

15º Simpósio Brasileiro de  
Computação Musical

SBCM

15th Brazilian Symposium on  
Computer Music

# Proceedings

Edited by:

Flávio Luiz Schiavoni

Rodrigo Schramm

José Eduardo Fornari Novo Junior

Leandro Lesqueves Costalonga

ISSN 2175-6759

Ficha catalográfica elaborada pelo Setor de Processamento Técnico da Divisão de Biblioteca da UFSJ

Simposio Brasileiro de Computação Musical (15. : 2015 : Campinas, SP)

Anais [recurso eletrônico] do 15º Simposio Brasileiro de Computação Musical = 15th Brazilian Symposium on Computer Music (SBCM), 23 a 25 de novembro de 2015, Campinas, SP / editado por Flávio Luiz Schiavoni ... [et al.]. – Campinas: UNICAMP, 2015.

Disponível em: <http://compmus.ime.usp.br/sbcm2015/files/proceedings-print.pdf>

ISSN: 2175-6759

ISBN: 978-85-76694-75-5

1. Música por computador. 2. Arte e tecnologia. 3. Multimídia (Arte). I. Schiavoni, Flávio Luiz (Ed.). II. Título.

CDU: 78:004

SBCM 2015 is organized by

University of Campinas (UNICAMP)

President: José Tadeu Jorge

Vice President for University Coordination: Álvaro Penteadó Crósta

Vice President for Research (PRP): Gláucia Maria Pastore

Coordination of Interdisciplinary Centers (COCEN)

Coordinator: Jurandir Zullo Junior

Interdisciplinary Center for Studies on Sound Communication (NICS)

Coordinator: Adriana do Nascimento Araújo Mendes

Art Institute, Department of Music

Director: Fernando Augusto de Almeida Hashimoto

Chief of the Department: Leandro Barsalini

Coordinator of Graduate Studies in Music: Alexandre Zamith Almeida

Coordinator of Undergraduate Studies in Music: Paulo J. Siqueira Tiné

Production Center Staff (Ceprod)

Visual programming: Ivan Avelar

Promotion

Brazilian Computer Society – SBC

## **Proceedings of the 15th Brazilian Symposium on Computer Music**

Cover design by Ivan Avelar and José Fornari

Edited by:

Flávio Luiz Schiavoni (UFSJ / BR)

Rodrigo Schramm (UFRGS / BR)

José Eduardo Fornari Novo Junior (UNICAMP / BR)

Leandro Lesqueves Costalonga (UFES / BR)

ISSN 2175-6759

Permission to make digital or hard copies of all part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page.

© Brazilian Computer Society (Sociedade Brasileira de Computação – SBC)

CNPJ nº 29.532.264/0001-78

<http://www.sbc.org.br>

Av. Bento Gonçalves, 9500 – Setor 4 – Sala 116 – Prédio 43424 – Agronomia

CEP 91501-970 – Porto Alegre – RS, Brasil

All papers published in this proceedings were edited based on the original files sent by the authors, without any modification.

## Apresentação

O SBCM - Simpósio Brasileiro de Computação Musical é um evento bianual com reconhecimento internacional no campo da computação musical, organizado pelo NUCOM, a Comissão Especial em Computação Musical da Sociedade Brasileira de Computação (SBC). Esta 15<sup>a</sup> edição será realizada de 23 a 25 de Novembro, em Campinas, e suas atividades planejadas para o SBCM 2015 estão organizadas de acordo com o tema Computação Musical: ultrapassando fronteiras de processamento de sinais e modelos computacionais. Desde os primórdios deste campo de pesquisa, transitando entre experimentos eletroacústicos, sistemas complexos de geração sonora e interfaces homem-máquina, processamento de sinais e modelos computacionais sempre tiveram um papel preponderante. Contudo, o tema proposto para este SBCM abre espaço para a discussão associada a outros aspectos também inerentes ao campo da computação musical, como matemática, criatividade, significado estético, realidade virtual e redes sociais.

## Carta da Organização

É sempre uma satisfação organizar um evento do renome do SBCM; o Simpósio Brasileiro de Computação Musical. Este é, sem dúvida, um dos eventos nacionais mais importantes na área de música e tecnologia, apresentando uma indubitável tradição acadêmica que remonta décadas de atuação e presença ativa de importantes nomes desta área de estudo acadêmico, que intersecciona a ciência, a tecnologia e a arte. A organização local iniciou suas atividades no primeiro semestre de 2015, quando foi formalmente definido que o SBCM 2015 de fato seria realizado na UNICAMP. Eu, como pesquisador do NICS (Núcleo Interdisciplinar de Comunicação Sonora) e participante de diversos SBCMs anteriores, me senti confortável e alinhado com os propósitos e interesses deste evento e devo dizer que a organização ocorreu de forma tranquila, produtiva e motivadora. Para que assim tenha ocorrido, contei com o indelével apoio de diversos setores da Universidade Estadual de Campinas, a UNICAMP, em especial, com as equipes do: Centro de Convenções (CDC), a Diretoria de Apoio à Produção do Departamento de Música do Instituto de Artes, bem como o NICS. Em termos de pessoal, agradeço ao inestimável auxílio da Elizabeth Fernandes (a Beth, do NICS), que participou ativamente de cada detalhe da organização do SBCM; à Prof. Dra. Adriana Mendes, coordenadora do NICS e também participante da equipe de organização do SBCM, bem como todos os demais membros desta equipe, a saber: Ana Lia, Gerson Abdala, Gisele Gomes, Francieli Moreira, Angelica Peña, Miguel Clementi, Djalma de Campos, Sandra Cielavin, Rafael de Almeida, Tiago Tavares, Charles Santana, Gabriel Rimodi, Alexandre Henrique e Matteo Ricciard. Como é de costume, este SBCM também contou com a promoção da Sociedade Brasileira de Computação, a SBC, a qual eu agradeço a presteza e disponibilidade. Também tivemos o privilégio de contar com o apoio financeiro da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, a CAPES, que novamente subsidiou a realização deste evento, bem como ao Fundo de Apoio ao Ensino, à Pesquisa e Extensão, a FAEPEX, da pró-reitoria de pesquisa (PRP) da UNICAMP. Por fim, agradeço à Comissão Especial de Computação Musical, o CECM, na pessoa do Prof. Dr. Leandro Costalonga, e aos respectivos coordenadores de seção técnica, Prof. Dr. Flávio Schiavoni; e musical, Prof. Dr. Rodrigo Schramm. Espero que o SBCM 2015 seja um evento que traga grandes contribuições à pesquisa, à divulgação e ao desenvolvimento da computação musical, que ainda permanece como a grande frente agregadora da análise, do processamento, da composição e da performance musical contemporânea.

Cordialmente

José Eduardo Fornari Novo Jr.

Coordenador geral do SBCM 2015

# Introduction

The SBCM Brazilian Symposium on Computer Music is a biannual event with international recognition in the field of computer music organized by NUCOM, the computer music interest group of the Brazilian Computer Society (SBC). This 15th edition will be held from November 23th to 25th, 2015, at Campinas, São Paulo, and is specially dedicated to the theme "Computer Music: beyond the frontiers of signal processing and computational models". From electroacoustic experiments to the most complex sound generation and human machine interfaces, signal processing and computational models always had an important role. The proposed theme for this SBCM brings up discussions about aspects that are also inherent to the computer music field, as math, creativity, aesthetic meaning, virtual reality and social networks.

## Welcome

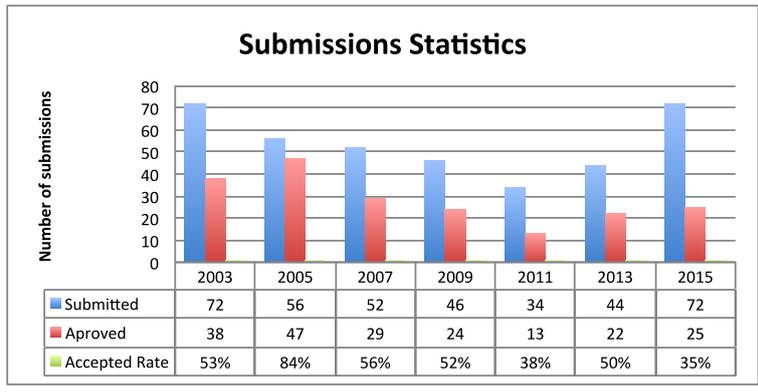
It is always a pleasure to organize an event with the status of SBCM; the Brazilian Symposium on Computer Music. This is undoubtedly one of the most important national events in the field of music and technology, with decades of academic tradition and the active presence of important names of this field of study, that intersects Science, Technology and Art. The local organization started its activities in the first half of 2015, when it was formally decided that the SBCM 2015 would indeed be held at the University of Campinas, UNICAMP. I, the local organizer, researcher at NICS (the Interdisciplinary Center for Sound Communication), and former participant of previous SBCM events, felt comfortable and aligned with the purposes and interests of this event. I must say that the organization process happened in a rather peaceful, productive and motivating way. In order to have that happened, the local organization had the privilege of counting with the support of various departments at UNICAMP, especially with the staffs of: Convention Center (CDC), the Department of Music, at the Institute of Arts, and NICS. In terms of people, I thank the invaluable cooperation of Elizabeth Fernandes (Beth, from NICS), who actively participated in each and every detail of SBCM 2015 organization; Prof. Dra. Adriana Mendes (NICS coordinator and also member of the local organizing team), as well as all other members of this staff, namely: Ana Lia, Gerson Abdala, Gisele Gomes, Francieli Moreira, Angelica Peña, Miguel Clementi, Djalma de Campos, Sandra Cielavin, Rafael de Almeida, Tiago Tavares, Charles Santana, Gabriel Rimodi, Alexandre Henrique and Matteo Ricciard. As usual, this SBCM also had the promotion of the Brazilian Computer Society, SBC, which I here acknowledge. We also had the privilege of having the financial support of CAPES, the Brazilian Coordination for Higher Education and Personnel Improvement, which once again supported an SBCM event, as well as FAEPEX, the Fund for Support Education, at PRP / UNICAMP. Finally, I also thank CECM, the Special Committee on Computer Music, in the person of Prof. Dr. Leandro Costalonga, as well as SBCM 2015 technical section coordinator, Prof. Dr. Flavio Schiavoni; and musical coordinator, Prof. Dr. Rodrigo Schramm. I hope that this SBCM 2015 turns to be an event that succeeds to bring about invaluable contributions for the research, dissemination and development of computer music field that still stands as one of the large aggregating fronts for the analysis, processing, composition and performance in contemporary music.

