn an acoustic music instrument.

8 Applications

The main application for this gesture interface is in controlling an algorithm synthesis during a live performance. So, for example, we can transmit the output values through a MIDI channel like control-change commands.

But the fact that the mouth gives the control makes the interface useful also in other fields. Persons with problems in using arms and disabled can connect to a computer using the interface like a mouse simulator, the first two dimensions will choose position and the third will make selections in a windows system.

After, considering that the interface makes a four dimensional map of the vocal tract postures, the

interface may be an aid in phonetics studies and in learning language for deaf subjects.

Further improvements of the EK Controller are possible. First of all it is possible to build the EK Controller on a stand-alone board using a DSP: in this way the use will be more practical for musicians who do not have a portable PC. It is also possible to analyse the training period results of many users (hundreds) to see if it is possible to subdivide the projection matrix into a small number of families. So, after this, new users could pass over the training period, they have only to choose the family in which they are.

But the much interesting improvement is about something to obtain an easier control. We are already studying how recognise some particular paths in the control space and to convert them in simple output configurations, for example the changing of only one element of the control array.

References

- N.H.Crowhurst, *Audio System Handbook*, Los Angeles: Blue Ridge Summit, 1975, pp. 1-54.
G.Gambolati, *Elementi di calcolo numerico*, Padova: Edizioni Cortina, 1985, pp. 159-170.
J.Johnston, *Econometrica*, Milano: Franco Angeli Editore, 1987, pp.354-366.
J.D.Markel, A.H.Gray: *Linear Prediction of Speech*, New York: Springer-Verlag, 1976.

J.Millman, *Circuiti e sistemi microelettronici*, Torino: Bollati Boringhieri, 1987, pp. 590-694.

N.Orio, *Progetto di un controllore gestuale azionato dal cavo orale*, Tesi di Laurea, Università di Padova; Relatori: Prof. G.De Poli, Ing. R.Bresin

J.Pressing, "Cybernetic Issues in Interactive Performance Systems", *Computer Music Journal*, vol. 14, n. 1, pp. 12-25, MIT Press.

A Fuzzy Approach to Performance Rules

Roberto Bresin, Giovanni De Poli, Roberto Ghetta
E.mail: rb@csc.unipd.it, depoli@dei.unipd.it

C.S.C. - Centro di Sonologia Computazionale, Università di Padova
Padua, Italy

Abstract

The realization of the Stockholm KTH musical performance rules with fuzzy technique is thereby explained. The technique allows the formulation of rules in an evocative manner because it uses words rather than mathematics.

1 Introduction

Computer musical performance has always had the shortcoming of being excessively dull, that is to say of not introducing those microvariations which give musicality to the performance of a musician. To overcome this shortcoming Sundberg and his colleagues of the Royal Institute of Technology (KTH) of Stockholm have proposed performance rules [5,8], which allow a better performance than that obtained without the interpretation of the score. These rules introduce certain microvariations in the duration (DR), in the intensity level (L) of the tone, and in the duration of rest between notes (DRO). Microvariations are generally given in terms of functions and always in mathematical terms. Performance rules follow an additive formulation, i.e. microvariations proposed by the rules are added to one another. Almost each rule has a multiplying coefficient (k) defined as emphasis parameter. This parameter allows the amplification or attenuation of the effect of the corresponding rule.

The use of fuzzy sets as a way to express uncertainty in a structured manner is now spreading in engineering [6,7]. Specific adjectives are associated to fuzzy sets so that linguistic deduction rules are obtained to establish adequate output values (from the fuzzy point of view). Numeric results can be obtained from these values so that a fuzzy controller can be build. Fuzzy methods allow the relatively easy implementation of communication aspects, typical of human reasoning where qualitative variables, unspecified concepts and subjective considerations play a fundamental role. Thus fuzzy logic leads to the creation of a controller able to "explain" its behaviour.

At Padua University we studied different approaches [1,2,4] to model performance rules.

based on neural networks and many-sorted-logic. Supposedly a linguistic approach rather than a mathematical one enables to obtain a more effective model for the concepts at the basis of musical performance. From this point of view a fuzzy controller could be particularly useful for the expert in the field of musical interpretation who deals with the problems of computer musical performance. To test the opportunities given by the fuzzy approach we developed a fuzzy controller based on KTH rules. This reformulation has to be taken into account not only as way to establish the rules according to the fuzzy technique but also as a starting point towards a further development of the Sundberg rules which are at the basis of this work.

2 Fuzzy Rules

The KTH rules have been reformulated keeping the original additional aspect. First of all a choice has been made to eliminate the rules whose formulation was not apt to a fuzzy realization. In particular, the rules with constant or linear deviations have not been considered because their fuzzy realization is possible but it is not interesting. Table 1 shows the rules that have been realized. The description of fuzzy reasoning ought not to be too complex in order to obtain a better interpretation. It is advisable that input and output variables should not be divided into a too large number of membership functions and the number of inference rules must be as limited as possible.

DDC 1: Durational Contrast	DPC 1B: High Loud	GMA 2A: Harmonic Charge	GMI 1A': Leap Articulation	GMI 1B: Leap Tone Duration
----------------------------------	-------------------------	-------------------------------	----------------------------------	----------------------------------

Table 1

The aim is to obtain a controller consistent with the linguistic aspect of its behaviour, therefore a

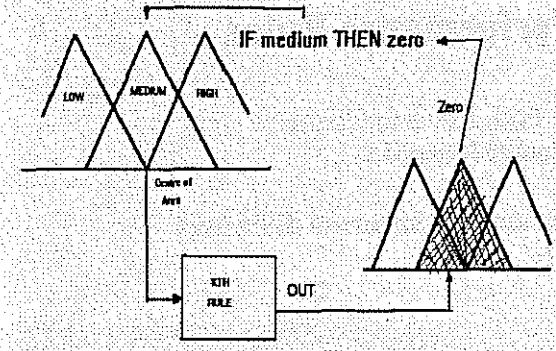


Figure 1

careful assessment of the adjectives to associate to membership functions is required so that meaningful inference rules can be achieved. It must be noted that to this aim the division into membership functions (and consequently the verbal characterization of an input set) must be identical for the same variables even though they are used in different rules. Although the formulation is still additional, the implementation of a rule is no longer independent from that of the others. In fact, implementing a rule with fuzzy method means choosing membership functions adequately and this choice will influence all the rules which use the same variable. The initial choice of membership functions has been made empirically; to obtain the final formulation a try and error procedure has been necessary, as it is for all fuzzy controllers. When building membership functions, it is better to analyse where output sensibility is greater: in this case it is better to thicken membership functions. In order to modify the controller behaviour, it is also possible to change membership functions shape. Figure 1 shows how we build fuzzy rules: from the central value of the input membership function we calculate the output value given by the KTH rules. The consequent of the fuzzy rule becomes the membership function "nearer" to this value. In order not to render the controller excessively complicated, with the introduction of a large number of membership functions, it is possible to use the following type of rules:

IF (DURATION IS "long")
THEN (DRO IS "very short" OR DRO IS "short")

These rules allow to obtain an intermediate result without introducing another membership function so that the interpretation of the controller is easier. From a logical point of view these rules imply that for a given input there are several correct output values (fuzzy values are meant here). We must remember that fuzzy rules have to be significant, i.e. they should have a linguistic meaning. So we must choice accurately our adjective. The inference method known as MAX-

MIN resulted the most suited for our fuzzy controller. The evaluation of the defuzzified value is obtained through the Center of Area method.

3 A Global Fuzzy Controller

Until now we described a fuzzy controller that considers rules one by one: final deviations are obtained adding deviations of each rule. Our previous experience with neural nets [2] showed that, using one global function of all the inputs instead that an additive formulation, a better model could be obtained. As showed in [3], it is possible to represent the KTH rules system with the following equation:

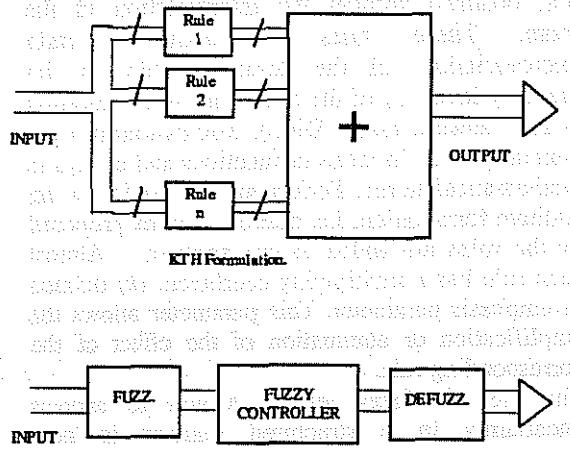
$$y_n = \sum_{i=1}^N k_i \cdot f_i(\bar{x}_n)$$

where the meaning of each parameter is the following:

y_n deviation of time or of loudness
 $f_i()$ function that represent the i-th KTH rule determined in an heuristic way

\bar{x}_n represent the vector of the parameters used by the KTH rules applied at the n-th note

k_i constant used to emphasize the deviation due to each i-th rule



The CSC model could be written in the following form:

$y_n = net(k, \bar{x}_n)$

where the meaning of each parameter is the following:
 y_n deviations of time or of loudness
 $net()$ Neural Net, non linear function that computes the deviations

- x_n is the same of equation
 k vector of constant or of variables which evolve very slow in time

DDC 1 "Durational Contrast"
DPC 2A "Melodic Charge"
GMI 4' "Articulation of Repetition"
GMI 1B "Leap Tone Duration"
GMI 1A' "Leap Articulation"
DPC 1 "High Loud"
GMA 1 "Phrase"

Table 2

As a result of these considerations we developed a new fuzzy formulation of the model of CSC. The result is a fuzzy global controller (GFC). In the previous paragraphs we choose rules that contain some uncertainty (i.e. rules that are good for fuzzy realization); so we avoid linear or deterministic rules. For the GFC point of view, however, we choose also linear or deterministic rules if their output is very meaningful. For the choice criteria we referred to [2]. Table 2 shows the rules taken into account.

From Table 2 is possible to observe that we need also information about the structure of the piece of music. For this reason, the GFC must also decide when a rule is to be applied. This task was not performed by the controller described in the previous paragraphs. Figure 2 shows the difference between the KTH model and the GFC. Each output variable is described by a non-linear function:

$$DRO = f_{DRO}(DR, \Delta N, PR, AR)$$

$$\Delta DR = f_{\Delta DR}(DR, \Delta N, PR, MC)$$

$$\Delta L = f_{\Delta L}(DR, N, MC)$$

where the meaning of each parameter is the following:

- DR duration in ms
 ΔN leap interval in semitones
PR end of phrase flag
AR articulation of repetition flag
N number of semitone (N=60 for C4)
MC melodic charge

To obtain these functions we used the standard value of KTH rules, i.e. we used rules with $k=1$. These functions are not obtained by adding rules of Table 2 because some saturation are inserted. Figures 3, 4 and 5 show a comparison between

deviations obtained by applying KTH rules and those obtained with the GFC. The score is Sonata K284 of W.A.Mozart. Generally, deviations given by the GFC follow correctly the deviations proposed by the KTH rules; the error obtained from the difference between the two performance is generally under the JND.

It should be noticed that GFC would require others input value to manage the quantity factor. This will increase the complexity of the controller as previously discussed. For the GFC there are two ways to insert quantity factor. The simpler is to multiply final value, in this case all effects that affect the same output are amplified at the same manner. If we don't want this we must insert quantity factor "inside" the controller.

Another choice is to not insert any quantity factor. In this case when we want apply the GFC to a piece with other values of k parameter, we must rebuild the controller (like neural nets where we must train another time the net). This method takes a lot of time but in this case the controller is simpler.

Also the tempo parameter affects the quality of performance. When tempo is very fast or very slow using, like us, absolute output value could determine some mistakes. For a further development it is possible to consider variables with their relative value instead with the absolute one. In this case we avoid the necessity of rebuilding rules when we want to perform scores with different tempo but is more difficult to consider rules (like Phrase) where deviations depend only by structure of score.

4 Conclusion

It has been shown that a fuzzy controller can quite accurately approximate the KTH performance rules both with the additive formulation than the global formulation.

By examining (we could also say "reading") the controller it is possible to quickly understand its functioning together with the reasons for the control action. In this sense fuzzy realization can be considered as a linguistic reformulation of the musical performance rules. However what is all the more useful is that it is possible to associate an algorithm that can be

used on computer. The most important aspect of a linguistic formulation is that it is comprehensible to everyone, it will be particularly useful as the basis for the cooperation with music expert in order to integrate or revise KTH rules.

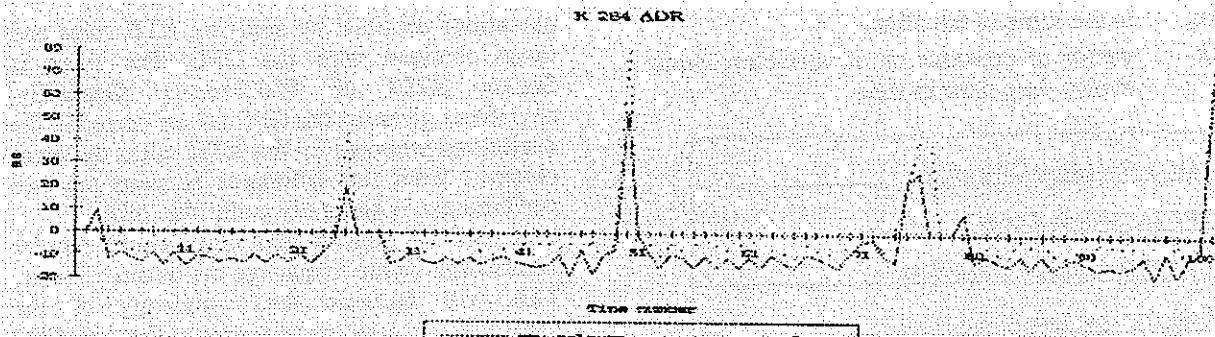


Figure 3

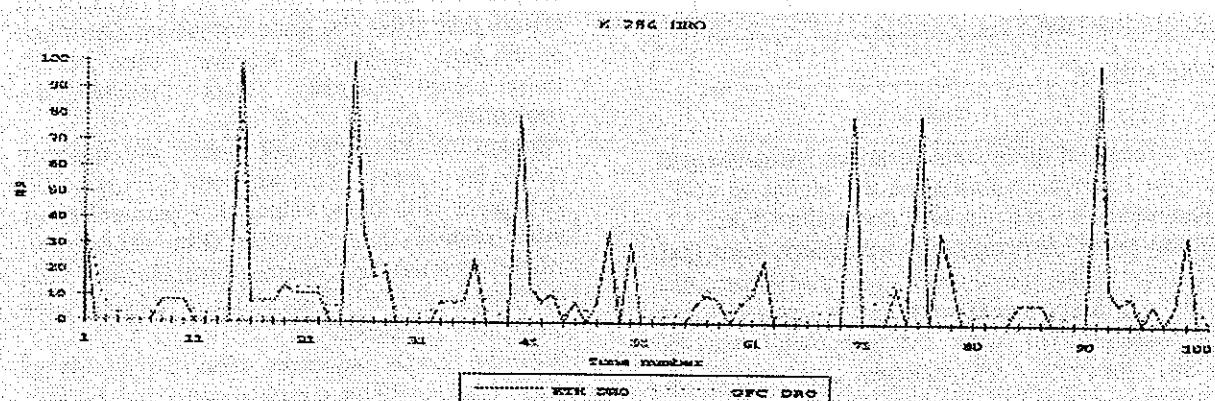


Figure 4

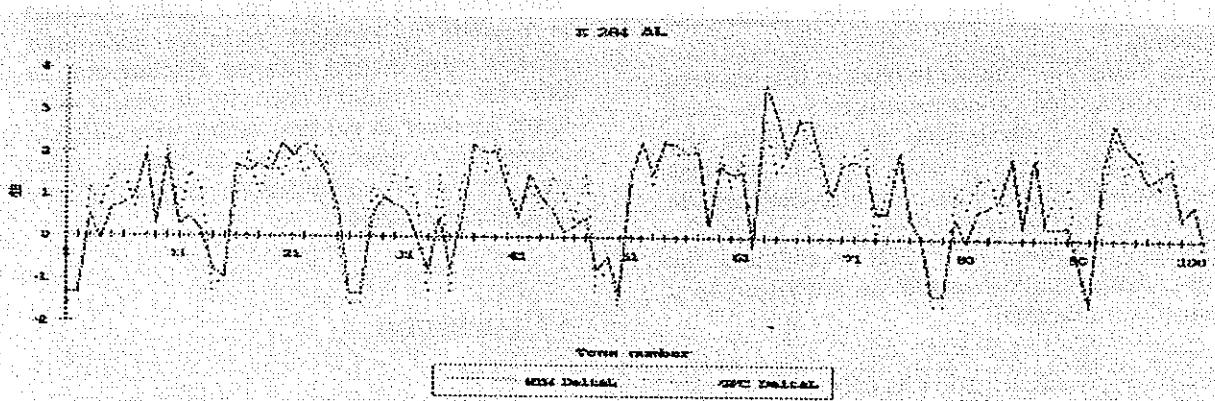


Figure 5

References

- [1] Bresin, R. - De Poli, G. - Vidolin, A. "Symbolic and Sub-Symbolic Rule System for Real Time Rule Performance". *Proceedings of ICMC 92*, pp. 211-214, 1992.
- [2] Bresin, R. - De Poli, G. - Vidolin, A. "Performance of Musical Scores by Means of Neural Networks". *Proceedings of Conference "Das Hör als Interpret"*. (Forthcoming) 1995.
- [3] Bresin R., Vecchio C. "Analysis and synthesis of the performing action of a real pianist by means of artificial neural networks". *Proceedings of the 3rd ICMPC*, University of Liege, pp. 353-354 1994
- [4] De Poli, G. - Ironi, G. - Vidolin, A. "Music Score Interpretation Using a Multilevel Knowledge Base". *Interface* (19), pp. 137-146, 1990.
- [5] Friberg, A. "Generative Rules for Music Performance: A Formal Description of a Rule System". *Computer Music Journal* (15) 2, pp. 56-71, 1991.
- [6] Kosko, B. "Neural Networks and Fuzzy System", Prentice Hall, 1992.
- [7] Lee, C. C. "Fuzzy Logic in Control System: Fuzzy Logic Controller". *IEEE Transaction on System, Man and Cybernetics* (20) 2, pp. 404-432, Feb 1990.
- [8] Sundberg, J. - Askenfelt, A. - Fryden, L. "Musical Performance: a Synthesis-by-Rule Approach". *Computer Music Journal*, 7 (1), pp. 37-43 Spring 1983.

Automated Optical Music Score Recognition and its Enhancement using High-level Musical Knowledge

Kia C. Ng, Roger D. Boyle and David Cooper *

Division of Artificial Intelligence, School of Computer Studies

* Department of Music

The University of Leeds,
Leeds, United Kingdom.

Email: kia@scs.leeds.ac.uk, roger@scs.leeds.ac.uk, d.g.cooper@leeds.ac.uk

Abstract

The aim of this project is to build a system which automatically transforms printed music scores into machine-readable format. In this paper, we outline a phased approach that disassembles complex interconnected musical symbols into low level graphical primitives before classification. After this, a reassembly process reconstructs the music score. This approach simplifies the task of recognition and provides an intermediate stage where musical syntax can be applied to refine recognition. A post-processing method that uses high-level musical knowledge to enhance recognition is presented, in which global information such as the time and key signatures are automatically estimated. With this information, earlier recognition can be reconfirmed and corrected if necessary.

1 Introduction

With the advance of electronic music, computer technology has increasingly become an indispensable tool in music. Notation, sequencing and sound synthesis programs have been developed and are widely used. Complicated and repetitive processes, for example transposition and part extraction, can be automated with the help of computers if the scores are represented in a machine readable format. Current general input methods, such as using an electronic keyboard, are time consuming and require human intervention. Optical Music Recognition (OMR), is an interesting and efficient input method for musical software (Carter 1988, Selfridge-Field 1994).

Trained musicians can sight-read musical scores at speed and with confidence; often the condition of these scores is not perfect however. Printing and scanning faults which are normally tolerable to the human brain can often complicate the recognition process. Examples include: broken symbols; broken and bowed stave lines; incompletely filled solid note heads; and flooding near junctions (Clarke 1990, Carter 1994).

2 Approach

To automate the process of music score recognition, we propose a phased approach in which features essential to an interpretation of the score are sought first, and verified by their mutual coherence, thereby allowing an intelligent search for more ambiguous features to be undertaken. The system takes a digitised music-score grey image as input, and automatically processes it and outputs a machine readable format of the score, such as a MIDI file.

2.1 Pre-processing

Firstly, an iterative thresholding process (Mardia 1986; Venkateswarlu) is used to find a threshold value and the image is binarised. The binary (black and white) image is then used to detect the skew. The original image is deskewed by rotation and the result is binarised with the threshold computed earlier (Ng and Boyle 1992) (see Figure 1).

The de-skewing is important because the breakpoints used by the sub-segmentation process (see Section 2.2) that disassembles complex musical features relies on correct vertical orientation of these features.

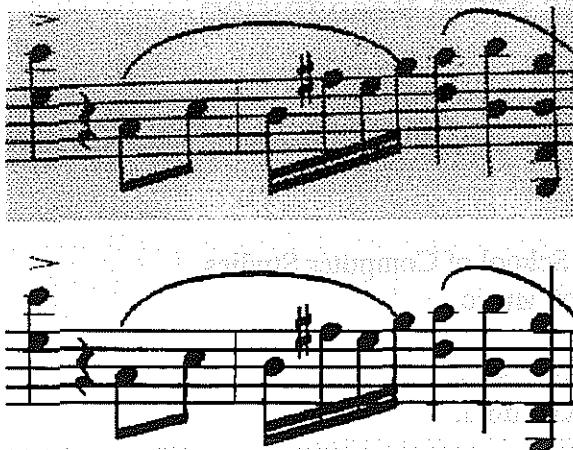


Figure 1. An example input grey image (upper) and the de-skewed and binarised version (lower).

Using the de-skewed image, individual stave lines are searched for and recorded to build the score layout. Musical symbols are generally written on five equally spaced lines that act as a kind of grid system, and hence the average height of a stave line and the average distance between two stave lines can be used as two important basic values for many subsequent processes. The sum of the average stave-line height and average inter-stave line distance form the fundamental unit (α) used by a classification process.

After stave detection, an initial segmentation using an image-labelling method (Sonka, Hlavac and Boyle 1993) is performed. Figure 2 shows an example output where stave pixels are marked with grey, and each object is displayed with its bounding box to show the result of the labelling process.

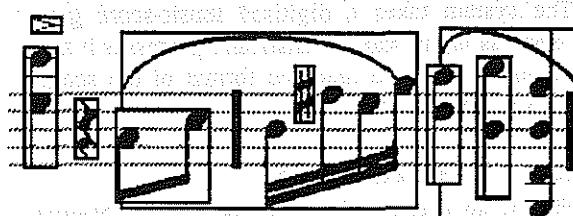


Figure 2. An example output after staves detection and initial segmentation.

After segmentation, a rule-based classifier (Ng and Boyle) recognises primitive musical symbols that are not connected, or symbols that are normally located at certain position with respect to their staff. For example, clef signs often appear near the beginning of a staff.

2.2 Sub-segmentation and Recognition

Musical notation is highly interconnected and there are many variations in shapes and sizes when symbols are grouped with one another. For example, consider the difference between the shape of a quaver and the shape of a beamed two-quaver group. We approach this problem with a process of sub-segmentation in which compound symbols are broken into groups of lower-level musical primitives such as note heads, vertical lines, horizontal lines (including slanted lines) and curves, before recognition. As an illustration, a beamed group of four semiquavers is sub-segmented into two horizontal lines (beams), four vertical lines (stems) and four note heads. The system can pick out curves (slurs, ties or phrase marks) which are overlaid or interconnected with other symbols.

The best break-point of a symbol is taken from the absolute second differential of the vertical projection histogram of the symbols (Ng and Boyle 1992).

After each sub-segmentation process, possible primitives are passed to a nearest-neighbour-classifier; some are then recognised with strong confidence and others with ambiguity. The two feature-vectors used by the classifier are the width and height (in α units) of the possible primitives.

The process of sub-segmentation and classification is iterated until the feature is recognised or the termination criteria are satisfied. Termination criteria for the sub-segmentation process include the size and the density of the symbols. If the density of a symbol is high (e.g. $> 75\%$), it is nearly solid and therefore not likely to decompose further. An example of a size-constraint is "less than 2α by 2α " because in this size range the symbol is likely to be a primitive or unknown.

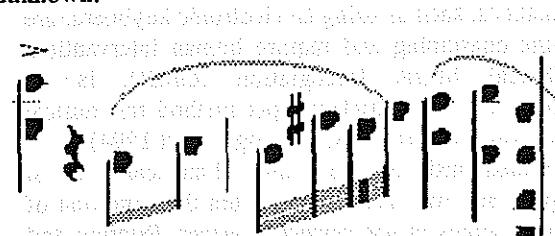


Figure 3. Sub-segmentation.

Figure 3 shows an example result of sub-segmentation. Sub-segmented primitives are relocated to illustrate the break-points, and traced primitives such as beams are greyed.

The primitives are then reassembled into the original score using basic musical notation syntax. Since many musical primitives complement each other, musical knowledge can be employed to enhance the recognition. For instance, an accidental is likely to appear before a note-head or another accidental, a solid note head should have its stem either at its lower left or higher right, beams must pass through stems. At this stage, symbols that carry multiple meanings, such as dots, can be clarified. If a dot is above a note head, the dot is an articulation sign, and if it is after a note head, it extends the duration of the note.

3 High-level Knowledge Enhancement

The process of OMR can produce ambiguities and omissions in its output, which require higher-level musical knowledge to correct. These may be due to the vagaries of typesetting in which symbols are connected with no clear break-points. Problem areas include the absence or incompleteness of key and time signatures, accidentals, irrational groupings, and broken symbols.

We have suggested a series of low-level algorithms which rely on basic musical syntax to determine primitives that have not been correctly recognised. The corrected primitives are reconstructed as musical symbols which can then be parsed using a high-level knowledge-base for more global information such as metre and grouping. As well as the total duration accumulated between two bar lines, other information tracked earlier, such as beam grouping, is used to identify the time-signature. The intermediate results are parsed using a list of common rhythmical patterns.

3.1 Key Signature

As reported in Ng, Boyle and Cooper (1995), key signatures may be automatically detected by using note distribution with a rule-based system. It was found that the Tonic and Dominant of a key are typically the most sounded notes for music written in the period of common-practice tonality, or governed by a single key centre defined as the tonic key.

For OMR, we can use the isolated accidentals recognised during the initial classification to assist the key signature detection. These accidentals can be used to assist the correction of the key signature, either directly or by implication. For example, a natural in a bar may suggest a sharp or flat in the key signature, or another note with the

same accidental sign before this particular note. The order of accidentals in the depiction of a key (i.e. F#, C#, G#, D#, A#, E#, B# for sharp keys and B \flat , E \flat , A \flat , D \flat , G \flat , C \flat , F \flat for flat keys) is a musical convention that can be used to interpolate accidental symbols that have not appeared or were mis-recognised. For example, if we find a C# in the key signature, one must assume that there is an F# before it. Courtesy accidentals (accidentals within brackets) may also be used as contextual clues.

3.2 Time Signature

For time-signature detection, assuming one time signature in the section under consideration, we take a semiquaver count of bars that have been recognised fully, and take the mean (Ng and Boyle). Table 1 shows the time signature and the number of semiquavers per bar. Semiquaver counts of 24, 16, 12, 8 and 6 cannot immediately identify time signatures, and further contextual information is required.

Number of semiquaver	Time Signature
48	6/2, 12/4
36	9/4
32	4/2
24	3/2, 6/4, 12/8
18	9/8
16	2/2, 4/4
12	3/4, 6/8, 12/16
9	9/16
8	2/4, 4/8
6	3/8, 6/16
4	2/8

Table 1. Number of semiquaver and the time signature.

In the case of semiquaver counts of 16 and 8, resolution of the two possible time signatures is difficult, indeed some composers may use the two options fairly indiscriminately, though we might expect, for example, the minim to be a more dominant musical feature of 2/2 than 4/4. Beaming information provides clues to the resolution of some of the other ambiguities.

If we consider semiquaver counts of 6, 12, 24, and 48, similar rules for beaming of groups within bars apply. In the case of compound-quadruple times (12/4, 12/8 and 12/16) beaming will tend to isolate four distinct groups, whereas compound-duple metres (6/2, 6/4, 6/8) will tend to promote two groups. Clearly in the cases of metres with

crotchets or minims (e.g. 12/4, 6/2) as their base units, some difficulties may arise when attempting to determine the time signature. However, quavers tend to be regular features in such metres, and their beaming can help in the resolution process. The resolution of metres with alternate simple or compound readings (e.g. 3/4 and 6/8) can usually be simply determined by a combination of beaming, and relative incidence of particular rhythmic patterns. Many compound metres are associated with specific dance forms which have their own characteristic rhythmic patterns (for instance, the gigue); some of these characteristic patterns remain as residual features of more recent works using the same time signature. A common example is the dotted quaver - semiquaver - quaver pattern. Some note types are extremely unlikely to appear within bars governed by certain time signatures. For example, the minim is seldom found in music in 6/8. Caution must be used with such heuristics, however, because some composers flout conventions in the area of rhythmic notation.

3.3 Enhancements

Automated time- and key-signature deduction is important because the information supplied provides many useful constraints, and can help determine specific features of the music.

Once the time and key signatures have been resolved, earlier low-level processes can be repeated to correct further remaining ambiguity with greater confidence.

An unclear note duration may be clarified by the difference of the total duration of a bar and that indicated by the prevailing time signature. With a known key signature, we can check all isolated accidental signs logically. For instance, a natural sign should only appear if there was an accidental sign before it, or if it is a sharp or flat of the key signature. There is a musical tendency to use the same type of accidental as that found within the key signature: sharp keys tend to use sharp accidentals and flat keys tend to use flat accidentals. However this is merely a tendency, and many examples of flat accidentals within sharp keys (and vice versa) may be found.

4 Conclusion

In this paper, we outline our pre-processing and recognition methods, and concentrate on the issue of high-level musical knowledge enhancement, particularly our strategies to deduce time and key signatures automatically, and the use of this information to correct and reconfirm recognition. With the feedback and confirmation from high-level musical knowledge we hope to recognise

complex music scores. We believe that musical knowledge enhancement is important and provides the way forward when dealing with complicated scores, and especially hand-written manuscripts in which symbols are often confused by ambiguities due to interconnection or incompleteness beyond isolated recognition.

References

- N. P. Carter, R. A. Bacon and T. Messenger: "The Acquisition, Representation and Reconstruction of Printed Music by Computer: A Survey", *Computers and the Humanities*, Vol. 22, 1988, pp 117 - 136.
- N. P. Carter: "Music Score Recognition: Problems and Prospects", *Computing in Musicology, An International Directory of Applications*, Vol. 9, Center for Computer Assisted Research in the Humanities (CCARH), 525 Middlefield Road, Ste. 120, Menlo Park, CA 94025-3443, USA, February 1994, pp. 152 - 158.
- A. T. Clarke, B. M. Brown, and M. P. Thorne: "Problems to be faced by Developers of Computer Based Automatic Music Recognisers", *Proceedings of the International Computer Music Conference*, Glasgow, UK, September 1990, pp. 345 - 347.
- E. Selfridge-Field: "Optical Recognition of Musical Notation: A Survey of Current Work" *Computing in Musicology, An International Directory of Applications*, Vol. 9, Center for Computer Assisted Research in the Humanities (CCARH), February 1994, pp. 109 - 145.
- K. V. Mardia and T. J. Hainsworth: "A Spatial Thresholding Method for Image Segmentation", *Technical Report*, Department of Statistics, University of Leeds, June 1986.
- K. C. Ng and R. D. Boyle: "Segmentation of Music Primitives", *Proceedings of the British Machine Vision Conference*, Leeds, UK, Springer-Verlag London Limited, September 1992, pp. 472-480.
- K. C. Ng and R. D. Boyle: "Reconstruction of Music Scores from Primitive Subsegmentation", To appear in *Image and Vision Computing*.
- K. C. Ng and R. D. Boyle and D. Cooper: "Automatic Key Signature Detection using Note Distribution", *Technical Report 95.23*, School of Computer Studies, The University of Leeds, Leeds, UK, 1995.
- M. Sonka, V. Hlavac and R. D. Boyle: "Image Processing, Analysis and Machine Vision", Chapman & Hall Computing, London, 1993.
- N. B. Venkateswaran and R. D. Boyle: "New Segmentation Techniques for Document Image Analysis", To appear in *Image and Vision Computing*.

Feature Extraction



Simulation of Listening to Bach's *Wohltemperierte Klavier*

Marc Leman
IPEM, University of Ghent

Francesco Carreras
CNUCE/CNR, Pisa

Abstract

In this paper, the basic assumptions of a schema theory of music cognition are tested for the first time in a realistic environment, using a powerful parallel computer. The model takes Book 1 of Bach's *Wohltemperierte Klavier* as input and develops a schema for tone center perception. The sampled sounds are processed for pitch extraction by an auditory model which produces images that are self-organized on a neural map. The results confirm the assumptions that have been obtained in previous studies with reduced datasets. A schema for tone center perception was produced with topological properties that reflect the circle of fifths and a close match with the results of psychological tests was obtained. In order to make this large scale simulation feasible, the use of supercomputing equipment was required. The model can be used to study music cognition in other realistic environments, such as music from other cultures or sounds for timbre studies.

1 Introduction

Recent developments in neurology, psychology, cognitive sciences, brain research and artificial intelligence have stimulated new directions of investigation in cognitive science in which music has become an important domain of application. Within this framework, a schema theory of music cognition has been developed and a computer model has been designed to test basic assumptions in the domain of tone center perception [8]. Thus far, however, the simulations have been based on limited data-sets such as a set of chords, a series of cadences, or a short piece of music. This paper discusses the results of testing the model within a large and realistic musical environment, using Book 1 of Bach's *Wohltemperierte Klavier* in a recording by K. Gilbert (CD ARCHIV 413439-2).

2 Simulation and Realistic Environments

Research in cognitive science is indeed often based on a small-scale artificial world. When models are

tested in a more realistic environment, the amount of the data involved is huge and the computational efficiency of the model and the computing facilities become a critical factor in research. The aim of this research is therefore twofold:

- to test the model under realistic conditions, using music as the input environment
- to exploit parallelism as a possible way to manage real-world simulations of cognitive models.

3 Overview of the Model

The perception module consists of an acoustical front end based on an auditory model [13] and a pitch extraction part based on a periodicity analysis of neural discharge patterns in the auditory channels of that model [8].

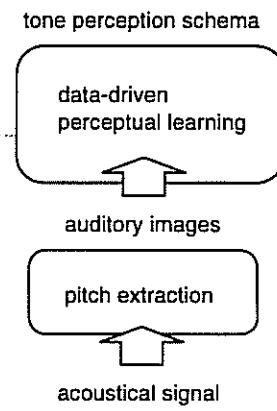


Fig. 1. Overview of the model

The cognitive model accounts for data-driven perceptual learning. This learning by self-organization is realized at long-term, it yields a schema of tone center perception. After learning, the model can be used for tone center recognition. Figure 1 gives an overview of the model.

As in previous simulations, the algorithm is based on the Self-Organizing Map (SOM)-architecture [5]. The SOM maps the N-dimensional auditory images onto a two-dimensional grid but it keeps the topo-

logical order of the implied relations that are imbedded in the auditory images.

In a number of pilot studies using reduced data sets it was shown that, in response to a set of musical chords, the SOM develops a structure which (in music theory) is known as the *circle of fifths*. The response structure of the map furthermore shows high correlations with data of mental representations obtained from psychological research [6, 8].

4 Simulation and Computational Limitations

Using Book 1 of the *Wohltemperierte Klavier* by J.S. Bach in an interpretation of K. Gilbert (clavichord) as the training environment, the auditory model generates a data set of 649264 images. Each image, called a tone context image contains 56 data points. The images, taken at a sampling rate of 100 sa/s, are the (leaky) integrated result of a summary-autocorrelation analysis of neural firing patterns (in frames of 30 ms). More details are in [8] or [7].

Running this data set during several epochs, the esteemed computing time required on a PC, workstation or even the parallel nCUBE system that was used for a previous experiment with a reduced data set, is beyond any reasonable limit for productive work in cognitive science. A complete revision of the original computer program [9] was started in order to

- optimize the time consuming parts of the code,
- reduce the impact of interprocessor communication,
- limit disk I/O activity by using large block operations.

5 An Implementation of SOM on the IBM SP2

The IBM SP2 system at CNUCE/CNR has eight processors, each with 64 Mbytes of memory, based on the POWER2 microprocessor technology and is provided with High Performance Switch Adapters that allow direct connection of each node to all the others. The SOM algorithm requires that for each input vector the most respondent one is found in the neuronal grid (which is actually a torus surface). Synaptic adjustment is then performed within a given radius around the most respondent neuron. In the parallel implementation, the neuronal grid is divided into subgrids of equal size that are assigned to each processor node. The size of the subgrid depends on the number of processor nodes that are used. In the present simulation, the size of the grid

is fixed to 20 x 20 neurons and a configuration of 4 processor nodes is employed. The SP2 implementation makes use of global communication operations. The partial information computed in each node is collected and the concatenated data is sent to all the nodes.

Speedup of the parallel implementation of the SOM

Starting from the assumption that there is a reasonable amount of redundant information in Bach's *Wohltemperierte Klavier*, we decided to run two times through all the data. In the algorithm, the data (1298528 vectors) are moved in advance to each processor node temporary disk area to speed up the disk I/O operations. They are divided into blocks of 40579 vectors. Each block is read and processed by all processor nodes. At the end of a block processing, each node writes to disk its assigned part of the computed neural grid and a new run is started with the subsequent block of data.

Learning Bach's Wohltemperierte Klavier

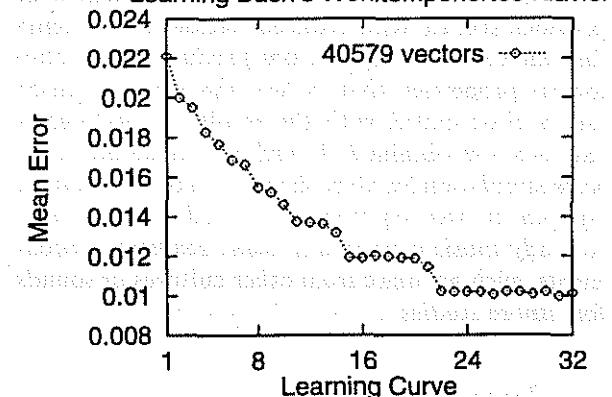


Fig. 2. Learning curve of Book 1 of J.S. Bach's *Wohltemperierte Klavier*. The learning curve is plotted for 40579 vectors. The error decreases rapidly and levels off after 16 iterations (learning curve 16).