Best regards,  
José Eduardo Fornari New Jr.  
General Chair of the SBCM 2015

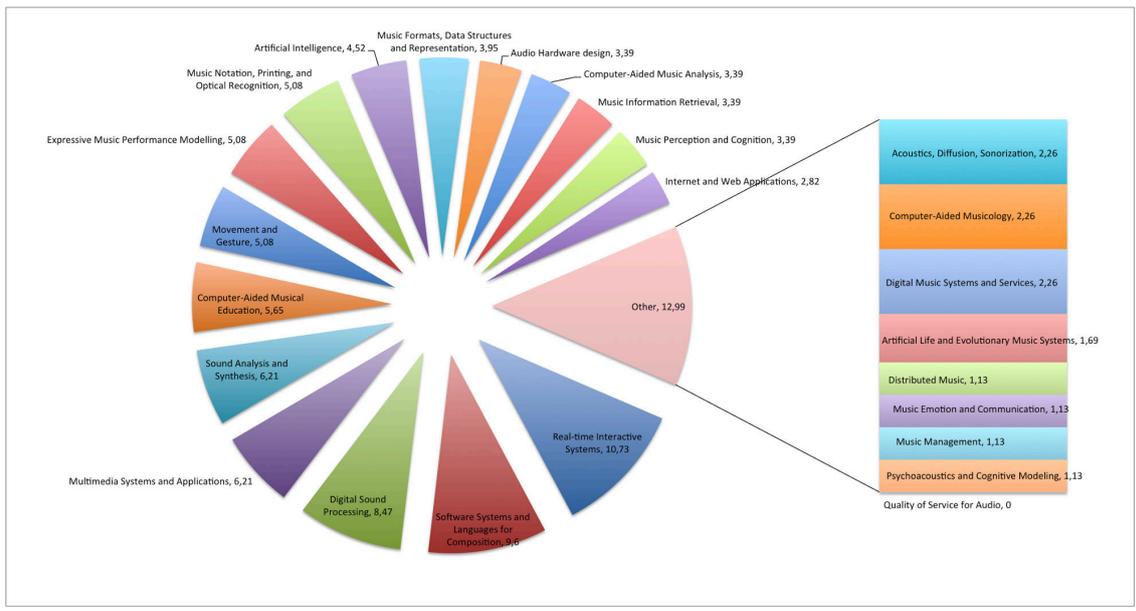
# Carta da CECM/SBC

Na sua 15ª edição, o Simpósio Brasileiro de Computação (SBCM), se volta para além das fronteiras dos processamento de sinais e modelos computacionais. Volta-se para sua intrínseca vocação: a interdisciplinaridade. O primeiro SBCM ocorreu em 1994 e até 2001 foi realizado anualmente, na maior parte das vezes como um evento satélite do Congresso da Sociedade Brasileira de Computação, se tornando bienal a partir de então. Em 2003, a CECM entendeu que o SBCM estava forte o suficiente para se tornar um evento independente. Desta forma, o SBCM 2003 foi o último a ser realizado como evento satélite, servindo como referência para efeitos de comparação com as edições que se sucederam.

O gráfico abaixo mostra os números de submissões, aprovações e respectivas taxas de aceitação das edições do SBCM desde 2003. Observe que após ser realizado isoladamente houve um declínio do número de submissões atingindo seu ponto mais baixo em 2011. Felizmente, o trabalho feito pela CECM permitiu reverter esse declínio e chegar 2015 com o mesmo número de submissões de 2003, mesmo aumentando o rigor do processo de revisão, demonstrado pela redução da taxa de aceitação.



Em 2015, um total de 158 autores de 12 estados brasileiros (além do Canadá e Irlanda), representando 26 centros de pesquisa, submeteram 72 trabalhos na forma de artigos completos técnicos e musicais, pôsteres técnicos e musicais, além de composições e demonstrações de produtos. Isso é um avanço inegável no número de pesquisadores que estão atentos a possibilidade de aplicação de técnicas musicais à música. Os tópicos que agregaram a maior parte dos trabalhos foram: Real-time Interactive Systems (≈ 11%), Software Systems and Languages for Composition (≈ 10%), Digital Sound Processing (≈ 8,5%). Isso está alinhado com nossa proposta de discutir para além das fronteiras dessas grandes áreas. Os desafios futuros do SBCM perpassam por uma ampliação da participação de pesquisadores internacionais no evento, além uma estratégia de massificação do ensino da computação musical nos cursos de computação e música no Brasil.



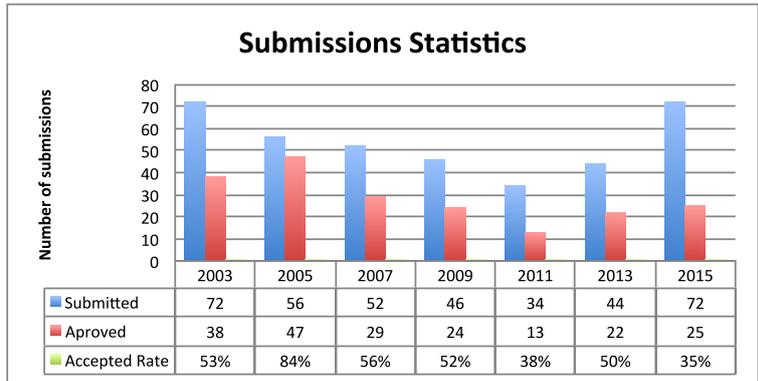
A CECM gostaria de agradecer a presença de todos os participantes e, em especial, aos nossos palestrantes convidados Prof. Dr. Adolfo Maia Jr, Prof. Dr. Rafael Santos Mendes e Prof. Dr. Marcelo Queiroz. Também é devido uma menção honrosa de agradecimento a equipe de organização liderada pelo Prof. Dr. José Tuti Fornari, e aos coordenadores técnico Prof. Dr. Flávio Schiovani e coordenador musical Prof. Dr. Rodrigo Schramm.

Leandro Lesqueves Costalonga,  
 Coordenador CECM/SBC

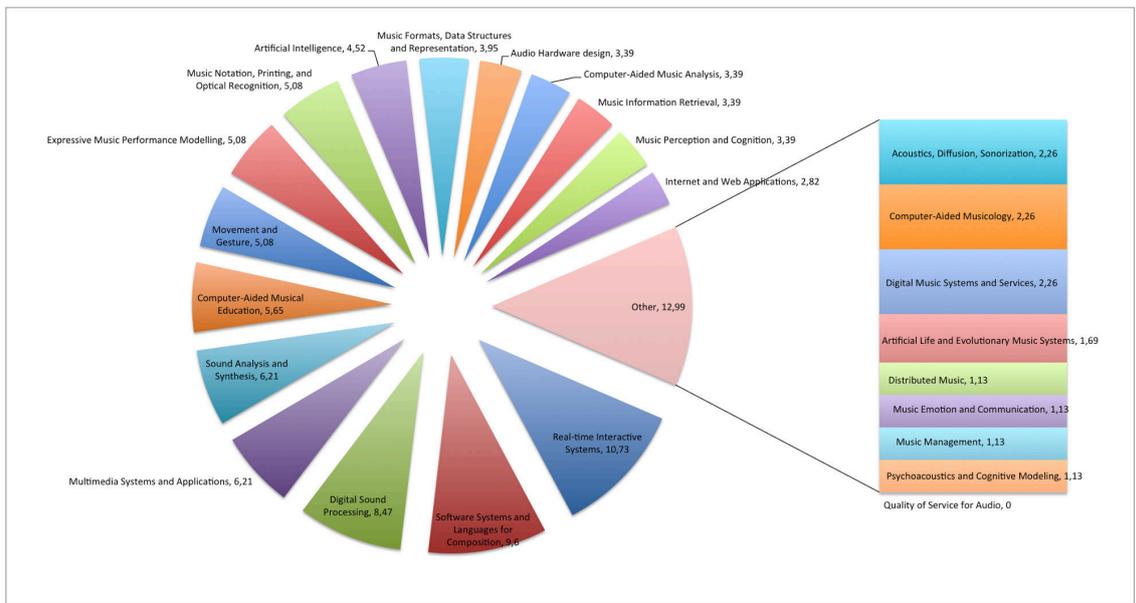
# CECM/SBC Letter

The XV Brazilian Symposium on Computer Music (SBCM) turns itself to its intrinsic interdisciplinary characteristics, reaching out for aspects beyond the signal processing algorithm and computer models. The first SBCM dates back from 1994 happening every year up to 2001, most of the time as a satellite event of the Brazilian Computing Society Symposium. From 2001 on, the organization commission (CECM) decided to promote the event every other year. The last SBCM organized as a satellite event was in 2003, when the CECM assumed the event was mature enough to go on a solo flight. Therefore, the 2003 edition serves as a good reference to the subsequent editions.

The graph below shows the number of submission, approved papers and the respective acceptance rate since 2003. Note that after the decision to become a main event, the number of submissions declined considerably edition after edition. Fortunately, due to the hard work of the CECM, the numbers of the 2015 edition went up to 2003 level, even though the reviewing process is more rigorous nowadays as demonstrated by the acceptance rate.