The number of vectors that have to be processed within the different radii, according to SOM, are defined at the start. The values of the current block number, the absolute vector count, the number of vectors already processed with the current radius, are maintained and written to disk at the end of a block processing. This procedure allows to stop and restart the run at any point, and so it realizes an indirect form of checkpointing that was useful for running the simulations. We selected 8 radius values and the distribution of the vectors over each radius was calculated according to an inverse exponential. We adopted the constraint that about one fourth of all vectors has to be processed with radius zero. The speedup of the parallel implementation with 4 processors, was with a factor 3.08. Other simulations, using 8 processors, showed a speedup of 5.37. More details on the parallel implementation are in [2].

6 Simulation results

The learning curve of the simulation is shown in Fig. 2. It shows the mean error evolution (calculated as the distance between the images and their representations on the neural grid) during the processing of 32 datablocks of 40579 vectors each. Thus far, the schema obtained by data-driven learning of Bach's *Wohltemperiertes Klavier* has only been tested with Shepard-prototype patterns. These test patterns, also used in previous simulations [8, 7], are based on Shepard-chords and they have a very different timbre from the clavichord sounds that were used in the training set. (Fig. 3). The neural grid of Fig. 3 shows a well-defined topological order, and the output confirms the earlier pilot studies that used limited data-sets. A correlation with the psychological data [6] is shown in Fig. 4a-b.

7 Discussion

Although a more detailed analysis of the clustering needs to be done, we may conclude that this simulation, using a realistic musical environment as input, confirms the assumptions of the schema theory. The self-organization principles, simulated by means of artificial neural network, can be regarded as a model of the self-organization capabilities of the human brain.

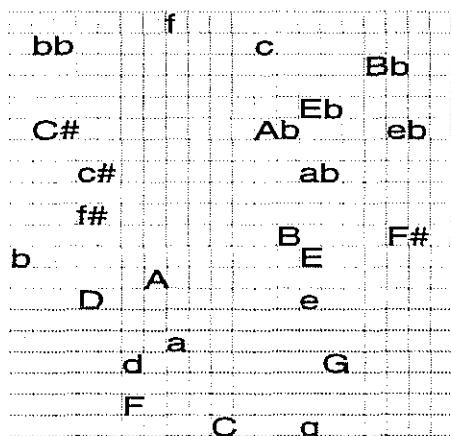


Fig. 3. Topological Map obtained by training of Book I of Bach's *Wohltemperiertes Klavier*. This is a 20 x 20 neurons network on a torus surface. The test patterns are images of Shepard-cadences that stand for tone centers (labeled C, C \sharp , D ..., bb, b). Although the test patterns are quite different from the training data, it is possible to map out the global ordering in terms of circles of fifths.

In that sense, the simulation is not just a test of a computer model, but it generates some hypothesis about human knowledge acquisition and ultimately also about the human brain. Two hypotheses are based on this finding:

1. that the schema (including its topological features) has a physiological basis in the sense that somewhere in the cortex, an area is assumed where neurons show that behaviour.
2. that the sense for tone center is a cultural (emic) phenomenon, mediated by auditory (etic) principles.

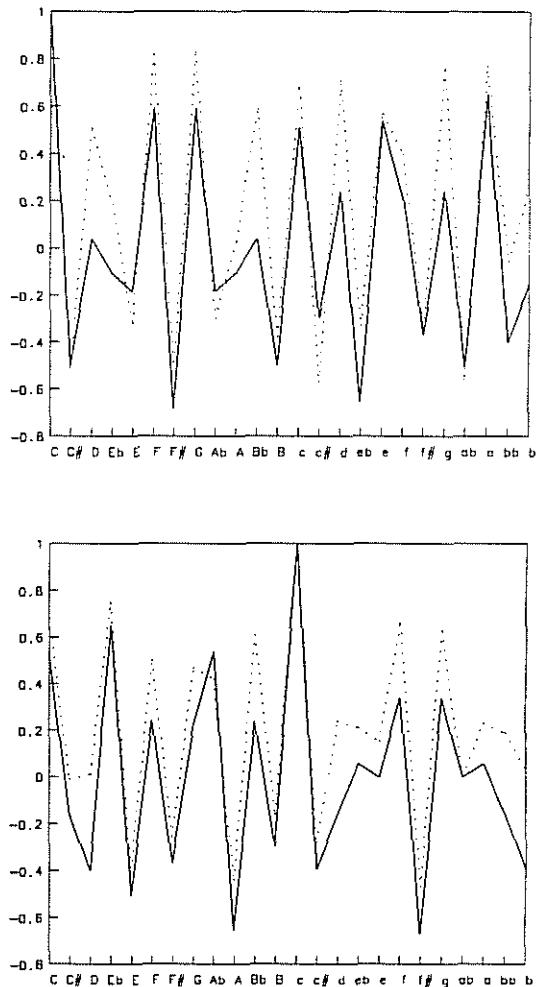


Fig. 4. Network response structure compared with psychological data. The dotted curve shows the correlation coefficients of the network response to the Shepard-test patterns that stand for tone center images. The full curve shows the correlation of psychological data. (a) Correlations with respect to the C-major tone center. (b) Correlations with respect to the C-minor tone center. The correlation of the dotted and full curves gives a value of 0.915 and 0.947 for major and minor tone centers, respectively.

If other musical instruments (e.g. with an inharmonic overtone structure) or other melodic and harmonic scales would have been used, then the sense for tone center would show a different picture. The approach therefore suggests a number of new directions of investigation:

- Does the modelling environment provide a suit-

able computational tool to study historical developments, such as the development of tonal music (see e.g. [4])?

- To what extend can ethnomusicological studies or in general, studies which assume different sound environments, be developed within the limited constraints of the present research paradigm?
- Can the hypothesis about the physiological basis of tone center perception be falsified or corroborated by neurophysiological research (e.g. using brain imaging techniques)?

In addition to the epistemological and methodological aspects of this study, there is a practical application of the obtained results in interactive computer music systems. The data-driven perceptual learning yields stable auditory images that can be used for recognition of tone centers or (by extension) chords. It may be useful to train such images taking as input realistic environments (e.g. noisy environments) that are going to be used in the interactive application [1].

8 Conclusion

The modelling environment developed in this study is not limited to tone perception studies. SOM is indeed a general purpose model of self-organization and it has already been used in studies of timbre [3, 11, 10, 12]. The present parallel implementation allows the use of much larger training sets and more realistic environments.

We finally believe that the application of cognitive modelling to real musical environments is a novel step towards the emancipation of music research as a fully independent and advanced discipline. Our study shows that the computational cost for simulation of cognitive processes in a (relatively small) realistic environment is high. Flexible working at present still requires access to supercomputer systems.

9 Acknowledgment
Part of this research was realized during research stages thanks to the support of N.F.W.O. and C.N.R.

- ## References
- [1] A. Camurri and M. Leman, *Applicazioni musicali basate su intelligenza artificiale*. In A. Di Scipio, editor, *Teoria e prassi della musica nell'era dell'informatica*. Giuseppe Laterza Editore, Bari, 1995.
 - [2] F. Carreras and M. Leman, *Distributed parallel architectures for the simulation of cognitive models in a realistic environment*. In *Poster Presented at Parco95*, Ghent, 1995. Parco95.
 - [3] P. Cosi, G. De Poli, and G. Lauzzana, *Auditory modelling and self-organizing neural networks for timbre classification*. *Journal of New Music Research*, 23(1):71-98, 1994.
 - [4] R. Eberlein, *Die Entstehung der tonalen Klangsyntax*. Peter Lang, Frankfurt am Main, 1994.
 - [5] T. Kohonen, *Self-organization and associative memory*. Springer-Verlag, Berlin, 1984.
 - [6] C. Krumhansl, *Cognitive foundations of musical pitch*. Oxford University Press, New York, 1990.
 - [7] M. Leman, *Schema-based tone center recognition of musical signals*. *Journal of New Music Research*, 23(2):169-204, 1994.
 - [8] M. Leman, *Music and Schema Theory - Cognitive Foundations of Systematic Musicology*. Springer-Verlag, Berlin, Heidelberg, 1995.
 - [9] M. Leman and P. Van Renterghem, *Transputer implementation of the Kohonen feature map for a music recognition task*. In *Proceedings of the Second International Transputer Conference: Transputers for Industrial Applications II*, Antwerpen, 1989. BIRA, Belgian Institute for Automatic Control.
 - [10] G. De Poli, P. Prandoni, and P. Tonella, *Timbre clustering bij self-organizing neural net-works*. In *Proceedings CIM 1993 Milano*, 1993.
 - [11] G. De Poli and P. Tonella, *Self-organizing neural network and grey's timbrespace*. In *Proceedings of the ICMC 1993 Tokyo*, 1993.
 - [12] P. Toivainen, *Timbre maps, auditory images, and distance metrics*. *Journal of New Music Research*, in press.
 - [13] L. VanImmerseel and J. P. Martens, *Pitch and voiced/unvoiced determination with an auditory model*. *The Journal of the Acoustical Society of America*, 91(6):3511-3526, 1992.

Marc Leman
IPEM, University of Ghent
Blandijnberg 2, B-9000 Ghent, Belgium
(Email:Marc.Leman@rug.ac.be)

Francesco Carreras
CNUCE/CNR
Via S. Maria 36, I-56126 Pisa, Italy
(Email:F.Carreras@cnuce.cnr.it)

Estrazione in Tempo Reale di Parametri di Controllo da Segnali Audio Complessi

Davide Alghi

Laboratorio di Informatica Musicale
Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
via Comelico, 39
I-20135 Milano (Italia)
fax +39 2 55006373
E-Mail: music@imiucca.csi.unimi.it

Abstract

Given the ever increasing necessity, in the professional field as much as in not strictly musical field, of a multimedia tool able to describe information in several forms in order to make more easily apprehensible the communication and exchange of ideas, we have designed and implemented a prototype able to give us more insight of a sequence of musical sounds.

Such a tool could be a very useful in those experimental performances where the multimedia has a strong influence.

In fact the description of the audio is done in real time and it could be used to control one or more devices to process images, audio, mechanic motion, text, movies.

The basis for this work are the fast Fourier transform and the relative sound description in terms of frequencies, the Motorola DSP56000 for real time processing, some imagination and love for sound.

Volume, complex pitch, amplitude envelope (attack, sustain, decay and release) and spectral density for fourths of octave are extracted and displayed on the monitor, allowing a wide pictorial representation of sound evolution.

1 Introduzione

Che bello sarebbe avere una precisa percezione visiva dell'audio. Uno strumento in grado di descrivere anche questo è quella che vogliamo raggiungere: per il momento ci siamo dedicati alla estrazione di valori descriventi il più precisamente possibile l'audio.

Premettiamo che il prototipo finora ottenuto lavora correttamente se il segnale ha una composizione frequenziale superiore ai 370Hz poiché ci siamo dedicati più alla stesura di algoritmi validi per il calcolo del pitch e dell'inviluppo lasciando per ultimo il problema della compatibilità del prototipo su tutta la banda udibile: l'espansione a tutta la banda è di facile implementazione.

Entrando nel tecnico: il segnale passando dall'ingresso analogico della scheda audio viene campionato e subisce una prima analisi per il calcolo dell'inviluppo e del volume. Entrambi i valori vengono calcolati nel dominio del tempo non richiedendo quindi una grossa sforzo computazionale.

Da questa prima fase di elaborazione escono cinque valori descrittivi: un valore (variabile da 0 a 15) rappresentante il volume, ed altri quattro (anch'essi variabili da 0 a 15) descriventi l'inviluppo con attacco, sostegno, decadimento e rilascio.

La fase successiva di elaborazione cui il segnale campionato è sottoposto è la FFT di 2048 campioni la volta utilizzando una FFT a 1024

complessa risparmiando il 25% di tempo e memoria del calcolatore rispetto agli ordinari algoritmi per il computo della DFT.

Ottenute le informazioni in frequenza si passa al calcolo della densità spettrale a quarti d'ottava e al calcolo del pitch complesso determinando quali sono le frequenze fondamentali che compongono il segnale.

Pitch e densità spettrale vengono calcolati sulla base di informazioni ricavate in frequenza per avere dati più precisi, sui quali è possibile fare maggiore affidamento che non la caratterizzazione del pitch nel tempo.

2 Inviluppo e volume

I parametri estratti per l'inviluppo sono attacco, sostegno, decadimento e rilascio. I valori che possono assumere variano da 0 a 15: un limite massimo per il sostegno per ovvi motivi di elaborazione.

Il calcolo del volume si basa su una media dei picchi più alti individuati nel segnale.

Il volume pone un punto di riferimento per l'algoritmo di calcolo dell'inviluppo che si avvale di quattro soglie: soglia del silenzio (SIL_SOGLIA), soglia del rilascio (RIL_SOGLIA), soglia inferiore del sostegno (INF_SGLIA:=VOLUME-GAP) e soglia superiore del sostegno (SUP_SOGLIA:=VOLUME+GAP).

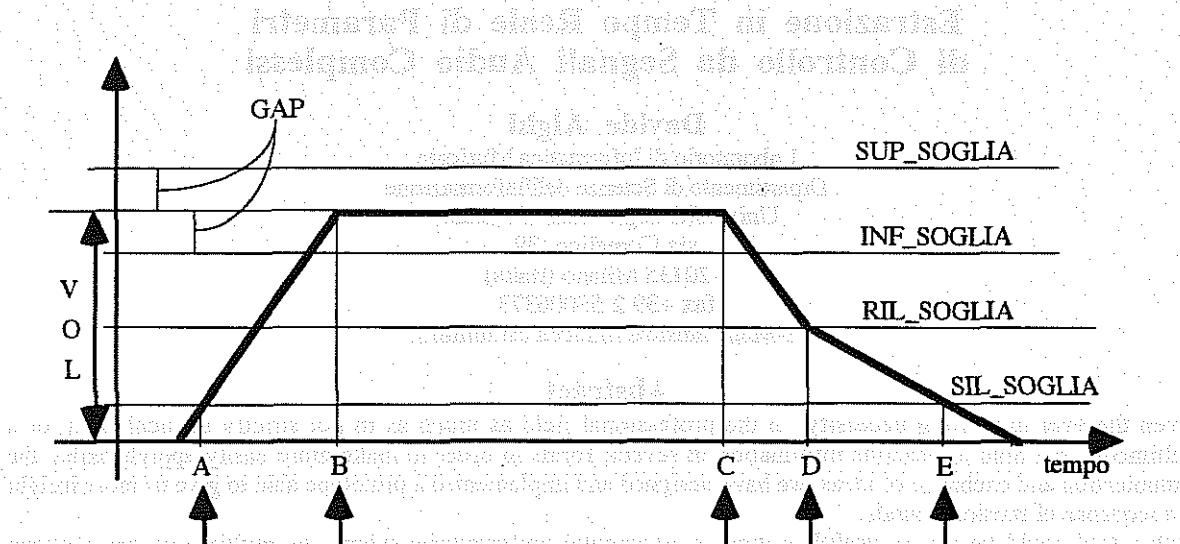


figura1: l'inviluppo

La figura 1 mostra la funzione di ogni soglia. Il segnale viene letto campione per campione e si considera la sua presenza solo se il campione supera la soglia del silenzio che è variabile dalla finestra dei controlli dell'applicazione a seconda dei gusti descrittivi che si vogliono imporre al software. Non appena si il segnale supera SIL_SOGLIA si fissa un marker (INI) in questo punto che servirà per il calcolo finale.

Conseguentemente si procede col calcolo dell'attacco che coprende il tempo a partire dal marker INI a quando il segnale non supera la soglia denominata INF_SOGLIA, calcolata sottraendo al valore del volume conosciuto finora, un valore GAP variabile dalla finestra di controllo dell'applicazione. Tale variabilità è stata introdotta per poter considerare eventuali fluttuazioni del segnale in fase di sostegno dovuti a vibrati più o meno rapidi e profondi. Ovviamente il valore GAP influenza anche superiormente al volume calcolando la SUP_SOGLIA che serve per verificare se siamo effettivamente in fase di sostegno o ancora in fase di attacco: ciò implica il fatto che le due soglie per il sostegno vengano ricalcolateogniqualvolta il segnale supera SUP_SOGLIA.

Ritornando al discorso dei marker: si era fissato il primo marker INI appena si superava la SIL_SOGLIA, ora si deve fissare il secondo marker ATT non appena si va oltre la INF_SOGLIA.

A questo punto si prosegue con il calcolo del sostegno fintantoché il segnale non scende al di sotto di INF_SOGLIA, creando la necessità di posizionare un terzo marker SOS definendo il punto di termine di fine del sostegno e di inizio del decadimento.

La soglia del rilascio RIL_SOGLIA viene calcolato pari alla metà del volume ed il gioco è lo stesso: non appena il segnale scende al di sotto di RIL_SOGLIA si fissa un quarto marker denominato DEC, che determina anche l'inizio del rilascio.

Per finire, al momento in cui l'audio scende alla soglia del silenzio si fissa l'ultimo marker RIL che definisce la fine dell'inviluppo.

Il calcolo finale dei quattro valori descriventi l'inviluppo richiede delle semplici sottrazioni:

attacco	$=$	ATT-INI
sostegno	$=$	SOS-ATT
decadimento	$=$	DEC-SOS
rilascio	$=$	RIL-DEC

Ovviamente i precedenti calcoli vengono effettuati su 2048 campioni la volta richiedendo quindi una memoria della storia passata nel calcolo dell'inviluppo almeno fintantoché non si conclude la descrizione di un inviluppo intero.

3 Pitch e densità spettrale

La parte più interessante e descrittiva riguarda appunto il pitch complesso e la densità spettrale a quarti d'ottava.

Questi valori si ottengono effettuando una FFT su gruppi da 2048 campioni la volta, determinando in modo molto semplice la densità spettrale che ci viene direttamente fornita dalla trasformata di Fourier, mentre il calcolo del pitch richiede un algoritmo un po' più complesso e messo a punto al L.I.M. di Milano.

Suddividiamo la banda dell'udibile in 126 intervalli teorici centrati ognuno sulla frequenza fondamentale di ogni nota (appunto 126), sapendo che la relazione tra le frequenze fondamentali di due note successive è pari alla radice dodicesima di 2.

Ogni nota è formata dalla frequenza fondamentale e dalle armoniche successive, date dalla fondamentale moltiplicata per 3, 4, 5, etc. fino ad un massimo pratico di 10. La nota sarà quindi formata dalla fondamentale sommata alla seconda armonica che dista dalla prima esattamente 1 ottava, ovverosia 12 intervallini di cui abbiamo appena parlato; la frequenza della terza armonica sarà inclusa nel 19-esimo intervallino dalla fondamentale, ovvero 7

intervalli dopo la seconda armonica. Via via possiamo constatare che le distanze tra armoniche successive misurate in intervallini darà il seguente vettore delle armoniche:

12 7 5 4 3 3 2 2 2

Supponiamo ora che in un qualche modo riusciamo ad ottenere una serie di intervallini, contenenti le frequenze del segnale che stiamo analizzando, dal risultato della FFT, ad esempio:

49, 53, 56, 61, 72, 77, 80, 81, 87, 89, 90, 94, 96, che altro non sono che alcuni dei 126 intervalli nei quali sono stati intercettati presenze di segnale.

Ad ognuno di questi valori assegnamo ora un peso per determinare da quante note è effettivamente composto il segnale audio che stiamo analizzando. Il criterio di assegnamento dei pesi fa uso del vettore delle armoniche:

- a) partendo dall'intervallo 49, gli sommiamo 12 ottenendo 61 che possiamo vedere all'interno della serie di intervalli (peso=2);
- b) sommiamo poi il valore 7 ottenendo $49+12+7=68$ che non troviamo nella serie;
- c) continuando a questo modo otterremo per l'intervallo 49 la seguente sequenza

+12 +7 +5 +4 +3 +3 +2 +2 +2
49 61 68 73 77 80 83 85 87 89
6 dei quali li ritroviamo all'interno della serie degli intervalli forniti dall'analisi del segnale tramite la FFT: il peso per l'intervallo 49 sarà allora 6;

d) procedendo allo stesso modo per tutti i 13 intervalli della serie otteniamo la seguente sequenza dei pesi dei rispettivi intervalli:

6 6 6 6 3 2 3 1 1 1 1 1 1

e) la determinazione del numero delle note componenti il pitch si effettua sommando i pesi delle note a partire dai più "pesanti" fintantoché la somma è inferiore al numero degli intervalli che

compongono la serie. Nel nostro caso il segnale è composto da 3 note ($49+53+56=158>13$)

4 FFT reale ridotta

Data la necessità di avere una elaborazione in tempo reale abbiamo dedicato tempo per la stesura di un codice assembler DSP per il calcolo di una FFT reale ridotta in grado di farci risparmiare il 25% di tempo di computazione e di memoria. L'algoritmo di base è stato tratto dal testo di R.A.Roberts e C.T.Mullis: Digital Signal Processor, nel quale viene suggerito il calcolo di una DFT reale come se fosse una trasformata complessa. In altre parole, consideriamo N campioni: quelli in posizione pari come parte reale e quelli in posizione dispari come parte immaginaria di un nuovo segnale, al quale applichiamo un normalissimo codice per il calcolo di una FFT, ma su un numero di punti pari alla metà ($N/2$). Poi con un semplice calcolo di integrazione riusciamo a risalire ai coefficienti come se avessimo eseguito una FFT su N punti.

Non diamo qui la spiegazione matematica dell'algoritmo per ovvi motivi di spazio.

5 Accordo politimbrico

I due accordi in questione sono un DO maggiore suonato con un organo a canne ed un intervallo di terza maggiore composto da RE5 e FA#5 prodotto con una chitarra elettrica. La Fig.2 riporta l'immagine durante la fase di sostegno. Il pitch è chiaramente visibile nell'apposito riquadro col quale possiamo vedere nella raffigurazione dello spettro la distribuzione del segnale o meglio delle armoniche fondamentali, che vengono tutte descritte attraverso i primi tre segmenti.

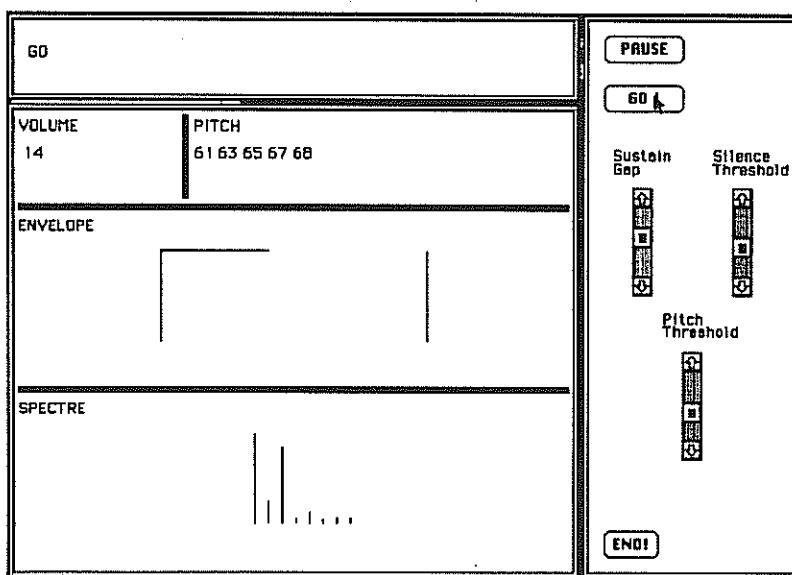


figura 2: organo e chitarra

Le fondamentali delle note 61 e 63 (DO5 e RE5) le possiamo vedere sommate nella prima linea dello spettro; la seconda rappresenta solamente il 65 (ovvero il MI5); mentre le armoniche fondamentali

di 67 e 68 (FA#5 e SOL5) sono contenute nella terza linea come somma algebrica.

Durante l'uso dell'applicazione abbiamo accertato che nel momento in cui alcuni strumenti, presi

singolarmente, venivano descritti correttamente dall'applicazione, se accoppiati tra di loro producevano un timbro descrivibile dal prototipo. Risulta evidente il fatto che se cercando di descrivere un segnale complesso, composto dalla somma timbrica di due strumenti, mantenessimo una differenza in intensità molto alta, ovverosia il volume in ingresso all'applicazione di uno strumento fosse molto più alto dell'altro, non riusciremmo a descrivere la somma timbrica ma solamente quella dominante, quella dello strumento con intensità maggiore. Viceversa se il volume di uno strumento preso singolarmente risulta essere talmente alto da produrre distorsione è ovvio che la descrizione diventerebbe alquanto impossibile. Si è accertato che la differenza tra le due intensità è la cosa importante, anche se entrambi gli strumenti hanno un buon grado di libertà nel variare il proprio volume; è necessario comunque che entrambi siano ad un volume interpretato simile "ad orecchio". Questo discorso fa capire quanto il prototipo non sia costretto dalle caratteristiche delle sorgenti sonore, che potrebbero altrimenti limitare le capacità descritte fino a renderlo praticamente inutilizzabile.

Bibliografia

- A.V.Oppenheim, R.W.Schafer**
Digital Signal Processing
Prentice-Hall, N.J. (U.S.A.), 1975
- R.A.Roberts, C.T.Mullis**
Digital Signal Processing
Addison Wesley, U.S.A., 1987
- F.R.Moore**
An Introduction to the Mathematics of Digital Signal Processing (Parts 1,2)
Computer Music Journal, Vol.2-Num.1/2, 1978
- B.A.Blesser**
Digitalization of Audio: A Comprehensive Examination of Theory, Implementation and Current Practice.
Journal of the Audio Engeneering Society, Vol.26-Num.10, 1978
- J.A.Moorer**
Signal Processing Aspects of Computer Music -- A Survey
Computer Music Journal, Vol.1-Num.1, 1977
- M.Piszczalski, B.A.Galler**
Automatic Music Transcription
Computer Music Journal, Vol.1-Num.4, 1977
- L.J.Millott**
Envelope Detection of Binary VSB Data Signals
IEEE Transactions on Comunications, Vol.33-Num.7, 1985
- D.B.Paul**
The Spectral Envelope Estimation Vocoder
IEEE Transaction on Acoustics, Speech and Signal Processing, Vol.29-Num.2, 1981
- H.V.Sorensen, C.S.Burrus**
Efficient Computation of the DFT with Only a Subset of Input or Output Points

IEEE transactions on Signal Processing, Vol.41-Num.3, 1993

L.R.Rabiner, M.J.Chang, A.E.Rosenberg, C.A.McGonegal

A Comparative Performance Study of Several Pitch Detection Algorithms

IEEE Transactions on Acoustics, Speech and Signal Processing, Vol.24-Num.5, 1976

J.E.Lane

Pitch Detection Using a Tunable IIR Filter

Computer Music Journal, Vol.14-Num.3, 1990

W.B.Kuhn

A Real-Time Pitch Recognition Algorithm for Music Applications
Computer Music Journal, Vol.14-Num.3, 1990

R.A.Sukkar, J.L.LoCicero, J.W.Picone

Design and Implementation of Robust Pitch Detector Based on a Parallel Processing Technique
IEEE Journal on Selected Areas in Communications, Vol.6-Num.2, 1988

A.S.Tanguiane
An Algorithm of Recognition of Chords

Proc. of the 14th Int. Computer Music Conference, 1988

Motorola Inc.

DSP56000/DSP56001 User's Manual

Prentice-Hall, N.J. (U.S.A.), 1989

Motorola Inc.

DSP Development Software Manual

Prentice-Hall, N.J. (U.S.A.), 1992

Motorola Inc.
MC68030 Enhanced 32-Bit Microprocessor User's Manual

Prentice-Hall, N.J. (U.S.A.), 1989

B.W.Kerningham, D.M.Ritchie

Linguaggio C
Jackson, Milano, 1989

Apple Computer Inc.

Inside Macintosh (Vol.1-6)

Cupertino, California (U.S.A.), 1985

SYRINX

Conversione automatica di processi audio analogici in processi MIDI: visualizzazione e confronto

A. Ballista, G. Fazzini, G. Haus

L.I.M. - Laboratorio di Informatica Musicale
Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
via Comelico 39
I-20139 Milano (Italia)
fax +39 2 55006373
e-mail: music@imiucca.csi.unimi.it

Abstract

Syrinx can be considered as a first step towards the formalization and the recognition of music interpretative structures. MIDI parameters depending on pitch, intensity, and duration, are detected within audio files and Standard MIDI Files are then generated. Couples of different executions of a music score can be visualized by means of a grid-like representation on the screen and compared one each other. Also, the original score from the composer can be compared with any particular execution of the score itself.

1. Introduzione

Un interprete, posto di fronte ad una partitura musicale, molto difficilmente (potremmo dire forse mai) produrrà una esecuzione corrispondente in modo assoluto a quanto scritto sulla carta. Di fronte a due esecuzioni del medesimo brano da parte di due diversi strumentisti si è in grado (perlomeno chi ha l'orecchio più allenato) di distinguere un musicista dall'altro e senza dubbio si trova molto più gradevole una esecuzione che si discosti leggermente da quanto scritto piuttosto che una esecuzione "meccanica". Il lavoro svolto in questo progetto rappresenta un primo passo verso il riconoscimento e la formalizzazione della interpretazione musicale. Per la prima fase sperimentale si è scelto di lavorare su file audio ricavati dalla digitalizzazione di registrazioni di brani per flauto solo; in una seconda fase saranno considerati anche brani polifonici, e infine, politimbrici. Con tecniche di elaborazione del segnale digitale vengono estratti dai file audio i seguenti parametri per ciascuna nota [Moorer, 75]:

- altezza,
- durata,
- intensità,

che vengono poi tradotti in accordo con il protocollo MIDI (Musical Instruments Digital Interface) [Anderson, 86], scelto in quanto rappresenta sicuramente il linguaggio di comunicazione musicale di maggior diffusione. A causa delle limitazioni intrinseche del MIDI [Loy, 85] [Moore, 88] si è deciso di produrre anche una rappresentazione grafica di quanto analizzato, utilizzando un sistema a griglia simile a quello di molti sequencer commerciali e, per motivi di puro spazio fisico disponibile su un monitor di un calcolatore, ci si è limitati al confronto di due sole esecuzioni alla volta.

2. L'applicazione Syrinx

L'applicazione Syrinx può essere considerata una sorta di traduttore da file audio a Standard MIDI File. I due file audio in ingresso devono essere nel formato Sound Designer II, campionati con una frequenza di 44100 Hz, a 16 bit e possono essere sia monofonici sia stereofonici (i file stereo vengono ridotti a mono dalla applicazione mediante una semplice semisomma dei due campioni contemporanei). L'elaborazione del segnale audio digitalizzato viene effettuata senza uscire dal dominio del

tempo attraverso l'applicazione della funzione di autocorrelazione [Rabiner, 77]:

$$\phi_l(m) = (1/N) \sum_{n=0, N-1} [x(n+l)w(n)] \cdot [x(n+l+m)w(n+m)]$$

con $0 \leq m \leq M_0 - 1$

che, applicata ad una finestra ampia 10 millisecondi (441 campioni nel nostro caso) della forma d'onda analizzata, presenta il massimo assoluto all'inizio ed il primo massimo relativo ad una distanza corrispondente al periodo della frequenza fondamentale presente nella finestra. Prima però di essere passato alla funzione di autocorrelazione il segnale viene pretrattato attraverso le funzioni non lineari:

$$y(n) = \text{clc}[x(n)] =$$

- se $x(n) \geq C_L$: $[x(n)-C_L]$
- se $|x(n)| < C_L$: 0
- se $x(n) \leq C_L$: $[x(n)+C_L]$

$$y(n) = \text{clp}[x(n)] =$$

- se $x(n) \geq C_L$: $x(n)$
- se $|x(n)| < C_L$: 0
- se $x(n) \leq C_L$: $-x(n)$

$$y(n) = \text{sgn}[x(n)] =$$

- se $x(n) \geq C_L$: 1
- se $|x(n)| < C_L$: 0
- se $x(n) \leq C_L$: -1

dove la soglia viene posta uguale al 68 % del più piccolo dei massimi assoluti del segnale nel primo e nell'ultimo terzo della finestra di analisi. Utilizzando gli output di queste tre funzioni e il segnale originale, si hanno quindi dieci diverse combinazioni possibili per calcolare la funzione di autocorrelazione:

1	$x(n)$	$x(n)$	$x(n)$	$x(n)$
2	$\text{clc}[x(n)]$	$\text{clc}[x(n)]$	$\text{clc}[x(n)]$	$\text{clc}[x(n)]$
3	$\text{clp}[x(n)]$	$\text{clp}[x(n)]$	$\text{clp}[x(n)]$	$\text{clp}[x(n)]$
4	$x(n)$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$
5	$\text{clc}[x(n)]$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$
6	$\text{clp}[x(n)]$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$
7	$x(n)$	$\text{clc}[x(n)]$	$\text{clc}[x(n)]$	$\text{clc}[x(n)]$
8	$x(n)$	$\text{clp}[x(n)]$	$\text{clp}[x(n)]$	$\text{clp}[x(n)]$
9	$\text{clp}[x(n)]$	$\text{clc}[x(n)]$	$\text{clc}[x(n)]$	$\text{clc}[x(n)]$
10	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$	$\text{sgn}[x(n)]$

Sperimentalmente si è visto che la combinazione numero 10 è quella che fornisce i migliori risultati e viene quindi utilizzata per default, ma l'utente volendo ha la possibilità di cambiare questa scelta. Una volta ottenuto un vettore delle altezze dell'energia, i dati ottenuti vengono organizzati in una struttura a lista in cui ciascun elemento è costituito da otto campi:

- un puntatore all'elemento successivo della lista;
- un puntatore all'elemento precedente della lista;
- l'altezza della nota;
- la durata della nota (in millisecondi);
- il momento di inizio della nota;
- il momento della fine della nota;
- un puntatore ad una lista con l'andamento dell'energia lungo la durata della nota;
- un puntatore ad una lista con l'andamento delle deviazioni dalla frequenza base lungo la durata della nota.

Le informazioni così organizzate vengono visualizzate sullo schermo attraverso due griglie che possono rappresentare le altezze, il volume oppure le deviazioni dalle altezze teoriche (Figura 1). Ciascuna griglia ha la possibilità di scorrere e di essere dilatata/contratta indipendentemente dall'altra in modo da facilitare il confronto fra le due esecuzioni.

Scendendo la lista precedentemente costruita vengono anche generati gli Standard MIDI File delle due esecuzioni analizzate. Viene quindi prodotto un messaggio di Note On ogni volta che si passa ad un nuovo elemento della lista ed il valore di velocity associato al Note On viene calcolato in base all'energia presente nel segnale dopo la fase transitoria dell'attacco. Poichè si è visto sperimentalmente che moduli sonori diversi presentano diverse funzioni dell'energia in funzione del valore di key velocity e che queste funzioni seguono andamenti che non si possono definire né lineari né esponenziali, si è scelto di mappare i valori di energia e di key velocity in modo lineare lasciando poi all'eventuale sintetizzatore utilizzato per la riproduzione del MIDI file il compito di interpretare secondo la sua logica questi valori. A questo punto è sorto il problema che per il flauto, come per tutti gli strumenti a fiato, il valore di velocity all'inizio non è assolutamente sufficiente per descrivere la dinamica della nota in quanto l'interprete, aumentando o diminuendo la pressione del fiato, può aumentare o diminuire l'intensità all'interno di una singola nota. Il protocollo MIDI fornisce, oltre al valore di key velocity, solamente un controller che va ad agire sul volume generale dello strumento (limitazione dovuta evidentemente al fatto che il linguaggio MIDI è nato prevalentemente per far comunicare tra loro strumenti a tastiera) per cui si è deciso di usare uno fra i controller non definiti "immagazzinando" l'informazione ottenuta in attesa che sia possibile in futuro tradurla più efficacemente. Per rappresentare le deviazioni dalla frequenza base della nota è invece presente il controller del pitch-bend il cui valore viene quindi modificato con la risoluzione temporale decisa dall'utente.

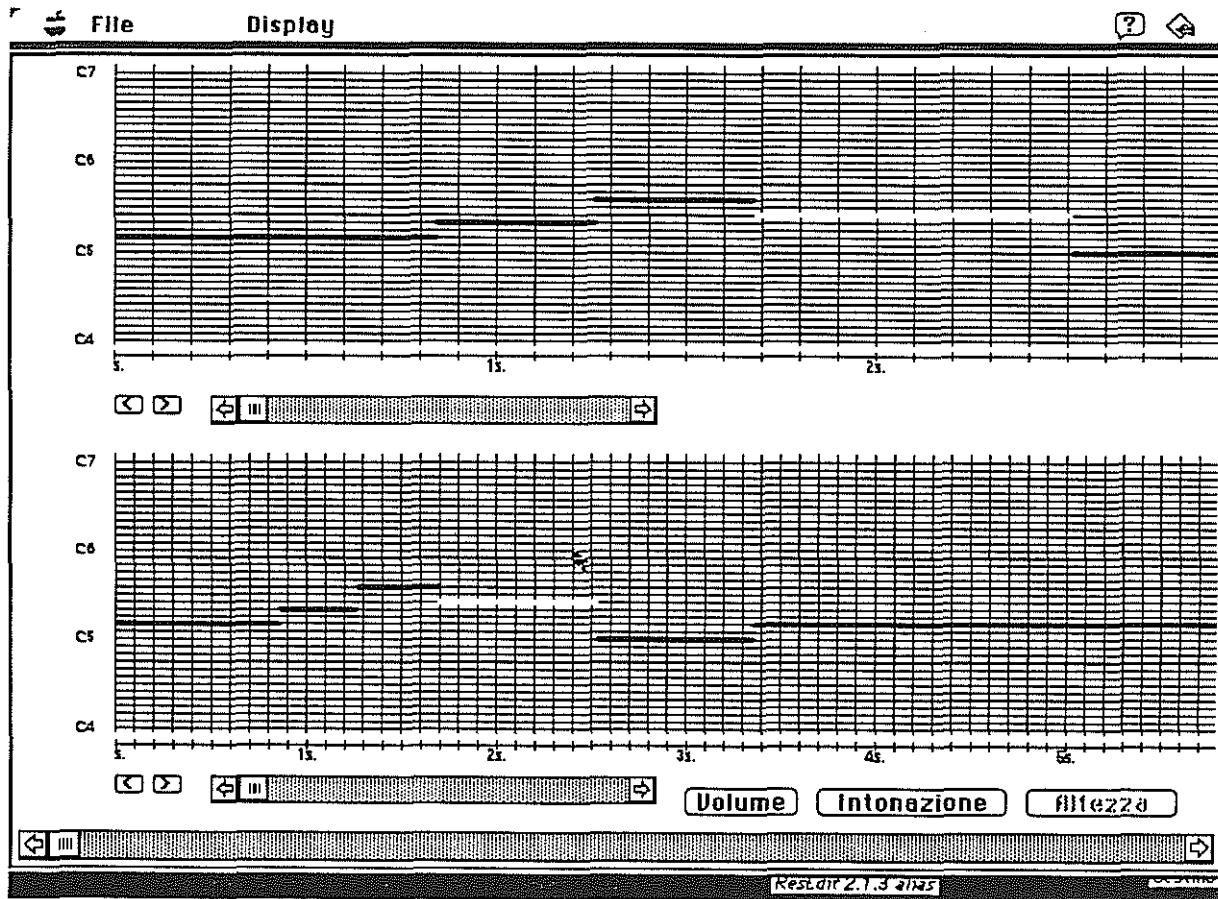


Figura 1

3. Un esempio

Come esempio di applicazione di Syrinx si è scelto di mostrare il confronto fra una esecuzione meccanica di un brano e l'esecuzione da parte di un interprete. Il brano scelto è la Allemande della "Suite per flauto solo" dello autore francese del '900 Jean Francaix, e precisamente le battute dalla numero 35 alla numero 40 (Figura 2). L'interprete del brano è la flautista austriaca Manuela Wiesler mentre l'esecuzione meccanica è stata ottenuta sfruttando i due software commerciali Finale 2.6.1 e Performer 3.6.1 che hanno pilotato un modulo sonoro YAMAHA TX81Z di cui è stata sfruttata la voce preimpostata di flauto. Finale, pur essendo un ottimo programma per l'editoria musicale, non offre molte funzioni per il controllo dei parametri MIDI per cui traduce con valori di key velocity diversi i segni di espressione tipo *p*, *mp*, *ff*, ecc. mentre ignora le variazioni di tempo e lascia all'utente il compito di definire come interpretare le forcille.