In 2015, a total of 158 authors, from 12 different Brazilian states (and also Canada and Ireland) submitted a total of 72 papers to SBCM. At least 26 research centres could be identified. This is an incredible increase of the number researchers that are starting to turn their attention to see in the music a rich field to apply computational techniques. The trending topics for the 2015 edition were Real-time Interactive Systems ( $\approx 11\%$ ), Software Systems and Languages for Composition ( $\approx 10\%$ ), Digital Sound Processing ( $\approx 8,5\%$ ). This is inline with our theme that intended to discuss the limits of those big areas in the research field. The challenges for the forthcoming SBCMs includes initiatives to increase the number of foreigner researchers, definitely placing the SBCM in the on international roadmap computer music events. Also, we aim to define a strategy to disseminate computer music knowledge at the computer and music undergrad curriculum.



The CECM would like to thanks the presence of all participants, especially, our invited keynote speakers Prof. Dr. Adolfo Maia Jr, Prof. Dr. Rafael Santos Mendes e Prof. Dr. Marcelo Queiroz. In addition, we must acknowledge all the effort done by the organization committee headed by Prof. Prof. Dr. José Tuti Fornari, the technical chair Prof. Dr. Flávio Schiovani, and the music chair Prof. Dr. Rodrigo Schramm.

Leandro Lesqueves Costalonga,  
CECM/SBC Coordinator

# Symposium Committee

## General Chair

José Eduardo Fornari Novo Junior (UNICAMP / BR)

## Program Chair

Flávio Schiavoni (UFSJ / BR)

## Music Chair

Rodrigo Schramm (UFRGS / BR)

## Local Arrangement Chairs

Tiago Fernandes Tavares (UNICAMP / BR)  
Charles De Paiva Santana (UNICAMP / BR)  
Gabriel Rimoldi De Lima (UNICAMP / BR)  
Adriana Do Nascimento ( UNICAMP / BR)  
Elizabeth Fernandes (UNICAMP / BR)  
Alexandre Henrique Dos Santos (UNICAMP / BR)  
Matteo Ricciardi (Faculdade Nazarena Do Brasil / BR)

## Technical Paper Committee

Adolfo Maia Jr. (Universidade Estadual de Campinas / BR)  
Andrew Horner (The Hong Kong University of Science & Technology / CN)  
Charles Nichols (Virginia Tech / US)  
Emilios Cambouropoulous (Aristotle University of Thessaloniki / GR)  
Edilson Ferneda (Universidade Católica de Brasília / BR)  
Evandro Milletto (Instituto Federal do Rio Grande do Sul / BR)  
Fabien Gouyon (INESC / PT)  
Flávio Luiz Schiavoni (Universidade Federal de São João Del Rei / BR)  
Flávio S. Corrêa da Silva (Universidade de São Paulo / BR)  
Geber Ramalho (Universidade Federal de Pernambuco / BR)  
Hugo de Paula (PUC Minas Gerais / BR)  
Jean Pierre Briot (CNRS Université Paris / FR)  
José Eduardo Fornari Novo Junior (UNICAMP / BR)  
Leandro Costalonga (Universidade Federal do Espírito Santo / BR)  
Luciano Vargas Flores (Instituto Federal do Rio Grande do Sul / BR)  
Luke Dahl (University of Virginia / USA)  
Luis Jure (Universidad de la República / UY)  
Marcelo Gimenes (University of Plymouth, UK)  
Marcelo Johann (Universidade Federal do Rio Grande do Sul / BR)  
Marcelo Queiroz (Universidade de São Paulo / BR)  
Marcelo Soares Pimenta (Universidade Federal do Rio Grande do Sul / BR)  
Mikhail Malt (Institut de Recherche et Coordination Acoustique/Musique / FR)  
Patrick McGlynn (Maynooth University / IE)  
Rodrigo Schramm (Universidade Federal do Rio Grande do Sul / BR)  
Roger Dannenberg (Carnegie Mellon University / USA)

## Music Committee

Bruno Ruviano (Santa Clara University / US)  
Damian Keller (Universidade Federal do Acre / BR)  
Duncan Williams (ICCMR, Plymouth University / UK)  
Eloy Fernando Fritsch (Universidade Federal do Rio Grande do Sul / BR)  
Federico Visi (ICCMR - Plymouth University / UK)  
Fernando Iazzetta (Universidade de São Paulo / UK)  
Flávio Luiz Schiavoni (Universidade Federal de São João Del Rei / BR)  
James Correa (Universidade Federal de Pelotas / UK)  
José Eduardo Fornari Novo Junior (UNICAMP / BR)  
Leandro Costalonga (Universidade Federal do Espírito Santo / BR)  
Luciano Zanatta (Universidade Federal do Rio Grande do Sul / BR)  
Luke Dahl (University of Virginia / USA)  
Marcelo Gimenes (ICCMR, Plymouth University / UK)  
Marcus Alessi Bitencourt (Universidade Estadual de Maringá / BR)

Ricardo Bordini (Universidade Federal do Maranhão / BR)  
Ricardo dal Farra (Universidade Nacional de Tres de Febrero / AR)  
Rodrigo Schramm (Universidade Federal do Rio Grande do Sul / BR)  
Tiago Tavares (Universidade Estadual de Campinas / BR)  
Victor Lazzarini (National University of Ireland / IR)

### **CECM/SBC Cordinator**

Leandro Lesqueves Costalonga (UFES/BR)

### **Webmaster**

Leandro Pessoa (UFV / BR)

# Table of Contents

## Technical Papers and Posters

Evaluating automated classification techniques for folk music genres from the Brazilian Northeast Jeronimo Barbosa, Cory Mckay, Ichiro Fujinaga . . . . .	3
Automatic gesture segmentation of a musical performance using a cumulative dissimilarity measure Hugo de Paula, Maurício Loureiro, Euler Teixeira, Raphael Borges . . . . .	13
Detecção de Refrão em Sinais de Áudio usando Extração de Características de Intensidade do Som Renato Rodrigues, Geber Ramalho, Giordano Cabral . . . . .	25
Análise do Vibrato e Bending na Guitarra Elétrica Tairone Magalhaes, Davi Mota, Aluizio Oliveira, Maurício Loureiro . . . . .	36
Copista - Sistema de OMR para a recuperação de acervo histórico musical Avner Paulo, Flávio Schiavoni, Marcos Laia, Daniel Madeira . . . . .	48
Utilização de Software Livre no ensino de Música da UFSJ Luan Gonçalves, Flávio Schiavoni . . . . .	60
A Study Design About the Influence of Spectrum Content in the Perception of Consonant and Dissonant Chords Guilherme Santo, Antonio Goulart, Micael Antunes . . . . .	68
A Low Cost Computing Interface to Speed Up Braille Music Notation Antonio Penteado, Vilson Zattera, Jose Fornari . . . . .	72
Árvore das Memórias: Instalação Multimídia Interativa Ayer Netto, Luan Casthologe, Amanda Oliosi, Ariane Mateus, Leandro Costalonga, Daniel Coura . . . . .	76
Reflexões sobre o Ensino de Computação Musical: Como é a disciplina de Introdução a Computação & Música no INF/UFRGS Marcelo Johann, Marcelo Pimenta . . . . .	84
Symbolic regression as a computer-aided music tool for analysis and composition José H. Padovani, Jônatas Manzolli . . . . .	96
Towards a Virtual Portable Radio Platform for Audio Transmission Esteveo Haeser, Evandro Miletto . . . . .	107
A modular platform for a subtractive synthesizer on Arduino Due Rodolfo Pirotti, Marcelo Johann, Marcelo Pimenta . . . . .	111
Composição Algorítmica em Redes Complexas Vitor Rolla, Luiz Velho . . . . .	115
Compilador musical: Uso da tecnologia de compiladores para validação de música simbólica. Leandro Pessoa, Flávio Schiavoni . . . . .	119
Entre notas e alaridos: perspectivas na bioacústica de mamíferos terrestres Regis Faria, Aline Gasco, Patrícia Monticelli . . . . .	123
Protótipo Robótico de Baixo Custo para Performance Musical em Violão Acústico Higor Camporez, Trajano Mota, Leandro Costalonga, Helder Rocha . . . . .	127
Representação de Conhecimento Musical e Programação Lógica Indutiva - Uma Revisão Sistemática Clenio Gonçalves Jr., Murillo Homem . . . . .	138
V-Conga: Um Estudo de Caso Para Avaliar as Capacidades de Captura de Nuances Gestuais do Leap Motion© em um Sistema Híbrido Eduardo Silva, Giordano Cabral, Geber Ramalho . . . . .	142
Investigação sobre Combinações de Frases Percussivas através de Levantamento de Hipóteses e Análise Computacional Luca Dias, Geber Ramalho, Giordano Cabral . . . . .	146
Um Estudo sobre as TICs como Ferramentas na Educação Musical de Alunos com Deficiência Visual Alexandre Santos, Jose Fornari, Vilson Zattera . . . . .	150

Tempot: criação, edição de funções no navegador em tempo de execução. Guilherme Lunhani, Flávio Schiavoni . . . . .	154
--	-----

## Musical Papers, Posters

Interação e convergência entre o instrumental e o eletrônico em Poussières Cosmiques Danilo Rossetti . . . . .	159
Comprovação; Abordagens Desde a Heurística Estética em Eco Composição Luzilei Aliel, Rogerio Costa, Damian Keller . . . . .	169
Metaflauta: design e performance de instrumento aumentado via suporte computacional Gabriel Rimoldi, Jônatas Manzolli . . . . .	181
GuitarAMI: um Instrumento Musical Aumentado que Transpõe Restrições Intrínsecas do Violão Eduardo Meneses, Jose Fornari . . . . .	193
Caminho das águas, instalação audiovisual interativa: implementação computacional e interação dos visitantes Clayton Mamedes, Artemis Moroni, Jônatas Manzolli, Denise Garcia . . . . .	201
SoMo: Um Instrumento Musical baseado em Movimento Características Estéticas e Técnicas e potenciais Usos Tiago Brizolara da Rosa, Marcos Moritz . . . . .	213
Colagem, recorte e erros em um processo composicional utilizando o Music21. Guilherme Lunhani . . . . .	225
A Música Visual como alternativa aos modelos computacionais utilizados no controle da Síntese Granular em Tempo Real Fernando Falci Souza . . . . .	235
Expert systems and genetic algorithms applied to the twelve-tone musical composition Anderson Rodrigo Barretto Teodoro, Flávia de Toledo Martins Bedê, Denilson Paulo Santos, Paulo André Lima de Castro . . . . .	247
GestAção I: A hybrid computer music guitar performance mediated by instrumental gestures Jorge Santos, Jose Fornari . . . . .	251

## Music Compositions

Bandas Críticas Ariane Stolfi . . . . .	256
Contrapontos Fernando Falci Souza . . . . .	258
Unangebracht (2015) Darwin Pillar Corrêa . . . . .	260

## Summary

Author Index . . . . .	262
Institution Index . . . . .	263

# Technical Papers and Posters

# Evaluating automated classification techniques for folk music genres from the Brazilian Northeast

Jerônimo Barbosa<sup>1</sup>, Cory McKay<sup>1</sup>, Ichiro Fujinaga<sup>1</sup>

<sup>1</sup>CIRMMT, Schulich School of Music, McGill University  
555 Sherbrooke St. West, Montreal, Canada

{jeronimo.costa,cory.mckay}@mail.mcgill.ca, ich@music.mcgill.ca

**Abstract.** *In this exploratory study, we investigate the problem of automated genre classification focusing on the particular context of folk music genres from the Brazilian Northeast—an issue still under-explored in the literature. As a contribution, we have: a) created a new public dataset with 75 samples equally distributed over 5 different genres (i.e., Cavalo Marinho, Ciranda, Coco, Maracatu de Baque Solto and Maracatu de Baque Virado); and b) evaluated 68 features and 10 classifiers, aiming to find those well-suited to this particular classification task. Our results demonstrated high classification accuracy rates: 60% before feature selection when using either the Naive Bayes or Support Vector Machine algorithms, and a very impressive increase to 100% classification after feature selection with our best-performing feature selection methodology.*

## 1. Introduction

“Brazilian cultural heritage, like racial identification in the country, is mixed, complex and diverse. Nowhere is this situation more evident than in the Brazilian Northeast” [Crook, 2005]. This heritage can also be observed in music. In Pernambuco (a state from the Brazilian Northeast), examples like ‘maracatu’, ‘cavalo marinho’, and ‘coco’ show us how these influences were incorporated over the years into unique musical genres, playing an important role on what is known today as ‘*Música Popular Brasileira*’ (MPB).

However, these genres are fairly unknown outside the “broadcast, recording, print, and electronic media that circulate” about Brazil [Crook, 2005]. The difference between them can be subtle and confusing for non-specialists. This problem is enhanced by the existing oral tradition in the Brazilian Northeast and by the relative lack of educational books to guide newcomers [Souza, 2011]. Perhaps those reasons might explain: a) the lack of studies investigating the suitability of traditional Music Information Retrieval (MIR) techniques in the context of folk music from northeastern Brazil; and b) the lack of public databases of folk music from northeastern Brazil, which could be used by MIR systems.

In order to address these issues, this exploratory study focuses on the usage of automatic genre classification in the context of folk music genres from the northeastern Brazil. As a contribution, we have: a) created a new public dataset consisting of 75 samples equally distributed over 5 different genres (i.e., *Cavalo Marinho*, *Ciranda*, *Coco*, *Maracatu de Baque Solto* and *Maracatu de Baque Virado*); and b) evaluated different features and classifiers, aiming to find which ones would be more suitable for the classification task. As tools, we used jMIR/jAudio [McKay, 2010], for the feature extraction, and Weka [Witten et al., 2011], for the classification and feature selection.

## 2. Background

Audio-based genre classification is a traditional task for MIR systems, included in the MIREX (Music Information Retrieval Evaluation eXchange) since its beginning in 2005<sup>1</sup>. As a result, we have today several different approaches—see [Scaringella et al., 2006] for a comprehensive survey—and some high-level support tools designed for the task, such as the jMIR [McKay, 2010] and the Marysas<sup>2</sup>.

Although there are no studies concerning the application of these techniques for the folk musical genres from Northeast Brazil, they have already been used in other very specific musical contexts. Recent examples include: a) 7 heavy metal music sub-genres [Tsatsishvili, 2011], in which the author achieved 45.7% accuracy with AdaBost algorithm; b) 9 Latin American Music genres [Völkel et al., 2010], in which the authors achieved 86.7% accuracy using a new technique based on the extraction of rhythmic patterns and template matching; and c) 8 electronic dance music sub-genres [Leimeister et al., 2014], in which the authors achieved 71% accuracy using a similar technique.

## 3. Methodology

In order to achieve the goals specified, we used the following methodology:

1. We listed and analyzed which musical genres could be considered for this project;
2. We created the necessary dataset for this project;
3. We extracted features from the audio files using jAudio;
4. We compared performances of different classifiers with 5-fold cross validation using Weka; and
5. We investigated different feature selection techniques and their impact on the classification using Weka.

Each one of these steps will be further discussed in the following sections.

### 3.1. Choosing the genres

Folk music genres from northeastern Brazil might share some common characteristics. Examples include rigid musical structure (e.g., instrumentation and rhythm could have little variation throughout songs), predominant use of percussion instruments, where rhythm plays an important role, and close relationships to traditional celebrations or rituals—as shown in Figure 1. They also have a great component of oral tradition [Souza, 2011] and—although a simple task for people experienced in music or those traditions—there is not much material dedicated to teaching how to identify these genres.

One of the few exceptions is the “Manual dos ritmos pernambucanos” [Souza, 2011], in which the author presents rhythmic characteristics of 9 genres from Pernambuco, a state in the Brazilian Northeast. Based on this material, we decided to focus our study on a subset of 5 genres. We believe these examples covers part of the cultural diversity on the Brazilian Northeast, without making our scope too broad. They are:

- Maracatu de baque solto;
- Maracatu de baque virado;
- Cavalo marinho;
- Ciranda; and
- Coco.



Figure 1: The chosen genres <sup>3</sup>

Audio examples for each of these genres can be found online <sup>4</sup>.

### 3.2. Our dataset

The lack of public databases focused on the selected genres motivated us to create a new one. Our dataset was composed of audio extracted from public videos available on Youtube. We gave preference to audio extracted from amateur videos recorded in the context/conditions those genres are often performed (i.e., in the streets, during rehearsals or presentations, with background noise). We did this in order to create a standard audio quality across the different genres, as we could not find recordings with good audio quality for all genres. In addition, public videos allows other researchers to have access to our dataset.

In total, 15 videos were selected for each genre. For each audio extracted, a small excerpt—30 seconds duration—was randomly selected by using a custom script <sup>5</sup>, resulting in 75 labeled samples. The 30-second duration was chosen for two reasons: a) Since the duration of the video files ranged from 40 seconds to 3 minutes, the 30-second selection process would cover all cases; b) We believe that 30-second is sufficient for a human specialist to recognize the genre.

The list of selected videos <sup>6</sup> and the final dataset are publicly available online.

### 3.3. Feature Extraction

Features were extracted using the jAudio tool, provided by jMIR framework [McKay, 2010]. As settings, we used the default window size of 512 with no overlap. In total, 13 low-level features were used. Except for the ‘Method of

<sup>1</sup><http://www.music-ir.org/mirex/wiki/2005>

<sup>2</sup><http://marsyas.info/>

<sup>3</sup>Image sources: (a) <http://goo.gl/0Rxwie> ; (b) <http://goo.gl/3BtIIZ> ; (c) <http://goo.gl/NWZsup> ; (d) <http://goo.gl/UmNEfo> ; (e) <http://goo.gl/2ymyvP> .

<sup>4</sup><https://github.com/jeraman/sbcm2015-dataset/tree/master/database>

<sup>5</sup><https://github.com/jeraman/sbcm2015-dataset/blob/master/mp3randomchopper.py>

<sup>6</sup><https://github.com/jeraman/sbcm2015-dataset/blob/master/database%20youtube%20list.txt>

Moments' (excluded from our study due to instability), these are the default features suggested by jAudio. They are (name followed by its respective jAudio description):

**Spectral Centroid:** "The centre of mass of the power spectrum";

**Spectral Rolloff Point:** "The fraction of bins in the power spectrum at which 85% of the power is at lower frequencies. This is a measure of the right-skewness of the power spectrum";

**Spectral Flux:** "A measure of the amount of spectral change in a signal. Found by calculating the change in the magnitude spectrum from frame to frame";

**Compactness:** "A measure of the noisiness of a signal. Found by comparing the components of a window's magnitude spectrum with the magnitude spectrum of its neighbouring windows";

**Spectral Variability:** "The standard deviation of the magnitude spectrum. This is a measure of the variance of a signal's magnitude spectrum";

**Root Mean Square:** "A measure of the power of a signal";

**Fraction Of Low Energy Windows:** "The fraction of the last 100 windows that has an RMS less than the mean RMS in the last 100 windows. This can indicate how much of a signal is quiet relative to the rest of the signal";

**Zero Crossings:** "The number of times the waveform changed sign. An indication of frequency as well as noisiness";

**Strongest Beat:** "The strongest beat in a signal, in beats per minute, found by finding the strongest bin in the beat histogram";

**Beat Sum:** "The sum of all entries in the beat histogram. This is a good measure of the importance of regular beats in a signal";

**Strength Of Strongest Beat:** "How strong the strongest beat in the beat histogram is compared to other potential beats";

**MFCC (split into 13 different features):** "MFCC calculations based upon Orange Cow code"; and

**LPC (split into 10 different features):** - 'Linear Prediction Coefficients calculated using autocorrelation and Levinson-Durbin recursion".