Non volendo introdurre decisioni arbitrarie si è deciso di non definire questi valori rendendo così l'esecuzione ancora più meccanica. Le battute analizzate, seppur esigue come quantità,

presentano variazioni sia di tempo sia di dinamica a causa della presenza di un *poco accelerando* alla battuta 35, di un *poco poco al tempo* alla fine della battuta 38 e di variazioni dinamiche che vanno dal *piano* dell'inizio della battuta 35 e della fine della battuta 40 al *forte* posto alla fine della battuta 37.

Sia dalla rappresentazione grafica (Figura 2) sia dall'analisi dei MIDI file prodotti si può notare innanzi-tutto che l'esecuzione umana risulta più lunga di circa un secondo e questo nonostante la presenza dell'accelerando. Si deduce che il tempo tenuto è più lento di quello indicato sullo spartito, ma quello che più sorprende e più appare evidente è la grande disparità di durata fra note che teoricamente dovrebbero essere identiche.

Per esempio la nota di un ottavo, che dalla esecuzione meccanica risulta avere una durata fissa di 435 millisecondi, può anche durare 600 millisecondi come il la nelle battuta numero 35 oppure raggiungere una durata minima di 250 millisecondi come il re posto alla fine della stessa battuta 35 che subisce quindi un deciso accorciamento dovuto al protrarsi altrettanto deciso della nota precedente. La battuta numero

40 è quella che dal punto di vista delle durate presenta il comportamento più distante dall'esecuzione meccanica. Per effetto sia della scritta *poco poco al tempo* sia della fine della frase musicale, i tempi subiscono un notevole allungamento per cui si incontrano il mi bemolle della durata di 1 secondo, il si bemolle

della durata di 1,65 secondi, il do naturale della durata di 90 millisecondi e il do diesis della durata di 1,20 secondi. Inoltre il passaggio dal si bemolle al do naturale non avviene bruscamente, ma attraverso un portamento che fa apparire ad un certo punto frequenze corrispondenti al si naturale.



Figura 2

Dal punto di vista della dinamica si nota immediatamente come nell'esecuzione umana più che in quella meccanica vengano utilizzati tutti i valori di key velocity disponibili. Mentre la macchina fa corrispondere al piano dell'inizio della battuta 35 il valore di 27, al forte della battuta 37 il valore di 60 ed al piano della battuta 40 il valore 27, l'esecuzione della flautista presenta valori minimi di key velocity pari a 15 e valori massimi pari a 110 ottenendo un'escursione ben maggiore.

4. Sviluppi futuri

Come già accennato l'analisi può essere estesa da strumenti monofonici a strumenti polifonici, e quindi a organici strumentali, costringendo però all'utilizzo di tecniche per l'elaborazione del segnale digitale più raffinate della funzione di autocorrelazione utilizzata finora. Sicuri miglioramenti potranno poi venire dall'adozione di un formato di codifica

dell'informazione musicale simbolica più completo del protocollo MIDI, che permettano la completa traduzione di tutte le informazioni ricavate dall'analisi del segnale audio.

Bibliografia

- [Anderson, 86] C. Anderson: *MIDI for Musicians*, AMSCO Publications, 1986.
- [Loy, 85] G. Loy: "Musicians Make a Standard: The MIDI Phenomenon", *Computer Music Journal*, Vol. 9, N. 4, 1985.
- [Moore, 88] F. R. Moore: "The Dysfunctions of MIDI", *Computer Music Journal*, Vol. 12, N. 1, 1988.
- [Moorer, 75] J. A. Moorer: "On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer", Stanford University, 1975.
- [Rabiner, 77] L. R. Rabiner: "On the Use of Autocorrelation Analysis for Pitch Detection", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 25, N. 1, 1977.

Real-Time Musical Rhythm Tapping: a Neural Networks - Numerical Algorithms Hybrid System.

Andrea Bertoni, Roberto Bresin
E.mail: rb@csc.unipd.it

C.S.C. - Centro di Sonologia Computazionale, Università di Padova
Padua, Italy

Abstract

The problem of rhythm tapping, as location of musical bar limits, is very difficult because parameters can change a lot and they can assume a very large number of combinations. We tried to solve the problem by realizing a system which combines a sophisticated use of neural networks with usual numerical algorithms. Neural networks have been used as elements of an array, called neural map. Using only the most important parameters of a note, this map suggests if a certain note is at the beginning of a bar. It is interesting the fact that neural map behaviour accords well to main results in psycho-cognitive studies about rhythm understanding. The flow of input data (pitch and duration of each note) and of the outputs of the neural map are processed by a sub-system that gets the absolute position of a past bar (phase) and its duration (period); these informations are enough to forecast if the note that is going to begin will be the first of a new bar.

1 Introduction

From the beginning, one of the main problems in computer music has been, the right recognition of musical scores from live performances. The problem is very difficult because parameters can change a lot and they can assume a very large number of combinations. In many works researchers tried to solve the problem in a simplified way. In all of these works the two main challenges are:

- transformation of durations of performed notes in musical durations (duration quantization); in this case you have either to provide song tempo or to extract it from the performance;
- location of starting points and length of bars; this implies the skill of recognizing musical accents and, as a consequence, the recognition of the rhythm.

The challenge of bar recognition has been faced by numerous authors that used songs either transcribed from the score or from performance (performance could be quantized or not quantized). Longuet-Higgins and Lee [1] and Povel and Essens [2] use data from score. The main idea of Longuet-Higgins and Lee's model is that listeners try to recognize the rhythm in order to minimize the number of syncopations, idea confirmed by various experiments. Instead Povel and Essens affirm that rhythmical analysis is helped by an internal clock, and their model, realized by a computer program, calculates the clock period that is best induced by input data. In Chung's model [3] there is not an intermediate phase in which data are quantized; the model proceeds directly from performance data to a notation that is similar to standard musical notation, and works in a strictly temporal way, so that it can be a model of the process used by human listeners to

find the rhythm. Rhythm perception is divided in two processes. The first one assigns to each note an importance parameter that resumes the properties of a note (loudness, pitch, expressive timings, etc.), influencing its interpretation as strong beat. The list of importance parameters is passed to the metrical hierarchization process that analyzes it in order to build the rhythmical sequence. Other systems that work directly from performances are described by Katayose et al. [4], and by Allen and Dannenberg [5]. Recently Rosenthal [6] published one of his works that is realized as a program. This program collects temporal positions of note attacks coming from the performance; then it looks for series of attacks equally spaced (potential rhythmic levels) and ranks each one according to rules that correspond to how human listeners choose rhythmic interpretations. Top ranked interpretation is taken as the right one. In this work we consider the following requirements:

- real-time rhythm recognition in performances;
- score generation;
- use of neural networks.

The necessity of real-time working influenced structure and code of algorithms. For score generation we needed a quantization of input data. Neural networks have already been used for note quantization [7] but they have been mainly applied in the field of tonal harmony.

We noted that we do not need using those systems for quantization, because we can put together traditional algorithms with good performances. On the contrary we must find out starting points of bars, which corresponds to particular musical accents, and we implemented it with neural networks. Our choice was justified by the wide data range and by the numerous non-linear relationship

among data. Infact typical characteristics of neural networks are robustness, flexibility, and ability to learn non-linear relationship. As durations we use temporal distances between two successive attacks (MIDI command: Note On) in hundredths of second because in our conception of rhythm only attack times are important, we do not consider notes with legato or followed by a pause (Johnson-Laird [8]). As loudness we use the Key Velocity (a MIDI parameter) that comes with note attack. Durations are then quantized and stored also as durations in 32th which will be called QD (Quantized Durations). Loudnesses and durations are processed and proposed to the neural map together with QD of last but one note. The map output represents the possibility that the present note is the first one of a bar, concept similar to that one of salience [6]. Salience values, supported by other data structures, allow another module (Phaper: phase and period finder) to compute the absolute position of a whole bar (phase), and the bar length (period) that best approximates collected data. After that we can forecast if the next note will be the first one of the next bar.

2 The Neural Map

The starting idea to build this module was Chung's work [3]. In his model Chung assigns to each note a parameter resuming the characteristics that make the note sounding as a strong beat. This parameter is computed in a heuristic way from loudness, pitch, expressive timings, etc. We substituted heuristic computation with a neural network because the first one seemed too much peculiar. Infact the main characteristic of supervised neural networks (used here) is to fit also complex non-linear relationship by only presenting a series of examples. Shaffer et al. [9] say that, since metrical sequences are directly determined, they are better predictors than melodic rhythm in synchronizing dance, clapping or other musicians' performance, because they can be extracted from the superficial rhythm of the execution. On the other hand Clarke [10] concludes that effects deriving from melodic changes in tonal melodies are marginal in comparison with those produced by the metric change. So, from these indications and from personal experiments, we decided not to take pitch into account because it has only a little influence. Analysis of piano executions revealed that events in strong metrical positions can be emphasized by playing the first event in each bar louder, longer, and more legato (Drake and Palmer [11]). In theory, to understand metrical structure, note durations should be enough; in the reality, there are many ambiguous situations. Numerous studies about rhythm perception found that loudness differences suggest beat positions reliably (Sloboda [12], Handel [13], Norris and Rosenthal [14]), while MIDI data from piano executions don't show that strong beats are reliably played louder. So, as

Rosenthal suggests [6], informations about loudness should not be ignored, but used judiciously. We will consider only duration and loudness of notes. From examined studies we understood that it is not important the absolute value of a parameter, but its value in relation with temporally near ones, that is the previous note and the next one. So we need a situation with a present note (N0), a future one (FN) and a past one (PN1). Since our system has to work in real-time, we only have temporally past notes. Therefore we consider the last but one note as present note, the last one as future and the others as past ones. In this way the neural map will give a result about the last but one input note. In order to reduce the chance of equal inputs and improve network training, we considered also another past note (PN2), the one before PN1. The map has been built with neural networks as multi-layer Perceptron, trained using Back-Propagation algorithm. As we said before, we do not consider absolute values of parameters, but their values in comparison with the center note (N0). So we used ratios of each parameter with its correspondent of N0, resulting in 3 values for duration and 3 for loudness. One of the most difficult problem in using this kind of networks is the normalization of input values, because their range is often very large. We did not find any generic approach to this problem and so one of the authors designed method that gave very good results: the MFMS (More Frequent More Sensitive) coding. This coding has more precision where data are more frequent. In this way we can better distinguish data exactly where they are concentrated and avoid ambiguous situations. As examples we chose 11 songs transcribed from the corresponding score but with loudness coming from real performances. Songs have been chosen so that they are various for age and musical style and they offer a large outline. We obtained over 1200 examples and they caused numerous problems. A first problem was the input congruency. This concept means that a neural network, as a function, given a certain input, can have only one output configuration. The problem has been solved with a specific program which, among examples with equal inputs and different outputs, keeps only that one with the more frequent output pattern. After this first processing the number of examples remained over one thousand. At the beginning we tried generic networks with 6 inputs, 1 output and 1 or 2 hidden layers, but we noticed a considerable computational charge and the impossibility to converge of the networks. We thought that this behaviour derived from the fact that inputs are ratios which do not take into account the absolute values: the same ratio situation is not the same thing, when N0 is a quaver or a demisemiquaver. This gave the idea to apply the "divide et impera" technique, well known in signal processing field (FFT). So the training examples have been divided

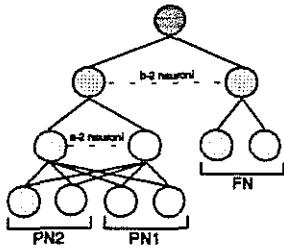


Fig. 1 Topology of used neural networks.

according to the N0 duration expressed in 32th: we found 9 groups of durations (1, 2, 3, 4, 6, 8, 16, 32). Since each group consisted of no more than 378 examples, we were able to use smaller networks. In order to further reduce the computational charge, we used a specific network topology (see fig. 1): not anymore 6 contemporary inputs, but only the 4 ones concerning past notes (PN1, PN2) are pre-processed by a set of neurons (semilayer) which are at the same level of inputs regarding future notes. The real map is composed by a vector of elements (one for each 32 durations) which can be:

- a neural network (durations 1, 2, 3, 4, 6, 8, 12, 16);
- a fixed value (duration 32 with value 0.9);
- an interpolated value (other durations).

From a mathematical point of view the map can be considered a function from a subset of R^6 ($[0.1, 0.9]^6$) to the interval $[0.0, 1.0]$; this function is clearly not linear and therefore an analytical description is nearly impossible. In these cases people often use a look-up table and find the data not in the table with an interpolation algorithm. If we keep present that the map has a correctness of 97.97 %, comparing it with a normal look-up table we can draw two great advantages:

- a data compression of about 26 times;
- "automatic" interpolation of input data not present in examples.

In order to do subsequent experiments we chose other 10 "check" songs among classical, and popular musics.

3 The Duration Quantizer

Score performance is never the simple mechanical translation of symbols into sounds with a certain pitch duration and loudness. Even agogique indications (allargando, ritenedo, accelerando, etc.), dynamical symbols (ppp, mp, fff, etc.) and expression symbols (legature, staccato, portato, etc.) coded in the musical grammar do not give a precise indication about how to generate those effects. If you add unavoidable microvariations of duration and loudness from nominal values, that are typical of personal performance style, you can understand how much a performance can deviate from its score. A determinant step toward note duration quantization is getting the right song speed. Speed could be inserted in beats per minute (bpm), but it

Brano	% ON	% OFF	% TOT
Musical moment (Schubert)	16.67	85.96	69.33
Minuetto from "Orfeo"	50.00	100.00	85.96
Waltz from "Der Freischütz"	61.11	98.67	91.40
Sad song (Cajikovskij)	33.33	90.91	75.96
Reverie (Schumann)	0.00	85.39	69.72
Toast from "La Traviata"	23.33	92.31	67.07
Funiculi funiculà (Denza)	70.59	63.16	65.93
Casta diva from "Norma"	57.14	90.98	87.50
Prima carezza (De Crescenzo)	100.00	96.40	96.92
La Paloma (de Yradier)	42.31	89.06	75.56

Tab. 1 System performance with check songs. Percents of correctness in forecasting:

% ON notes starting bars;
% OFF notes not starting bars;
% TOT all the notes.

would be restrictive for executors. We tried to find out a method as automatic as possible. In fact, our algorithm asks, at the beginning of a performance, to choose among three metronome ranges. Furthermore the algorithm uses a moving average mechanism which let the system follows also little speed variations.

4 Phase and Period Search

The modules previously described give the flow of possible accents with a one-note delay. In this subsystem we took our ideas from works by Chafe et al. [15], and Rosenthal [6]. In their work, Chafe et al. introduced the concept of anchorage: the metrical structure of a song is anchored, and determined by accents in it. Theoretically this lets you find perfectly metrical structure, but in the reality, or better in computer music, accents can be found only within a certain approximation. In our work we consider as anchorage a temporal position corresponding to a high output of the neural map. As a consequence we have three kinds of anchorage:

- true anchorage: given anchorage corresponds to an accent;
- false anchorage: anchorage does not correspond to an accent;
- ghost anchorage: anchorage has not been found but there is an accent.

So we need algorithms that eliminate false anchorages and interpolate the ghost ones. We define also IAI (Inter Anchorage Interval) as the time interval between two subsequent anchorages. In this work we are interested in finding phase and period of a song. With phase we mean the instant when a whole bar starts; period is the duration of a generic bar, that is the bar time. During an execution, from the list of pointed out anchorages we can extract hypothesis (phase, period) representing possible interpretations. These interpretations are ranked as in Rosenthal [6], but here ranking is based on a "score" computed from frequencies of coincidence of anchorages with the current hypothesis. The interpretation with higher score or, with same scores, with longer period is

chosen as current interpretation. According to Palmer and Krumhansl [16], "in order that a metrical level (or a temporal periodicity among events with same accent strength) is recognized in a musical composition, it is necessary that a sufficient number of subsequent (adjacent) beats are played at that level". So it is not enough a period repeated many times in a song, but it is also necessary to have subsequent repetitions. We developed a mechanism with two hypothesis (phase, period): the current one, used for prediction, and the substitution one, changing continuously in time. For each hypothesis, if the map confirms the forecasted beat a posteriori, then a counter is increased by the relative period, otherwise is decreased by the same value. When the two counters show that current hypothesis is not valid anymore and that the substitution one behaves better, substitution hypothesis becomes the current one. Using a counter, a hypothesis confirmed for enough time remains valid even if it is denied for a short time. With the above-mentioned algorithm it is possible to have phase and period evaluations updated to the last but one note and therefore valid for future forecasting. In fact, from those values and an elapsed time counter, it is simple to forecast if next note will be accented.

5 Algorithm Application

Concepts expressed in previous chapters have been implemented in suitable libraries and they have been used to create two versions of a graphical program called TAPPER. The first version of the program (GTAP) get data from a file, and allowed algorithm checking and parameters tuning. With songs used in neural network training, the neural map obviously behaves nearly perfectly (98%) and the whole system recognizes phase and period correctly. With "check" songs (see tab. 1), with half of them we obtain good percent of correctness ($\geq 50\%$) on every kind of notes, but bar recognition is not so satisfactory. These not good performances can be attribute to the following facts:

- a) examples for neural networks training should be chosen more carefully, especially about situations contained in them;
- b) sometimes in a song the program does not find enough elements to distinguish among multiple values of the bar length;
- c) bar beginnings falling during a note are not revealed.

On the contrary the second version of the program (MTAP) has been conceived for real-time analysis through a MIDI interface. MTAP gets MIDI data from a group of input channels and, after decoding them, applies them to algorithms; forecasted bar startings are sent in output to a MIDI channel as rhythmical sequence. The use of events scheduling allowed particularly to eliminate the behaviour seen in the previous c) point and therefore to improve system performances.

6 Conclusions

The problem of real-time rhythm recognition is complex but, with other authors' suggestions and the original idea of the neural map, we succeeded in building a system that tries to solve it in the best way possible. In fact we tried to use neural networks where they are really useful, that is to simulate the complex mechanism of rhythmical accents recognition. Numerical algorithms succeeded well in the other tasks. During the creation of each system block we kept in mind numerous results of cognitive psychology researches, and, as a consequence, we note that our system behaves similarly to human mechanisms, and we can take it as an approximated model of those behaviours. In particular it is interesting to notice that from cognitive psychology we drew only starting ideas and not psychofisiological mechanisms that allow the recognition of rhythmical accents. These mechanisms have been "extracted" automatically by neural networks during their training and using only duration and loudness of each musical note.

References

- [1] Longuet-Higgins, C. Lee "The Rhythmic Interpretation of Monophonic Music" *Music Perception*, 1, pp. 424-441, 1984.
- [2] Povel, P. Essens "Perception of Temporal Patterns" *Music Perception*, 2, pp. 411-440, 1985.
- [3] Chung, An Agency for the Perception of Musical Beats M.S. Thesis EECS, MIT Media Laboratory, 1989.
- [4] Katayose, H. Kato, M. Imai, S. Inokuchi "An Approach to an Artificial Music Expert" *Proceedings of the 1989 International Computer Music Conference*, San Francisco: Computer Music Association, 1989, pp. 139-146.
- [5] Allen, R. Dannenberger "Tracking Musical Beats in Real Time" *Proceedings of the 1990 International Computer Music Conference*, San Francisco: Computer Music Association, 1990, pp. 140-143.
- [6] Rosenthal "Emulation of Human Rhythm Perception" *Computer Music Journal*, vol. 6, n. 1, Spring 1992.
- [7] Desain, H. Honing "Quantization of Musical Time: a Connectionist Approach" *Computer Music Journal*, 13 (3), pp. 56-66, 1989.
- [8] Johnson-Laird "Rhythm and Meter: a Theory at the Computational Level" *Psychomusicology*, 10, pp. 88-106, 1991.
- [9] Shaffer et al. "Metre and Rhythm in Piano Playing" *Cognition*, 20, pp. 61-77, 1985.
- [10] Clarke "Structure and Expression in Rhythmic Performance" *Musical Structure and Cognition*, London: Academic Press Inc. Ltd., 1985.
- [11] Drake, C. Palmer "Accent Structure in Music Performance" *Music Perception*, vol. 10, n. 3, pp. 343-378, Spring 1993.
- [12] Sloboda "The Communication of Musical Metre in Piano Playing Performance" *Quarterly Journal of Experimental Psychology*, 35A, pp. 377-396, 1983.
- [13] Handel, Listening, Cambridge Massachusetts: MIT Press, 1989.
- [14] Morris, D. Rosenthal "The Effect of Varying Loudness and Duration on Human Rhythm Perception" Technical Report, Cambridge Massachusetts: MIT Media Laboratory, 1990.
- [15] Chafe, B. Mori-Reynaud, L. Rush "Toward an Intelligent Editor of Digital Audio: Recognition of Musical Constructs" *Computer Music Journal*, vol. 6, n. 1, Spring 1982.
- [16] Palmer, C.L. Krumhansl "Mental Representation for Musical Meter" *Journal of Experimental Psychology: Human Perception and Performance*, vol. 16, n. 4, pp. 728-741, 1990.

Soul: un sistema per la classificazione di segnali musicali basato su reti neurali

Claudio Cioffi

DIST - Università di Genova. - Laboratorio di Informatica Musicale
Via Opera Pia 13 - 16145 Genova - Italy

Abstract

The object of this description are the development and implementation of a model based on an architecture able to imitate the processes performed in the human auditory system in order to recognize and classify sounds. This paper is intended to revise a system presented at the IX CIM, the approach of which proved to be inadequate to reach the biological capability of distinguishing a sound source (typically a musical instrument) from noisy backgrounds which can not be defined in terms of quantities typical of the signal theory (frequency, amplitude, correlation...). Most models developed in literature treat either the analytical properties of peripheral auditory system or the properties of synthesis of tone completion processes. Such processes have rarely been integrated. The SOUL (Self Organizing Universal Listener) system, instead, employs neural nets whose input data are high-level features extracted from devices modelling the human ear. Such devices employ output information produced by an algorithm (proposed by E. Terhardt) which simulates the tone completion processes. This information is used to perform an adaptive filtering by making recourse to a bank of variable central frequency filters, the form of which derived from L. Van Immerseel e J.P. Marteens's studies.

1. Introduzione

I problemi principali da risolvere progettando sistemi di riconoscimento riguardano di solito lo sviluppo di algoritmi in grado di estrarre caratteristiche "ad alto livello" dall'input, nel nostro caso consistente in suoni rappresentati da sequenze di campioni binari a 16 bit. Si è in particolare dovuto affrontare il problema di formalizzare dei concetti di "timbro" e "pitch".

Come è noto il timbro può essere descritto *matematicamente* da un vettore contenente le ampiezze delle varie armoniche. Da un punto di vista *psicoacustico*, è invece quella proprietà di un suono che determina nell'ascoltatore una data sensazione di *qualità* ("colore").

È risaputo che tali due definizioni sono *in contrasto tra loro*, in quanto *timbri che dal punto di vista psicoacustico possono essere considerati identici sono assolutamente diversi fisicamente*. Ciononostante, il sistema uditivo è in grado con apparente facilità di trovare nei suoni provenienti dallo stesso tipo di strumento delle caratteristiche comuni che sembrano sfuggire ad un'analisi matematica.

Molto è stato scritto in letteratura su come l'informazione proveniente dall'inviluppo possa aiutare: ho eseguito per verifica un esperimento facendo ascoltare a persone dotate di orecchio non allenato solo la fase di "sustain" di tre strumenti: pianoforte Steinway, pianoforte Bechstein, tromba. Ebbene: la totalità dei soggetti ha suddiviso correttamente i suoni in due classi (la classe "pianoforte" e la classe "tromba"), sebbene, salvo qualche eccezione, non sia riuscita a stabilire di quali strumenti si trattasse.

Sembra dunque che l'assenza d'inviluppo ostacoli solamente la connessione tra percezione e base di conoscenza acquisita, ma che non ostacoli il processo di raggruppamento e classificazione di un dato insieme di timbri.

Appare quindi chiaro che la chiave di soluzione del problema risiede nella determinazione di quelle proprietà dei timbri che sfuggono a un'analisi matematica, ma che sono molto ben individuate dal sistema uditivo. Per rendere un calcolatore in grado di eseguire una siffatta classificazione occorrerà un approccio non tanto matematico quanto il più possibile *antropomorfo*.

Il primo problema è stato rendere la risposta del sistema invariante rispetto alla nota eseguita dallo strumento, nonostante il contenuto armonico cambi in funzione di essa.

Tale processo consiste nella formalizzazione del concetto di "pitch" e nella sua determinazione. Dal punto di vista psicoacustico, il "pitch" non è altro che la sensazione di "altezza" che

un dato suono armonico genera nel nostro apparato uditivo. È misurato in "unità di pitch" e definito come il valore numerico della frequenza di una sinusoide pura avente intensità pari ad un livello di pressione sonora di 40 dB.

A causa di questa definizione si tende spesso a confondere il concetto di "pitch" associato ad un dato suono con quello di "frequenza del suono". Ciò non è corretto, in quanto:

1. Non ha senso associare una frequenza ad un suono non sinusoidale, composto dunque da più frequenze.
2. Non ha neppure senso considerare il pitch come la frequenza più bassa tra quelle che compongono il suono in questione, dal momento che il nostro orecchio può fornire sensazioni di pitch che in certi casi (pitch "virtuale", o ricostruzione della fondamentale mancante) non corrispondono ad alcuna frequenza fisicamente presente nel suono percepito, come è stato dimostrato dagli studi di E. Terhardt.
3. Una stessa sinusoide pura può evocare sensazioni di pitch differenti secondo il suo livello d'intensità sonora ed a seconda che siano presenti o meno altri stimoli sonori simultanei ("masking").

Dal punto di vista matematico, ciò che si avvicina di più al concetto di pitch è l'inverso del periodo di una forma d'onda. Questa definizione non è tuttavia applicabile al caso pratico, in quanto un suono reale è una forma d'onda continuamente affetta da svariate imprecisioni rispetto alla formulazione teorica, che non permettono l'utilizzo di un approccio matematico. Tra le più rilevanti possono essere citate:

- ✓ Micro-variazioni nel periodo della forma d'onda che, non essendo percepibili dal sistema uditivo umano, non influenzano la sensazione di pitch prodotta dal suono.
- ✓ Armoniche la cui frequenza non coincide esattamente con un multiplo della fondamentale, ma se ne discosta di minime quantità.

Vi è infine un ultimo fattore da considerare: essendo il pitch un prodotto del sistema uditivo, questo può essere percepito solo quando il suo valore rientra nel range di udibilità dell'orecchio. Il risultato di ciò è che una serie di sinusoidi, pur avendo frequenze comprese nel range 20-20000 Hz, potrebbe essere tale da formare un'onda sonora il cui periodo è molto più lungo di 1/20 Hz. In tal caso l'ascoltatore, pur udendo il suono, non riuscirà ad associarvi alcuna intonazione. Si parla in tal caso di "residui anarmonici".

2. Il sistema SOUL

Nell'elenco seguente sono fissate alcune delle linee guida che si è cercato di seguire durante lo sviluppo del sistema. Esse sono:

- Poiché una persona distingue tra sequenze di campioni relative a suoni emessi da strumenti diversi, in esse deve esserci tutta l'informazione necessaria ad effettuare la distinzione.
- L'analisi in frequenza non dà luogo ad alcuna perdita di informazione (da una FFT infatti si può sempre riottenere il suono originario mediante antitrasformazione), per cui lo spettro di un suono contiene tutta l'informazione originaria.
- Lo spettro d'un suono contiene molta più informazione di quanta ne occorre (né esiste, d'altra parte, un computer abbastanza potente da poter ricevere come input di una rete neurale le migliaia di campioni che costituiscono una FFT): occorre pertanto riuscire ad estrarre solo i dati "rilevanti".
- Tale processo di estrazione deve uniformarsi il più possibile a ciò che avviene nell'orecchio umano.
- Il sistema finale deve poter "riconoscere" strumenti musicali suonanti note singole, ossia deve essere in grado di raggruppare in un'unica classe note diverse suonate dallo stesso strumento o da strumenti differenti ma dello stesso tipo.
- Il riconoscimento dovrebbe preferibilmente avvenire anche in condizioni "rumorose", ossia quando il suono emesso dallo strumento da classificare è mescolato con altri suoni o rumori qualsiasi che mascherano parzialmente il segnale.

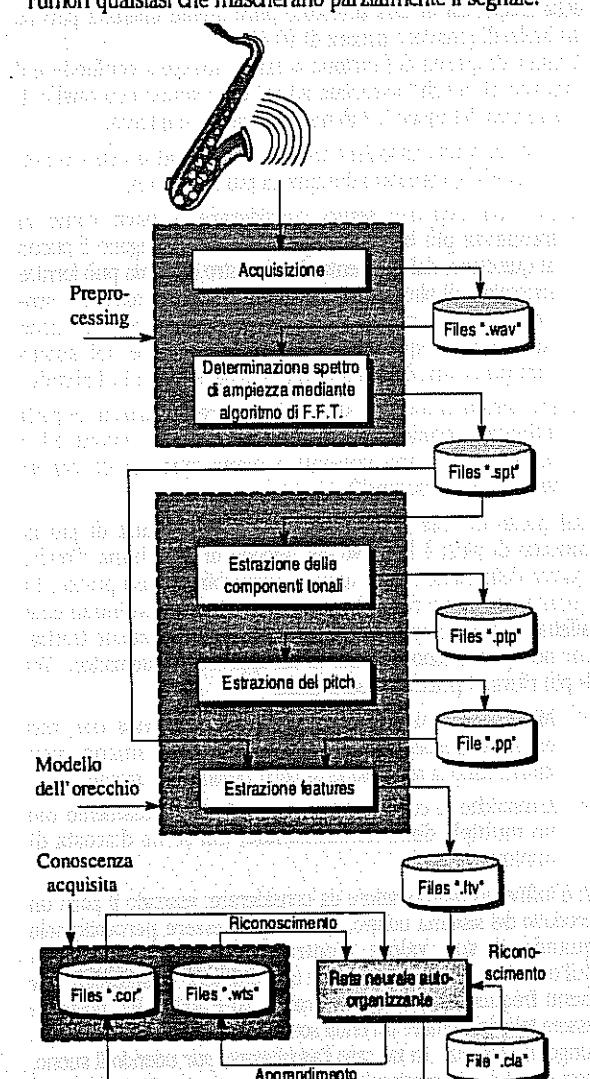


Fig. 1: Struttura a blocchi del sistema SOUL

Lo studio accurato delle problematiche che stanno dietro a tali punti ha condotto ad un sistema strutturato come in fig. 1. Esso può suddividersi essenzialmente in tre moduli principali:

- Preprocessing, che acquisisce il suono reale mediante campionamento, normalizzazione ed analisi in frequenza. In uscita fornisce un vettore rappresentante il modulo dello spettro d'ampiezza del suono normalizzato al valore max.
- Un modello dell'orecchio, che simula il comportamento del sistema uditivo periferico (orecchio esterno, medio e interno).
- Una rete neurale, che simula il comportamento della parte centrale del sistema uditivo (corteccia uditiva).

I capitoli seguenti contengono le descrizioni del funzionamento e delle problematiche relative ad ogni singolo blocco.

2.1. Acquisizione dei segnali

Si è optato in SOUL per una rappresentazione a 16 bit per limitare al massimo la distorsione da errore di quantizzazione.

Per quanto riguarda il campionamento, la soluzione ideale sarebbe stata usare la frequenza standard di 44100 Hz utilizzata nei CD audio, che garantisce la ricostruzione di tutte le componenti del segnale percepibili dal sistema uditivo umano. Tale soluzione si è rivelata però troppo onerosa, sia per l'occupazione di spazio su disco dei file contenenti i campioni sia per l'aumento eccessivo dei dati da trattare (specialmente nella FFT). Si è d'altra parte constatato che apparecchi per la riproduzione del suono non ad "alta fedeltà" (radioline AM, telefoni, ecc.), pur avendo bande passanti molto limitate, non ostacolano significativamente il riconoscimento di uno strumento musicale da parte del sistema uditivo. Si è allora concluso che un'estesa banda passante non fosse fondamentale per un buon comportamento del sistema in fase di riconoscimento, per cui si è scelto di utilizzare una frequenza di campionamento pari a 22050 Hz.

2.2. Scelta del Training Set

Il problema da risolvere è stato descrivere esaurientemente ogni classe con un numero di esempi limitato: l'occupazione di spazio disco e il tempo materiale per eseguire l'acquisizione sconsigliavano l'uso di un Training Set costituito da più di un centinaio di suoni. Gli expander a disposizione (Roland U-220, E-MU Proteus; Akai S900) contenevano inoltre un ristretto numero di suoni diversi per ciascun tipo di strumento musicale, per cui il campionario disponibile era piuttosto limitato.

Tenuto conto di ciò, si è deciso d'utilizzare un Training Set composto da esempi di cinque classi di strumenti ben distinguibili:

- Bassi elettrici ed acustici pizzicati (18 esempi)
- Pianoforti (15 esempi)
- Strumenti ad arco (9 esempi)
- Sassofoni (9 esempi)
- Ottoni (27 esempi)

Ogni strumento è stato acquisito tre volte, nell'atto di suonare rispettivamente le note: D01, F01 e S11. Sono stati acquisiti inoltre alcuni suoni di test, emessi da strumenti appartenenti ad una delle classi sopra citate, "sporcati" con rumori di sottofondo (altri strumenti, percussioni, ecc.). Per l'acquisizione ci si è serviti di una scheda sonora Sound Blaster AWE 32, stabilendo che ogni file sonoro dovesse rispettare le seguenti specifiche:

- ✓ Il suono va registrato al massimo volume possibile senza che insorgano fenomeni di "clipping", così da sfruttare tutta l'informazione disponibile nella dinamica a 16 bit.
- ✓ Il suono deve essere completo: devono cioè essere presenti tutte le fasi dell'inviluppo, dall'attack al release. La fase di attack deve esaurirsi entro il campione N° 5000 e quella di sustain deve avere durata minima di 16384 campioni.
- ✓ La lunghezza del file deve essere sfruttata al massimo, per ottenere il miglior rapporto possibile tra mole di dati da elaborare ed informazione effettivamente processata.

2.3 Analisi in frequenza

L'algoritmo di FFT è stato implementato con particolari accorgimenti che hanno permesso di minimizzare i tempi di calcolo. La forma della finestra temporale adottata è quella di Hanning, poiché è risultata la più efficiente nell'evidenziare i picchi dovuti alle componenti armoniche del segnale.

2.4 Filtraggio

Seguendo le indicazioni di Van Immerseel, la trasmissione sonora di un segnale attraverso la catena esterna e media dell'orecchio è stata modellata con un filtro passa basso del secondo ordine avente una frequenza di risonanza di 4 kHz.

La coclea è invece modellata essenzialmente da un banco di filtri passa banda, le cui frequenze centrali non sono però spaziate in maniera equidistante. Si è ritenuto allora utile definire una unità di misura, detta *unità di banda critica*, (Terhardt 1980) rispetto alla quale le frequenze centrali dei filtri fossero distribuite in maniera equidistante. Il range di frequenze per le quali un dato filtro viene eccitato prende il nome di *banda critica* relativa al filtro in questione. Il mapping tra frequenze f (in kHz) e unità di banda critica u risulta, da dati sperimentali, essere ben approssimato dalla funzione:

$$u(f) = \begin{cases} 11.82 \operatorname{arctg}(1.21f) & f \leq 1.5 \text{ kHz} \\ 5 \ln(f / 1.5) + 12.61 & f > 1.5 \text{ kHz} \end{cases}$$

Tale formula permette di calcolare quale filtro andrà a risuonare sotto lo stimolo di una data frequenza f .

Per essere in accordo con i dati psicofisici sperimentali, i filtri passa banda hanno le seguenti caratteristiche:

- ✓ una risposta compresa tra +0dB ÷ -3dB nel range di frequenze in cui il filtro è quello che risuona più forte.
- ✓ una pendenza di circa 10 dB/unità di banda critica dal lato delle basse frequenze e di circa 20 dB/unità di banda critica dal lato delle alte frequenze.

L'innovatività del sistema consta tuttavia nel fatto che **non tutti i filtri si attivano contemporaneamente in presenza del segnale in ingresso**: si è infatti ipotizzata la possibilità che processi conoscitivi derivanti da fenomeni d'apprendimento possano influenzare la risposta dei singoli filtri del banco cocleare. In particolare, ad es., la continua esperienza uditiva di suoni di tipo armonico potrebbe generare nell'ascoltatore un meccanismo che, non appena venga individuato il pitch del suono percepito, predisponga il sistema a "udire" usando in particolar modo, tra tutti i filtri disponibili nella coclea, quelli la cui distanza in frequenza corrisponde a quella presente tra le componenti di una serie armonica.

Ciò che si è fatto in SOUL è proprio il simulare l'attivazione solo di quei filtri che si trovano esattamente in corrispondenza dei punti dove dovrebbero trovarsi la fondamentale del suono e le sue armoniche. Per fare ciò occorre ovviamente conoscere preventivamente quale sia il "pitch" del suono in input: a questo provvede il modulo descritto di seguito.

2.5 Estrazione del pitch

La procedura è basata sulla teoria del pitch virtuale (Terhardt, 1974), ed in particolare sul principio che la percezione del pitch dipende sia da un ascolto analitico (generante un pitch spettrale) che da uno a più alto livello (generante un pitch virtuale).

La più o meno ambigua percezione del pitch governata da questi due tipi di ascolto è descritta da due corrispondenti pitch pattern: spettrale e virtuale, consistenti ognuno in 8 valori di pitch, cui sono associati 8 pesi corrispondenti, teneri conto della relativa "predominanza" di ciascuno degli 8 pitch. Il pattern di pitch spettrali è determinato mediante un'analisi spettrale, una estrazione delle componenti tonali, una valutazione di effetti di mutuo mascheramento e un aggiustamento dei pesi in accordo col principio della dominanza spettrale, mentre il pattern dei

pitch virtuali è ottenuto da quello dei pitch spettrali mediante un algoritmo di stima della coincidenza dei subarmonici. Al termine di questo calcolo, il pitch (spettrale o virtuale) avente il peso più elevato sarà assunto essere "il" pitch del suono in ingresso. L'algoritmo è descritto in (Terhardt et al., 1982).