These low-level features generated in total 68 summary features (i.e., overall standard deviation and average for each low-level-feature), which were used for the classification. A file containing the values of all extracted features, for each sample in our database, is available online <sup>7</sup>.

### 3.4. Classification

For the classification task, we have used Weka. No default parameters were changed (except for the k-NN, for each we used k=3). The classifiers tested were:

- k-NN;
- NaiveBayes;
- j48;
- NBTree;
- MultiClassClassifier;
- Classification via regression;
- Support Vector Machine (SMO);
- Neural Network (Multilayer Perceptron);
- AdaBoost; and
- DecisionTable.

<sup>7</sup><https://github.com/jeraman/sbcm2015-database/tree/master/extracted%20features>

It is important to stress that there is a large diversity of approaches towards classification in literature, and some might have been omitted due to scope constraints. We reasoned that the algorithms selected are among the most basic and popular, covering at the same time part of the diversity.

The classifiers were tested with all 68 summary features using 5-fold cross-validation. We chose 5-fold cross validation due to the limited size of our dataset. We emphasize that there was no overlap in terms of the samples in matching training and testing folds during cross validation. In order to provide basis for comparison, we also highlight that the success rate for a random classifier is 20%. Results are summarized in Table 1.

**Table 1: Results for each classifier tested with all 68 features, using 5-fold cross validation. The best result was achieved with the Naive Bayes and the Support Vector Machine (60%).**

Classifier	Correctly Classified Instances
k-NN (k=3)	58.6667%
<b>NaiveBayes</b>	<b>60%</b>
j48	45.33%
NBTree	50.67%
MultiClassClassifier	50.67%
NBTree	54.67%
Classification via regression	46.67%
<b>Support Vector Machine</b>	<b>60%</b>
Multilayer Perceptron	57.33%
AdaBoost	33.33%
DecisionTable	44%

The best results were achieved with the Naive Bayes and the Support Vector Machine (both 60%). A complete report for each one of the classifiers can be found online <sup>8</sup>. Considering these cases, we present the detailed accuracy and confusion matrixes respectively in Tables 2, 3, 4, and 5.

**Table 2: Detailed accuracy for the Naive Bayes classifier using all 68 features.**

Class	Precis.	Recall	F-Meas.
<i>Cavalo Marinho</i>	0.692	0.6	0.643
<i>Coco</i>	0.417	0.333	0.37
<i>Ciranda</i>	0.545	0.4	0.462
<i>Maracatu Solto</i>	0.6	0.8	0.686
<i>Maracatu Virado</i>	0.684	0.867	0.765
<b>Weig. Avg.</b>	0.588	0.6	0.585

By analysing these results, we can notice that Coco was the genre which received the larger number of misclassifications (10) throughout the scenarios (i.e., Coco samples were wrongly classified as another genre), followed by Ciranda (9 misclassifications with

<sup>8</sup><https://github.com/jeraman/sbcm2015-database/tree/master/stage%201%20-%20classification>

**Table 3: Confusion Matrix for the *Naive Bayes* classifier using all 68 features.**

<b>Cav. Marinho</b>	<b>Coco</b>	<b>Ciranda</b>	<b>Mar. Solto</b>	<b>Mar. Virado</b>	<i>&lt;-classified as</i>
<b>9</b>	2	0	4	0	<b>Cav. Marinho</b>
2	<b>5</b>	4	1	3	<b>Coco</b>
1	2	<b>6</b>	3	3	<b>Ciranda</b>
1	1	1	<b>12</b>	0	<b>Mar. Solto</b>
0	2	0	0	<b>13</b>	<b>Mar. Virado</b>

**Table 4: Detailed accuracy for the *Support Vector Machine* classifier using all 68 features.**

<b>Class</b>	<b>Precis.</b>	<b>Recall</b>	<b>F-Meas.</b>
<i>Cavalo Marinho</i>	0.9	0.6	0.72
<i>Coco</i>	0.357	0.333	0.345
<i>Ciranda</i>	0.5	0.6	0.545
<i>Maracatu Solto</i>	0.625	0.667	0.645
<i>Maracatu Virado</i>	0.706	0.8	0.75
<b>Weig. Avg.</b>	0.618	0.6	0.601

**Table 5: Confusion Matrix for the *Support Vector Machine* classifier using all 68 features.**

<b>Cav. Marinho</b>	<b>Coco</b>	<b>Ciranda</b>	<b>Mar. Solto</b>	<b>Mar. Virado</b>	<i>&lt;-classified as</i>
<b>9</b>	3	1	2	0	<b>Cav. Marinho</b>
0	<b>5</b>	5	2	3	<b>Coco</b>
0	2	<b>9</b>	2	2	<b>Ciranda</b>
1	2	2	<b>10</b>	0	<b>Mar. Solto</b>
0	2	1	0	<b>12</b>	<b>Mar. Virado</b>

Naive Bayes, 6 with Support Vector Machines). Their accuracy are considerably below the average. In the case of Coco, for example, variables such as ‘Precision’, ‘Recall’ and ‘F-Measure’ are almost half of the average. Further studies are needed in order to address this issue.

### 3.5. Feature Selection

To improve the classification rates, we explored two different feature selection approaches: the wrapper method and filter method. Again, we reasoned that those are among the most basic and popular approaches employed in literature. For the wrapper method-based approaches, we tested:

- ‘ClassifierSubsetEval’ as attribute method, and ‘BestFirst’ as search method;
- ‘ClassifierSubsetEval’ as attribute method, and ‘GreedyStepWise’ as search method;
- ‘WrapperSubsetEval’ as attribute method, and ‘BestFirst’ as search method; and
- ‘WrapperSubsetEval’ as attribute method, and ‘GreedyStepWise’ as search method.

Regarding the filter method-based approaches, we tested:

- **‘InfoGainAttributeEval’** as attribute method, and **‘Ranker’** as search method.

In both cases, we used default parameters as provided in Weka. The only exception was the default classifier used to guide the feature selection, which was replaced by the Support Vector Machine classifier. The reason was its performance, as described in the previous subsection (*Classification*).

We trained these approaches with 80% of our original dataset (our training set), validating their performance with the remaining 20% (our validation set). Our intention was to “use an independent set to evaluate feature selection’s efficacy”, avoiding previously mentioned pitfalls in music classification [Fiebrink and Fujinaga, 2006]. Thus, the validation set did not include samples also found in the training set. The Support Vector Machine classifier was used again as a benchmark to compare improvement on accuracy. Without feature selection, its accuracy was 60%.

It is important to highlight the reduced size of our validation set (i.e., 15 samples, or 20% of the original dataset). Such small size may cause uncertainty about the precision of the results achieved. In order to minimize this issue, we have:

1. Randomly repartitioned the our training set into 5 new subsets. Each subset was composed of 48 (i.e., 80% of our training set) randomly selected samples from our training set. This step was done by using a custom script <sup>9</sup>;
2. For each subset created, we performed an independent feature selection. All selected features were then independently validated (with our validation set). In the particular case of the ‘InfoGainAttributeEval’ based approach (which ranks all features by giving it a score), we only selected the features with scores greater than zero. Both selection and validation were performed in Weka;
3. For each tested approach, we first calculated the average and the standard error over the results achieved in each subset <sup>10</sup>, as presented in Table 6. Finally, we have analysed the occurrence of the selected features over the subsets—summarized in Table 7. Features were listed if they appeared at least in half of the subsets <sup>11</sup>.

**Table 6: Average accuracy and standard error for each feature selection approach tested. Average calculated over the results achieved in each subset.**

Feature Selection Approach	Avg. Accuracy	Avg. Standard Error
ClassifierSubsetEval + BestFirst	86.67%	3.4%
ClassifierSubsetEval + GreedyStepWise	88%	3.27%
InfoGainAttributeEval + Ranker	100%	0%
WrapperSubsetEval + BestFirst	86.67%	3.65%
WrapperSubsetEval + GreedyStepWise	85.33%	2.49%

As presented in Table 6, we noticed a significant improvement when using feature selection—no matter the approach chosen. The best result was achieved when using the ‘InfoGainAttributeEval’ (100% average accuracy with the Ranker search method). Even the worst scenario (‘WrapperSubsetEval’ plus ‘GreedyStepWise’) presented a significant improvement on accuracy (around 25.33% increase, with 2.49% of standard error).

<sup>9</sup><https://goo.gl/uJpQTH>

<sup>10</sup><https://goo.gl/uI9b52>

<sup>11</sup><https://goo.gl/KFNhmJ>

Regarding the most common features—among the 68 tested—for the classification task, results are presented in Table 7. We note the predominance of ‘Mel-frequency cepstral coefficients’ (MFCC) related features. Considering the 15 features suggested by ‘InfoGainAttributeEval’, for example, 12 are MFCC related. Further studies are needed in order to address this issue.

**Table 7: The most common features selected over the subsets used for training. Features presented below appeared at least in half of the subsets. Approaches are represented as: (a) ClassifierSubsetEval + BestFirst; (b) ClassifierSubsetEval + GreedyStepWise; (c) InfoGainAttributeEval + Ranker; (d) WrapperSubsetEval + BestFirst; (e) WrapperSubsetEval + GreedyStepWise.**

Feature Selection Approach	Selected features	Occurrence over the folds
(a)	MFCC Overall Standard Deviation9	80%
	Fraction of Low Energy Wind. Overall Standard Deviation0	60%
	MFCC Overall Standard Deviation0	60%
	MFCC Overall Standard Deviation10	60%
	Spectral Variability Overall Average0	60%
	Beat Sum Overall Average0	60%
	MFCC Overall Average3	60%
	MFCC Overall Average8	60%
(b)	MFCC Overall Standard Deviation0	60%
	MFCC Overall Standard Deviation9	60%
	Spectral Variability Overall Average0	60%
	MFCC Overall Average3	60%
(c)	MFCC Overall Standard Deviation7	100%
	MFCC Overall Standard Deviation8	100%
	MFCC Overall Standard Deviation9	100%
	MFCC Overall Standard Deviation10	100%
	MFCC Overall Standard Deviation11	100%
	MFCC Overall Standard Deviation12	100%
	LPC Overall Standard Deviation0	100%
	Spectral Variability Overall Average0	100%
	MFCC Overall Average3	100%
	MFCC Overall Average4	100%
	MFCC Overall Average5	100%
	MFCC Overall Average9	100%
	MFCC Overall Average7	80%
LPC Overall Standard Deviation8	60%	
MFCC Overall Average8	60%	
(d)	MFCC Overall Average5	80%
	LPC Overall Standard Deviation3	60%
(e)	LPC Overall Standard Deviation3	60%
	MFCC Overall Average9	60%

These results suggest that feature selection plays an important role in improving accuracy on automated classification of folk music genres from the northeastern Brazil. Details (for both training and validation stages) are available online <sup>12</sup>.