2.6 La rete neurale auto-organizzante

Il banco di filtri adattivi, opportunamente pilotati dal modulo di determinazione del pitch, estraggono dunque le feature ad alto livello, rappresentate nella fattispecie dalle potenze in uscita dai dieci filtri che si trovano in corrispondenza delle dieci prime armoniche del suono in input.

Poiché non è noto a priori come elaborare certe informazioni, e si vuole in un certo senso che la rete "scopra" da sola concetti interessanti in una popolazione di dati non strutturati, si è optato per l'utilizzo di una rete "auto organizzante", facente dunque uso di un algoritmo di learning di tipo non supervisionato.

È stato scelto l'algoritmo di Kohonen, sia per la sua semplicità, sia per la possibilità che offriva di testare immediatamente, senza sviluppare algoritmi complessi di cui non era dimostrata l'efficacia, la bontà del modello dell'orecchio implementato.

In SOUL si è optato, in mancanza d'indizi specifici riguardanti la dimensionalità dello spazio dei timbri, per una rete bidimensionale. Sono state effettuate in particolare le seguenti scelte:

1. Il criterio di vicinanza topologica per la definizione della bolla di attivazione è di tipo euclideo e la bolla è di forma sferica.
2. Sono definiti due valori *max* e *min* sia per il tasso di apprendimento α che per il raggio della bolla di attivazione r ed i valori di $\alpha(t)$ e di $r(t)$ vengono fatti diminuire linearmente tra *max* e *min* durante il ciclo di learning.
3. I valori di *min* e di *max* per α sono stati fissati a 0.2 e 0.8, mentre quelli per il raggio della bolla sono stati fissati rispettivamente a 0 e alla più grande tra le dimensioni della rete.

3. Risultati ottenuti

Nella fig. 2 è riportato un esempio di mapping. Ogni cella rappresenta un neurone della rete. Le sigle nelle caselle corrispondono ai nomi dei file ".wav" contenenti i suoni utilizzati nel training set. È utilizzata la seguente convenzione:

- Le prime tre lettere indicano il tipo di strumento, e sono:
"bas" → basso; "str" → strumento ad arco
"trp" → tromba "pno" → pianoforte;
"sax" → sassofono "trb" → trombone
- La quarta lettera indica il sintetizzatore da cui è stato effettuato il campionamento:
"p" → E-MU Proteus "u" → Roland U-110
- Le due cifre successive indicano il preset interno ai moduli di sintesi sonora da cui è stato effettuato il campionamento.
- Le ultime lettere indicano la nota eseguita dallo strumento.

Come si può notare, la clusterizzazione ottenuta nello spazio di uscita è soddisfacente, eccetto per due soli campioni, uno di sax e uno di violoncello, i quali vanno a mapparsi in zone "estranee" al loro gruppo di appartenenza. In conseguenza di tale clusterizzazione, la percentuale di riconoscimento è molto buona. Tutti i campioni del training set sono naturalmente riconosciuti. Per ottenere una stima delle prestazioni del sistema, sono stati condotti i seguenti esperimenti:

1. È stata ripetuta più volte la fase di apprendimento usando ogni volta un training set in cui era escluso, a turno, ciascuno dei 78 campioni utilizzati. Tale campione era poi presentato alla rete come suono incognito da classificare. Così facendo si sono ottenute 71 classificazioni corrette e 7 errate. Due di queste ultime si riferiscono ai due campioni sopra citati (STRU67SI e SAXP16SI), mentre i rimanenti si riferiscono tutti a campioni che in fig. 2 si trovano in zone "di confine" (PNOP03SI, PNOU07FA, PNOP04FA e TRBP23SI).

2. Sono stati utilizzati 15 campioni incogniti, tre per ognuna delle cinque classi, acquisiti dall'Akai S-900 (il training set proviene tutto dal Roland U-110 e dall'E-Mu Proteus). Il sistema è riuscito a classificare correttamente 13 campioni, mentre ha fallito negli altri due casi.
3. Gli stessi campioni utilizzati al punto (2) sono stati "sporcati" con suoni estranei in sottofondo (rispettivamente un colpo di piatti e un accordo di chitarra elettrica per ogni campione). Dei 30 campioni così ottenuti, il sistema è stato in grado di riconoscerne 24, fallendo in 6 casi.

Un interessante risultato consiste comunque nel fatto che *in tutti i casi*, anche quelli in cui la classificazione è stata errata, il sistema è stato in grado di riconoscere correttamente quale fosse la nota suonata dallo strumento in esame, a riprova dell'ottimo funzionamento del modulo di estrazione del pitch.

3.1 Prestazioni del sistema e possibilità di utilizzo in tempo reale

Al fine di valutare le possibilità di utilizzo in tempo reale di SOUL, sono state eseguite delle misurazioni sui "tempi di risposta" di ogni modulo del sistema, eseguite su due calcolatori differenti, uno basato su un processore i80486 dx funzionante ad un clock di 33 MHz ed uno basato su un processore Pentium™ funzionante ad un clock di 90 Mhz. Sono stati scelti i seguenti calcolatori perché rappresentano rispettivamente il sistema base, al di sotto del quale non è consigliato scendere per poter utilizzare SOUL, ed il sistema dalle massime prestazioni in questo momento disponibile.

La durata dell'acquisizione è fissa, poiché corrisponde alla durata del suono stesso. La durata della fase di learning dipende linearmente dal n° di passi. Per "istantaneo" si intende di durata così breve da non essere misurabile manualmente.

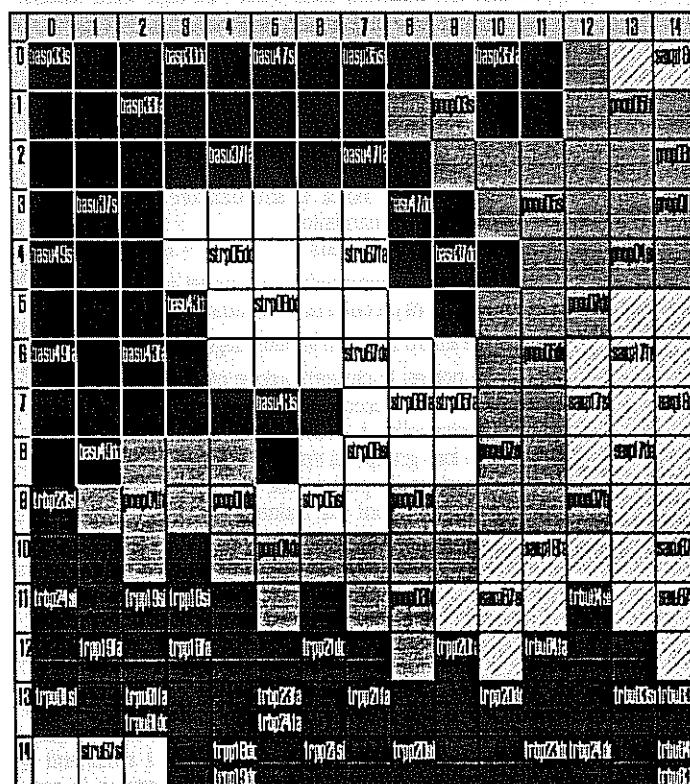


Fig. 2: esempio di mapping ottenuto su rete 15x15, learning 100 passi.

4. Osservazioni e commenti ai risultati

- Dalla distribuzione dei timbri nello spazio di uscita si nota come gli strumenti che si raggruppano meglio sono gli ottoni e i bassi, mentre il cluster relativo al pianoforte si presenta spesso di forma allungata, a volte addirittura divisa in due "sottocluster", a conferma della complessità armonica di questo tipo di strumento.
- Chiaramente i problemi di riconoscimento si hanno soprattutto nelle regioni di transizione tra una classe ed un'altra. L'influenza delle zone "perimetrali" è piuttosto marcata a causa del limitato numero di campioni del training set. Si è dovuto tenere limitato tale numero per i motivi esposti in precedenza, ma è indubbio che aumentandolo la percentuale di errore dovrebbe diminuire sensibilmente.
- È stato notato che gli errori avvengono soprattutto in corrispondenza delle note più acute (Fa-Si). Ciò può essere dovuto all'aumentare dell'errore di campionamento, tuttavia tale problematica andrebbe approfondita.

- La disposizione finale dei cluster cambia al variare dei valori random iniziali dei pesi dati ai neuroni. Il sistema è tuttavia consistente, in quanto mantiene le caratteristiche di base riguardanti la posizione relativa dei campioni del training set, indipendentemente dalla loro posizione assoluta. Ad es. certe coppie di campioni si trovano sempre vicine tra loro, bassi e ottoni sono ai lati opposti della rete, il cluster relativo al sax confina sempre con quelli relativi a tromba e pianoforte. Anche in questo caso sarebbe interessante determinare il rapporto causa-effetto alla base di tale tipica disposizione.

- Il sistema sembra infine essere invariante rispetto alla particolare nota suonata. A riprova di ciò, eseguendo tre fasi di apprendimento separate per ogni nota (una con i 26 campioni che suonano il Do, una con i 26 che suonano il Fa ed una con i 26 che suonano il Si), non si nota alcun miglioramento nella clusterizzazione, segno che la nota differente non disturba in maniera significativa il funzionamento del sistema.

Bibliografia

- Bertelli et al. - "Soul: un senso acustico adattivo per il riconoscimento di sorgenti sonore", Atti IX CIM, 1991.
- C. Ciolfi - "Un sistema per la classificazione di segnali musicali basato su reti neurali", Tesi di Laurea. Dist Università di Genova, 1995.
- G. De Poli, P. Tonella - "Self-Organizing Neural Networks and Grey's Timbre Space", Proc. ICMC, 1993.
- J.M. Grey - "Multidimensional perceptual scaling of musical timbres", in *The Journal of the Acoustical Society of America*, Vol. 61 N° 5, 1977.
- T. Kohonen - "Self-Organization and Associative Memory", Springer-Verlag, 1988.
- M. Lerman - "Emergent Properties of Tonality Functions by Self-Organization", in *Interface*, Vol 19 N. 2-3
- R.J. Ritsma - "Frequencies dominant in the perception of the pitch of complex sounds", in *The Journal of the Acoustical Society of America*, Vol. 42, 1967.
- E. Terhardt - "Pitch, consonance and harmony", in *The Journal of the Ac. Society of Am.*, Vol. 55, N° 5, 1974.
- E. Terhardt et al. - "Algorithm for extraction of pitch and pitch salience from complex tonal signals", in *The J. of the Acoustic Society of America*, Vol. 71, N° 3, 1982.
- E. Terhardt et al. - "Analytical expressions for critical band rate and critical bandwidth as a function of frequency", in *The J. of the Ac. Soc. Am.* Vol. 68, 1980.
- L.M. Van Immerseel et al. - "Pitch and voiced/unvoiced determination with an auditory model", in *The Journal of the Acoustic Society of America*, Vol. 91, N° 6, 1992.

A Hybrid System for the Analysis of Musical Signals: HARP/TCAD

Giorgio Allasia, Antonio Camurri

University of Genova

DIST - Department of Communication, Computer and System Sciences

Computer Music Laboratory

Via Opera Pia 13, 16145 Genova (Italy)

Marc Leman

University of Gent

IPEM - Institut for Psychoacoustic and Electronic Music

Blandijnberg 2, B-9000 Gent (Belgium)

Abstract

This paper presents a model of hybrid system for the analysis of musical signals, based on both symbolic and subsymbolic representations. Our work starts from TCAD - *Tone Center Attractor Dynamics* (Leman 1995), a subsymbolic model of schema-based tone centers and chords recognition of musical signals, and the HARP system (Camurri et al. 1995). The HARP model is here used to integrate the knowledge of the two different domains of TCAD, the former extracting the tonality out of a music signal and the latter extracting the chords. HARP operates at two levels: a symbolic one, where concepts, their taxonomic relationships, production rules and high-level temporal knowledge are represented, and a subsymbolic one, where agents operating on music signals and auditory images are hooked to concepts and dynamically connected according to the relations defined in the symbolic level among concept instances. We have developed a HARP model as an evolution of the one described in (Camurri and Leman 1992) which embeds agents implementing TCAD, and we completed it with further symbolic agents in order to improve the analysis capabilities. Results of the current implementation are described in the paper. Further research regards the analysis of the functionality of a musical piece.

1 Introduction

AI-systems for musical signal applications that allow multiple representation levels (including symbolic and subsymbolic) are here called *hybrid systems*. As stated elsewhere (Camurri et al 1994), with the term "symbolic" we refer to representation systems whose "atomic" constituents are in turn representations. Such systems have a syntax (which defines well-formed expressions), and a semantics (which defines what the symbol represents and how the logical relationships between them should be interpreted). A "subsymbolic" representation is made of constituent entities that are *iconic* representation, e.g., signals, sound samples, auditory images. The main property of iconic representations is that the form is identical to the content.

2 The HARP Hybrid System

HARP (Camurri et al 1995) is a hybrid system for the representation and real-time processing of music and multimedia, which allows the specification of one-to-one connections between symbolic and subsymbolic agents in a hybrid architecture. Subsymbolic agents typically operate on auditory images and acoustical signals. Symbolic agents are here delegated to the manipulation of higher level entities emerged from TCAD subsymbolic agents (e.g., chord instances).

From the point of view of the representation and reasoning architecture, the formal realisation of the HARP model consists of a scheme combining

different formalisms, where it is possible to make a distinction between:

- a) A *Long Term Memory* (LTM), containing two knowledge-bases: one is symbolic and it is based on a semantic network language of the family of KL-ONE (Woods and Schmolze 1992), a simple temporal logic language and production rules. The other is subsymbolic, based on a library of "behaviors", and can be learned by the system by means of self-organisation;
- b) A *Short Term Memory* (STM), which is a working memory representing a single context. The STM is formed by a symbolic component to represent individual entities, based on a sub-set of first-order logic, and a subsymbolic component, which can be related to the signal or to the acoustic and perceptual levels.

The symbolic and subsymbolic entities defined in the STM have a one-to-one correspondence. The subsymbolic part of the STM consists of a network of subsymbolic agents based on reasoning by metaphors and dynamic systems. The symbolic STM basically works as an interface between the LTM symbolic component and the subsymbolic STM: it is the core of the "fluent" connection between the representation levels.

The symbolic components are characterized by the classification and recognition algorithms typical of term-subsumption languages (Woods and Schmolze, 1992), integrated with temporal reasoning and rule-

based inference mechanisms. In more detail, given a taxonomy, the symbolic classification algorithm allows to find the correct position of a newly introduced concept, starting from its properties. Recognition algorithms allow the classification of individual constants.

Subsymbolic agents can be linked to real physical signals, images, or can be metaphors of the external word. Subsymbolic reasoning is performed by the execution of networks of agents performing simulations and measurements. Agents act either directly on the real environment or on a mental model, i.e., an internal representation, a metaphor of the real word.

The subsymbolic agents - the *experts* and the *icons* - are hooked to symbols, and are implemented in HARP as Microsoft OLE Automation clients and servers respectively. They are dynamically loaded/released by the HARP hybrid planner according to the context update. The HARP hybrid planner is therefore able to instantiate, connect and control the communications between agents and icons in a given context.

3 The TCAD Ontology

In this section the structure of the high level taxonomy, which is the symbolic part of the LTM and the starting point of the music knowledge base in our system, is described. Figure 1 shows a part of such long term knowledge, which can be part of a more general music knowledge base. The complete KB for the HARP-TCAD model is described in detail in (Allasia 1994). The graphical notation is the same as in KL-ONE: ellipses represent concepts, double arrows represent IS-A specialization links (e.g.: *fragment* IS-A *music_situation*), and boxes connected by single edges represent relations (*roles*) between concepts. The concept *music_situation* is described as a specialization (i.e. a subclass) of the more general concept *situation*. Therefore, *music_situation* inherits the roles *sit_beg* and *sit_end*. Grey concepts are aliases for a better graphical presentation of the symbolic KB.

The concepts *action* and *situation* subsume all concepts representing entities characterized by a temporal location; *situation*'s represent states of the world in which no significant change occurs: a *situation* holds for a certain interval, delimited by the instants corresponding to the fillers of the *sit_beg* and *sit_end* temporal roles. An *action* produces some kind of change in the world: for this concept, the roles *input* and *output* are defined. For each action, the filler of *input* is the state of the world before the action is performed, whereas the filler of *output* corresponds to the state of the world after the action is performed.

In the part of the taxonomy under the *situation* concept, we find the *music_situation* concept which subsumes, through *compound_music_situation*, the *music_piece* concept. The *music_piece* concept

represents a general music signal that we consider as composed by one or more *fragments* (the *sit_p_of1* aggregate role) and one *conclusion* (the *sit_p_of2* aggregate role). The *fragment* concept represents a short fragment of an acoustic music signal that we want to analyse according to the TCAD outputs. The *fragment* concept has three feature roles that have as value restriction the concepts *tone_center*, *chord*, and *degree*. The first two correspond directly to the TCAD outputs and, since TCAD outputs are specified with respect to all tone center images in terms of correlation coefficients, these concepts have two feature roles (*has_tone_value* and *has_chord_value*) that represent the value of these coefficients. The *degree* concept represents the evaluation of the functionality of a certain chord in a particular tonal context, and its value restriction concept *degree_value* is a kind of "merit factor" of a possible interpretation of functionality. The concept *cadence* is directly subsumed by *music_situation* and represents particular musical phrases that is possible to extract from the musical signal. The *cadence* is formed_by a set of chords and *has_tonality* a *tone_center*.

On the other side of the global taxonomy we have the concept *action* which subsumes the concepts *compound_action* and *atomic_action*. A *compound_action* represents complex actions which can be splitted in single actions or *atomic_actions*. In our example the concept *analysis*, which represents the complete analysis of the musical signal, has two aggregate roles with *tcad* and *cadence_analyzer*, that are concepts subsumed by *atomic_action* and representing the two modules of our analysis on the musical signal. To complete the taxonomy we must add the concepts that represent the schema used by TCAD for its recognition. For the schema of the tone center we have 24 concepts representing the 12 tonalities both in the major and in the minor scale. These 24 tone centers are the stable points of attraction inside the schema. These concepts are subsumed by the concept *tone_center*. For the schema of the chords we have 115 concepts representing the chords used by the TCAD agents working in the domain of chords recognition as stable points of attraction. These concepts are subsumed by an alias of the concept *chord*. Finally we consider the seven possible degrees of the functionality, represented by seven concepts subsumed by an alias of the concept *degree*.

4 Musical Signal Analysis

The *experts* and *icons* subsymbolic agents are counterparts of *actions* and *situations*, respectively. Let us now see how the system operates. Starting from a user-defined action instance and some related assertions, a corresponding initial context for either action simulation or plan execution is built by the system. This is accomplished by completing the set of

initial assertions with the possible further assertions of actions, situations and other concepts which are involved. Then the system finds out and instantiates their corresponding agents in the subsymbolic STM: the communication links between them are defined on the basis of topology of the KL-ONE net. The mechanisms of context completion, of its updating while performing actions and the management of the links between subsymbolic entities are supervised by the hybrid planner, which corresponds to a subsymbolic inference mechanism, described with more details in (Camurri et al., 1995).

In our example an icon is hooked to the concept *fragment* and an agent to the concept *tcad*. The icon is implemented as a class in an object oriented concurrent environment. Precisely, the class is contained in a Windows application (built in Visual C++) behaving as a server in the Microsoft OLE Automation environment. When the class is instanced, it contains the data filtered from the TCAD output files. The class has an interface of functions (methods) that allow to manipulate its data, and that can be invoked by other OLE objects. The TCAD agent is a class contained in another Windows application acting as a OLE Automation client. In this way the agent "knows" of the existence of the icon and can invoke its methods.

Our analysis starts with a user-defined instance of a *fragment* concept. Since the *tcad* action is triggered on the *fragment* concept, the system instantiates an agent hooked to *tcad* which begins to process the data contained in the icon. When the agent detects that a new chord emerged from the data, it asserts a new situation that corresponds to that new fragment, and terminates. At this point, calls to the methods in the icon's interface that identify the parameters of the new situation are invoked, and new assertions are added as instances of the corresponding concepts (*tone_center*, *chords* and their taxonomy). After this, HARP invokes a symbolic method (a production rule) which infers the functionality of the fragment and assert it. At this point, a new agent hooked to *tcad* is instantiated, which searches for a new chord and so on until the end of the music signal data.

The last agent terminates and assert an instance of the concept *conclusion* on which is triggered the concept *cadence_analyzer*. A new instance of this concept invokes symbolic agents which, by applying simple rules, try to recognize different types of cadences from the symbolic data instanced. When an agent "recognizes" a cadence, it asserts it with the links to the chord instances that form it.

Once that the knowledge database referring to a particular musical piece object of the analysis is created, it is possible to query it, by means of the *Query* module of the HARP system. This module allows the user to define queries, using a Prolog-like syntax (see figure 2). In this way it is possible to extract analysis details we are interested in. It is also

possible to introduce and activate symbolic agents whose execution can update the database with new assertions. These symbolic agents can be invoked either by the user, or they can be "triggered" automatically, e.g., on a particular feature instance extracted by a subsymbolic feature extractor agent.

5 The Case Study

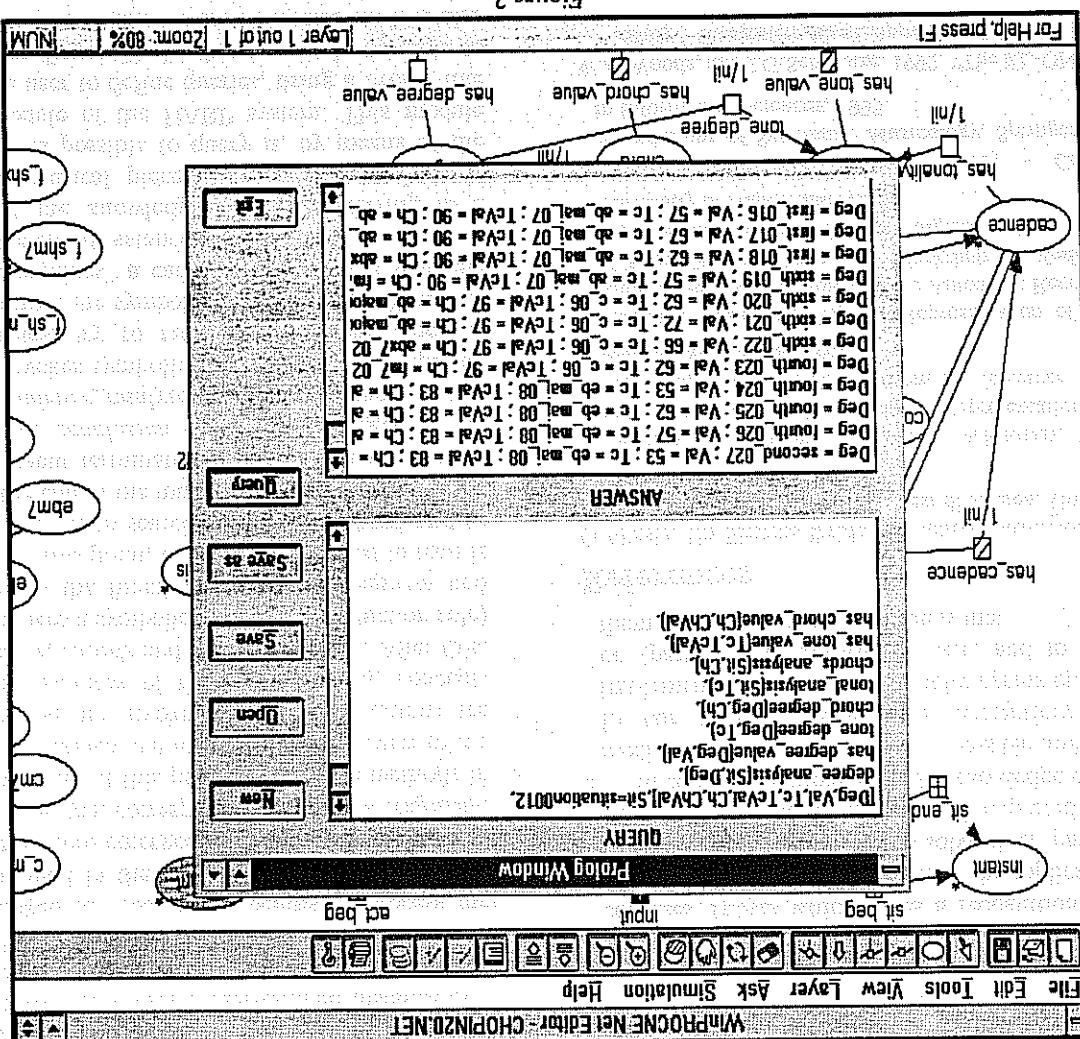
The musical piece chosen as example and test of the system is Chopin's Prelude N.20. Here we consider only the first four measures of the piece, since there is one interpretation of their functionality commonly accepted from the musicologists, but they contain some stylistic ornaments originating possible ambiguities. In particular we have found that the results coming out from the analysis performed by the hybrid system were very interesting in the following cases:

- During the modulation from one tonality to another, TCAD, due to the attractor dynamic's inertia, perceives this transition few seconds later and this creates some initial uncertainty in the recognition of a new tonal context in the piece. However, exploiting the fact that the recognition of the chords is quite correct, the system executes some rules that do not match with the misleading tonal context, and allow a clear recognition of the right tonality to be established;
- In these four measures of the piece we find some *appoggiatura*: the presence of this ornaments confuses TCAD which gives a recognition of the chords pattern not correct. Again, the application of symbolic agents dedicated to solve these particular cases helps in finding the correct interpretation;
- In certain situations only the knowledge of some music rules can help to solve a possible ambiguity. In our hybrid system it is straightforward to implement these rules as symbolic agents operating on (part of) the knowledge base, and to invoke them when a doubtful situation is met.

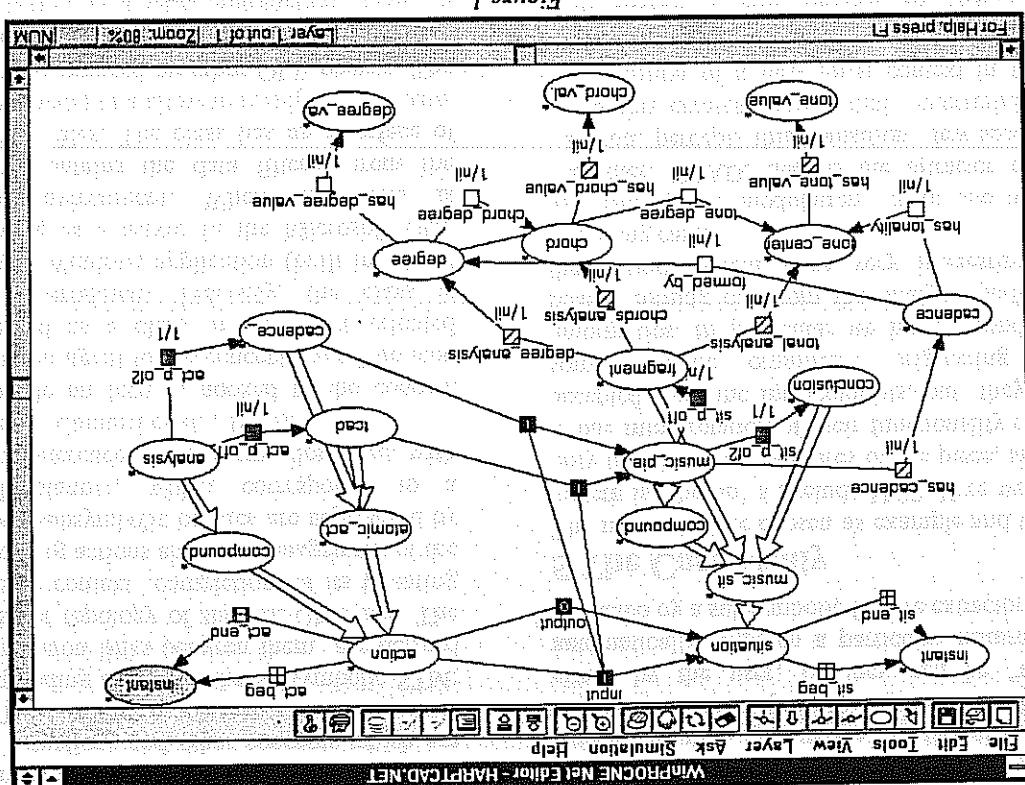
References

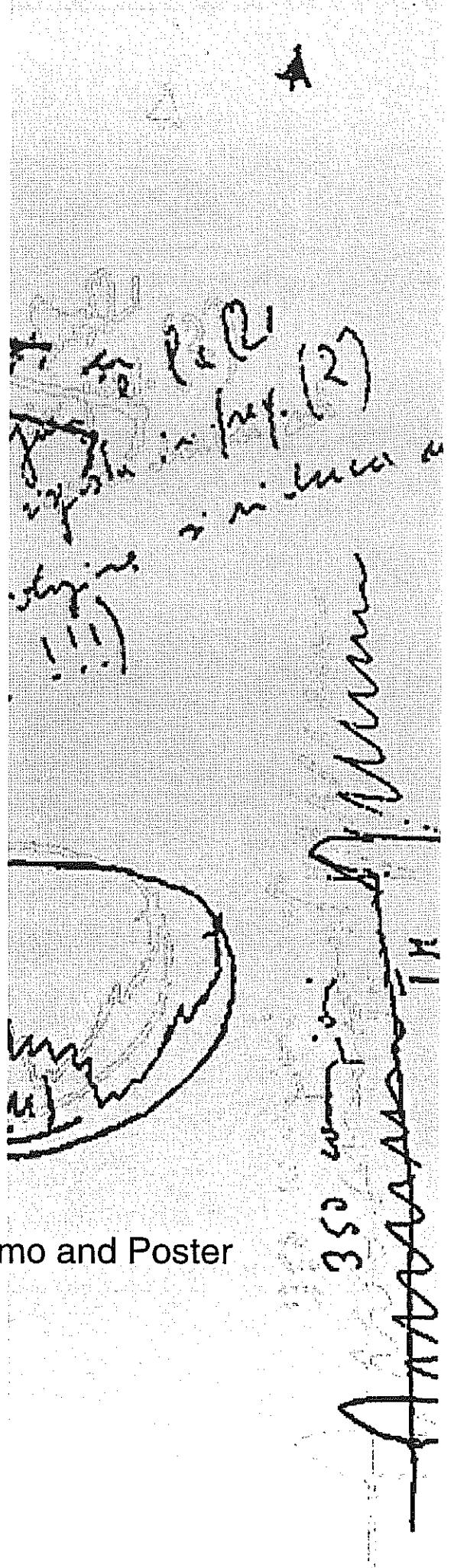
- G.Allasia, Un Sistema Ibrido Simbolico Subsimbolico per l'Analisi di Segnali Musicali, Tesi di Laurea, Università di Genova, Oct. 1994.
- A.Camurri, A.Catorcini, C.Innocenti, A.Massari, "Music and Multimedia Knowledge Representation and Reasoning: the HARP System", *Computer Music Journal*, MIT Press, 1995.
- A.Camurri, M.Leman, "Hybrid Representation of Music Knowledge - A Case Study on the Automatic Recognition of Tone Centers", Proc. *Intl. Workshop on Models and Representations of Musical Signals*, Capri, 1992, University of Napoli and AIMI.
- M.Leman, *Music and Schema Theory - Cognitive Foundations of Systematic Musicology*, Springer Series in Information Sciences, 1995.
- W.A.Woods, and J.G.Schmolze. 1992. 'The KL-ONE family.' *Computers Math. Applic.* 23(2-5):133-177.

Figure 2



181





Demo and Poster

GamePlan Project

Ambiente integrato di composizione assistita dal calcolatore e di sintesi del suono: una libreria Patchwork™ per il controllo di strumenti CSound

Giovanni Cospito, Matteo Pennese

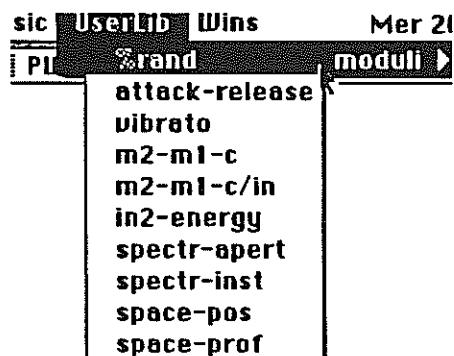
Laboratorio per la Sperimentazione e la Didattica dell'Informatica Musicale (LSDIM) c/o Sezione di Musica Contemporanea della Civica Scuola di musica di Milano
Corso di Porta Vigentina, 15 20122 Milano - Italy
tel.: +39 2 58314433 fax: +39 2 58303122

Abstract

The developed project is based on three guiding ideas: 1. the composer must be able to fix his synthesis model or sound treatment (sound composition) 2. the composer must be able to fix his control parameters at the abstraction level and with the symbolic form regarded as most suitable to develop his musical ideas 3. the composer must be able to operate in the same context of computer-assisted composition for acoustic instruments, synthesis and sound treatment. A Patchwork library has been programmed (Lisp language) permitting, by the Patchwork graphic interface, to control and to compile automatically CSound scores.

In Patchwork™ (ambiente grafico programmabile in linguaggio Lisp per la composizione assistita dal calcolatore, sviluppato presso l'IRCAM di Parigi) è stata programmata una cosiddetta "libreria" che permette, tramite interfaccia grafica, il controllo e la compilazione automatica di partiture CSound.

fig.1, libreria Patchwork:



Affrontandolo da un punto di vista generale, il progetto sviluppato si basa su tre idee guida:

1. il compositore deve poter fissare il suo modello di sintesi o di trattamento del suono (composizione del suono). Lo sviluppo di un algoritmo di sintesi fornisce solo un controllo procedurale sulla produzione dei suoni, che è comunque una parte fondamentale nel processo compositivo. Invece, il risultato musicale percepito è determinato da un insieme complesso di fattori che, pur se sistematicamente indagabile, va comunque valutato sperimentalmente e successivamente, in qualche modo, formalizzato.

2. sperimentalmente, il compositore deve poter fissare i suoi parametri di controllo al livello di astrazione e nella forma simbolica ritenute più efficaci allo sviluppo delle sue idee musicali; l'interfaccia per l'utente, dunque, deve, da un lato, permettere un utilizzo intuitivo ed empirico ma, d'altro lato, permettere anche un controllo raffinato dei singoli parametri. A questo livello viene fissato l'insieme dei parametri usando un qualsiasi insieme simbolico ritenuto il più opportuno, dove ogni

parametro è, a sua volta, un insieme di altri parametri, di funzioni e leggi di relazione.

3. il compositore deve poter operare in un ambiente integrato di aiuto alla composizione dove gli è possibile manipolare nello stesso tempo e in un'unica partitura i parametri tradizionali per la composizione con strumenti acustici e quelli per la sintesi ed il trattamento del suono.

Nell'ambiente sviluppato, è stato elaborato in CSound un algoritmo di sintesi partendo dai modelli proposti da Schottstaedt e Morrill, con un insieme complesso di parametri per il controllo numerico della sintesi. Successivamente si è fissato sperimentalmente l'insieme dei parametri musicali, le loro funzioni e le leggi di dipendenza e relazione. Questo insieme, che è un insieme di nomi a cui viene associato un opportuno "range" di variabilità, è l'insieme che va ad integrarsi, in Patchwork, con i parametri tradizionali controllati nella composizione assistita dal calcolatore.

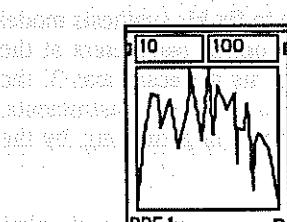
Sono stati scelti i seguenti parametri di controllo per la sintesi del suono, i quali, a loro volta, si sono trasformati in moduli Patchwork programmati in Lisp:

- inviluppo d'ampiezza
- mobilità
- vibrato
- energia spettrale
- apertura spettrale
- instabilità spettrale
- armonicità/inarmonicità spettrale
- posizione del suono nello spazio

Tali parametri possono variare con continuità nel tempo e vanno ad integrarsi con gli altri moduli Patchwork per la definizione delle altezze, del ritmo, delle dinamiche e di quant'altro sia di necessità per il singolo utente. In tale maniera, dunque, il compositore agisce ed interagisce fra le funzioni di composizione assistita dal calcolatore (che potremmo anche chiamare "attività di scrittura della partitura tradizionale") e la composizione del suono tramite sintesi, senza soluzione di continuità d'ambiente di lavoro. I moduli permettono la scelta di due tipologie di dati come "input", in relazione alle esigenze dell'utente:

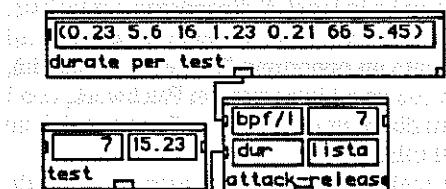
1. funzione per segmenti (Break Point Function), ovvero funzione a linee continue; già implementata in Patchwork, permette la visualizzazione grafica dei dati tramite curva nonché, nel caso specifico, un controllo intuitivo dei parametri. L'utente, ad esempio, potrà disegnare empiricamente la curva che andrà a fornire i dati dell'evoluzione dell'inviluppo d'ampiezza.

fig.2, i dati per l'inviluppo d'ampiezza vengono estratti dal profilo della BPF:



2. lista/liste di valori; la lista, tipo di dato prevalente in Patchwork, permette altresì il collegamento dei moduli di controllo della sintesi ad altri moduli preesistenti.

fig.3, i dati per l'inviluppo d'ampiezza vengono definiti con precisione dall'utente tramite una lista contenente i dati medesimi:



A questo punto si allarga ancora l'ambito di applicazione del Patchwork System al live control.

Una futura estensione di questo ambiente prevede la connessione con sistemi in tempo reale (Mars™ dell'Iris) ed in generale con processori per il "live electronics".

Bibliografia

- J. Grey, "Multidimensional Perceptual Scaling of Musical Timbre", *Journal of the Acoustical Society of America*, Vol. 61, N. 5, 1977.
- D. Morrill, "Trumpet Algorithms for Computer Composition", *Computer Music Journal*, Vol. 1, N. 1, 1977.
- B. Schottstaedt, "The Simulation of Natural Instrument Tones Using Frequency Modulation with a Complex Modulating Wave", *Computer Music Journal*, Vol. 1, N. 4, 1977.
- D. Wessel, "Timbre Space as a Musical Control Structure", *Computer Music Journal*, Vol. 3, N. 2, 1979.
- G. DePolis, "A Tutorial on Digital Sound Synthesis Techniques", *Computer Music Journal*, Vol. 7, N. 4, 1983.
- W. J. Slatson, "Sound-Color Dynamics", *Perspectives of New Music*, Vol. 25, N. 1 & 2, 1987.
- B. Vercoe, *Csound User Manual*, Massachusetts: MIT Media Lab, Cambridge, 1990.
- G. Steele, *Common Lisp: The Language*, Bedford: Digital Press, 1990.
- AA.VV.: *La composition assistée par ordinateur*, Les cahiers de l'IRCAM, Paris, 1993.
- AA.VV.: Patchwork documentation, "Reference", "Programming guide", IRCAM, Paris 1994.
- R. Ethington, B. Punch: "SeaWave: A System for Musical Timbre Description", *Computer Music Journal*, Vol. 18, N. 1, 1994.

Driving a Musical Sound Synthesis System with a Digital Signal Processor : A Design Exercise

P.J. Comerford and L.K. Comerford

Department of Computing, University of Bradford
Bradford, England

Abstract

This paper explores the potential of using Digital Signal Processors to replace custom hardware in a musical sound synthesis system. The system is described and two implementation options which make use of DSPs and a third option which uses only custom hardware are examined.

1 Introduction

Real time musical sound synthesis with many independent sound sources is a computationally intensive operation. Traditionally most practical digital synthesisers have been built with microprocessor-driven custom hardware, but recent developments in high speed digital signal processors, together with considerable reductions in their price per MHz, have made them look like an attractive alternative to part or all of the custom hardware [Wilkinson 1995].

This paper is concerned with exploring the potential of using a digital signal processor (DSP) for driving a musical sound synthesis system which employs a modified form of the additive synthesis algorithm [Comerford, 1992, 1993] [Eaglestone and Oates, 1990].

2 The Musical Sound Synthesis System

The system was originally designed in the early nineteen eighties and has been recently updated. It has enjoyed considerable commercial success, notably as a classical organ simulator. It is constructed on a modular basis with each sound generating module (music module) able to support 64 independent musical note generators. The note generators are implemented in custom hardware which is microprocessor-driven. Each note generator can look up 2 single or multiple cycle waveforms stored in common waveform memory and each waveform has its own independent amplitude envelope, which makes spectral interpolation with a single note generator possible. Amplitude values for each note generator can be updated under hardware control every sample look-up period.

Pitch for each note generator is determined by the magnitude of a phase-angle increment which is added to the note generator's current phase angle value every time a sample is looked up. Phase angle increments have fractional parts so that accurate pitch resolution can be obtained. Resulting sample magnitude errors, which occur when phase angle values fall between stored samples, are reduced by linear interpolation between adjacent samples.

Note generator outputs can be distributed as required between up to 8 audio output channels per music module. Channel outputs are normalised to 16 bit resolution with consequent gain adjustments being made by a digital gain controller following the D to A converters.

The common waveform memory may be RAM or ROM. If it is RAM it is loaded with waveforms generated in a waveform synthesiser as organ stops or instruments are selected. There is a waveform synthesiser, implemented in custom hardware, on each music module.

All of the above operations are carried out in real time inside each music module of the synthesis system. To perform them while maintaining a sample output rate for each channel of up to 45KHz, the music module needs fast access internal memory for note generator control data, plus facilities for fast multiplication, fast addition and shifting, fast multiplexing and decoding, fast control action choice, and fast bit manipulation.

3 Replacing Custom Hardware with DSPs

All DSPs can provide fast access internal memory and facilities for fast multiplication and addition, and many have fast internal barrel shifters. A limited amount of internal parallel processing is usually available too. Operations requiring multiplexing, decoding, control action choice or bit manipulation are, however, more difficult. For example, an operation in the synthesis system such as the computation of multipliers for sample interpolation arithmetic requires, because of the complex way in which the waveform memory is partitioned, bit extraction, decoding, shifting, two's complementing and multiplexing. All these operations can be implemented in a compact combinational logic circuit in 25 nanoseconds or less. A DSP, because of the sequential nature of its operation, would require at least 10 instruction times of 25 to 50 ns each for the same task.

The architecture of the synthesis system makes extensive use of parallel operations, mainly in the form of pipelining. This arrangement cannot be emulated inside a DSP with the limited amount of

parallel processing at present incorporated in DSP architectures.

4 A Mostly DSP Solution

An analysis of the situation in 1992, when the recent update was being planned, made it clear that, if all the features of the synthesis system were to be retained, the best that could be achieved with a state of the art 33MHz Motorola 56001 DSP or 33MHz Texas 320C31 DSP was 6 generators per DSP. The equivalent of a music module would require 11 DSPs, plus additional logic to interface them to a common waveform synthesiser and to the D to A converters and controlling microprocessors. This solution was too expensive and, in spite of the reductions in cost per MHz of DSPs since 1992, remains so. The alternative of organising DSPs in a pipeline was also examined, but offered no improvement.

5 A Hybrid Solution

An alternative approach, which overcomes many of the problems of the mostly DSP solution, transfers tasks more suitable for combinational logic circuitry to field programmable gate arrays (FPGAs) external to the DSP. With this solution it is also possible to organise a pipeline arrangement with the DSP and the FPGAs performing tasks in parallel, and the use of custom logic to interface the two. A design study, again carried out in 1992, found that the most efficient way to divide the tasks was to allocate note generator control data storage and look-up, pitch arithmetic and amplitude envelope multiplication to the DSP and all the other tasks to the FPGAs. Using a 33MHz Motorola or Texas DSP, 32 note generators could be supported with this arrangement at a sample output rate for each output channel of 37.2KHz. The DSP instruction loop, traversed once for each note generator, is 13 two clock period instructions long. To reduce the use of jump instructions, code for 32 loops is written as a continuous block with a single jump at the end. Within each block there are 2 slots for the controlling microprocessors to access the DSP's RAM. The DSP is supported by 4 Actel FPGAs and a 2k x 8 fast access RAM.

For 64 note generators, two 32 bit generator units are interfaced to the common controlling microprocessors, to a common waveform memory and to a common waveform synthesiser, which is implemented in custom logic. An additional FPGA is needed for these interfaces and 2 others for interfacing the units to the serial 16 bit D to A converters in the output channels.

6 A Full Custom Solution

With this solution, which makes no use of DSPs, there is no problem in supporting 64 note generators in a single unit at a sample output rate of

40 to 45KHz, and the unit needs only two more FPGAs than a 64 generator hybrid solution. It needs more fast access RAMs, but overall it was in 1992 significantly cheaper, only a little less compact and the inconvenience of splitting the note generator complement into 2 blocks of 32 is avoided. There is no loss in flexibility involved in abandoning the DSPs.

Accordingly the full custom solution was chosen for the recent update and full custom modules have been in commercial use for over a year, during which time they have performed satisfactorily and reliably.

7 1995 Update

During 1995, the hybrid solution was reviewed in the light of the availability of faster and more powerful DSPs at a lower price per MHz and faster and more powerful FPGAs at a lower price per gate. It is now possible to support 64 note generators with a single DSP, but the availability of the improved FPGAs weighed in the price/performance balance again in favour of the full custom solution, but this time utilising a DSP-like architecture.

References

- P.J. Comerford: "Playing with Numbers" - parts 1 and 2, *Hong Kong Engineer*, Vol. 20, Nos. 8 and 9, 1992, pp 40-44 (No. 8) and pp 42-44 (No. 9).
- P.J. Comerford: "Simulating an Organ with Additive Synthesis", *Computer Music Journal*, Vol. 17, No. 2, 1993, pp 55-65.
- R.A. Wilkinson: *Architectures for Additive Synthesis*, University of Bradford Ph.D. Thesis, 1995, Chapters 4, 6 and 7 (pp 43-74, 91-140)
- B.M. Eaglestone, and S. Oates: "Analytical Tools for Group Additive Synthesis", *Proceedings, International Computer Music Conference*, Glasgow, 1990, pp 270-273

P.J. Comerford and L.K. Comerford, Microcomputer Music Research Unit, Department of Computing, Phoenix Building, University of Bradford, BRADFORD, West Yorkshire, BD7 1DP, U.K. Email: p.j.comerford@comp.brad.ac.uk, l.k.comerford@comp.brad.ac.uk

A Multimedia HyperText Environment in Computer Music

A. Katrami, T. Ingall, and L. Comerford

Electronic Imaging and Media Communications Unit, and Department of Computing, University of Bradford
Bradford, England¹

Abstract

This paper describes the concept and implementation of a package of tutorials introducing a range of major topics in Computer Music, realised in an interactive multimedia environment for self-paced learning. Based on the Authorware Professional authoring tool on the Macintosh, these tutorials have proved to be successful as an integral part of course modules in an educational setting, and are of wider application to introduce the interested enquirer to many aspects of this broadly based field in a relevant and innovative way.

1 Introduction

Those who work in the field of Computer Music education will be aware of the need for suitable non-specialist literature to introduce the student to different aspects of this broadly-based field. This paper describes an original multimedia solution to this problem, developed at the University of Bradford - a package of computer-based tutorials named *Musical Odyssey* which offers the student an interactive environment for self-paced learning about Computer Music.

2 Implementation

Musical Odyssey has been developed using the platform of the "Authorware Professional" package. Running on the Macintosh, Authorware is an object-oriented authoring tool for creating multimedia presentations by integrating pictures, text, sound and video. No scripts or programming languages are required; instead, logic is represented visually by connections between different applications. The overall application design can be made visible through a hierarchical map structure. By using Authorware, those developing computer-based training can preview interaction between the system and the user to evaluate the response of the application.

3 Objectives

Musical Odyssey was developed for use in the Electronic Imaging and Media Communications Unit at the University of Bradford. The B.Sc. course in Electronic Imaging and Media Communications at Bradford has been designed to educate and equip professionals who can work innovatively in both the creative arts and the engineering sciences [1]. As the subject matter covered is relatively new, there is a shortage of suitable information in a readily available format for the student undertaking such a course. In order to meet the needs of students who are inexperienced in the subject, tutorials have been designed for use

as an integral part of 3 course modules (that is, Audio-Visual Systems, Computers and Music, and Sound Processing). The second of these modules is also used by students in the Department of Computing [2]. It is natural that a learning tool for use in such a setting should be multimedia in concept. As such, the aim is to provide an interactive multimedia environment for self-paced learning to initiate the novice into many of the major areas of the subject. The content and presentation of *Musical Odyssey* make it equally useful for the non-specialist with serious interest in the field.

4 Content

The tutorials in *Musical Odyssey* cover the following six major topics, which are selected by the user from the main menu:

- History and development of Computer Music
- Computer Music Software and Hardware Packages
- MIDI
- Studio Technology - Equipment, Architecture & Special Effects
- Sciences in Composition
- Different Musical Styles

From each topic on the main menu, the user can access the following categories of information:

History and development of Computer Music

- Historical review, including nineteenth century musical background
- Progression from tonality to atonality
- Summary of work of key composers

Computer Music Software and Hardware Packages

- Tools for composition
- Hard disc recording systems
- Sequencers

- Music and Fractals

MIDI

- An introduction to MIDI
- The configuration and use of MIDI, including practical examples of MIDI connections
- MIDI and Multimedia

Studio Technology

- The use of studio equipment
- Techniques of analogue and digital recording
- Creating special effects
- Studio Architecture

Sciences in Composition

- The evolution of musical science
- Synthesis techniques and synthesis algorithms
- Electroacoustic composition

Different Musical Styles

- A library of musical examples illustrating a range of different musical styles

5 Presentation

When each of the categories is accessed, the user is presented with a wide-ranging multimedia information package. This includes many musical illustrations taken from the works of a range of composers. A comprehensive explanation is offered of how different equipment works, and practical information is given on what equipment configurations are necessary to achieve particular results. There are also simple illustrations of sound manipulation and processing techniques such as filters, echo, chorus, flange and time/pitch stretching. All these examples have been created using a suitable combination of text, pictures, diagrams and voice-overs, and include sections of videos. For example, one of these video sections provides guidelines to users on the interconnection of different MIDI equipment, another gives examples of how fractal images can be used to create electroacoustic compositions, while another video section shows extracts from Jean Michel Jarre's concert in Houston to demonstrate multimedia applications.

6 Results

Musical Odyssey tutorials have been used as part of the modules of the Electronic Imaging and Media Communications B.Sc. course, and in the Computer Music option module in the Department of Computing, since June 1994. They have helped

students to grasp the basics of many different aspects within the field of Electronic Music, and have allowed them to create their own simple sound examples at a very early stage. The tutorials have proved an excellent introduction to the different equipment which they are to encounter in the music studio, so that they have been able to utilise this equipment with confidence and efficiency from the start, even though they had no previous practical experience in such a setting.

7 Future development

The content of the tutorial package was tailored specifically for the requirements of the courses at Bradford. Authorware provides a suitable environment for the efficient expansion of the tutorials' content as an increased number of relevant major topics are identified, and as it is judged useful to allow more or different information to be accessed on each topic.

At present, the EIMCU has a complementary set of tutorials based on C-Sound, which introduce students to basic techniques of sound creation including additive synthesis, sampling, AM and FM synthesis and subtractive synthesis. It would be useful to integrate the content of these C-Sound tutorials onto the Authorware platform.

Future versions of the tutorial package would benefit from the inclusion of facilities for self-assessment of progress and learning. Further topics which might be useful additions to Musical Odyssey include:

- the physics of musical sound,
- the human auditory system,
- psychoacoustics and music perception, and
- environmental acoustics

References

- [1] A. Katrami, T. Ingall, O. Nash and M. Chester: Computer Music Education at Bradford University, *Proceedings, Computers in Art and Design Education*, Brighton 1995 pp289
- [2] A. Katrami, P. Comerford and B. Eaglestone: Computer Music at the University of Bradford, *Proceedings, International Computer Music Conference*, Banff 1995

¹ Anna Katrami (EIMC Unit), and Lucy Comerford (Computing Department), University of Bradford, Richmond Road, Bradford, West Yorkshire, BD7-1DP, U.K. E-mail: A.Katrami@brad.ac.uk, L.K.Comerford@comp.brad.ac.uk

Musical Synthesis of DNA Sequences

Peter Gena, Ph.D., Charles Strom, M.D., Ph.D.

School of the Art Institute of Chicago, Illinois Masonic Medical Center
Chicago, Illinois USA

Abstract

As a consequence of the Human Genome Project, there has been an explosion of primary DNA sequencing data available on CD ROM. This includes complete genomes of viruses, partial genomes of bacteria, and complete sequences for hundreds of human proteins. Consequently, we began to envision a type of computer-generated music that would take cues for its musical parameters directly from the physiological ones present in DNA. A DNA sequence consists of a specified order for the production of amino acids. The physical properties of amino acids (dissociation constant, molecular weight, and chemical class) combined with the properties of the individual bases (melting temperatures) provide the basis for inheritance and evolution and our musical compositions. The converted results, one for each codon, represent distinct musical actions in MIDI note events. Thus far, we have generated musical compositions from several human, viral, and bacterial sequences. This paper outlines our research.

1 Introduction: DNA

The genetic code is an alphabet made up of four chemical compounds which form the nucleotide bases—adenine (A), cytosine (C), guanine (G), and thymine (T). These bases are linked in a specific order to form the double helical structure known as deoxyribonucleic acid, or DNA. Each individual living organism has a unique order of bases that completely determines its physical structure. The four nucleotides are arranged in three-letter units known as codons. Each codon specifies one of nineteen amino acids. When they are grouped by chemical type, there are eight such categories. The DNA template, located in the nucleus of each cell, acts as a blueprint that directs the production of proteins. DNA is translated into messenger ribonucleic acid, or mRNA that is in turn serially scanned by ribosomes, organelles located in the cell's cytoplasm. Ribosomes use the mRNA as a template to direct the synthesis of proteins.

2 Physio-musical conversion

The initial programming task was to write an algorithm that converts the list of sixty-four codons into distinct musical events according to physical properties. A look-up table of the codons and their corresponding amino acid types, followed by the dissociation constant or pK(a) and molecular weight, was constructed as a data-base. There are eight basic

musical timbres; one for each of the eight classes of amino acids. Each of the nineteen amino acids has a distinct pK(a) that helps define pitch. Additional modifications involve physical properties of the molecular bonding occurring in the codon itself, independent of what amino acid it codes for. Using 7.0 as the neutral point in acid/base equilibrium point, pK(a)'s below 7.0 are acidic while those above are basic. Hence, there are two equations for each codon: one correlates higher pitch with acidity, the other with base. The algorithm makes a binary choice with each selection.

2.1 Pitch

$$f = ((p(4G + 2T) + 12) + k$$

$$f1 = (((p - 7.0))(4G + 2T) + 12) + k$$

where:

f & $f1$ = MIDI pitches

p = pK(a)

G = $\sum G + C$ per codon

T = $\sum A + T$ per codon

k = Constant (Hydrogen Bonds): [AA = -2, TT = -1, CC = +1, CG = +2, GC = +3, GG = +4].

Additional pitch-bend commands for each note place the music in just intonation.

2.2 Intensity

Intensities (velocity) are also adjusted according to the hydrogen bonding occurring in each codon. As with pitch, there are two corresponding equations for each codon and a binary choice is made with each selection.

$$I = 6H$$

$$II = 109 - I$$

where:

I and II are MIDI velocity levels

H is proportional to codon melting temperature and

Hydrogen-bond strength per codon (each G = +8, C =

$$= +6, A = +4, T = +1).$$

2.3 Duration

The pK(a) and atomic weights of the amino acids

determine durations.

$$D = 0.01pM + 0.1Sk$$

where:

D = duration in clock ticks

p = pK(a)

M = molecular weight of amino acid

S = f (sum of hydrogen bonds per codon)

k = tempo constant (>0), higher number = slower tempo.

3.0 Programming and realization

All of the preliminary programming is scripted in Hypercard. The scripts prepare all the necessary data, that is, the table of codons, and the genomes as collections for the MAX object code language (Copyright by IRCAM and Opcode Systems). The initial table data contains the index number followed by the codon, amino acid, pK(a), amino acid class number, and the molecular weight of the amino acid.

Codons 11 - 16 from the table:

- 11, TGT, CYS, 1.900, 6, 121;
- 12, CCG, PRO, 1.952, 7, 155;
- 13, CCC, PRO, 1.952, 7, 155;
- 14, CCT, PRO, 1.952, 7, 155;
- 15, CCA, PRO, 1.952, 7, 155;
- 16, ACG, THR, 2.088, 1, 120;

Each codon is transformed into a list in a collection. The list specifies the address, MIDI pitch, velocity, channel number, pitch bend, and duration of the event for the corresponding codon.

MAX collection:

- 11, 26 6 24 23
- 12, 38 120 7 1 31
- 13, 35 108 7 19 30
- 14, 32 78 7 26 30

15, 32 96 7 26 30
16, 34 108 1 28 25

A second series of algorithms reads the raw DNA strings for a genome, searches for the start and stop codons, and then forms the three-letter codon sequences. Uncoded filler, ubiquitous extraneous material bearing no significance to amino acid production, is ignored. In addition, each codon from the genome is checked in the look-up table and its index number is put into another collection.

Beta Globin sequence:

ATG ATG ATG GTG CAC CTG ACT CCT GAG
GAG AAG TCT GCC GTT ACT GCC CTG TGG
GGC AAG GTG AAC GTG GAT GAA . . . etc.

Corresponding MAX collection:

1,32;2,32;3,32;4,33;5,56;6,42;7,17;8,14;9,23;10,23;
11,64;12,24;13,46;14,36;15,17;16,46;17,42;18,54;1
9,52;20,64;21,33;22,30;23,33;24,20; . . . etc.

Actual genomes of human or bacterial proteins, or complete viruses can then be scanned by a MAX patch so that each of the codons is culled from the data-base table and then played in real-time linear sequence as MIDI events. This process is analogous to the scanning of the mRNA by the ribosomes as it adds amino acids sequentially to make proteins—a process not unlike several cars (ribosomes) on a roller coaster negotiating the identical track (mRNA), but at different locations, speeds, and spacings. Polyphonic voices can occur just as multiple ribosomes run along a single strand of mRNA. At this point in our work, the computer performs the music on a Yamaha TX802 digital synthesizer according to a duration constant (the greater the constant, the longer each relative MIDI event).

4 Future work

Thus far, we have generated musical compositions for blood and liver cells, the polio virus, botulinin toxin (botulism), measles, rubella, four distinct common cold viruses, and the HIV virus (we have presently avoided most human proteins because of large amounts of uncoded filler found in between sequences). The next major goal is to realize the Smallpox (Variola) Virus (now extinct save for two vials in Atlanta and Moscow respectively). Because of its many distinct sequences and extreme length (20,000 base pairs), the MAX patch presently being used will require some modifications. Future plans also include the investigation of replacing MIDI events with real-time synthesis programming.

PROMENADE

Un Sistema Software per la Sincronizzazione di Partiture e Processi MIDI

Agostino Inzoli, Isabella Pighi

L.I.M. - Laboratorio di Informatica Musicale
Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
via Comelico 39
I-20135 Milano (Italia)
fax +39 2 55006373
e-mail: music@imiucca.csi.unimi.it

Abstract

This project, called PROMENADE, has been developed to carry out a Macintosh prototypal system for synchronization between MIDI musical processes and multimedia objects. The original idea was to create an user-friendly environment which allows to perform multimedia objects in temporal instants previously selected. User can manage follow kinds of multimedia objects: PostScript score pages; AIFF audio files; PICT image files; QuickTime movies. PROMENADE consist of two software applications: EDITOR and VIEWER. The first one allows the user to have an editing session based on traditional staff notation to plan synchronized performance; the second one, perform it during a MIDI playing session. Generally, PROMENADE could realize synchronization between MIDI song and any kind of MIDI device (hardware and software), which must be able to receive and recognize MIDI messages.

1. Introduzione

Il presente progetto, denominato PROMENADE e sviluppato presso il Laboratorio di Informatica Musicale del Dipartimento di Scienze dell'Informazione dell'Università degli Studi di Milano, è stato rivolto alla realizzazione di un sistema prototipale per ambiente Macintosh che consenta la sincronizzazione tra processi musicali MIDI e processi multimediali. Lo scopo del progetto è stato quello di creare un ambiente di semplice utilizzo, capace di gestire immagini video, suoni, grafica e qualsiasi altro strumento (oggetto multimediale) che rappresenti un canale comunicativo, il tutto sincronizzato rispetto ad una partitura MIDI (Standard MIDI File) [1] [2].

L'utente sarà quindi in grado di realizzare una performance che integri vari tipi di media, associando a diversi istanti temporali della partitura MIDI degli oggetti multimediali che verranno attivati solamente quando l'esecuzione della base musicale sarà giunta nei punti di sincronismo stabiliti.

Con il termine oggetto multimediale intendiamo definire con un unico termine alcuni dei processi audio e video più comuni come immagini grafiche in formato standard PICT, singole pagine di partitura in formato PostScript, file audio in formato AIFF (Audio Interchange File Format) e

QuickTime movie, anche se in realtà questa definizione può essere estesa a qualsiasi altro tipo di oggetto. Quindi per attivazione di un oggetto multimediale intenderemo d'ora in poi riferirci alla visualizzazione sullo schermo di tale oggetto (nel caso di PICT, file PostScript e tracce video di movie) oppure alla sua riproduzione sonora (per quanto riguarda i file audio AIFF e le tracce audio o MIDI di movie).

2. Soluzioni innovative e possibili applicazioni

Un'attenta analisi di ciò che oggi offre il mercato del software multimediale in campo musicale ha ispirato alcune idee innovative che successivamente sono state implementate durante la fase realizzativa di PROMENADE. Molti tra i sistemi commerciali visionati, seppure molto potenti, sono molto ingombranti a livello di occupazione di memoria, non sono semplici da utilizzare e, nonostante consentano l'import di diversi tipi di oggetti multimediali, non permettono operazioni di sincronizzazione che abbiano come riferimento una partitura MIDI. Sistemi di sviluppo, come ad esempio Macromind Director 3.1 [3], consentono la sincronizzazione tra oggetti multimediali e Standard MIDI File, ma non permettono all'utente di decidere gli istanti temporali per il sincronismo, basandosi su una partitura tradotta nel formato tradizionale a

pentagrammi; mentre esistono altri tipi di ambienti che riguardano più specificatamente il campo musicale e che quindi permettono una semplice scelta degli istanti temporali di attivazione degli oggetti multimediali considerando direttamente lo spartito corrispondente al file musicale MIDI, come accade, ad esempio, utilizzando Cubase Score 2.0 [4]. Purtroppo questi ambienti non si adattano ai nostri scopi, essendo molto complessi e non consentendo la sincronizzazione di diversi tipi di media.

In qualità di ambiente per la realizzazione di performance multimediali, PROMENADE non è molto potente poiché non si propone a livello di sistema autore, ma tuttavia consente, sfruttando alcune applicazioni commerciali e compatibilmente con il tipo di macchina utilizzata, l'adattamento a diverse esigenze che riguardano, nello specifico, il campo musicale. Lo scopo principale non è dunque stato quello di realizzare un ambiente per la definizione di performance multimediali, in quanto abbiamo già verificato l'esistenza di strumenti avanzati che ottemperano in maniera esaurente a questa necessità, ma bensì ha riguardato, in modo più preciso, l'integrazione di vari tipi di media rispetto all'esecuzione di una partitura musicale MIDI.

Quindi PROMENADE realizza un ambiente musicale innovativo in cui l'utente, che tipicamente è un musicista e non un informatico, è in grado di decidere facilmente i tempi di sincronizzazione indicando i punti sulla partitura in notazione tradizionale e associandovi oggetti multimediali di qualunque tipo (immagini, grafiche, pagine di partitura, file audio e movie) senza dover utilizzare sistemi complessi che prevedano la conoscenza di codici speciali per la sincronizzazione audio-video come lo standard SMPTE (Society of Motion Picture and Television).

Inoltre con PROMENADE, potendo importare oggetti del tipo pagine di partitura create con altri ambienti per l'editoria musicale, è possibile realizzare un volta-pagina automatico durante una performance musicale live, che preveda l'utilizzo di altre parti strumentali sintetizzate da un sistema basato su di un sequencer MIDI. Il musicista potrà in questo modo utilizzare PROMENADE come una specie di "leggio automatico", ovvero come uno strumento che gli consenta di sincronizzare, rispetto alla base musicale MIDI, i tempi di "voltata" e di posizionamento delle pagine di spartito della parte strumentale che dovrà eseguire.

3. Architettura software e specifiche funzionali di PROMENADE.

Il sistema PROMENADE è composto da due ambienti principali ben distinti, EDITOR e

VIEWER, che consentono all'utente di compiere tutte le operazioni necessarie per la realizzazione di una performance multimediale in modo molto semplice e sistematico.

La scelta di dividere il sistema nelle due applicazioni citate ha reso l'intero ambiente più semplice dal punto di vista dell'utilizzo e meno "ingombrante" per quanto riguarda l'occupazione della memoria: infatti risulterebbe inutile e dispendioso, in termini di allocazione in RAM, avere una singola grossa applicazione che esegua tutti i compiti affidati ai due ambienti, dato che la parte di modifica dello Standard MIDI File realizzata utilizzando EDITOR, non riguarda direttamente la performance multimediale, di cui invece si occupa esclusivamente e in un secondo momento l'applicazione VIEWER.

Mediante l'utilizzo dell'applicazione EDITOR, l'utente è in grado di stabilire facilmente in quali istanti temporali della partitura MIDI attivare gli oggetti multimediali selezionati.

L'applicazione EDITOR è a sua volta costituita da due parti principali, ciascuna riguardante una precisa fase nel processo di definizione di una performance.

La prima parte di EDITOR si occupa della visualizzazione e della fase di editing del file contenente la partitura MIDI tradotta nella notazione grafica del pentagramma. L'utente è in grado di selezionare graficamente gli istanti temporali nella partitura che corrispondono ai punti di inizio-fine battuta (fig. 1) e associare ad essi gli oggetti corrispondenti che si desidera vengano attivati nel corso della performance.

La seconda parte di EDITOR riguarda la modifica dello Standard MIDI File, ovvero si occupa dell'inserimento degli opportuni messaggi di sincronismo negli istanti temporali indicati dall'utente durante la precedente fase di editing.

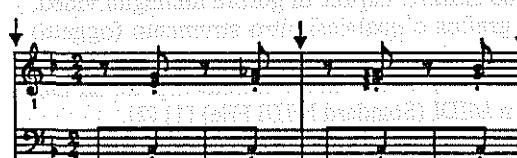


fig.1 Istanti temporali della partitura associabili all'attivazione di un oggetto multimediale.

L'applicazione VIEWER è invece interamente dedicata alla gestione vera e propria del processo di sincronizzazione tra gli oggetti multimediali assegnati e la performance musicale MIDI. Tale ambiente realizza l'interfacciamento di tutto il sistema PROMENADE con il protocollo di comunicazione MIDI, permettendo così l'acquisizione e la verifica dei dati provenienti da un sequencer, cioè da un'applicazione in grado di riprodurre partiture MIDI, e occupandosi inoltre

dell'attivazione degli oggetti multimediali relativi nel corso della performance musicale.

Quindi, l'ambiente PROMENADE, deve essere necessariamente integrato con altre opportune applicazioni commerciali, allo scopo di fornire un sistema completo per la realizzazione di performance multimediali sincrone rispetto ad un brano musicale MIDI (fig. 2).

L'utente, utilizzando un ambiente integrato per l'editoria musicale (Score Editor) che sia in grado di memorizzare le proprie sessioni di lavoro nel formato PostScript, può ricavare da un qualsiasi Standard MIDI File (SMF) la rispettiva partitura grafica, oppure viceversa, può scegliere di realizzare una propria partitura ottenendone il corrispondente

file MIDI. La partitura, che come già detto, verrà a sua volta memorizzata su file nel formato di descrizione grafica PostScript (PostScript Score), andrà quindi caricata nell'EDITOR allo scopo di decidere in quali istanti temporali, durante la performance, dovranno essere attivati gli oggetti multimediali stabiliti; al termine dell'operazione di editing verrà creato uno Standard MIDI File (SMF*) che, rispetto alla versione originale, conterrà in più dei messaggi chiave negli istanti temporali selezionati dall'utente [5] (fig. 3). Verrà inoltre fornito un elenco sequenziale di tutti gli oggetti coinvolti nella performance (Performance Objects List) in modo da avere i corretti riferimenti tra i messaggi inseriti nella partitura MIDI e gli oggetti multimediali corrispondenti.

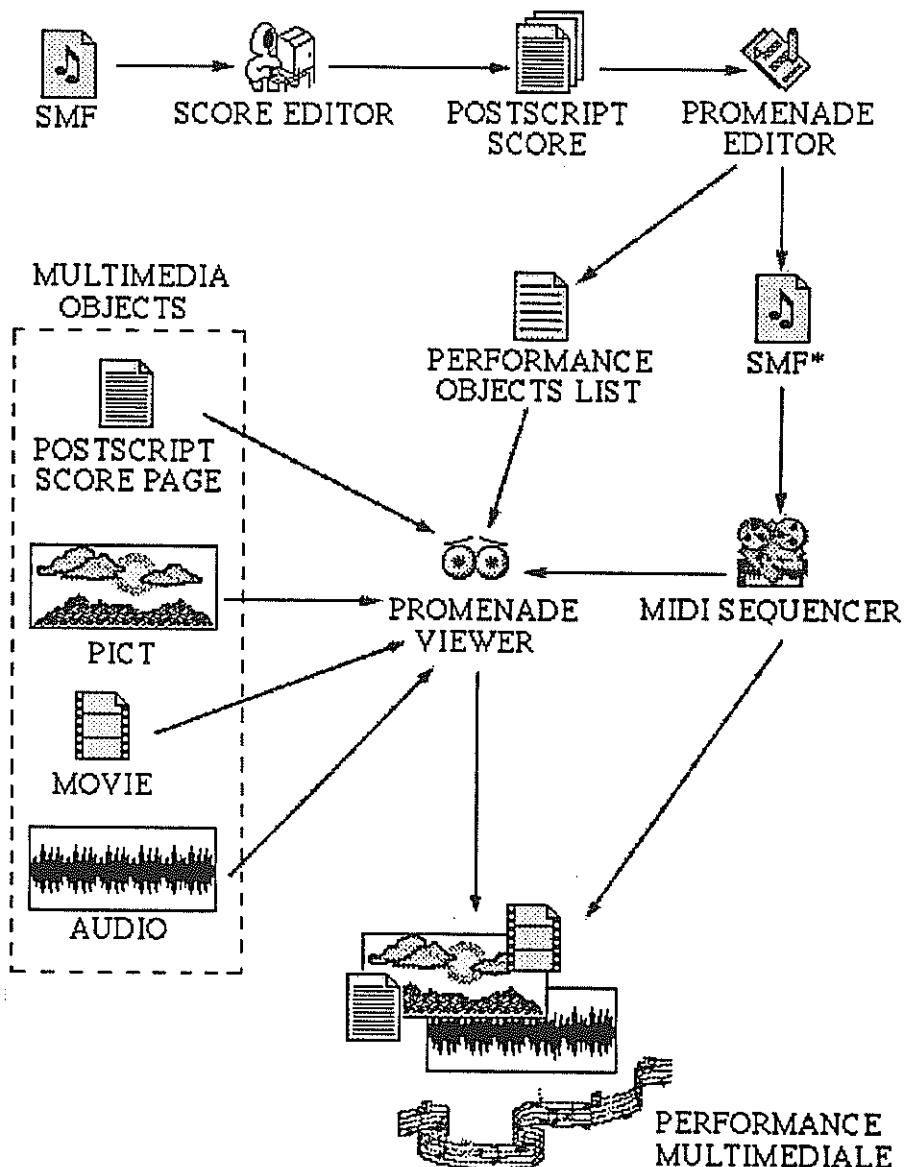


fig.2 Ciclo di funzionamento di PROMENADE.

A questo punto, utilizzando un qualsiasi sequencer MIDI, sarà possibile riprodurre lo Standard MIDI File modificato e, allo stesso tempo, i dati in uscita dal sequencer andranno indirizzati verso l'applicazione VIEWER, la quale provvederà, a sua volta, al riconoscimento dei messaggi di sincronizzazione inseriti nel file durante la fase di editing e inoltre si occuperà dell'attivazione degli oggetti multimediali corrispondenti, consultando la Performance Object List relativa alla performance in questione.

Standard MIDI File modificato (SMF*)

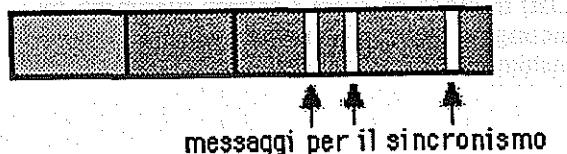


fig. 3 Inserimento dei messaggi di sincronismo nella partitura MIDI

Una volta realizzate le connessioni illustrate tra le porte logiche MIDI dei vari ambienti del sistema, tramite l'applicazione PatchBay del pacchetto MIDI Management Tools [6], sarà possibile dare inizio alla performance multimediale semplicemente avviando, tramite il sequencer, la riproduzione del file musicale MIDI contenente i messaggi di sincronismo.

4. Sviluppi futuri

Numerose sono le idee che sono nate durante lo sviluppo del progetto PROMENADE, legate in modo particolare al campo del multimediale.

In primo luogo, sicuramente, citiamo la possibilità di allargare l'insieme degli oggetti multimediali che possono essere trattati dall'ambiente. Sarebbe interessante poter sincronizzare ad una base musicale MIDI, oltre che oggetti di tipo software, anche apparecchiature hardware esterne, come centraline per il comando luci e, più in generale, effetti speciali riguardanti gli spettacoli dal vivo, utilizzando per tale scopo comandi basati sull'estensione MIDI Show Control del protocollo MIDI [7].

Inoltre, un'estensione prossima del sistema, può riguardare il potenziamento dell'ambiente per quanto concerne la sincronizzazione di oggetti multimediali di tipo dinamico (movie o file audio), in modo da

potere realizzare un processo di sincronizzazione che non abbia come unico punto di riferimento l'istante di attivazione dell'oggetto, ma che tenga conto anche degli istanti intermedi riguardanti l'evoluzione temporale che l'oggetto multimediale subisce durante la sua riproduzione.

In generale, PROMENADE potrebbe essere utilizzato per realizzare un qualsiasi sistema di sincronizzazione che riguardi partiture musicali MIDI e dispositivi hardware o software, i quali, a loro volta, devono essere in grado di ricevere e riconoscere messaggi appartenenti al protocollo di comunicazione MIDI.

Ringraziamenti

Si desidera ringraziare Goffredo Haus, direttore scientifico del Laboratorio di Informatica Musicale del Dipartimento di Scienze dell'Informazione di Milano, per avere ideato e supervisionato il progetto.

Bibliografia

- [1] MIDI Manufacturers Association: "MIDI 1.0 Detailed Specification", The International MIDI Association, Los Angeles, California, 1983.
- [2] International MIDI Association: "Standard MIDI Files 1.0", The International MIDI Association, Los Angeles, California, 1988.
- [3] G. Papetti: "Macromind Director 3.1", Bit, rivista n.149, pp. 110-115, Febbraio 1993.
- [4] A. Massazza: "Cubase Score 2.0 per Macintosh", Midi Songs, rivista n.25, pp. 34-37, Luglio/Agosto 1995
- [5] P. Betttoni: "Standard MIDI File Library", The Intelligent Music Workstation (IMW) CD-ROM, IEEE Computer Society Press, 1994.
- [6] Apple Developer Products: "MIDI Management Tools 2.0", Apple Computer Inc., Cupertino, California, 1989.
- [7] International MIDI Association: "MIDI Show Control 1.0", The International MIDI Association, Los Angeles, California, 1991.

TrikTraks: Structural Waveform Transformations with Composed Paths

Arun Chandra
Computer Music Project, School of Music,
University of Illinois, Urbana, IL 61801

Abstract

To describe a sound in terms of the structure of its waveform, and not in terms of its acoustic appearance, is the compositional premise of this program. As a composer of music, I am interested in describing precisely the *path of transformation* I desire, then discovering what its acoustic consequences are, instead of describing precisely the *acoustic consequences* I desire, then doing what is needed to achieve them.

TrikTraks is a program wherein a waveform is specified as an iterated sequence of elements. Each element has two variables: an *amplitude* (+32767) and *number of samples* (1-100). Each variable is given a *path* that determines its magnitude at each moment in time. Each path can be specified as: 1) a sinewave; 2) a triangle wave; 3) a sawtooth wave; 4) a square wave; 5) a polynomial between degrees 3 and 10; 6) or as an amplitude modulated FM wave.

Every variable of every element can be given a path that is independent of all other paths.

A sequence of elements may have up to 64 elements. Upon each iteration of the sequence of elements, each variable of every element changes its magnitude as determined by its path of transformation.

Given an arbitrary duration and the specification of the sequence of elements and their transformational paths, the sound is created for a duration specified by the user, and written to disk for playback.

Two compositions (*smear pulse no sneer* (9 minutes) for solo tape, and *the thin red line of subject matter* (6 minutes) for solo tape) have already been written using this program, and others are underway. The program is written in C, and currently runs on both a NeXT computer running NS3.2 (compiled with the GNU compiler 2.6.3), and a 386PC with a CardD digital i/o board, running DOS6.2 (compiled with the Borland compiler 4.0).