<sup>12</sup><https://github.com/jeraman/sbcm2015-database/tree/master/stage%20%20-%20features%20selection>

#### 4. Conclusion & Future Work

In this exploratory work, we have investigated the problem of automated genre classification considering the particular context of folk music genres from northeastern Brazil. As a contribution, we have: a) created a new public dataset with 75 samples equally distributed over 5 different genres (i.e., *Cavalo Marinho*, *Ciranda*, *Coco*, *Maracatu de Baque Solto* and *Maracatu de Baque Virado*); and b) evaluated different features and classifiers, aiming to find ones well-suited to this particular classification task.

Classification results—evaluated using 5-fold cross validation—showed high accuracy rates (e.g., 60% for both Naive Bayes and the Support Vector Machine classifier with all features) when compared to a random classifier (around 20% accuracy). We also highlight that *Coco* was the genre which received the larger number of misclassifications (10) throughout the best two scenarios. Further studies need to be performed in order to assess this issue.

Regarding the feature selection, a significant improvement (tested with an average accuracy of 89.33%) was found in all five approaches tested. As a highlight, 100% accuracy was achieved with the Support Vector Machine using ‘InfoGainAttributeEval’ as attribute method, and ‘Ranker’ as search method. The most common features were the ones related to the MFCC.

Finally, concerning the size of our dataset, it is important to stress its limitations. Previous works [McKay et al., 2006] have suggested that MIR databases should “include many thousands of recordings”. The authors argue that this would “allow sufficient variety”, and also “avoid research overuse of a relatively small number of recordings, which can result in overtraining”. Such considerations must be taken into account when analyzing our results—especially the feature selection, as already pointed out. However, given the lack of other open databases of folk music from the Brazilian Northeast, we believe that the dataset we present is a valuable and relevant initial contribution to the research community.

As future work, we plan to compare the results achieved in this work with the ones achieved by human evaluators (both specialists and non specialists). In addition, over the long term we plan to include other genres in our analysis, such as *Frevo* and *Baião*.

#### References

- Crook, L. N. (2005). *Brazilian Music: Northeastern Traditions and the Heartbeat of a Modern Nation*. ABC-CLIO, Santa Barbara, California.
- Fiebrink, R. and Fujinaga, I. (2006). Feature Selection Pitfalls and Music Classification. In *International Conference on Music Information Retrieval*, pages 340–341.
- Leimeister, M., Gaertner, D., and Dittmar, C. (2014). Rhythmic Classification of Electronic Dance Music. In *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*.
- McKay, C. (2010). *Automatic music classification with jMIR*. PhD thesis, McGill University.
- McKay, C., McEnnis, D., and Fujinaga, I. (2006). A large publicly accessible prototype audio database for music research. In *Proceedings of the International Conference on Music Information Retrieval*, pages 160–163.
- Scaringella, N., Zoia, G., and Mlynek, D. (2006). Automatic genre classification of music content. *IEEE Signal Processing Magazine*, 23(2):133–141.

- Souza, F. (2011). *Esquentando tambores: manual de percussao dos ritmos pernambucanos - escrita e tecnica*. Funcultura, CEL, Recife, Brazil.
- Tsatsishvili, V. (2011). *Automatic subgenre classification of heavy metal music*. Master's thesis on music, mind and technology, University of Jyväskylä.
- Völkel, T., Abeßer, J., Dittmar, C., and Großmann, H. (2010). Automatic genre classification of Latin American music using characteristic rhythmic patterns. *Proceedings of the 5th Audio Mostly Conference on A Conference on Interaction with Sound - AM '10*, pages 1–7.
- Witten, I. H., Frank, E., and Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, 3rd edition.

# Automatic gesture segmentation of a musical performance using a cumulative dissimilarity measure

Hugo B. de Paula<sup>1</sup>, Euler C.F. Teixeira<sup>2</sup>, Raphael Borges<sup>2</sup>, Maurício A. Loureiro<sup>2</sup>

<sup>1</sup>Instituto de Informática – Pontifícia Universidade Católica de Minas Gerais (PUC Minas)

<sup>2</sup>Center for Research on Musical Gesture and Expression (CEGEM)  
Universidade Federal de Minas Gerais,

hugo@pucminas.br, {mauricioloureiro, eulercft}@ufmg.br

**Abstract.** *This paper presents a methodology for gesture segmentation of musical performances. It uses a novel approach based on cumulative dissimilarity (CDIS) to determine the velocity of the movement, as well as the local displacement of the musician during a musical gesture. A segmentation algorithm is applied along the dissimilarity vector to provide information about musical gestures. These movement segments are then compared to previous studies of movement analysis and correlated to onsets, and velocity data. During experiments, acoustical and motion data of clarinet performances were analyzed. Results obtained from motion analysis of the clarinet bell showed that, despite CDIS being a 1-dimensional descriptor, it retains information about movement direction and shape and is correlated to expressive musical content.*

## 1. Introduction

According to Mitra and Acharya, (2007), “gestures are expressive, meaningful body motions involving physical movements of the fingers, hands, arms, head, face, or body with the intent of conveying meaningful information or interacting with the environment”. Gestures are studied in many different fields. In music, effective gestures are those that play a direct role in the production of sound, while ancillary gestures refer to body movements that are not involved in production of sound (Wanderley, 1999). During clarinet performances, it has been shown that ancillary gestures are not only frequent, but correlate with several audio cues that suggest its relation to musical expressive content (Teixeira et al., 2014).

Many studies have tackled expressiveness throughout analysis of acoustical data, such as energy envelope, pitch, note onsets and offsets, attack duration and spectral centroid. Some examples of such an approach can be seen in Campolina et al. (2009), Maestre and Gómez (2005) and De Poli et al. (2004). These studies have shown that musicians make use of small deviations, regarding note durations, articulations, intensity, pitch and timbre, in order to convey their musical intentions (Juslin, 2000).

Fenza et al. (2005) used a 3 layer motion processing to segment movements based on Quantity of Motion, and investigated gesture mapping in a 3D expressive space. Camurri et al. (2004) presented the gesture as a conveyor of information related to the emotional domain in dance or music performances. They identified descriptors

related to specific features of trajectory patterns, such as angularity, spatial occupation, symmetry, and others. Several other fields of research have successfully employed multi-modal analysis to study the coupling between the acoustical and visual components, such as in speech analysis (Barbosa et al., 2008) and dance (Naveda and Leman, 2009).

Wanderley et al. (2005) and Vines et al. (2006) showed that body movements are part of a performance, and some gestures are not produced for the purpose of sound generation, which they designated as ancillary gestures. They also observed that these gestures were closely related to the musician's expressive intentions in a particular performance.

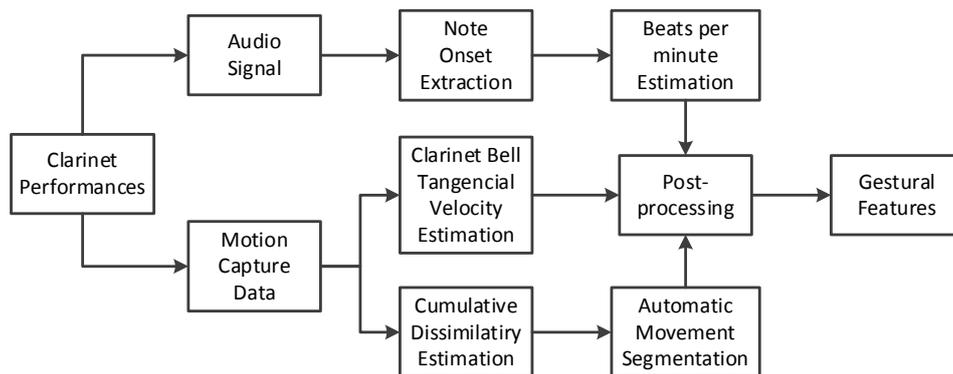
Recent studies have searched for methods to extract and analyze these movements in detail. Teixeira et al. (2010) proposes a tangential velocity based model to represent, segment and analyze expressive movements based on local gestural parameters. More recently, Teixeira et al. (2014) presented recurrent sequences of clarinet gestures in regions of the excerpts that related to musical structure. Teixeira et al. (2015) showed direct correlations between the recurrence pattern of clarinetists' ancillary movements and expressive timing associated with melodic phrasing and harmonic transitions. Several studies search for methodologies for segmenting musical gestures, in order to establish correspondence to musical structure. However, there are still no basic units established to segment body movements of music performances, unlike the audio data.

Caramiaux et al. (2012) used Hidden Markov Models (HMM) to segment movements into primitive shapes, selected from a base dictionary. Desmet, Nijs, Demey, Lesaffre, Martens and Leman (2012) proposed another statistical model for body movement segmentation and pointed to subjective links between these segments and the musical score. Rasamimanana (2012) defined a conceptual framework considering performer-instrument relationships that can provide ground to model expressive gestures using a space of possibilities.

In this study, we present a procedure for automatic movement segmentation in order to extract clarinet players' physical gestures during performances of pieces of the classical repertoire. We aim at evaluating gesture segmentation of several clarinetists, and with the purpose of relating their gestural patterns to the music structure.