1. Predecessors

The significant predecessors of are the SAWDUST program, (designed by Herbert Brun and written by Gary Grossman, Jody Kravitz, and Keith Johnson at the University of Illinois), and SSP (Sound Synthesis Program), (designed by Gottfried Michael Koenig, and realized by Paul Berg, Robert Rowe, J.D. Banks, and David Theriault at the Institute for Sonology, Utrecht). These programs can be classified as "non-standard sound synthesis programs," following S. R. Holtzman's definition. Paul Berg's description of SSP is appropriate for TrikTraks: "... [it is] best suited for a user who wants to define structures and listen to the results rather than a user who at all costs must have a certain sound."

```
duration(10)                                # total duration in seconds

# initial, final, type, [range() or periods()]
20 40 sin                                     # element 1, samples
1000 -1000 sin periods(2)                   # element 1, amplitude

40 20 tri periods(4) range(1,50)             # element 2, samples
-2000 2000 saw range(10000, -10000)         # element 2, amplitude

10 1 saw periods(75)                         # (etc.)
10000 -10000 sin periods(5) range(20000, -20000)
```

2. Standard Paths

A *Standard Path* is one of the four basic waveforms: sine, square, triangle, and sawtooth. A user specifies the *initial* and *final* magnitudes of a variable, and the *path* by which the initial and final values are connected. By default, a path is given one period of the controlling waveform. The number of periods can be specified by the user. The *range* of the controlling waveform is also given default values that can be overridden by the user. For example, here is an input data file of 3 elements, each of which is given distinct values. When either the *range* or *periods* is not explicitly given, default values are used: range(1,100), periods(1). Anything that follows a # mark is considered a comment.

The samples take the default values for range and number of periods, whereas for the amplitudes, the range and periods are both specified. The default

The first element has an initial sample length of 20 samples, and a final sample length of 40 samples. Its amplitude begins at 1000 and ends at -1000. Both samples and amplitude use a sine wave for their paths. The samples follow one period of a sine wave (the default), and the amplitudes follow 2 periods of a sine wave, as specified. Both samples and amplitude use the default values for their range.

The second element samples begin at 40 and end at 20, the amplitude begins at -2000 and ends at 2000. The samples follow the path of a triangle wave, with 4 periods, and a range of 1 to 50, while the amplitudes follow the path of a sawtooth wave, with one period (the default) and a range of +/-1000.

The settings for the third element should be understandable from the description of the first two.

This sequence of three elements is iterated, and at each iteration the samples and amplitude of each element change their value as controlled by their respective paths, range, and number of periods. The iteration process continues, until the specified duration is reached. There can be up to 64 elements in a sequence, which means there can be up to 128 distinct paths of transformation. By choosing distinct values for the type of path, its range and number of periods, each variable of each element can have a path independent from all other variables paths.

The sounding frequency of this waveform is the sum of the elements' samples at every iteration. Since this number is constantly changing, the frequency is constantly changing. The square and sawtooth paths have a periodic jump from the minimum to the maximum value (or vice versa). This results in an immediate and drastic frequency change, when applied to the variable number of samples. The amount of the change depends on the user specified maximum and minimum values for that variable, and the content and behavior of the neighboring elements. The sine and triangle paths in obvious contrast, have smooth rises and falls.

3. Polynomial Paths

A *polynomial path* connects the initial value of a variable to its finally value following the path of a polynomial between degrees 3 and 10. The specification is done in this way:

```
10 50 poly(7) range(0, 100000)
20000 -20000 poly(3) range(30000, -30000) periods(7)
```

The samples take the default values for range and number of periods, whereas for the amplitudes, the range and periods are both specified. The default limits are the same as described above for the

Standard Paths

The *amfm path* is the following function: $f(x) = \cos(2\pi amf / sr t) * \sin(2\pi fc / sr t + fc/fm * \sin(\pi fm / sr t))$. The specification is as follows:

```
10 50 amfm(am=0.2, fm=0.3, fc=0.5)
periods(2) range(10,50)
10000 -10000 amfm(seed=6)
periods(3) range(20000,-20000)
```

For the samples, the values for the amfm function are given explicitly, and for the amplitude, the values are given by a random number generator whose seed is set to 6. Again, the periods and the range can be given specified values, or use default values.

Arun Chandra
arunc@ux1.cso.uiuc.edu
Computer Music Project, School of Music,
University of Illinois, Urbana, IL 61801

University of Illinois, Urbana, IL 61801

Un software per la gestione in concerto di dispositivi
per il campionamento e l'elaborazione del suono

Franco Degrassi

gruppo musica-informatica c/o Dipartimento di Fisica dell' Universita'
via Amendola 70100 Bari
e.mail : cufaro@bari.infn.it

The present contribution is about a pc software controlling sampling boards, sound spatialization and elaboration devices.

This software, suitable for live applications, by means of a very simple interface, allows one perform both to sample sounds in real time, and to spatialize and elaborate already sampled sounds by m.i.d.i. devices.

Il punto di partenza del lavoro che da alcuni anni conduco assieme al trombonista Michele Lomuto e' costituito da una duplice sfida: l'estensione tecnologica delle potenzialita' dello strumentista e la "sfida economica".

Siamo infatti partiti, da un lato, dalla necessita' di estendere le potenzialita' compositive ed espressive dello strumento acustico senza allontanarsi dai tratti timbrici essenziali che lo caratterizzano, e dall'altro dall'impegno di utilizzare apparecchiature elettroniche non esoteriche ma di facile reperibilita', di uso comune e di costi non elevati, apparecchiature non concepite dai produttori in funzione della musica di ricerca (come quasi tutti i dispositivi presenti sul mercato a relativamente basso costo) ma "forzate" a questo scopo dal rapporto tra lo strumentista acustico e il compositore di computer music.

Si e' scelto il campionamento come forma principale di intervento sul suono.

Il musicista elettronico puo' operare in concerto in due modalita' diverse: o gestendo files di suoni precedentemente campionati dallo strumento acustico e assegnati alle rispettive icone sullo schermo, alla maniera di qualunque sintetizzatore a campionamento in commercio; oppure campionando dal vivo e gestendo suoni appena campionati, in un flusso continuo.

Il campionamento e l'emissione dei suoni campionati e' affidato a schede audio per p.c. (Duetto dell'Audiologic). Sia i suoni acustici che quelli digitalizzati possono essere elaborati in tempo reale mediante processori m.i.d.i. coordinati in una unica stazione di lavoro mediante un programma che gestisce con una interfaccia estremamente semplice e intuitiva per l'utente sia le schede audio per il campionamento che i moduli Niche Audio Control per la spazializzazione del suono e gli elementi per la riverberazione e la trasposizione dei suoni (serie spx Yamaha, ad esempio).

In tal modo il software si presta ad un uso interessante sia per la "musica strumentale contemporanea" (ad esempio esecuzione di versioni particolari di composizioni strumentali, come "Averla e Grifone" di S.Bussotti, in cui i suoni campionati sostituiscono gli strumenti acustici) che per la musica improvvisa (essendo possibile campionare in tempo reale con una scheda mentre le altre contemporaneamente eseguono frammenti precedentemente campionati). Il controllo della spazializzazione, realizzato utilizzando l'approccio descritto in [3], consente di determinare sul momento i percorsi del suono e la velocita' di spostamento utilizzando sia dei presets predefiniti che, in alternativa, un semplice controllo manuale mediante movimento del mouse in piccoli spazi "virtuali"; il controllo dei processori midi consente operazioni di riverberazione o di trasposizione (harmonizer) sempre gestite con estrema semplicita' atta a situazioni concertistiche.

Naturalmente i parametri controllati dal musicista elettronico sono in qualche modo limitati (istante d'inizio e di conclusione del campionamento o della riproduzione, presets di spazializzazione, di riverberazione o di trasposizione, controllo "manuale" della posizione del suono con riferimento a quattro punti di diffusione) poiche' egli, dovendo concentrarsi nell'ascolto dei suoni emessi dallo strumentista acustico e nella scelta dei frammenti da campionare (con una scheda) e contemporaneamente di quelli, precedentemente campionati da riprodurre (anche piu' frammenti con piu' schede), e' assorbito essenzialmente dalla necessita' di seguire l'intreccio "polifonico" o di tipo "strutturale".

Lo strumentista acustico si trova ad operare, in conclusione, come se interagisse con piu' linee di ritardo "intelligenti" e prive di limitazioni temporali nelle durate dei frammenti su cui esercitare il delay.

Riferimenti bibliografici

[1] Audiologic: Broadcasting advanced kit-manuale d'uso v.3.6.

[2] Music Quest: Programmer's toolkit version 4.2.

[3] A.Belladonna, A.Vidolin: "spAAce: un programma di spazializzazione per il Live Electronics" in Proceedings Ciarm 95 Ferrara pp. 113-118

Franco Degrassi via Toti , 78 70125 Bari

tel.(080) 5573029-5563487

Attenuazione Numerica dei Disturbi nei Segnali Audio

Rossi Domenico

L.I.M.

Dipartimento di Scienze dell'Informazione
Università degli studi di Milano
via Comelico 39
I-20139 Milano (Italia)
fax +39 2 55006373
e-mail: music@imiucca.csi.unimi.it

Abstract

The background noise in musical signals, is a very important problem in many applications. In this paper, we present a new numerical method to attenuate this type of noise. The method use, like other similar methods, a noise example (a portion of audio file with only noise) to attenuate; this is used to "clean" the original music signal (an audio file). The attenuation has performed with original signal magnitude modification using an attenuation function depending from the noise example. After the magnitude modification, the original signal was rebuilded with the inverse Fourier transformation.

1 Introduzione

Il presente lavoro, sviluppa una nuova tecnica di attenuazione del "Rumore di Fondo" nei segnali audio: partendo da uno studio sulle tecniche numeriche esistenti, si è cercato di valutarle sia in senso positivo che in quello negativo in modo da poter definire un nuovo algoritmo che ne amplificasse i pregi e ne limitasse i difetti.

Tutti i metodi considerati, utilizzano una stima del disturbo da attenuare in modo da poter specializzare l'attenuazione su quel particolare tipo di disturbo.

L'importanza di utilizzare una stima del rumore di fondo, deriva dal fatto che differenti tipi di sorgenti hanno differenti tipi di rumore di fondo, e occorre quindi valutare attentamente "caso per caso".

Per poter utilizzare il metodo proposto, occorre avere il segnale da attenuare in forma digitale, rappresentato da una sequenza di campioni. Occorre poi determinare quale parte della successione utilizzare per determinare la stima del rumore di fondo (in genere si utilizza una porzione di silenzio come l'intervallo fra due brani musicali).

2 Attenuazione

Supponiamo di avere identificato quella parte della successione di ingresso $y(n)$, da utilizzare per la determinazione della stima dello spettro del rumore di fondo, ovvero:

$$y(i), \quad r \leq i \leq r + S$$

($S >$ dimensione minima)

Occorre notare che il numero di campioni che vengono utilizzati per la determinazione della stima, devono essere in numero sufficiente per permettere una valida valutazione del rumore; logicamente più campioni vengono utilizzati e migliore sarà la stima ottenuta. La sezione della successione di ingresso determinata, viene divisa in blocchi di B (potenza di 2) campioni in modo da

poterli analizzare singolarmente e successivamente mediare i risultati ottenuti dall'analisi.

Dato che il nostro scopo è quello di determinare la stima dello spettro, ogni blocco di campioni verrà sottoposta ad una trasformata di Fourier discreta (DFT) per permetterne il passaggio nel dominio delle frequenze; successivamente, verrà calcolata l'ampiezza di ognuna delle $B/2$ frequenze dello spettro. Ottenute le ampiezze di ogni frequenza di ogni blocco, verrà effettuata la media matematica dei valori ottenendo la stima dei livelli di ognuna delle $B/2$ frequenze dello spettro.

Formalizziamo il procedimento precedente per la determinazione della stima dello spettro. Ogni blocco di segnale di B campioni, viene definito da:

$$y_j(i), \quad j \in [1, \dots, S/B] \text{ e } i \in [0, \dots, B - 1]$$

dove j è l'indice del blocco di appartenenza e i è l'indice del campione all'interno del blocco.

Lo spettro di ogni blocco di segnale viene determinato tramite la trasformata di Fourier, nel modo seguente:

$$N_j(k) = \text{DFT}[y_j] \in C, \quad k \in [0, \dots, B - 1]$$

dove il blocco j -esimo di segnale è stato indicato con y_j .

I blocchi degli spettri così ottenuti, vengono sottoposti alla procedura di media aritmetica ottenendo i B valori dei livelli medi degli spettri dei blocchi considerati:

$$|\hat{N}(k)| = \frac{1}{B} \sum_{j=0}^{B-1} |N_j(k)|$$

Considerando la simmetria dello spettro, allora i valori utili per la stima dei livelli dello spettro sono solamente i primi $B/2$ ovvero:

$$k \in \left[0, \dots, \frac{B}{2} - 1 \right]$$

Capitolo 10 - Filtraggio dei segnali

Una volta ottenuta la stima dei livelli dello spettro del rumore di fondo, è possibile procedere alla elaborazione del segnale. Il segnale da elaborare è costituito da una N-upla di valori o campioni che abbiamo chiamato $y_j(n)$.

Per effettuare l'elaborazione, anche in questo caso occorre dividere la successione in blocchi di B campioni che verranno elaborati in sequenza. Utilizzando la notazione precedente, supponiamo di avere diviso il segnale in ingresso come segue:

$$y_j(i), \quad j \in [1, \dots, N/B] \quad i \in [0, \dots, B-1]$$

Di ogni blocco y_j di campioni, viene calcolato lo spettro tramite la trasformata di Fourier discreta:

$$Y_j(k) = \text{DFT}[y_j] \in C, \quad k \in [0, \dots, B-1]$$

Si noti che $Y_j(k)$ è un valore complesso e mantiene sia l'informazione sull'ampiezza che quella sulla fase della componente k-esima.

$$\text{Amp}(k) = |Y_j(k)| = \sqrt{\text{Re}^2[Y_j(k)] + \text{Imm}^2[Y_j(k)]}$$

$$\begin{aligned} \text{Fase}(k) &= \text{pha}(Y_j(k)) = \\ &= \arctg\left(\frac{\text{Imm}[Y_j(k)]}{\text{Re}[Y_j(k)]}\right), \\ &\quad k \in [0, \dots, \frac{B}{2}-1] \end{aligned}$$

Dopo avere determinato le caratteristiche frequenziali del blocco di segnale sotto analisi, il nostro scopo diventa quello di modificare i livelli dello spettro determinato per poter generare, tramite la trasformata inversa, il segnale attenuato. Ciò equivale a creare un blocco di un nuovo segnale che ha come fase la stessa del segnale originale, $\text{Pha}(k)$, e come ampiezza una successione di valori che viene determinata in funzione dell'ampiezza del segnale, $\text{Amp}(k)$, e dell'ampiezza della stima del rumore $|\hat{N}(k)|$.

Sia x_j il blocco del segnale attenuato e $X_j(k)$ il suo spettro, allora:

$$|X_j(k)| = f(|Y_j(k)|, |\hat{N}(k)|) = \alpha_k \cdot |Y_j(k)|,$$

$$\text{pha}(X_j(k)) = \text{pha}(Y_j(k))$$

La parte innovativa del metodo proposto, consiste nel come vengono calcolati i livelli dello spettro di ogni blocco; questi vengono determinati con

10.1.1.1 Attenuazione del segnale

l'ausilio di una *funzione di attenuazione* $f(.,.)$ che esprime se e come deve essere diminuito il singolo livello di ogni componente $|Y_j(k)|$. La funzione di attenuazione utilizzata, è del tipo:

$$|X_j(k)| = f(|Y_j(k)|, |\hat{N}(k)|) = \alpha_k \cdot |Y_j(k)|, \\ k \in [0, \dots, \frac{B}{2}-1]$$

dove $\alpha_k \in [0, 1]$, rappresenta il coefficiente di attenuazione da applicare al livello del segnale della frequenza k-esima dipendente dalla differenza tra il livello del segnale e il livello della stima anch'essi alla frequenza k-esima. Il coefficiente di attenuazione può essere determinato tramite differenti funzioni di attenuazione; il nostro studio si è rivolto ad una particolare famiglia di funzioni che, opportunamente parametrizzata, permette l'adattamento a differenti tipi di segnale e rende possibili differenti gradi di attenuazione:

$$\alpha_k = p \sqrt{\frac{(|Y_j(k)| - |\hat{N}(k)|)}{M}} \cdot s$$

dove M rappresenta il fattore di normalizzazione, p indica l'andamento della curva di attenuazione (lineare, quadratica, cubica ecc.) e s fornisce un controllo sulla pendenza della curva stessa.

La figura 1, rappresenta un grafico dove, per valore fissato del parametro s , vengono proposte quattro curve di attenuazione. L'asse delle ascisse viene utilizzato per la differenza normalizzata dei livelli dei due spettri; in questa differenza si possono identificare due casi particolari:

- la differenza è inferiore a zero nel qual caso l'attenuazione è massima ($\alpha_k = 0$);
- la differenza normalizzata è superiore a $1/s$, allora l'attenuazione è nulla ($\alpha_k = 1$).

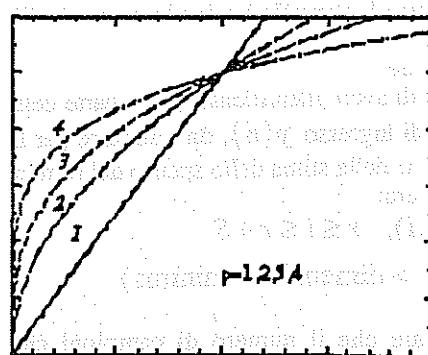


Fig. 1: $s=1000$

Una volta che sono stati determinati i livelli di tutte le $B/2$ frequenze del blocco di segnale, occorre procedere alla ricostruzione del segnale a partire dai livelli determinati e dalla fase del segnale originale.

Per fare questo si determinano sia la parte reale che la parte immaginaria dello spettro avente livelli e fase determinati:

$$\text{Re}[X_j(k)] = |X_j(k)| \cdot \cos[\text{pha}(X_j(k))],$$

$$\text{Imm}[X_j(k)] = |X_j(k)| \cdot \sin[\text{pha}(X_j(k))].$$

$$k \in \left[0, \dots, \frac{B}{2} - 1\right]$$

Per quanto riguarda i rimanenti $B/2$ valori, si utilizza la simmetria dello spettro in modo che:

$$\text{Re}[X_j(k)] = \text{Re}[X_j(B-k)],$$

$$\text{Imm}[X_j(k)] = -\text{Imm}[X_j(B-k)].$$

$$k \in \left[\frac{B}{2}, \dots, B-1\right]$$

Per poter tornare nel dominio temporale, viene effettuata una trasformata di Fourier inversa sullo spettro $X_j(k)$ determinato:

$$x_i(k) = \text{IDFT}[X_j] \in R, \quad k \in [0, \dots, B-1].$$

Lo schema in figura 2, rappresenta il procedimento appena spiegato.

Al metodo proposto, possono venire applicate alcune varianti che permettono un migliore adattamento al segnale sotto analisi. Una prima variazione, consiste nel modificare solo i livelli delle componenti la cui frequenza risiede in un determinato range (f_1, f_2) definito a priori.

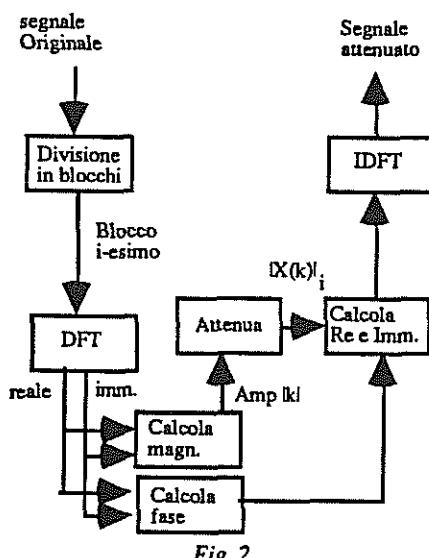


Fig. 2

A causa della discretizzazione operata dal campionamento, la modifica dei livelli dello spettro avverrà solo per quelle frequenze multiple di

f_{camp}/B , ovvero per quei valori di k che soddisfano alla seguente relazione:

$$\frac{f_1}{f_{camp}/B} < k < \frac{f_2}{f_{camp}/B}.$$

Un'altra variazione possibile, consiste nel determinare il parametro di attenuazione α_k non in funzione della differenza dei singoli livelli degli spettri, ma utilizzando un insieme di livelli corrispondenti ad un gruppo (o range) di frequenze successive.

A tale scopo supponiamo di dividere le $B/2$ frequenze in gruppi di R frequenze ognuno, e di determinare un unico coefficiente di attenuazione per tutte le frequenze del gruppo.

Le $B/2$ frequenze vengono divise in $B/(2R)$ gruppi di R frequenze, e chiamando i l'indice del gruppo di appartenenza, allora il generico α_k , con $i \cdot R \leq k \leq (i \cdot R) + R - 1$, verrà sostituito da $\bar{\alpha}_i$ con $0 \leq i \leq B/(2R)$:

$$\bar{\alpha}_i = \frac{1}{R} \sum_{k=i \cdot R}^{(i \cdot R) + R - 1} \alpha_k, \quad 0 \leq i \leq \frac{B}{2R}.$$

Il k -esimo livello dello spettro del generico blocco di segnale da elaborare diventerà:

$$|X_j(k)| = \bar{\alpha}_i \cdot |Y_j(k)|,$$

$$\forall k \in [(i \cdot R), (i \cdot R + R - 1)]$$

Come si è visto, l'attenuazione dipende esclusivamente dalla differenza tra il livello del segnale e il livello della stima del rumore ad ogni frequenza; è possibile modificare il metodo proposto permettendo che la determinazione del coefficiente di attenuazione dipenda, oltre che dai livelli, anche dalla frequenza in cui viene calcolato. Questo si ottiene facilmente scalando il livello della stima del rumore per un ulteriore coefficiente determinato in funzione della frequenza del livello stesso:

$$|X_j(k)| = f(|Y_j(k)|, \beta_k \cdot |\hat{N}(k)|) = \alpha_k \cdot |Y_j(k)|,$$

$$k \in \left[0, \dots, \frac{B}{2} - 1\right]$$

dove β_k , rappresenta la dipendenza della attenuazione dalla frequenza k -esima.

Si noti che questa variazione può essere vista anche come determinazione dell'attenuazione tramite una funzione di due variabili, di cui una è il livello dello spettro ad ogni singola frequenza e l'altra è la frequenza stessa. L'utilizzo di questo ulteriore coefficiente, può essere utile sia nel "centrare" il procedimento in particolari frequenze rispetto ad

altre, che nell'attenuare o amplificare l'attenuazione a secondo delle frequenze considerate.

Una funzione di interessante utilizzo per la determinazione del parametro β_k , potrebbe essere quella che fornisce la sensibilità dell'orecchio umano alle differenti bande del segnale; questo permetterebbe di effettuare l'attenuazione del segnale solo in quelle bande "comprensibili" all'udito umano.

3. Valutazione

Una valutazione del metodo, può essere data sia tramite l'incremento del rapporto Segnale/Rumore che tramite un esame visuale della forma d'onda attenuata. L'SNR viene calcolato considerando il valore più alto in una sezione di solo rumore prima dell'attenuazione e valutando il cambiamento di tale valore dopo l'elaborazione.

Per quanto riguarda l'incremento del SNR, la tecnica proposta riesce ad arrivare fino a un incremento di 10 dB; occorre notare però che tale incremento non sempre misura un effettivo incremento di qualità. Infatti tramite un'attenuazione molto aggressiva, è possibile aumentare notevolmente l'SNR causando però nel segnale elaborato delle degenerazioni del segnale stesso a causa della perdita di alcune componenti troppo attenuate. Comunque, nelle prove effettuate, si è sempre riusciti ad ottenere degli ottimi risultati di incremento di SNR e quindi di qualità del segnale audio, senza incappare in degenerazioni del segnale stesso.

Nelle figure successive è possibile visualizzare una sezione di segnale audio sia prima che dopo l'attenuazione.

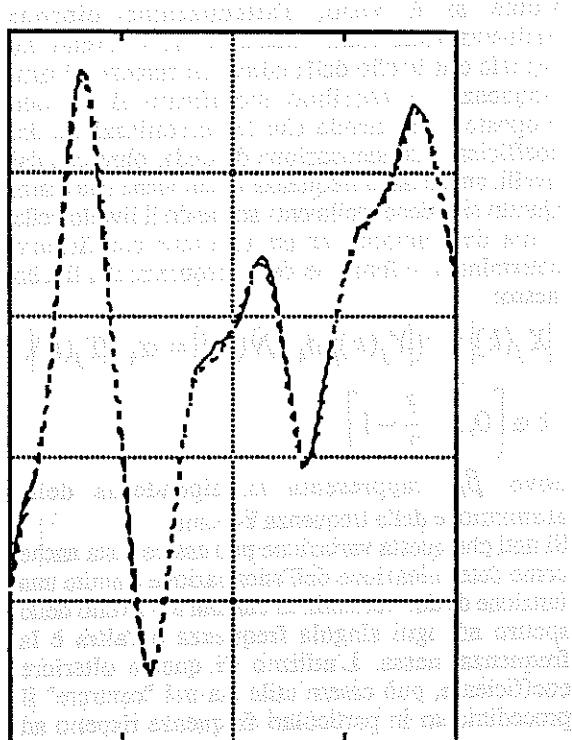


Fig. 3

Si può notare come vengano smussate quelle parti di segnale che presentano delle variazioni pseudo-casuali, mentre viene mantenuto l'andamento generale della forma d'onda.

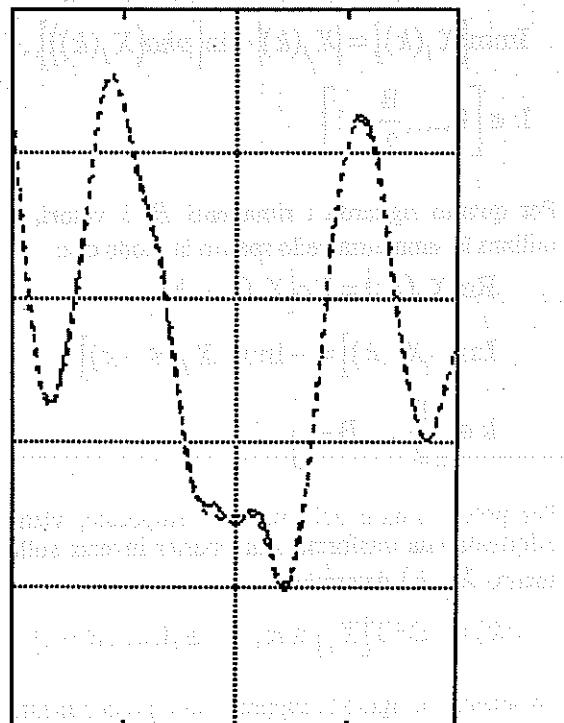


Fig. 4

Bibliografia:

Rossi D.; "Tecniche numeriche per il restauro dei segnali audio";

Tesi di Laurea in Scienze dell'Informazione
Università degli studi di Milano; 1993-1994.

F.R.Moore; "An introduction to the Mathematics of Digital Signal Processing";
Computer Music Journal, Vol. II N° 1 e 2; 1978.

A.V.Oppenheim, R.W.Schafer; "Elaborazione numerica dei segnali"; Franco Angeli Editore; 1993.

T.G.Stockham, Jr.; "Restoration of Old Acoustic Recordings by Means of Digital Signal Processing";
Audio Engineering Society: Preprint, 41st Convention; Ottobre 1971.

S.V.Vaseghi, R.Frayling-Cork; "Restoration of Old Gramophone Recordings";
Audio Engineering Society Journal; Vol. 40 N° 10 pp.791; Ottobre 1992.

An Environment for Music Performance

Paolo TORRENTE, Vittorio COLOMBO, Vincenzo LOMBARDO

Dipartimento d'Informatica, Università di Torino
Torino - Italy

Abstract

The purpose of this work is to offer an interactive environment, called FRIBERG, suitable for testing and validating rules for music performance. The system is basically centered on a core of symbolic rules worked out at KTH Institute in Stockholm. FRIBERG has allowed the empirical study of model details and offers a set of graphic tools for displaying score analysis and performance.

Introduction

Music performance has been widely explored during the last years both in its musicological and scientific aspects (A.A.V.V., 1994).

If performance task in music is left to a computer, the result sounds lifeless and dull. Computer mechanically maps score to sound, while human players introduce a plethora of expressive deviations from score nominal parameters such as duration and loudness, which have a fundamental role in the process of communication of music information from the composer to the listener. Deviations are due both to physical limitations of the instrument/ performer, and to interpretative needs: they indeed make the difference between human and computer performance.

A possible way to improve music acceptability in computer performance is based on a rule model. One of the best known models of this kind has been carried out by KTH researchers in Stockholm since 80s' (Friberg, 1991; Sundberg et al., 1989).

A generative model for music performance is based on interpretative rules whose aim is to simulate the behaviour of a gifted musician. Each rule is applied with a strength parameter k , that acts as a weight determining the amount of the corresponding deviation. A deviation alters a note parameter -loudness, duration, onset time - or introduces a rest.

The reader interested to syntax and semantics of the rules is pleased to refer himself to the specific sources (Friberg, 1991).

KTH rules have been the source of some implemented works, like the system MELODIA (Bresin et al., 1991). The author, pursuing a subsymbolic approach, has built a neural network-based system to produce a real time performance

of any music score. His main goal is to reproduce the performing style (staccato, legato, grace-notes) of a piano player.

Also our system FRIBERG is based on the KTH grammar for music performance. Our main interest focuses on the analysis of stylistic features in musical repertoires, via a flexible rule system.

The system FRIBERG

The rule-based system FRIBERG works directly on a symbolic level, where each rule interacts with a strength given by the parameter k .

FRIBERG offers an interactive environment with a custom graphic interface, for setting the values of the features that contribute to the expressive deviations computed by the rules and for checking the results of rule applications. Rules can be tested and validated individually or in group, in order to provide a fine characterization of the various performance styles and musical repertoires. System input is a symbolically coded version of a score. The output is the MIDI file resulting from the computation of the rules.

The overall system (figure 1) is composed of two parts:

- a graphic editor that displays the score and allows one to add interactively different types of musical information - such as current key, harmony context, phrase structure - which are not explicitly written in the common practice notation, but are requested by some rules. This new information is immediately displayed.
- the main module that, after the selection of a subset of the rules and the setting of each rule strength, applies them to the nominal values provided by the score and produces a MIDI file that can be listened in the same environment. This module also produces some graphical plots of the resulting deviations.

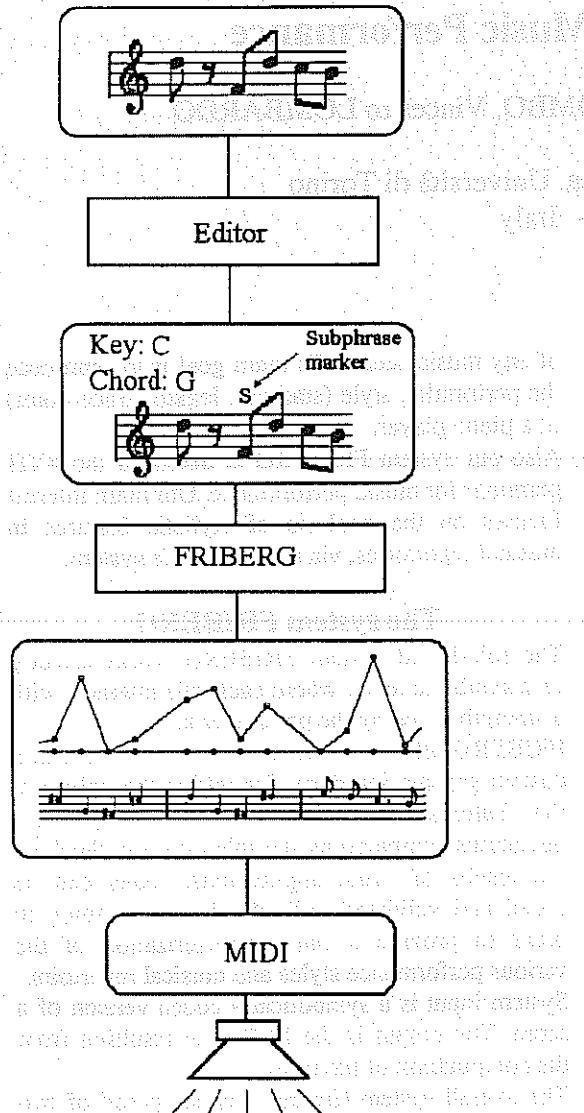


Figure 1: System architecture diagram showing the flow from musical score to MIDI output.

The rules we have employed are basically the KTH kernel set with the following arrangements. As far as polyphonic music, synchronization problems arise after the application of rules to each voice individually. The solutions practically approached are based on the election of a "main" voice that dictates the time table for the remaining voices. The rules are uniquely applied to this elected voice and the remaining voices are consequently synchronized. The method proposed by Friberg had been the introduction of a "virtual voice", artificially constructed from the other voices, following the principle of the "shortest note". We adopted an alternative solution, hinted by Friberg himself: to select an actual "leading" voice (e.g. a soloist) among the set of voices. Moreover, we have augmented the rule set by introducing a new rule inspired to Rash's work (1988), on ensemble performance. Since human

players can never be perfectly synchronized while playing together, the time onset of simultaneous notes is desynchronized of a small amount. We can think of these changes as phase differences among voices in order to reproduce a sort of rough "chorus effect".

We have conducted several listening tests to validate our model, by selecting scores from different repertoires. For each score, we have experimented several k-weight vector setups in order to achieve a result both pleasant and suitable from a stylistic point of view.

Conclusion

In this paper we have presented a rule-based system for music performance. The system architecture is flexible enough to be easily extended both in the rule set and in the possible applications of the environment.

Now we are going to improve model versatility, and to augment and calibrate the rule set.

References

A.A.V.V. Proceedings of the Aarhus Symposium on Generative Grammars for Music Performance; J. Sundberg ed.

BRESIN R., DE POLI G., VIDOLIN A.(1991). "Un approccio connessionistico per il controllo dei parametri nell'esecuzione musicale", Atti IX Colloquio di Informatica Musicale, Genova, 1991

FRIBERG Anders (1991). "Generative Rules for Music Performance: A Formal Description of a Rule System", Computer Music Journal, vol. 15 No.2, Summer 1991, pp. 56-71.

SUNDBERG, J. FRIBERG, A. & FRYDEN L. (1989). "Rules for automated performance of ensemble music", Contemporary Music Review, vol. 3, pp. 89-109.

RASH Rudolf (1988) "Timing and synchronization in ensemble performance", in Sloboda J. (Ed.), Generative Processes in Music. Clarendon Press, Oxford, pp. 70-90.

Dipartimento di Informatica

Università di Torino

Corso Svizzera, 185

10149 - TORINO

Tel.: 011/7429111

Fax : 011/751603

E-mail: music@di.unito.it

GRAINS

a software for real-time granular synthesis and sampling
running on the IRIS-MARS workstation.

Raffaele de Tintis

MM&T, via Aosta 2, 20155 Milano, Italia
fax 02/33105679
e-mail: detintis@ghost.sm.dsi.unimi.it

1. Introduction

This paper describes Grains, a tool for sound synthesis based on granular techniques with some open possibilities for processing and spatializing the generated grains.

Due to the characteristics of the MARS workstation, the system is completely manageable by midi controllers allowing the composer to choose his own way to work with it.

On one side we can have a master keyboard and/or midi faders, on the other we can automate the system using a sequencer or a software like Max.

In general the composer will take advantage of having both the approaches: realtime feedback playing the system, or the possibility of saving and editing a performance in successive times.

2. Data Reduction

Due to the interest that this kind of synthesis has obtained in the last years from both composers and researchers, many important criteria have been presented in order to better organize the high number of variables involved, from the hierarchical approach to the tendency masks introduced by Truax.

Especially in a real-time implementation, the reduction of data is a fundamental goal for the effectiveness and the efficiency for the composer who can work with high level parameters.

This was the primary aim I tried to achieve with the current implementation in order to reach an improved real-time feedback in a live performance.

The general approach has been that of creating an open system where the composer decides each time which are the parameters to work with and how to organize them, so, only those will be connected to any hardware sending midi continuous controllers and kept close at hand.

In general their number depends on how many different executors and external devices are involved. In the simplest configuration every variable is connected with one midi continuous controller so there is a 1:1 ratio.

Otherwise, the composer can reproduce a hierarchical approach organizing the parameters at different levels and fixing the relations between them. In this way, only the parameters at the highest level will be directly controlled by midi events, the other ones will be updated depending on the chosen relationships.

This kind of organization is possible due to the midi management system of the MARS. Infact, every variable used by the algorithms is controlled depending on some function of different midi events that can be chosen by the user.

3. Sound Generation

Sound computation is made by four identical algorithms that calculate quasi-synchronous granular synthesis, with the output from every algorithm being a stream with the grains following one after the other.

The parameters that determine the sound production, independently for each algorithm, are the classical ones for this kind of synthesis so, before activating each grain, a frequency, length, waveform and amplitude are generated.

A fifth parameter, the horizontal density, determines if, when one grain is terminated, a new one has to be activated, controlling how near two successive grains are on the stream, or in other words, the number of generated grains in the time unit at a given length.

At each sample after the passing of each grain this variable is compared with the output of a gaussian random signal generator and, if it is higher than it, a new grain is generated so, the law that directs the activation of successive grains is based on probabilistic evaluations.

For these reasons we are uncertain on the time that separates two successive grains, resulting in micromodulations in the generated spectra that disappear at high densities, when the level of the density variable is always higher than that of the gaussian generator and there is no time left between two adjacent grains.

Grain length, amplitude and the horizontal density are generated with normal distribution in a range

that is controlled via midi independently for each parameter. For the frequencies and waveforms, on the contrary, there are two variables each forming the final value as follows:

$$\text{Frequency/Wave N} = \text{Offset} + \text{Rand}(\text{RndLevel})$$

The values are extracted randomly from one window whose dimension is RndLevel, the other variable is an offset to shift the same window to the desired level; in this way we can control the level of randomness and the range of the generated values.

With this organization we set the overall density of the grain production in two ways. The horizontal density by the relative variable and the vertical density deciding how many algorithms are active at the same time controlling the stratification of the sonic material.

The sound synthesis is calculated by four identical algorithms developed in the EDIT20 graphical editor.

Due to the characteristics of this software, all of the resources used, as the waveforms and envelopes, can be easily changed. For this reason when the user runs the application, the default waveforms and grain envelope are automatically loaded into the MARS hardware, but, if we want to use different waveforms, all we have to do is to build them (for example with the WaveLoader application included in the MARS standard software) and give them the names grain1.wt...,grain20.wt for the waveforms and grainenv.wt for the curve used as the grain envelope.

Infact, what Grains does when it's first initialized is to load the files with these names in the current directory making the composer free to work with his own waves.

Included in the developed algorithms is a real-time continuous sampling line that adds the possibility to use an external signal as a base for the granulation process.

The possibility to select a fixed or a random waveform for each grain allows us to create a grain stream with each grain being an energy contribute from signals with different natures, having been synthetized at an earlier step or belonging to the natural landscape.

Different criteria for the granulation of continuous input sounds have been presented by composers and, in general, using the computational cost of the algorithms to build different images of the input sound or new formants to add to it has provided good results.

For this task, with Grains we can work with up to five separated outputs, one being for example a real-time continuously sampled voice and the other four being the outputs from the algorithms building new spectral zones or formants to add to it, each sound then, can be spatialized on a different output of the MARS.

4. Grains Processing

Some criteria for the processing of the grains, in the specific case of quasi-synchronous granular synthesis, are obtained by observing that the grain stream can be seen also as an impulse train with completely shapable impulses.

In this way, with minor additional costs, we can simulate some of the physical models that can be described as a simple exciter source and a resonator. Here the exciter source is the impulse train, so, we only need to add the resonating part as a particular case of processing for the grains. For these reasons, in Grains, every algorithm has three outputs, each with an independent gain controllable in real-time.

The first one contains the "clean" result from the granular process, the second one goes into a 3 filters bank reproducing the resonating part of the Vosim model with one low pass and two high pass second order filters. The third output goes into a waveguide resonator for the simulation of one-dimensional wave propagation.

It's clear now, that with the output from the algorithms being a grain stream going into the filters bank, we are very near to reproducing the Vosim model with the possibility to produce up to three different formant zones in each of the four algorithms.

Using the resonator it is possible to obtain formants too, in any case, the most noticeable effect is the loss of the perception of the distinct grains into the sonic texture and the possibility to vary a lot from the classical timbres obtained with granular synthesis. In some cases string like sounds can be obtained and, in general, has been shown (Fricke) that in the process of simulation of instrumental sounds with pulse trains, the pulse shape is characteristic of the interaction among the mechanical parts of the instrument so, controlling it we can simulate the stationary part in the spectra: we have to work with the grain stream as with an impulse train.

At last we notice that, the easy task of obtaining the concentration of the energy in particular spectral zones producing formants, gives good results when the aim is to add some new characteristics to continuous sampled sounds using the input line available. In this case we can add to the sampled sound both the results from the filters bank and the resonator reaching more complex spectras and use the eighth outputs of the MARS to have a real-time spatialization of the different images obtained.

5. User Interface

Grains has been developed primarily as a tool for real-time synthesis so that all of the parameters used to control the sound are midi variables manageable with external devices.

In any case, a user interface has been written in C for the ATARI platform allowing the complete control of the system via mouse. The only constraint with respect to the use by external controllers is the possibility to control only one variable at a time due to the mouse driven interface. There are, however, some features of the graphic interface that can be helpful, for example, when the application is first run, the algorithms and all of the resources needed for the granular synthesis such as the waveforms and the envelope are automatically loaded on the MARS. Otherwise they should be loaded one by one by the user at the right address (a control word) of the MARS fun memory.

Another feature included in the interface is the possibility to sample in real-time the mixed output from all of the algorithms for a maximum time of about 25 seconds for each sample.

In this way, we have the possibility to save in a simple binary file some granular synthesis sounds and then re-use them in different ways.

For example, various sample management softwares make the conversion from the binary to the Midi Sample Dump format allowing us to use the synthesized waveforms on any commercial sampler. Notice that, in this case, the sounds will be accelerated or decelerated because the MARS works at about 39KHz and, in general, the nearest playback rate on commercial samplers is 44,1 KHz. Another possibility could be that of using the waveforms in the binary format by any board with DA converters or re-use them on the MARS as .wt files (the extension used in the MARS ambient for the files containing a waveform).

References:

Roads, C. 1988. "Introduction to Granular Synthesis." Computer Music Journal 12(2):11-13.

Truax, B. 1988. "Real-Time Granular Synthesis with a Digital Signal Processing Computer." Computer Music Journal 12(2):14-26.

Jones, D., Parks, T. 1988. "Generation and Combination of Grains for Music Synthesis." Computer Music Journal 12(2):27-33.

MacKay, J. 1984. "On the Perception of Density and Stratification in Granular Sonic Textures: An Exploratory Study." Interface 13:171-186.

Mitsuhashi, Y. 1982a. "Musical Sound Synthesis by Forward Differences." Journal of the Audio Engineering Society 30(1/2):2-9.

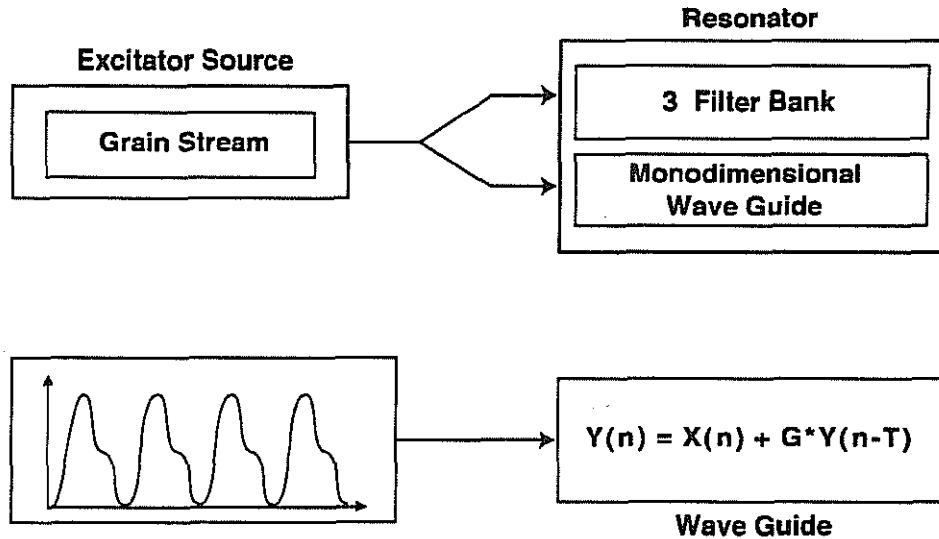
Roads, C. "Asynchronous Granular Synthesis." Representation of Musical Signals, MIT Press, 1991.

Fricke, J.P. "Transients with Irregular Pulses in the Pulse Forming Sound Generating Process." Proceedings of the 2nd International Conference on Acoustics and Musical Research, Ferrara, 1995.

De Poli G., Piccialli A. "Forme d'onda per la sintesi granulare sincrona." Atti del CIM, Roma, Marzo 1988, pp.70-75.

De Poli G., Piccialli A. "Pitch Synchronous Granular Synthesis." Representations of Musical Signals, MIT Press, 1991.

Smith, J.O. III, "Waveguide Filter Tutorial." Proc. ICMC, 1987, pp.9-16.



.....

Progetto e realizzazione di un sistema di sintesi del suono in tempo reale nel dominio della frequenza

Ing. M. Giachi

V. Monte Priore 3, I-60131, Ancona, Italy
Tel.: +39(0)71 43048

Abstract

This paper describes a real-time synthesis realization of musical instruments sounds. The parameters, in the frequency domain, are codified in the memory and after, at synthesis time, are interpolated with a linear technique. The system is based on a PC board with two microprogrammable DSPs (Orion) and the PC itself controls the polyphonic generation of 4 notes sended from a MIDI keyboard. The control program runs under Windows 3.1. The algorithm allows a synthesis of the sound with 27 partials controlled in frequency and amplitude. The timbre dynamic modelling and temporized advancement of the spectral windows that can be looping are possible.

1 Introduzione

Il progetto realizza *real-time* la generazione di suoni musicali codificati nel dominio della frequenza con un criterio basato principalmente sulla sintesi additiva. Si riescono a generare 4 note costituite ciascuna da 27 parziali controllati in frequenza ed ampiezza. Uno studio futuro sarà quello di eseguire il calcolo polifonico direttamente nel dominio della frequenza e antitrasformare successivamente con l'uso dinamico di un certo numero di sinusoidi fissate.

2 Descrizione

Il microprogramma è progettato per generare due note per ogni singolo DSP con la possibilità di trasferire il risultato ad altri DSPs connessi in cascata.

Nel nostro caso, utilizzando una scheda con due DSPs (Orion), potremmo quindi avere 4 note di polifonia e ascoltare quindi la qualità timbrica di determinati strumenti suonando le tre note che costituiscono un accordo e la quarta nota per la linea cosiddetta melodica (canto).

Il microprogramma permette:

- Sintesi additiva con 27 sinusoidi interpolate in ampiezza ed in fase con controllo delle microvariazioni in frequenza dei singoli parziali.

- Avanzamento controllato dello spettro tempo variante con gestione dello *start*, *end*, *loop point*.
- Politimbricità e multisplit.

Interagisce con i DSP un mini sistema operativo scritto in ambiente Windows che si dedica al controllo dei vari processi di *real-time* quali:
- Elaborazione degli eventi MIDI (MPU 401).
- Controllo dell'involucro.
- Monitor della ram del DSP.
- Caricamento dei suoni.

3 Analisi

L'analisi per l'estrazione dei parametri necessari alla successiva sintesi è stata realizzata con una tecnica basata sulla DFT.

Questa teoria di analisi si basa sulla conoscenza della frequenza della nota musicale da analizzare e sulla quasi stazionarietà in frequenza di quest'ultima.

La conoscenza della frequenza nominale (in realtà segnale pseudoperiodico) ci permette di analizzare il segnale con la DFT applicata ad una finestra di N campioni. L'analisi verrà ripetuta, spostandosi ogni volta in avanti nel tempo con uno shift di 270 campioni, fino al termine del suono memorizzato senza necessità di finestratura perché la quasi stazionarietà ci consente il calcolo senza troncature del segnale.

Con gli a_k e b_k calcoleremo il modulo dell'ampiezza e la fase. Utilizzeremo

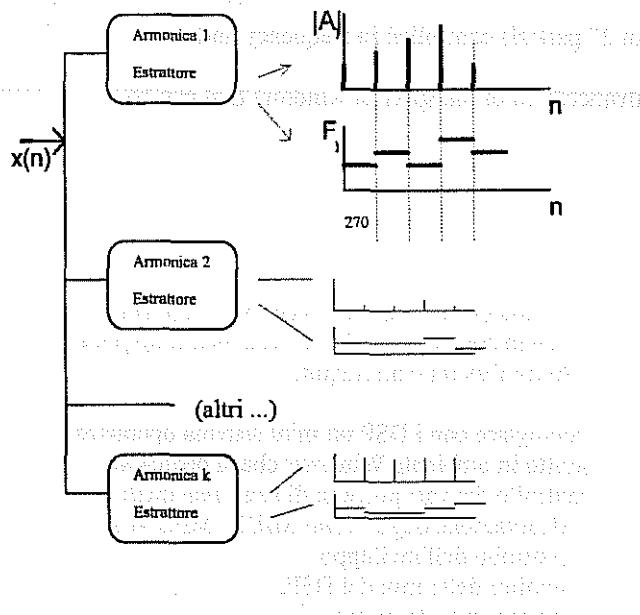
successivamente un algoritmo di ricostruzione della fase *unwrapped* per l'estrazione della frequenza media all'interno del frame di 270 campioni.

$$\theta_t = \operatorname{tg}^{-1} \left(\frac{b_t}{a_t} \right)$$

$$F_t = \frac{1}{2\pi} \frac{\theta_t(n+270) - \theta_t(n)}{270 \cdot TC}$$

TC = tempo di campionamento.

Schematicamente per ampiezza e frequenza:



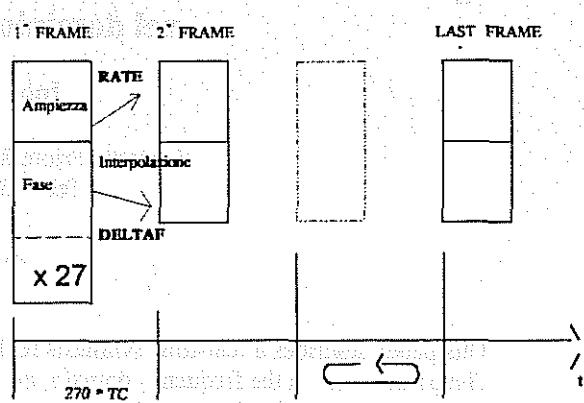
4 Sintesi

Una volta ottenute le frequenze medie di ogni frame ricostruiremo la fase di ogni armonica integrando la frequenza che, nel caso di funzioni integrate costanti in determinati intervalli, diventa:

$$\theta = \theta_0 + 2\pi \left(\int_{t-270TC}^t F_1 dt + \int_{t-270TC}^t F_2 dt + \dots + \int_{t-270TC}^t F_k dt \right)$$

La sintesi in oggetto è costituita quindi dalla sommatoria di 27 parziali controllati ciascuno sia in frequenza che in ampiezza. L'aggiornamento dei parametri, letti da una ram (1 Mw), viene effettuato in maniera automatica dal DSP ogni 270 campioni e questo corrisponde, per la frequenza di campionamento ottenuta, ad un tempo di circa 7 ms.

Andamento temporale della sintesi:



Il microprogramma è stato implementato con la possibilità di riscalare in ampiezza, attraverso un *level* ogni armonica indipendentemente dalla rappresentazione spettrale memorizzata. Questo ha permesso la filtratura nel dominio della frequenza e l'aggiustamento del timbro nello spostamento dell'esecuzione di una nota sulla tastiera (necessità di filtrare le armoniche superiori con l'aumento della frequenza della fondamentale).

Il medesimo controllo in ampiezza consente l'equalizzazione diversa del timbro in funzione della dinamica del tasto della nota premuta. Potremo praticamente definire una funzione di trasferimento $|H(jw)|$ variabile nel tempo controllata dal PC.

Il numero di cicli di microprogramma occorsi per la generazione della sintesi descritta è risultato essere 429 ed avendo il DSP un ciclo macchina di 62.5 nS si ottiene una frequenza di campionamento di 37.296 KHz.

5 Conclusioni

Il risultato è stato quello di una macchina funzionante che ha permesso di ascoltare i suoni e di capire l'eventuale realizzabilità di uno strumento con questa tecnica.

Abbiamo notato un buon controllo della fase con la tecnica dell'integrazione della frequenza media ed un buon inseguimento dell'ampiezza delle varie armoniche. Questo ha consentito, con alcuni suoni, di riprodurre quasi fedelmente il campione originale. Le maggiori differenze si hanno ovviamente in strumenti che hanno un'elevata densità spettrale che, anche se risintetizzati con armoniche non contigue per accrescere la brillantezza, si discostano abbastanza dal suono di partenza. Stiamo lavorando alla generazione di nuovi suoni, non derivati da strumenti musicali tradizionali, per ascoltare possibili risultati sonori. Questa ricerca è stata uno dei motivi di spinta nella realizzazione di questo lavoro.

Visual Composing System: a new approach to compositional real-time systems

Claudio Rosati[†], Fabio Armani[†], Giorgio Nottoli[‡], Riccardo Santoboni[†]

[†]IRIS s.r.l.
Paliano (FR), Italy

[‡]Conservatorio di Musica Licinio Refice
Frosinone (FR), Italy

Abstract

This paper summarizes a computer-aided composition package that allows the user/composer to intuitively represent and describe his/her own musical thinking in terms of objects (either sonic or musical) and actions upon those objects.

Visual Composing System allows a continuous feedback between compositional thinking and perceptual results, by means of an extensive usage of graphic tools and a real-time engine featuring audio control, generation and processing.

1 Introduzione

In questo articolo viene presentato, nei suoi punti chiave, un sistema software per la composizione musicale assistita che permette al compositore/utilizzatore di rappresentare e descrivere intuitivamente il proprio pensiero musicale in termini di oggetti (sonori e musicali) e azioni su di essi.

Visual Composing System, mediante un uso estensivo di strumenti grafici e un motore real-time di controllo, generazione ed elaborazione audio, permette un feedback continuo tra pensiero compositivo e risultato percettivo.

2 Modello Compositivo

Visual Composing System si basa su un modello compositivo che prende in considerazione tre *scales temporali* sulle quali si attua l'intero processo musicale:

Even rate, relativo alla trasformazione da oggetto a evento sonoro;

Control rate, corrispondente alla soglia di integrazione dell'orecchio, rappresentativa del rate di informazione;

Audio rate, legato alla massima frequenza audio che si vuole ottenere.

Il modello del *processo musicale* prende in considerazione l'insieme di azioni che agiscono sugli oggetti musicali e sonori, sia in termini di definizione e/o scelta dell'insieme di regole che li governano e gestiscono, sia in termini di appli-

cazione nel tempo (costruzione e decifrazione) di tali regole.

Per *oggetto sonoro* si intende "un oggetto (acustico) della percezione umana e non un oggetto matematico o elettroacustico di sintesi" [1] ovvero l'insieme di eventi sonori che da un punto di vista percettivo sono riconducibili alla stessa sorgente.

"(Tale) oggetto è definito dall'orecchio umano come la più piccola particella autonoma di un *paeaggio sonoro* ed è analizzabile secondo il suo profilo" [2], ossia tramite la variabilità dei parametri percettivi caratteristici del suono. "L'oggetto sonoro non va confuso con il corpo che lo ha prodotto, poiché ogni singolo corpo sonoro può produrre una quantità disparata di oggetti sonori, la cui varietà non può essere ricondotta ad una origine comune" [3].

L'*evento sonoro*, invece, è un suono di una data durata, che inizia in un determinato istante, ed è comprensivo dell'evoluzione temporale dei suoi parametri, in tutte e tre le scale dei tempi.

Caratteristiche percettive dell'evento sonoro, sono:

- la sua *evoluzione temporale* (attacco, corpo, caduta),
- *durata, frequenza e dinamica*,
- *massa* (o densità percettiva, corrispondente all'ampiezza di banda predominante),
- *grana* (ovvero fluttuazione interna del suono percepito),
- *trama* (generata dalla grana con effetti di ruvidità della superficie sonora).

Il passaggio da oggetto ad evento sonoro avviene assegnando nel tempo valori ai parametri dell'oggetto sonoro, ossia scegliendo uno degli elementi dell'insieme "oggetto sonoro".

L'*oggetto musicale* può essere considerato come l'insieme di eventi musicali che da un punto di vista percettivo sono riconducibili alla stessa unità morfologica. La descrizione di un oggetto musicale riguarda i parametri percettivi caratteristici della forma, la loro variabilità ed il loro legame con il processo generativo.

L'*evento musicale* è una forma (dalla figura, alla macroforma) strutturata nel tempo, comprensiva dell'evoluzione temporale del suo *profilo*, ed è un elemento dell'insieme "oggetto musicale".

Nella prassi tradizionale, l'evento musicale organizza parametri appartenenti alla scala dei tempi degli eventi sonori (event rate), come un profilo melodico che organizzi nel tempo le altezze (frequenze) degli eventi sonori (note).

Esempi nell'ambito del sistema temperato sono forniti da Mâche nell'analisi della sonata per pianoforte "Appassionata" di Beethoven [4] e da Schaeffer parlando della musica di Webern [5].

Nel caso più generale in cui l'evento musicale organizza parametri appartenenti a tutte e tre le scale dei tempi, esso (l'evento musicale) può coincidere con l'evento sonoro: per esempio una "fascia", caratterizzata da un profilo timbrico organizza nel tempo le relazioni fra le componenti parziali dell'evento sonoro. (Si udrà un solo suono, ma la sua variazione interna sarà percepibile come forma musicale.)

3 Visual Composing System

Il modello compositivo utilizzato rappresenta il *paradigma compositivo* di riferimento.

In questo ambito si è definita una notazione simbolica per rappresentare e descrivere gli elementi di tale modello e le loro interazioni, prendendo come riferimento la notazione di Booch ampiamente utilizzata nel paradigma di programmazione ad oggetti [6].

Il sistema proposto permette, mediante un approccio di tipo "Visual Programming", l'intervento trasparente e coerente ai vari livelli del processo musicale. L'utilizzo della notazione simbolica permette, con un solo linguaggio, di controllare e/o definire elementi alle tre scale temporali con le stesse modalità per tipi di parametro simili (sia come azione, sia come loro rappresentazione).

Questo è possibile grazie a un approccio object-oriented e gerarchico applicato sia alla definizione/costruzione degli oggetti musicali e sonori,

sia alla rappresentazione in partiture simbolica, operativa ed esecutiva delle azioni e delle regole del processo musicale.

Le caratteristiche dell'architettura di Visual Composing System sono state definite in modo da ottenere un sistema:

- *aperto*, al quale si possano aggiungere facilmente nuove funzionalità;
- *scalabile*, in modo da poter essere configurato a vari livelli di complessità/costo;
- *modulare*, ossia scomponibile in unità autonome sia hardware, sia software;
- *portatile* tra piattaforme differenti (Macintosh, Windows, ...) e dispositivi h/w di I/O specifici (MARS [7,8,9], ORION [10], General MIDI, ...), in modo tale che l'utente possa utilizzare Visual Composing System e svilupparne nuove funzionalità sul proprio computer, qualsiasi esso sia;
- basato su un approccio non-monopolistico e che sfrutta le più recenti tecnologie informatiche (OpenDoc, OLE, SOM, ...).

Le caratteristiche di scalabilità, modularità (sia h/w, sia s/w) e di architettura aperta implicano che il sistema possa:

- funzionare in presenza di hardware/software specializzato (come la stazione musicale MARS o il sistema ORION);
- funzionare anche in assenza di un hardware specifico (ossia non real-time, come ad esempio C-Sound);
- interagire con gli standard esistenti e futuri (MIDI, ZIPPI, NIFF, ...);
- essere adattato facilmente dall'utente alle proprie esigenze a tutti i livelli (e scale temporali).

L'ambiente principale di Visual Composing System offre come primo strumento di interazione una pagina di score generalizzato, nella quale vengono manipolati e visualizzati in modo specializzato tutti gli *eventi* gestiti dal sistema (ad esempio tracce MIDI, eventi di trigger o di controllo di oggetti musicali, e campioni sonori). Il sistema notazionale permette la realizzazione di partiture sia in forma *simbolica*, sia in forma *operativa*. Il prodotto finale del sistema è una partitura *esecutiva* che agisce direttamente con i dispositivi di I/O.

Il successivo strumento di interazione permette la creazione di oggetti musicali mediante il patching di *materiali grezzi*, *operatori*, *porte di I/O* e *performer* (tutti elementi della notazione simbolica), e altri oggetti musicali in modo gerarchico. I materiali musicali così realizzati interagiscono direttamente e con diversi livelli di "intelligenza" alle tre scale temporali con i dispositivi di I/O (vedi figura), producendo i dati della partitura esecutiva.

Il sistema verrà costruito attorno ad un motore real-time, disponibile sia su Macintosh, sia su Windows, che permetterà lo sviluppo di applicativi autonomi interagenti tra loro e con Visual Composing System.

4 Conclusioni

In questo articolo è stato presentato un sistema per la composizione musicale assistita che permette al compositore di interagire in modo trasparente, coerente e algoritmico ai vari livelli del processo musicale e alle diverse scale temporali.

Il sistema, sia nella struttura architetturale sia nel suo ambiente, è stato pensato per soddisfare le esigenze di una ampia categoria di applicazioni, comprendenti la composizione musicale, le performance arts, la post-produzione, la didattica musicale e la ricerca. In particolare:

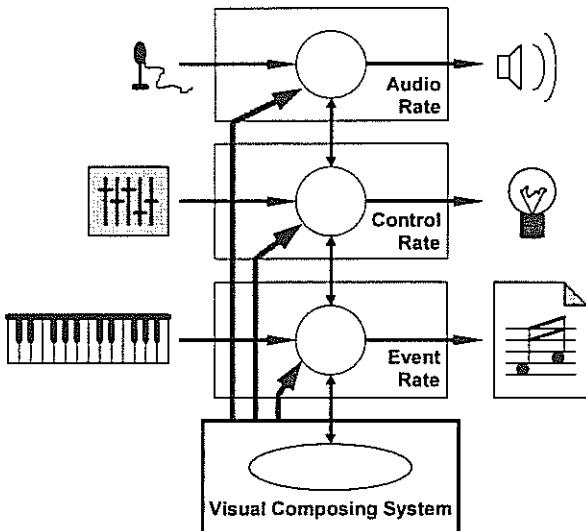
Composizione Musicale: come strumento di progettazione e documentazione sia per la musica colta [11], sia per quella più commerciale (colonne sonore, musica leggera, ...).

Didattica Musicale: in quanto versatile strumento di esplorazione sonora per l'infanzia, e metodologica per l'apprendimento della musica mediante manipolazione di oggetti sonori in corsi superiori (conservatori, università, ...).

Performance Arts: come "processore" real-time in spettacoli di live-electronics o teatro/danza, e progetti multimediali.

Post-Produzione: come "centro di controllo" di sistemi digitali.

Questo progetto, avallato dall'Istituto di Ricerca per l'Industria dello Spettacolo (IRIS), è stato ideato e sviluppato autonomamente dagli autori. Attualmente è in avanzata fase di specifica, ed è previsto per il prossimo anno il completamento della fase prototipale.



- [7] P. Andrenacci, E. Favreau, N. Larosa, A. Prestigiacomo, C. Rosati, S. Sapir: "MARS: RT20M/EDIT20 Development tools and graphical user interface for sound generation board", *Proc. of the ICMC*, S. Jose, 1992, pp. 340-343.
- [8] F. Armani, L. Bizzarri, E. Favreau: "MARS: DSP environment and applications", *Proc. of the ICMC*, S. Jose, 1992, pp. 344-347.
- [9] P. Andrenacci, F. Armani, A. Prestigiacomo, C. Rosati: "APPLI20: a development tool for building MARS application with an easy to use graphical interface", *Atti del X Colloquio di Informatica Musicale*, Milano, 1993, pp. 277-285.
- [10] G. Nottoli: "ORION: un sistema per la sintesi e l'elaborazione del suono integrato in un singolo chip", *Atti del IX Colloquio di Informatica Musicale*, Genova, 1991, pp. 220-230.
- [11] R. Santoboni, A. Paladin: "Apparizioni fugitive: una composizione realizzata sulla MARS con sintesi per modelli fisici", *Atti del CIARM, II conferenza internazionale di acustica e ricerca musicale*, Ferrara, 1995, pp. 127-132.

Bibliografia

- [1] P. Schaeffer: "La musique et l'ordinateur", *Musique et tecnologie*, Paris, 1970, pag. 84.
- [2] R. Murray Shafer: *Il paesaggio sonoro*, Ricordi, 1985, pag. 181.
- [3] P. Schaeffer: *Trois microsillons d'exemples sonores*, Paris, 1967.
- [4] J. J. Nattiez: *Il discorso musicale*, PBE, 1977, pag. 29.
- [5] M. Pierret: *Entretiens avec Pierre Schaeffer*, Belfond, Paris, 1969, pp. 54-55.
- [6] G. Booch: *Object-Oriented Analysis and Design with Applications*, 2nd ed., Benjamin/Cummings, Redwood City, 1994.

Contatti

IRIS s.r.l.
Parco La Selva, 151
03018 Paliano (FR), Italy
Tel. +39 (0)775 533 441
Fax +39 (0)775 533 343
e-mail: mc2842@mclink.it

Fuzzy Control of Sound Synthesis Parameters

Piergiorgio Sartor, Alvise Vidolin
CSC-DEI, University of Padova. Italy

1 Introduction

Synthesis algorithms require many control parameters which are directly related to the physical or numerical variables of the sonological model. The musician usually controls the sound manipulating few parameters musically expressive and often connected to the sound perception features.

Usually an electronic synthesizer has predefined relations between the inputs (eg. frequency and intensity) and the sounds parameters of the outputs (eg. the amplitude of the note); real instruments are quite different. Analyzing the sounds of this ones, with different frequencies and intensities of the note, it was seen how the spectrum is full of partials in the "grave" register and it becomes simpler to the "high" one. The spectrum also depends on the intensity: a *FORTE* sound has more partials than a *PIANO* one, and also a *FORTE* sound has a shorter attack time than a *PIANO* one.

Fuzzy theory can be a solution to this problem, because it can be seen as a nonlinear interpolator, but trimmed with few, simply rules given by an expert.

2 What is Fuzzy Logic?

Fuzzy Logic is an extension of the conventional (boolean) logic, which lets to deal with the concept of "partial truth", that is values of truth between "full true" and "full false". In classical logic a variable can be 1 or 0, that is it indicates "true" or "false", in Fuzzy Logic a variable can get values between 0 and 1, so it associates *fuzzies* of truth.

Classical logic operation become:

$\text{NOT } x: 1 - x$

$x \text{ AND } y: \min(x, y)$ or $x \cdot y$

$x \text{ OR } y: \max(x, y)$ or $x + y$ (with precautions)

There are other definitions for the logical operations, but this are the most successful ones.

As written in [3] we have to focus our attention on the "fuzzification" problem, that is how to transform "discrete" logic into "continuos" logic.

Classical algorithm is:

The input and output variables are defined.

The *UNIVERSE OF DISCOURSE* is defined, that is the range of the input and output variables, eg. *Frequency* from 100 to 780 Hz.

The "linguistic variables" are defined, that is the input/output variables sets are divided and

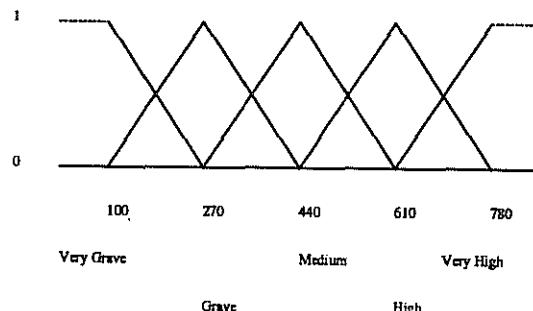


Figure 1: Fuzzy set of the input frequency.

named, eg. *HIGH*, *MEDIUM*, *GRAVE*.

The fuzzy sets of the linguistic variables are defined. For example *GRAVE* from 100 to 440 Hz, usually it is 1 in 270 Hz, it is 0 in 100 and 440 Hz. It rises and falls linearly from 100 to 270 and from 270 to 440 (Fig. 1). Fuzzy sets must overlap for proper operations.

For each input variable it is associated its fuzzy value to the linguistic variables. Eg., from previous point, if *Frequency* is 168 Hz then *GRAVE* is set to 0.4. This operation is repeated for each linguistic variables.

The rules (given by an expert) are executed mapping the input into the output (see "inference" in [1]), which is a fuzzy set; eg. "if *Frequency* = *GRAVE* then *Amplitude* = *MEDIUM*", if *GRAVE* is 0.67 then *MEDIUM* is set to 0.67.

The output value is calculated using the center mass method or the maximum method.

As all technology Fuzzy Logic has its advantages and disadvantages. Advantages are: better control, robustness, "linguistic" approach, that is the control is trimmed using a natural language (*HIGH*, *GRAVE*, *PIANO*, *FORTE*). Disadvantages are: more computational load, it needs an expert which defines the sets and the rules.

Fuzzy Logic can be seen as a nonlinear interpolator, and so it can be useful for the control of sound synthesis parameters.

3 A sonological model

As an example of this approach it was chosen to control the synthesis parameters of a percussion instrument realized using the additive technique.

Each partial has the following features.

Amplitude envelope with exponential attack

transitory (variable), short sustain time of 0.05s (constant) and exponential decay transitory (variable), with a 60dB dynamics.

Frequency (usually with harmonic law), which can be shifted and compressed or expanded, follows the law: $f_i = f_0(t+i^\alpha)$, where i is the number of the partial, f_0 is the fundamental frequency, t is the shift factor ($t = 0$ harmonic law), α is the compression or expansion factor ($\alpha = 1$ harmonic law, $0 < \alpha < 1$ compression, $\alpha > 1$ expansion).

The duration of each partial is fixed by the decay time which follows the law: $\frac{c}{f_i}$, with $c = 1000\text{Hz}$ and f_i = frequency of the i -th partial. The total number of partials is ruled by a lowpass filter with one real pole in $f_t = 500\text{Hz}$ (so with a slope of -20dB/dec). Partials with less than -60dB are not considered.

4 The Fuzzy controller

The attack time and the maximum amplitude are the outputs of the Fuzzy controller which has as inputs the frequency and intensity of the note.

Input and output sets are divided into 5 linguistic variables, this simplifies the software without performance losses. In figure 1 is reported the fuzzy set of the input frequency, the others are similar.

The rules which map the input into the output can be given as tables, as an example the rule:

if Frequency = HIGH AND Intensity = FORTE then Amplitude = MEDIUM

can be seen as a function:

$$f(HIGH, FORTE) \rightarrow MEDIUM$$

and so as a map from the sets *Frequency*, *Intensity* to the set *Amplitude*, this map can be implemented as a table.

The algorithm will be: for each pair of input linguistic variables an AND operation (as product) is performed and the result is assigned to the linguistic variable associated by the table, taking, as final result, the maximum between all the calculated values (see [2]), that is if more than

pp	pp	p	pp	pp
p	p	m	p	p
m	m	f	m	p
f	f	ff	m	m
f	ff	ff	f	m
Vg	g	m	h	Vh
Frequency				

Table 1: Rules map for the output amplitude

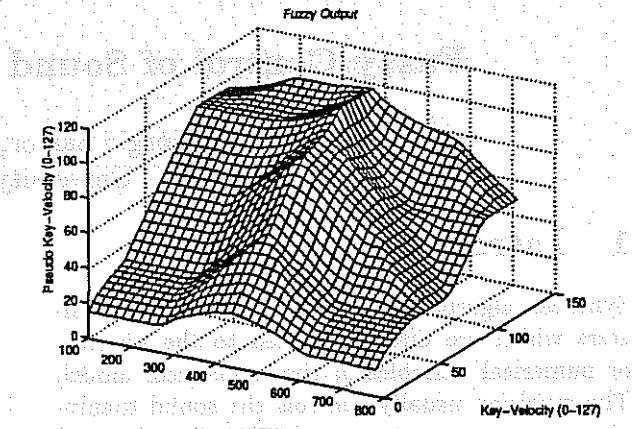


Figure 2: Controller output: Amplitude

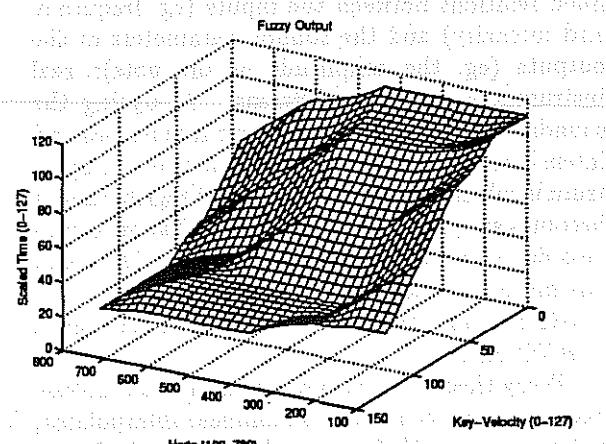


Figure 3: Controller output: *Attack time*.

one pair (*Frequency*, *Intensity*) is mapped into the same linguistic variable only the maximum result is taken.

As an example the rules for the set *amplitude* are reported in table 1. The controller outputs are in figure 2 and figure 3, where it can be appreciated how the rules table is mapped on a surface and how the interpolator works.

A possible program output is a CSound score. The system results were suitable to the expectations.

References

- [1] Y.F. Li and C.C. Lau. Development of fuzzy algorithms for servo systems. *IEEE Control Systems Magazine*, pag. 65-71, April 1989.
 - [2] W. Pedrycz. *Fuzzy Control and Fuzzy systems*. John Wiley & Sons Inc., 1989.
 - [3] Zadeh and Lotfi. Outline of a new approach to the analysis of complex systems. *IEEE Trans. on Sys. Man and Cyb.* 3, 1973.

Un'indagine sulle regole preferenziali di raggruppamento di Lerdahl e Jackendoff

Paolo Principi

Laboratorio di Informatica musicale del Dipartimento Musica e Spettacolo

Università degli Studi di Bologna

Tel/Fax: +39 71 883075/889850 E-mail: paoloprincipi@fastnet.it

Abstract

This paper describes a software tool (thesis at University of Bologna, referee prof. Mario Baroni) for the analysis of melodic lines based on the Lerdahl and Jackendoff's Grouping Preference Rules. Our approach, that we call "interactive", allows observation and control of the varius phases of the music analysis: the recognition of all potential group boundaries, the management of rules conflicts and the final segmentation.

With the experimentation on a large and heterogeneous melodic repertory we can: check the correctness of the software functioning; verify the possibility to apply the rules in different melodic contexts; attempt to compile, by the analysis results, a domain's hierachycal scale of the rules, as well as extrapolate same critical points of the rule system. The software, named IMAP (Interactive Melodic Analysis Processor) has been implemented in C language, running on Macintosh system.

1 Introduzione

L'idea alla base del lavoro trae origine da una considerazione di carattere teorico-metodologico che troviamo nel testo "A generative theory of tonal music" (d'ora in poi GTTM). Nel descrivere la teoria, gli autori motivano la scelta di un formalismo espresso in un linguaggio ordinario, non matematico, con la impossibilità di quest'ultimo di produrre una analisi definita. Nel caso infatti di ambiguità di intuizione nel flusso musicale, cioè di un conflitto tra regole, la soluzione cadrebbe nella regola "numericamente" più forte (dopo aver redatto ovviamente una scala gerarchica di dominio tra le regole stesse). Il risultato ottenuto sarebbe quello di una analisi apparentemente corretta ma che non consente l'individuazione e lo studio dei casi di dubbia interpretazione (sicuramente i più interessanti!). Per questo motivo Lerdahl e Jackendoff manifestano tutta la loro perplessità circa la possibilità di implementare la teoria in un software per calcolatore.

Considerando questo aspetto di fondamentale importanza teorica abbiamo realizzato uno strumento in grado di consentire un'analisi del testo a fasi distinte, di mostrare cioè i singoli passaggi con possibilità di intervento da parte dell'utente.

2 Il Sistema

Per quanto concerne la sua struttura globale, il sistema è articolato in cinque blocchi funzionali, ciascuno dei quali in grado di svolgere un determinato compito nell'analisi.

Editor: permette la immissione e/o la modifica dei dati in ingresso grazie ad un codice studiato appositamente per rappresentare la melodia in un formato alfanumerico.

Analyser: è il primo motore dell'analisi; applica a tutta la melodia l'insieme delle regole implementate

al fine di individuare i potenziali confini tra gruppi. Si ottiene in uscita la melodia con tutte le regole individuate, indipendentemente dal fatto che siano poi confermate. La conferma si ha con i successivi due blocchi del programma.

AutoGrouper: In questo blocco funzionale avviene una prima ricerca, sui dati ottenuti dall'analyser, mirata all'individuazione delle regole poco significative, regole che, secondo la teoria, dovrebbero esser soppresse perché dominate da altre adiacenti aventi forza di applicabilità maggiore. L'eliminazione di alcune regole, in modo automatico, avviene solo nel momento in cui non ci troviamo di fronte a casi di dubbia interpretazione.

InteractiveGrouper: qualora dovessero presentarsi delle situazioni in cui la scelta di sopprimere una regola sia difficile, il sistema permette l'individuazione del caso e la possibilità di scelta manuale del confine. Si possono così avere più ipotesi di risultati da uno stesso brano analizzato, dando la precedenza ora ad una regola, ora ad un'altra.

Output Monitor: è lo strumento che visualizza i risultati parziali dell'Analyser e dell'AutoGrouper, nonché quelli finali dell'InteractiveGrouper.