## **2. Methodology**

The main goal of this study is to evaluate a new method for movement segmentation of musical performances by a group of clarinet players contained in the same dataset presented by Teixeira et al. (2014). Figure 1 presents a schematic diagram of the procedure. As seen in the diagram, acoustical and motion data were obtained from clarinet performances. Several features were then calculated based on these data. After calculation of the cumulative dissimilarity vector, several procedures such as thresholding and filtering are applied to segment the musicians' body movement into gestures. This section presents how these features are obtained and combined to allow a fully automatic detection of musical gestures.



**Figure 1. Movement segmentation procedure diagram.**

Ten professional clarinet players performed a short excerpt of six bars extracted from the first movement of the Quintet for Clarinet and Strings in A Major, Kv 581 by W.A. Mozart (Figure 2). The musicians were asked to play according to two distinct experimental conditions: expressive performances as in a real concert situation (standard condition) and following a metronome, set to a tempo estimated from his/her previous standard performances (metronome condition). Each of the ten clarinet players performed this excerpt six times without accompaniment, three performances for each of the two experimental conditions. Motion tracking was done with high-end 3D motion capture devices, the NDI Optotrak Certus and the NDI Optotrak 3020. Motion capture markers were placed on their bodies and instruments. Detailed information and pictures about experiment setup, such as marker positions can be obtained in Teixeira et al. (2014).

Motion was captured at a sampling rate of 100 frames per second. Audio was recorded synchronously at a sampling rate of 44.1 kHz using a condenser microphone positioned one meter away from the clarinet. Pitch and energy envelope curves were extracted, and from these all note onsets and offsets were detected using the system described in Campolina, Loureiro and Mota (2009).



**Figure 2. Main theme from first movement of Mozart's Quintet for Clarinet and Strings in A Major, Kv 581, performed in the first experiment.**

## 2.1. Movement representation

Movement analysis in this study is based on the clarinet bell movement. The clarinet movement has been the object of previous studies (Caramiaux, Wanderley and Bevilacqua, 2012; Wanderley, 2002; Wanderley, Vines, Middleton, McKay and Hatch, 2005) and it is believed to be an important indicator of expressive movements made by the musician. The segmentation methods presented by these approaches are mainly based on mapping or recognition from a dictionary of shapes (Caramiaux et al., 2012; Vatavu et al., 2009), or low level features such as velocity or Quantity of Motion (Fenza et al., 2005).

The present study proposes a novel approach that abstracts from shape mapping but considers movement direction in a cumulative dissimilarity measure of movement

which might be more effective for establishing correspondence between gesture and audio musical content. Previous studies suggested the clarinet bell movement as an indicator of expressive movements of clarinet performance (Wanderley, 2002; Wanderley et al., 2005). Thus, the movement of the clarinet bell was taken relative to a coordinate system, located in the center of the Optotrak tracker. With a static reference, instead of a dynamic one, such as the mouthpiece, the clarinet bell movement incorporates any movement performed by the musicians with their feet, knees, torso, neck and arms, and can thus be seen as a general indicator of the players' movements. Optical flow techniques were already used to define a general motion indicator (Barbosa, Yehia and Vatikotis-Bateson, 2008). The analysis of a single tracking point, the clarinet bell, offered more precision and allowed to include its 3D trajectory, in order to define recurrent gestures and many associated gestural features.

The clarinet bell's tridimensional motion is represented by a matrix of dimension  $T \times M$ , where  $T$  represents the number of time frames and  $M$  represents the 3-dimensional markers' positions. We used three markers (top, left and right) to capture bell's translation and rotation in X, Y, and Z coordinates (Figure 3). Previous analyses showed that these markers are highly correlated and do not add in precision to the expressive gesture segmentation, probably due to the fact that the extent of rotation is small compared to the extent of translation. X and Y axes presented a correlation coefficient of 99.6% among markers and Z axis 92%. Correlation index of combined X and Y axes ranged between  $\pm 4\%$ .

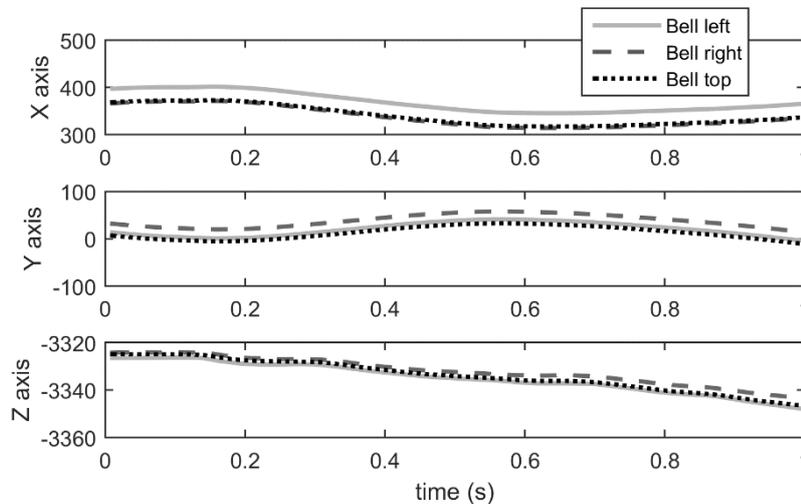


Figure 3. Trajectories of clarinet bell's top, left and right markers.

## 2.2. Movement processing

In order to analyze the evolution of the clarinet bell's tridimensional motion in conjunction with the acoustical data, we need an effective scalar representation of the motion data in time. Fenza et al. (2005) call these low-level features related to motion description. A simple solution presented by Teixeira, Loureiro and Yehia (2010) is to use the tangential velocity of the clarinet bell marker's trajectory, estimated by the Euclidian distance between subsequent samples of the positions of this marker. Although this unidimensional parameter captures a large amount of information from the musician's movements (Teixeira, Loureiro and Yehia, 2010), it does not contain information about the direction or the shape of the movement. For instance, the

difference between an up and down movement and a long ascending trajectory made by the clarinet player would not be revealed solely by the tangential velocity. The same problem would happen with the low-level feature Quantity of Motion (QoM) from Fenza et al. (2005). However, Fenza and colleagues combine other low-level features to help describe shape and direction of movements, which are: Contraction Index (CI), Movement Length (ML), Straight trajectory Length (SL), and Directness Index (DI).

While the estimation of note onsets and offsets from pitch and energy envelope curves extracted from the audio signal, allows the segmentation of the acoustical data into musical notes and phrases, there are no basic units established to segment the movements into gestures. It is possible to develop a procedure to segment the movement data accordingly, by subdividing those movements into representative segments, based on their geometrical and temporal attributes.

### 2.3. $r$ -Cumulative dissimilarity ( $r$ -CDIS)

To measure the amount of movement within a timeframe we present a new strategy based on a cumulative dissimilarity measure. Patrocínio Jr. et al. (2010) originally presented cumulative dissimilarity to detect gradual transitions on video sequences. The main idea behind cumulative dissimilarity is that small movements around the same region in the feature space are meaningless, but small movements consistently driving towards a new region in that space might mean a transition between different scenes.

Given a vector of elements ( $v$ ) and a  $2r$ -sized sliding window centered at position  $k$ , the  $r$ -cumulative dissimilarity  $r$ -CDIS $_k$  can be calculated as:

$$r\text{-CDIS}_k = \sum_{i=k-r+1}^k \sum_{j=k+1}^{k+r} \text{DIS}(v_i, v_j),$$

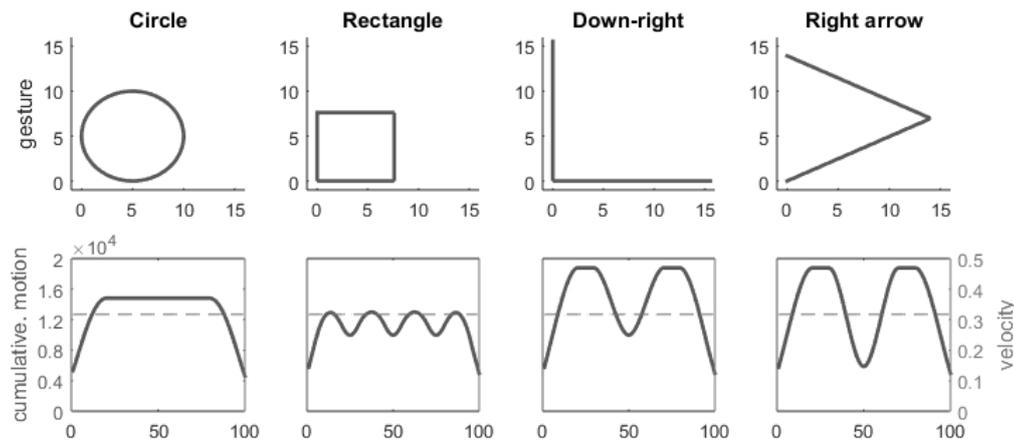
where  $\text{DIS}(v_i, v_j)$  is a dissimilarity measure between vector elements  $v_i$  and  $v_j$ .

Several different dissimilarity measures can be employed to calculate cumulative distance curve. In this work, Euclidean distance will be used as a dissimilarity measure calculated as:

$$\text{DIS}(v_i, v_j) = \sum_{m=1}^M (v_{j,m} - v_{i,m})^2$$

where  $M$  is the number of marker positions as stated in Section 2.1.

The main advantage of using a cumulative dissimilarity approach is that it can detect long gradual displacements during a fixed period and is resilient to small constant movements that go back and forth around the same region. Figure 4 clarifies this character. Figure 4 (upper line) presents four of the ten common gestures defined by Vatavu, Grisoni and Pentiu (2009). All movements were designed to have exactly the same overall length (100 points) and same tangential velocity. The only parameter that changes among them is the direction of the movement. Lower plots show tangential velocity and cumulative dissimilarity measures for all the four movements. Cumulative dissimilarity was calculated based on a 20-point sliding window and first and last points were replicated to avoid border effects.