3 L'analisi

Il corpus di melodie scelto come campione è costituito da semplici frammenti di temi tratti da una vasta letteratura musicale che spazia dalla musica del rinascimento italiano a compositori del nostro secolo come Schoenberg ed Hindemith. Lo scopo è quello da un lato di rappresentare la massima varietà di situazioni microstrutturali, dall'altro di analizzare repertori molto diversi tra loro. Nella teoria GTTM infatti, la struttura di raggruppamento è descritta come quella, tra le rappresentazioni gerarchiche, con minori vincoli di dipendenza da uno specifico idioma musicale. Il secondo obiettivo è quindi quello di verificare l'applicabilità delle regole in repertori non

contemplati nel testo di Lerdahl e Jackendoff. Da una ulteriore analisi dei risultati possono altresì scaturire alcune considerazioni sulla gerarchia di dominio delle regole. Crediamo infatti che l'efficacia delle regole sia in stretta relazione non solo con la maggiore o minore rispondenza alle intuizioni musicali, ma anche con il numero delle correzioni cui si è costretti a ricorrere al fine di risolvere situazioni di conflitto. Un basso numero di correzioni (cancellazioni di regole individuate nella prima fase dall'Analyser) sarà indice di buona validità delle regole e viceversa. I frammenti tematici analizzati sono stati suddivisi, per le ragioni accennate, in GTTM (gli stessi esempi presenti nel testo originale), Classici, Pre-classici e Post-classici.

4 I risultati

Le analisi sui corpus dei frammenti melodici sono state classificate, in base al numero delle correzioni effettuate, in: A) corrette (meno del 25% di correzioni); B) mediamente corrette (tra il 25 e il 45%) e C) non corrette (superiore al 45%). Il risultato generale della tripartizione, mantenendo la suddivisione nei quattro repertori proposti, è quello rappresentato in tabella.

Esempi	tipo A	tipo B	tipo C
GTIM	5	2	2
Classici	6	3	0
Pre-classici	2	3	3
Post-classici	5	3	3
Generale	18	11	8

La tendenza generale, indicata nell'ultima riga, denota un indice di validità del sistema di regole pari circa al 50% delle melodie analizzate (18 su 37). Meno di un terzo del totale ha bisogno di interventi di correzione, mentre 8 melodie sembrano segmentate non correttamente. I margini di differenza tra i quattro repertori sono di lieve entità; non crediamo quindi sia possibile una rigida delimitazione cronologica della validità del sistema teorico. Tuttavia la tendenziale maggior correttezza delle analisi nei repertori GTTM e Classico ci sembra da non sottovalutare. Da un'indagine più particolareggiata, volta alla osservazione delle singole regole, è possibile trarre indicazioni sulla ipotetica scala gerarchica di dominio delle regole in base al rapporto tra numero di occorrenze e numero di cancellazioni di ciascuna regola. Ordinando le regole in base alla minor percentuale di errore avremmo il seguente risultato:

regola	n.occurred	n.cancelled	% errore
3b dinamica	15	1	6.6%
3c articolazione	15	3	20.0%
2a pausa	106	22	20.7%
2b attacco	125	28	22.4%
3a registro	170	70	41.1%
3d lunghezza	36	16	44.4%

Si può notare nella precedente tabella come le sei regole di dettaglio locale implementate possano essere raggruppate in tre categorie: regole che riguardano rispettivamente l'espressione (3b, 3c) le durate (2a, 2b, 3d) e le altezze (3a). L'analisi dei singoli casi tuttavia denota, a conferma della nostra impostazione metodologica, come la gerarchia non sia sempre rispettata.

La perplessità, emersa nella fase di analisi, circa la non corretta formulazione della regola del cambio di registro (3a) risulta fondata constatato l'alto numero sia delle occorrenze sia delle cancellazioni. È questo uno degli aspetti da prendere in considerazione per gli sviluppi futuri del lavoro, insieme alla rivalutazione della questione delle regole di ampio livello (intensificazione, parallelismo e simmetria) non contemplate in questa prima fase del lavoro.

Ringraziamenti

È nostro desiderio ringraziare sentitamente, per la loro disponibilità nel fornire materiale accompagnato sempre da utili consigli, i gentilissimi: prof. Mario Baroni (Università di Bologna), prof. Lelio Camilleri (Conservatorio di Bologna), dott. Francesco Carreras (CNUCE/C.N.R. di Pisa), prof. Goffredo Haus (Università di Milano).

References

- M. Baroni, C. Jacoboni: "Verso una grammatica della melodia", Bologna, 1976.
- L. Camilleri, F. Carreras, C. Duranti: "An expert system prototype for the study of musical segmentation", *Interface* 19 (23), 1990.
- N. Chomsky: "Syntactic structures", Den Haag, Mouton (trad. it., *Le strutture della sintassi*, Bari, Laterza 1970), 1957.
- F. Giomi: "Applicazioni di tecniche computazionali all'analisi musicale", tesi di Laurea, Università di Pisa, 1992.
- F. Lerdahl, R. Jackendoff: "A generative theory of tonal music", Cambridge, 1983.
- F. Lerdahl, R. Jackendoff: "An overview of hierarchical structure in music" (trad. it. "Grammatica generativa e analisi" in *Il senso in musica* a cura di L. Marconi e G. Stefani, Bologna), 1987.
- P. Principi: "Analisi musicale e computer: un'indagine sulle regole preferenziali di raggruppamento di Lerdahl e Jackendoff", tesi di Laurea, Università di Bologna, 1985.
- Roads C.: "Grammars as representations for music", *Computer Music Journal*, 1979.

Graphical Display of Musical Information

Douglas J. E. Nunn^φ Alan Purvis^φ Peter D. Manning[#]

Durham Music Technology
^φSchool of Engineering / [#]Department of Music
Durham University
Durham, UK

Abstract

This paper examines the graphical display of musical audio, and gives examples of graphical outputs used in developing a music analysis system. It is argued that graphical display is a vital tool for analysis and also has much to offer for synthesis.

1 Introduction

Music has long relied on a wide range of visual representations. Standard notation uses a symbolic representation of notes having three primary attributes - onset, pitch, and duration. Conventionally, instruments do not change during the performance, so the global timbre is specified simply by writing the name of the instrument on the first page, and local timbral inflections are specified by symbols such as 'staccato', 'down bow', and 'stopped'. However, it has long been recognised that such notation cannot fully define the timbre or timbral nuances of the sound.

These drawbacks are also apparent in computer music, where composers wish to use an unrestricted musical vocabulary. Graphical display is a practical alternative to symbolic notations for the characterisation and specification of complex sounds. Computer analysis allows us to examine musical information in microscopic detail, and graphical display is the most useful method of presenting the large amount of data. Composers of electroacoustic music typically use complex control methods that cannot be reduced to a simple symbolic language, and graphical representations have proved to be powerful tools (see, for example, [Horry]). A graphical representation of audio thus has many potential applications to both analysis and synthesis.

2 System Overview

The graphical displays described below were developed as parts of a hybrid system for analysis and transcription, described more fully in [Nunn]. A common way to analyse and visualise sound is to examine or plot the magnitude-squared STFTs. However, this gives bins equally spaced in frequency, whereas the characteristic of interest is

log-frequency. In addition, this gives insufficient resolution at the bass end of the spectrum. A practical solution is to use a multirate system, where the sample rate is halved every octave. As a result, the spectra are more clearly defined. The multirate filtering and FFTs are implemented on the Texas Instruments TMS320C40, a high-performance DSP chip, but subsequent processing stages run on a standalone PC.

3 Graphical Display

Various graphical outputs were developed as part of the analysis system, and were found to be essential, both for presenting the results and for debugging the system itself. A typical spectrum is shown in Figure 1. Note that the fundamentals of the bass notes are quite clearly defined.

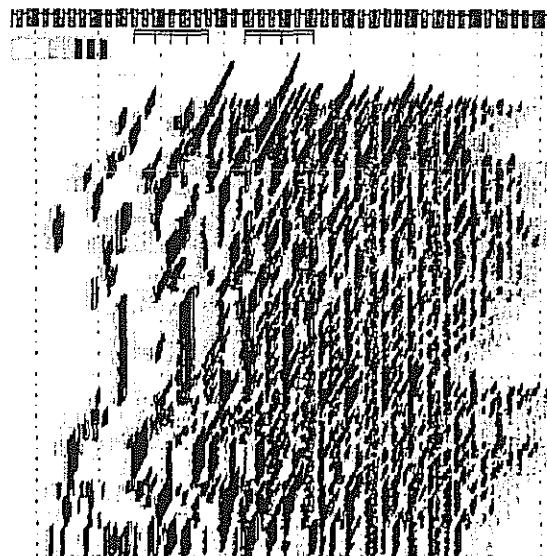


Figure 1 - Logarithmic spectrogram showing approximately 30 seconds of Mendelssohn's Sonata 3 for Organ. The horizontal axis is log-frequency; the vertical axis is time, from top to bottom.

Here, half a minute of sound is displayed on the screen, but a more typical use is to expand the time axis by scrolling the display. Two axes are needed for time and log-frequency, so amplitude must be plotted using a third 'dimension'. This is done both by using the colour of the block to denote the logarithm of the amplitude, and by drawing the blocks three-dimensionally, with the 'height' proportional to the amplitude of the transform bin.

The graphical displays operate several times slower than real time, making it difficult to relate the visual display to the sound. The solution adopted is to capture a large set of individual frames and then convert them into a movie format. The movie can then be played at the maximum speed permitted by the graphics hardware. For scrolling 640*480*256 graphics, this is typically 3-4 frames per second, which is jerky but still effective. This animation technique has also proved useful for examining the effects of varying the analysis parameters.

The constituent partial lines in Figure 1 are apparent to us, due to our innate ability to look at a series of points and see a line. For the task of transcription, further processing picks out these lines, which can also be displayed graphically. We have now abstracted the data from samples to short sinusoids to long chains, but the target entities are the notes of individual instruments. The process of extracting parallel entities from a single data stream is the most complex. The final stages attempt to form notes by grouping together chains that are harmonically related, and removing notes that are too short or too weak, or have insufficient harmonics.

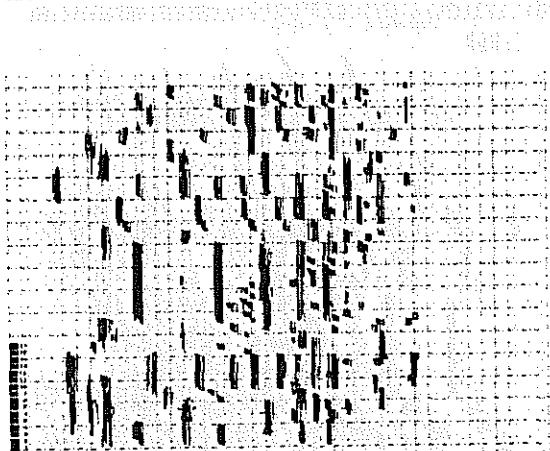


Figure 2 - Representation of individual notes extracted from the Mendelssohn example.

Figure 2 depicts the notes thus formed. The amplitude envelopes of the partials are plotted together at the position corresponding to the

fundamental. This time colours are used to distinguish different harmonics.

4 Discussion

Like computer music, computer graphics demands high processing performance, forcing trade-offs between accuracy and speed. Hardware advances have always been countered by increasing demands for processing power. Both disciplines require the manipulation of large complex objects, and in both fields the user interface is considered to be the most critical part of the system.

In developing the transcription system, it was found that graphical output was vital as a method of presenting the large amount of data, and also offered a method of creating crude real-time animations.

For analysis, graphical display offers the ability to depict attributes of the sound in a more intuitive manner. However, it also forces us to ask the difficult question: what does 'the sound of a trombone' look like? For synthesis and composition, graphical display allows an easily understandable interface. Perhaps most importantly, graphical representations could facilitate the linking of analysis to resynthesis, with the intervening transformations being carried out in the graphical domain.

Acknowledgements

The author (DJEN) has been funded by a research studentship from the Engineering and Physical Sciences Research Council.

References

- Y. Horry: "A Graphical User Interface for MIDI Signal Generation and Sound Synthesis", *Proceedings of the International Computer Music Conference*, Aarhus, 1994.
- D. J. E. Nunn, A. Purvis, and P. D. Manning: "Source Separation and Transcription of Polyphonic Music", *Proceedings of the International Conference on New Music Research*, Gent, 1994.

Profile

libreria per il controllo del profilo melodico

Mikhail MALT
m.malt@ircam.fr

Jacopo BABONI SCHILINGI
baboni@ircam.fr
IRCAM
1, Place Igor Stravinsky
75004 Paris
France

Questo lavoro rappresenta la prima parte di un nuovo progetto dove l'idea della trasformazione geometrica viene applicata alle note musicali. In altre parole abbiamo considerato il profilo melodico come un parametro indipendente nel processo compositivo di un brano musicale.

Introduzione

Nato come prima parte di un nuovo progetto per lo sviluppo del materiale musicale, Profile rappresenta l'idea della trasformazione geometrica applicata alle note musicali. Fra le caratteristiche percettive di una "melodia" (intesa come successione lineare di note nel tempo) il profilo musicale può essere considerato come uno dei parametri più importanti sia per la memorizzazione di una entità musicale, sia per la rappresentazione grafica di una idea musicale, sia per la "riconoscibilità" di un andamento melodico. Nel nostro caso specifico, abbiamo considerato il profilo melodico come una successione lineare di direzioni di intervalli musicali, equidistanti dal punto di vista temporale, rappresentata graficamente da una curva per segmenti (figura 1). Come diretta conseguenza di una concezione parametrica della composizione musicale, di chiara derivazione seriale, attualmente diversi compositori¹ hanno utilizzato l'idea di profilo come elemento indipendente dagli altri parametri musicali. In questo modo viene conferito al profilo un ruolo ed una importanza specifica per la composizione stessa delle opere musicali. In altre parole il profilo viene considerato parametro indipendente nel processo compositivo di un brano musicale.

Infine, Profile permette di utilizzare questo parametro sia come materiale di base, sia come processo compositivo, sia come elemento per controllare l'evoluzione di altri processi di trasformazione musicale.

La libreria Profile è interamente scritta in Common-Lisp nell'ambiente informatico PatchWork. Il programma PatchWork è stato concepito da M.Laurson e sviluppato all'IRCAM di Parigi da C.Rueda e J. Duthez per la composizione assistita dal calcolatore. Profile è un'estensione delle funzionalità di PatchWork applicate, in modo specifico, al profilo musicale.

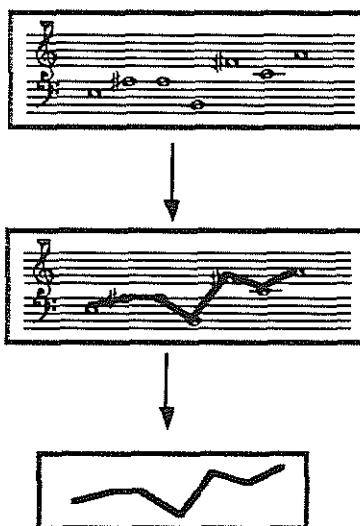


Figura 1

Obiettivi della libreria Profile

Questa libreria è stata concepita con lo scopo di rendere il controllo e la generazione del materiale compositivo il più "musicale" possibile. Inoltre, si è cercato di proporre alcuni processi di trattamento del profilo melodico, generalizzando, il più possibile, i meccanismi di azione sui profili musicali. Dal punto di vista pratico, l'utilizzatore può controllare direttamente: le direzioni degli intervalli di un profilo, gli intervalli stessi, le altezze assolute, sia in senso orizzontale che verticale (controllo armonico), l'andamento globale di ogni processo attraverso dei grafici e, infine, il grado di profondità di un processo attraverso la ricorsione e la iterazione, implementate direttamente nelle funzioni.

Struttura della libreria Profile

Profile è suddiviso in sei famiglie di funzioni ognuna delle quali permette un controllo specifico sulla manipolazione del profilo melodico.

¹Baboni-Schilingi, Brian-Ferneyhough, Fedele, Malt, Manoury, Melchiorre, Murail .

- 1) Gruppo di funzioni che opera un'azione elementare di perturbazione di un profilo melodico: la funzione *alea-perfb* perturba aleatoriamente un profilo e *compor/expan* comprime o espande gli intervalli di un profilo melodico.
 - 2) Gruppo di funzioni che permette di fare interagire due profili tra di loro: *control-perfb* permette di distorcere un profilo A in funzione di un altro profilo B; *prof-change* permette di incrociare i parametri di due profili differenti attribuendo la "forma" di un profilo B sul materiale di un profilo A.
 - 3) Gruppo di funzioni che applica delle operazioni di simmetria su un profilo: *reflexion* permette il ribaltamento di un profilo in funzione di un asse orizzontale; *double-reflexion* permette un doppio ribaltamento di un profilo in funzione di due assi orizzontali; *multi-reflexion* permette una riflessione multipla che conserva le direzioni del profilo di partenza.
 - 4) Gruppo di funzioni che permette di semplificare o di rendere più complesso un profilo musicale: *mean-derivation* semplifica un profilo attraverso la media di due intervalli contigui; *interlock* è la funzione complementare di *mean-derivation* e trascrizione musicale dell'algoritmo della generazione dei frattali chiamato *midpoint-displacement*; *derivation* è la traduzione musicale della derivata matematica; *integration* è la funzione complementare di *derivation*.
 - 5) Gruppo di funzioni (*inter-dyn*, *multi-interpol* e *interpol-prof*) per il controllo dinamico dell'interpolazione fra due profili musicali, con il controllo di tutti i parametri dell'interpolazione: numero dei passi dell'interpolazione, controllo del numero di elementi per ogni passo, controllo dell'evoluzione del processo attraverso grafici, controllo locale e globale del contesto armonico dell'interpolazione.
 - 6) Gruppo di funzioni di utilità di base.
- Conclusione**
- Durante la realizzazione di questo lavoro, che è servito come laboratorio di ricerca per la sperimentazione di alcuni principi di trattamento del materiale musicale, abbiamo potuto verificare la validità effettiva dell'idea di rappresentare e manipolare graficamente il profilo melodico. Un esempio è dato dal problema dell'interpolazione, che fino ad oggi è sempre rimasta la mera trasposizione di una procedura grafica e geometrica verso il mondo musicale.

Infatti, abbiamo potuto verificare che, musicalmente parlando, il percorso armonico risultante da una semplice interpolazione, non era isomorfo alla rappresentazione grafica. Una delle soluzioni proposte da Profile è quella di poter controllare esattamente il percorso armonico per ogni passo dell'interpolazione. Inoltre, la possibilità di controllare graficamente l'andamento di un intero processo d'interpolazione, dimostra che un profilo geometrico può essere lui stesso un processo musicale. Abbiamo potuto constatare, infine, che la potenzialità specifica di Profile risiede nel fatto di poter controllare, manipolare e strutturare, nel dominio della composizione, un intero processo a partire da uno degli elementi più semplici della rappresentazione simbolica: il profilo.

Riferimenti bibliografici

- BABONI SCHILINGI J.** (1995) Composizione per Modelli Interattivi, di prosima edizione
- BARRIERE, Jean Baptiste** (1984) "CHRÉODE I: chemins vers une nouvelle musique avec ordinateur", in L'IRCAM: Une pensée Musicale. Éditions des archives contemporaines, Paris.
- KANDINSKY W.** (1968) Punto, linea, superficie, Adelphi 16.
- MALT M.** (1993) - Introduction à Patchwork, IRCAM, Paris.
- MALT M., BABONI SCHILINGI J.** (1995) Profile- libreria per il controllo del profilo melodico per Patchwork, manuel de référence, IRCAM, Paris.
- Mc ADAMS, Steve et A. Bregman** (1987) "L'audition des flux musicaux", in Marsyas, Institut de pédagogie musicale et choréographique, La Villette, Paris (3-4) décembre 1987, PP 97-118.
- MURAIL T.** (1984) - "Spectes et Lutins" in L'IRCAM: Une pensée Musicale. Éditions des archives contemporaines, Paris.
- MURAIL T.** (1989) - "Questions de cible" in Entretemps n° 8, Paris, France.
- SAARIAHO K.** (1991) - "Timbre et Harmonie", in Le Timbre. Métaphore pour la Composition, Jean-Baptiste Barrière Editeur, Christian Bourgois Éditeur, IRCAM , Paris.

HyperCSound: ambiente grafico di ausilio alla sintesi del suono ed alla generazione di eventi sonori complessi

A. Belladonna

Conservatorio B. Marcello
Venezia - Italy

Abstract

HyperCSound is a group of HyperCard™ Stacks developed for create complex sounds in several kinds of sound synthesis. Each stack allows to define several sound parameters via an intuitive graphic user interface, and subsequently, to create the CSound text files (orchestra and score) necessary for the sound samples creation. Basically, HyperCSound was developed only as an "help-user tool" to simply describe and compose complex sounds and, except some parts, (see later) isn't a tool for musical composition.

1 Introduzione

HyperCSound è una serie di Stack di HyperCard™, ambiente di sviluppo e di authoring su piattaforma Apple Macintosh. HyperCSound è stato realizzato per consentire un comodo approccio grafico e gestuale ad alcune tecniche di sintesi del suono (in particolare additiva, sottrattiva e modulazione di frequenza) nell'impiego del linguaggio di sintesi CSound^[1]. In pratica, le varie componenti dell'ambiente HyperCSound si occupano, una volta impostati i parametri mediante un'interfaccia utente estremamente intuitiva, di generare i file di testo di "orchestra" e "partitura", file che poi saranno processati con il programma CSound per generare i file di campioni.

Essenzialmente, il Sistema HyperCSound è stato creato per aiutare nel processo della descrizione e composizione di suoni complessi, e, ad eccezione di alcune parti (vedi in seguito), non costituisce un ausilio per la formalizzazione di processi compositivi.

2 Perché HyperCard™

Le ragioni per cui è stato scelto l'ambiente HyperCard™ sono diverse e riassumibili nei seguenti punti.

1) l'ambiente HyperCard™ consente di lavorare direttamente "all'interno" dell'interfaccia utente, disegnando quest'ultima nel modo più idoneo e aggiungendo tutte le funzionalità al momento opportuno senza dover riprogettare l'intero programma nel caso di cambiamenti anche sostanziali.

2) HyperCard™ permette una programmazione procedurale estremamente flessibile utilizzando un linguaggio interno (HyperTalk™)^[2] molto intuitivo ma nello stesso tempo ricco di comandi e funzioni.

3) La notevole perdita di prestazioni computazionali nell'utilizzo di un linguaggio di tipo

"interpretato" quale HyperTalk™, in luogo di un linguaggio in grado di generare applicazioni compilate, non rappresenta in questo caso problema alcuno, in quanto i file creati da HyperCSound devono essere successivamente processati dal CSound, operazione questa che annullerebbe tutti i benefici di una maggior velocità nella creazione dei file di testo da sintetizzare.

4) Ho calcolato che la realizzazione in C o Pascal dell'intero lavoro mi sarebbe costata un tempo circa 5 o 6 volte superiore, senza tenere conto che le eventuali successive modifiche alle funzionalità e/o all'interfaccia avrebbero comportato dispendi di tempo notevoli e non commisurati allo scopo del lavoro, essendo HyperCSound un costante "work in progress".

3 Le componenti di HyperCSound

Attualmente il sistema HyperCSound è formato dai componenti sottoelencati.

- HCS Shell: palette con pulsanti per il lancio dei programmi elencati a seguito: permette di accedere immediatamente a tutte le componenti del sistema.

- HyperSpectra: tool per la sintesi additiva che consente di passare dinamicamente tra diversi spettri con frequenze ed ampiezze variabili. HyperSpectra genera suoni dinamici nei quali si passa da uno spettro all'altro nel tempo desiderato mediante interpolazione lineare dei valori di ampiezze e/o di frequenze delle diverse componenti sinusoidali.

- HyperZFilter: tool per la sintesi sottrattiva che permette di passare dinamicamente tra diversi banchi di filtri composti ognuno da 32 filtri passa banda con frequenze centrali e larghezze di banda variabili. L'utente può configurare fino a 16 diversi banchi aventi ognuno proprie frequenze centrali e larghezze di banda. Per ogni filtro vengono definiti graficamente sia il livello di uscita (equalizzazione) che la larghezza di banda, mentre le frequenze centrali vengono generate con le stesse leggi viste

in HyperSpectra. HyperZFilter filtra rumore bianco, treni di impulsi o file audio in input; permette di generare suoni dinamici nei quali si passa da un banco all'altro (morphing) nel tempo desiderato mediante interpolazione lineare dei valori di frequenze centrali e/o larghezze di banda.

- HyperBuildFM: tool per la sintesi in Modulazione di Frequenza che consente la creazione di eventi sonori formati da 1 a 16 blocchi FM semplice in parallelo, ognuno con parametri di f_0 , n_1 , n_2 e indice di modulazione indipendenti. Nella pagina dedicata alla configurazione dei blocchi FM, l'utente può impostare i parametri relativi ad ogni oscillatore nonché il livello di uscita di ogni blocco FM tramite una interfaccia grafica estremamente intuitiva; la costruzione dell'evento sonoro vero e proprio invece avviene in una apposita finestra di tipo "arrange", nella quale l'utente disegna ogni sotto-evento generato dai singoli blocchi FM. Gli inviluppi di ampiezza e di indice di modulazione vengono scelti sempre graficamente attraverso una serie di template.
- HyperGen07: tool editor / librarian per la creazione e memorizzazione di funzioni a segmenti di retta (routine GEN 07 del CSound). Le funzioni vengono disegnate con il mouse previa definizione del range dei valori delle ascisse e delle ordinate; mentre la funzione viene disegnata si generano automaticamente i valori opportuni.
- HyperOrcLib: data-base per la creazione, memorizzazione ed editing di file <orc> di CSound.
- HyperScoLib: data-base per la creazione, memorizzazione ed editing di file <sco> di CSound.
- HyperGenLib: data-base per la creazione, memorizzazione ed editing di routine "GEN" di CSound.

4 Paint Composer

Paint Composer è l'unica parte del sistema HyperCSound che può costituire un ausilio per la formalizzazione di processi compositivi. Paint Composer traduce in partiture CSound eventi grafici di tipo "bitmap" creati liberamente dall'utente oppure generati algoritmicamente con gli appositi moduli. Per la traduzione dei pixel in eventi di nota è stato necessario programmare in Pascal una funzione esterna (XFNC)^[3] in grado di "far riconoscere" ad HyperCardTM lo stato di un pixel sullo schermo.

Il sistema Paint Composer è attualmente composto dai moduli sottoelencati.

- Paint Composer Heart: cuore del sistema, è il programma che si occupa di creare le partiture CSound scansionando gli eventi grafici presenti in un'area dedicata; tali eventi grafici vengono tradotti in note CSound nel dominio tempo/frequenza. L'utente può impostare sia il range del tempo che quella frequenziale in modo da decidere come gli eventi grafici debbano essere decodificati in istruzioni di nota. La generazione sonora utilizza un

semplice oscillatore sinusoidale, in quanto la complessità del suono generato è normalmente determinata dalla densità e dall'andamento degli eventi grafici. Sempre da questo programma è possibile disegnare una funzione di controllo globale che agisce sull'ampiezza generale della partitura generata, permettendo un accurato controllo del volume nel tempo. L'inviluppo di ampiezza dei singoli eventi viene scelto graficamente in una apposita pagina software.

- Score Board: è la "lavagna" utilizzata dai programmi di generazione grafica algoritmica.
- Trigon: modulo per la generazione di eventi grafici basati sulle funzioni trigonometriche fondamentali.
- Chaos: programma per la generazione di eventi grafici basati sulla formula logistica

$$x(n+1)=x(n)*r*(1-x(n))$$

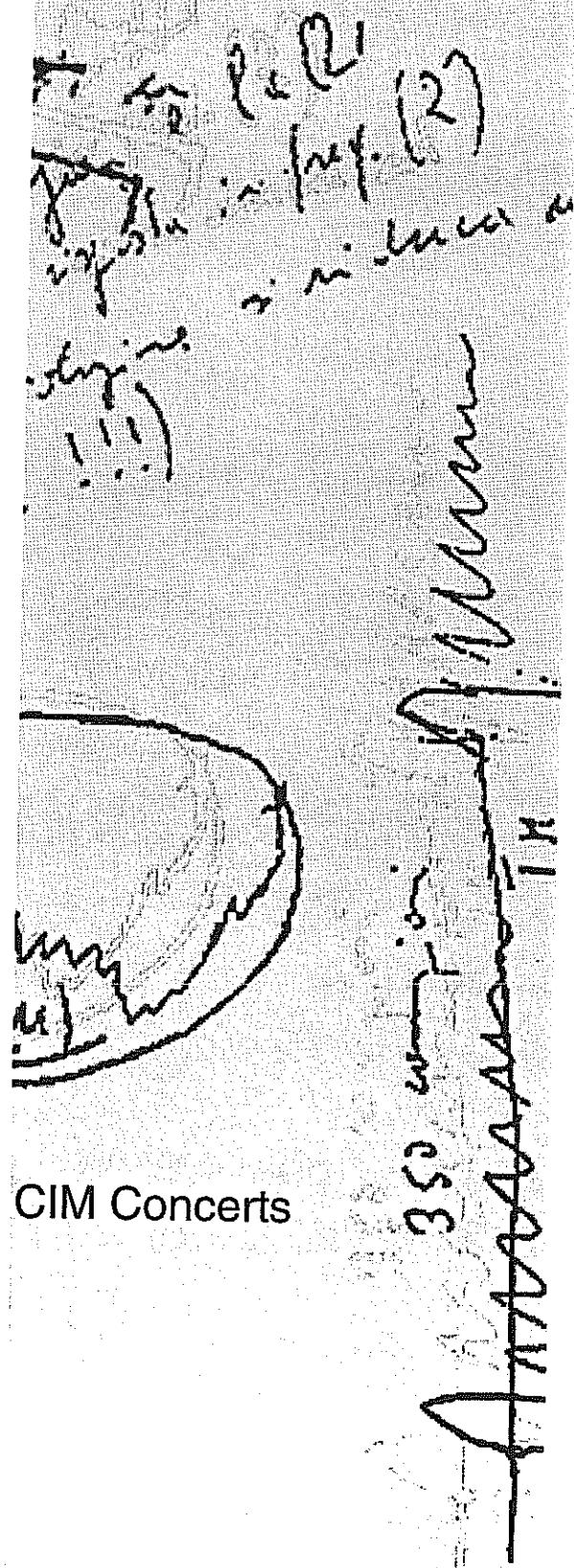
Tale formula, mediante opportuna scelta dei parametri, permette di generare eventi grafici complessi particolarmente adatti a situazioni "di fascia" sia cangianti che statiche.

5 Finalità del lavoro

HyperCSound è nato per esigenze personali, tuttavia l'interfaccia con la quale vengono definiti i parametri necessari alla implementazione delle tecniche di sintesi del suono trattate permette un comodo approccio anche a chi non conosce perfettamente il linguaggio CSound, in quanto creazione dei file di orchestra e partitura, nonché lo stesso processamento di tali file dal CSound, avvengono interamente dall'interno dell'ambiente stesso. Credo, inoltre, che HyperCSound possa rappresentare un esempio di come sia possibile da parte del musicista, purché ovviamente in possesso di determinate conoscenze, crearsi un ambiente personalizzato che permetta di concentrarsi molto più sull'aspetto "creativo" del suono, definendo i molteplici parametri in una forma a lui molto più congeniale di quella matematica, senza necessariamente ricorrere ai classici linguaggi di programmazione.

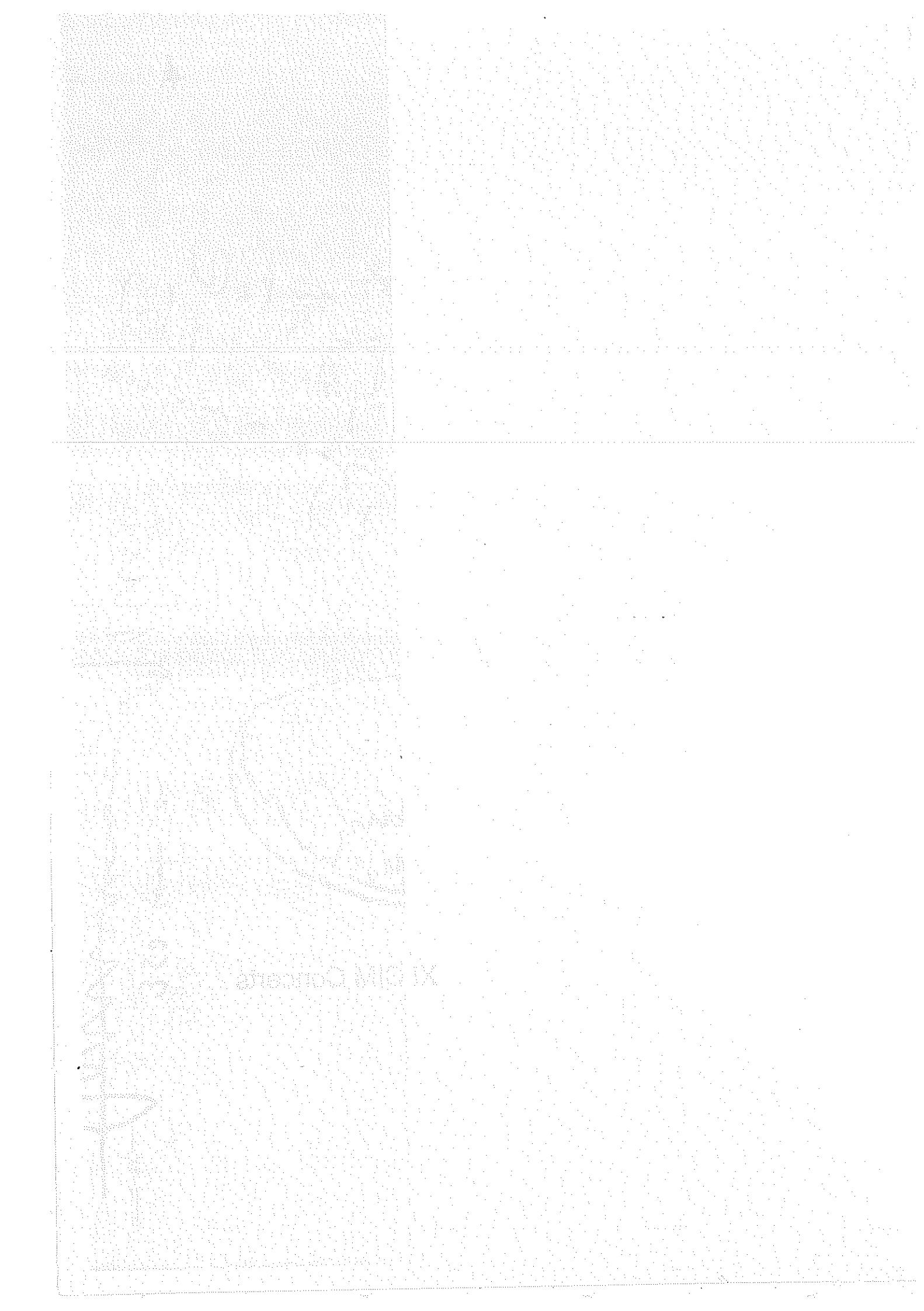
Riferimenti bibliografici

- [1] B. Vercoe: "CSound Manual", Media Lab, M.I.T.
- [2] D. Goodman: "The Complete HyperCardTM 2.0 Handbook", Bantam Books, 1990
- [3] "Inside Macintosh" voll. I - V, Addison-Wesley, 1985-1988



XI CIM Concerts

30.03.1991
António Mamede



Concerti dell' XI CIM

8-nov ore 21

Conservatorio G.B.Martini - piazza Rossini, 2

C. AGUIAR	<i>Piece of mind</i>	per nastro
P. PACHINI	<i>Butterfly crusade</i>	per flauto dolce e nastro
F. GIOMI	<i>That's all folks!</i>	per nastro
A. CIPRIANI	<i>In memory of a recorder</i>	per flauto dolce e nastro
F. DHOMONT	<i>Novars</i>	per nastro
E. PAPPALARDO	<i>Oltre</i>	per flauto dolce e nastro
M. MESSIERI	<i>Rapaci</i>	per flauto, fagotto, pianoforte e nastro

9-nov ore 21

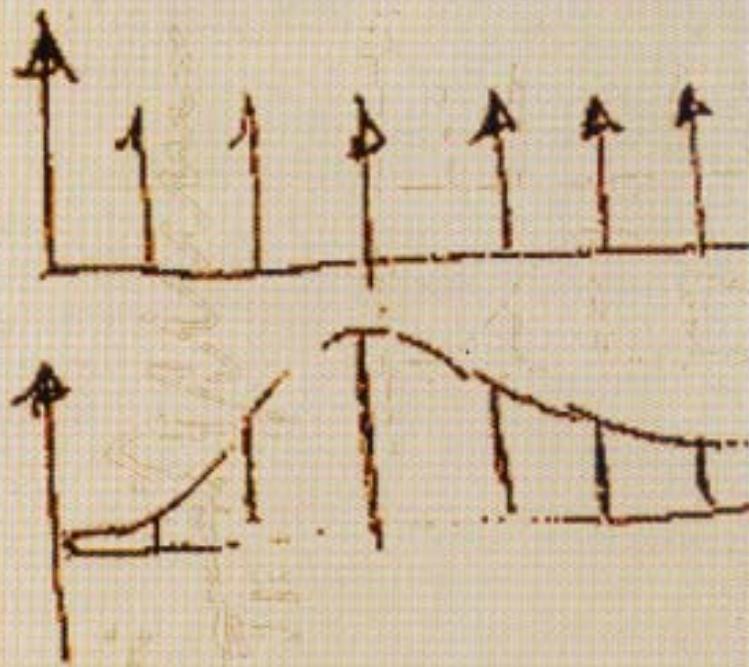
Conservatorio G.B.Martini - piazza Rossini, 2

G. PATELLA	<i>Linee agitate</i>	per nastro
R. DAPELO	<i>Kindergarten</i>	per pianoforte e nastro
J. HARRISON	<i>....et ainsi de suite...</i>	per nastro
L. CECCARELLI	<i>Quanti</i>	per clarinetto e nastro
E. PANCALDI	<i>Dialoghi</i>	per nastro
A. DI SCPIO	<i>Essai du Vide. Schweigen</i>	per nastro
Z. SETTEL	<i>Taudrium</i>	per contrabbasso e live electronics

10-nov ore 21

Conservatorio G.B.Martini - piazza Rossini, 2

G. COSPITO	<i>Ombre e riflessi</i>	per flauto, sintetizzatore e nastro
L. PAVAN	<i>Code</i>	per nastro
M. SERRA	<i>Tityrus</i>	per soprano, nastro e live electronics
W. PRATI	<i>Grains #3</i>	per contrabbasso e Mars workstation
D. SMALLEY	<i>Valley flow</i>	per nastro
M. DE AMICIS	<i>Indeterminato</i>	per flauto, nastro e live electronics
B. TRUAX	<i>Sequence of later Heaven</i>	per nastro
M. BISCARINI	<i>Studio</i>	per 13 strumenti e nastro



35

35

35

35

35

allow to DF
at impulsive