**Figure 4.** Lower panel show cumulative dissimilarity (10-CDIS) and tangential velocity (dashed line) of common gestures (top panel). Adapted from Vatavu, Grisoni and Pentiu (2009), Fig. 7, p. 9.

It can be seen in Figure 4 that longer movements result in higher cumulative dissimilarity on average. This feature also captures the change in direction: the higher the change in direction the lower the cumulative dissimilarity value. These properties allow for better characterization of movements and provide a base for gesture segmentation.

## 2.4. Movement segmentation

Movement segmentation is performed in four steps. Firstly,  $r$ -CDIS<sub>*k*</sub> of Euclidean distance is estimated in every point of the  $T \times M$  matrix of markers positions. The result is a vector of length  $T$  of  $r$ -CDIS<sub>*k*</sub>, for  $k \in [1..T]$ . The second step is the thresholding: every point in vector below a certain threshold is set to zero. This step is important to eliminate small movements that are constrained in space, especially those movements from balance and posture control (Winter, 1995). Afterwards, the 1<sup>st</sup> derivative is calculated and used to estimate limits, considered as movements onsets and offsets, analogous to notes onsets and offsets, upon which movement lengths are estimated. A final step filters movement segments that were considered too short according to a pre-defined note duration related to the estimated musical tempo in beats per minute (BPM).

## 3. Experimental results

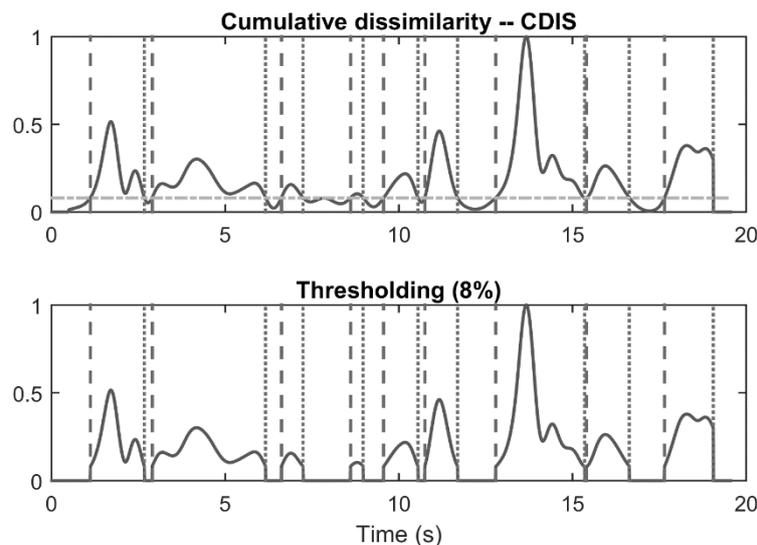
Table 1 summarizes the recording dataset of 60 performances, as previously described, of an average length of 18.7 seconds, summing up almost 23 minutes of recorded music.

**Table 1. Description of the recording dataset.**

Recordings		Recording length (seconds)		Tempo (BPM)	
# Clarinetists	10	Max	22.1	Max	128.6
# free expressive performances	3	Min	16.5	Min	85.6
# metronome restricted performance	3	Mean	18.7	Mean	113.9
Total # of recordings	60	Total	1,126.6		

Figure 5 shows the thresholding procedure. Cumulative dissimilarity vector is represented by a 50-CDIS ( $r = 50$ ), which means that a window of length 100 was used. Since the motion capture sampling rate is 100 samples/second, window length

corresponded to exactly 1 second. According to Patrocínio Jr. et al. (2010),  $r$ -CDIS is specially good in finding segments of length  $2r$ . Larger values of  $r$  flattens the CDIS curve, making it harder to segment and losing precision in gesture position estimation, since it might sum up more than one gesture altogether. The threshold value used in all 50-CDIS curves was 8%, which was empirically obtained. Larger threshold values reduce the average length of motion segments, eliminating small segments and increasing the number of medium sized segments. On the other hand, smaller threshold values produce several meaningless micro-movements, and extra long macro-movements.



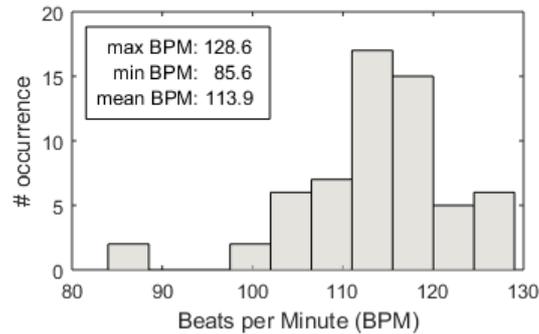
**Figure 5. Thresholding procedure. Horizontal dashed line represents threshold (8%). Vertical lines represent onsets (dashed lines) and offsets (dotted lines). Continuous line represents normalized 50-CDIS as a function of time (s).**

After the thresholding, CDIS vector was segmented and movement onsets and offsets were determined. First two rows of Table 2 show the results obtained from segmentation. The smallest detected movement has only 30 milliseconds, which is shorter than a sixty-fourth note in an *Allegro* tempo of 120 BPM. The largest segment was more than 15 seconds long. At first, we will consider that long movements occur when the musician keeps moving continuously and we will not force any further segmentation. However, we considered that too short segments do not correlate to music expressiveness and therefore should be removed by considering that ancillary gestures shorter than a sixteenth note are not long enough to convey expressive content.

**Table 2. Segmentation and filtering results**

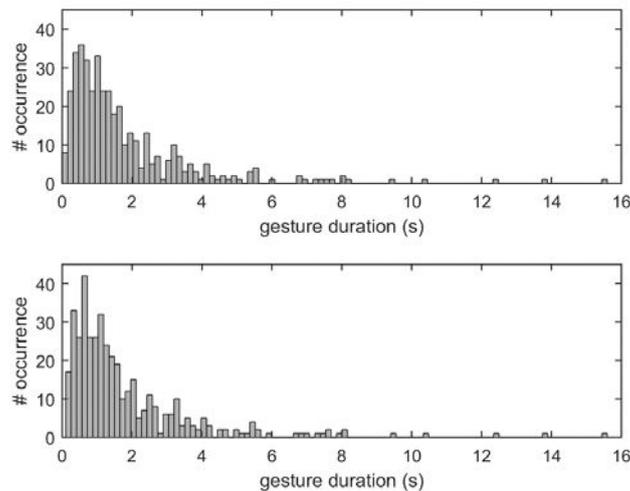
Variables	Minimum value	Maximum value	Mean value
Duration all segments (s)	0.03	15.53	1.79
# Segments / performance	1	11	6.9
Duration long segments (s)	0.13	15.53	1.81
# Long segments / performance	1	11	6.8

The cut-off length was estimated by the duration of the sixteenth note for each performance tempo in BPM, estimated from note onsets. Figure 6 shows a histogram of the estimated tempi. One performer played the excerpt in tempo *Andante*, much slower than the expected *Allegro*.



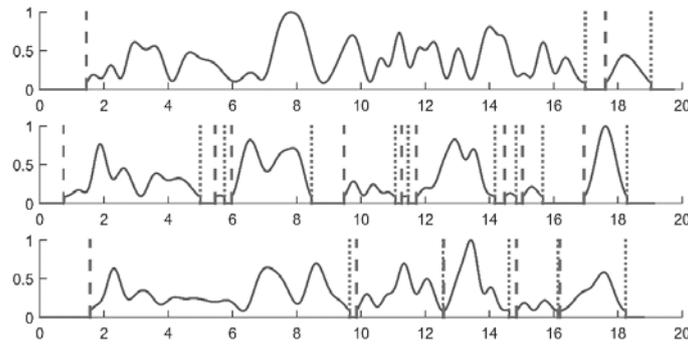
**Figure 6. Histogram of recording tempos.**

Figure 7 shows histograms of gesture duration. Upper plot presents unfiltered gesture segments and lower plot segments after removal of short segments. It can be clearly seen that there are 5 large segments above 9 seconds long, in a total of 413 detected gestures. 127 segments (30%) have duration between 1 and 2 seconds. Lower plot shows results after filtering. Only 5 segments were removed and length distribution remained almost the same.

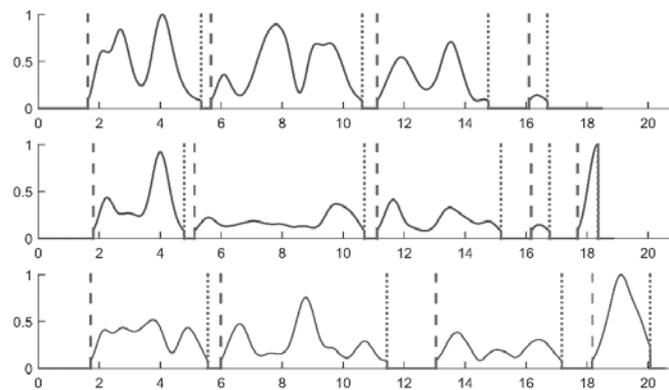


**Figure 7. Histogram of gesture durations. Upper plot shows 413 unfiltered segments. Lower plot shows 408 BPM-filtered segments.**

Figure 8 shows a sample segmentation of three expressive performances played by the same clarinetist. This clarinetist produced the largest (upper plot) and the smallest (middle plot) segments compared to all recordings. It shows how sensitive to the thresholding the segmentation procedure is. However, Figure 9 show other three expressive performances played by another clarinetist. It can be seen that this clarinetist has a steady behavior that was perfectly captured by the segmentation. In this case, all performances were segmented in four parts (the fifth part in the middle plot is actually silence, after the musician has finished playing). The four segments were also proportionally alike.



**Figure 8. Sample segmentation of three free expressive performances from the same performer showing limitations of the model. Vertical lines represent movement onsets (dashed lines) and offsets (dotted lines). Continuous line represents normalized 50-CDIS as a function of time (s).**

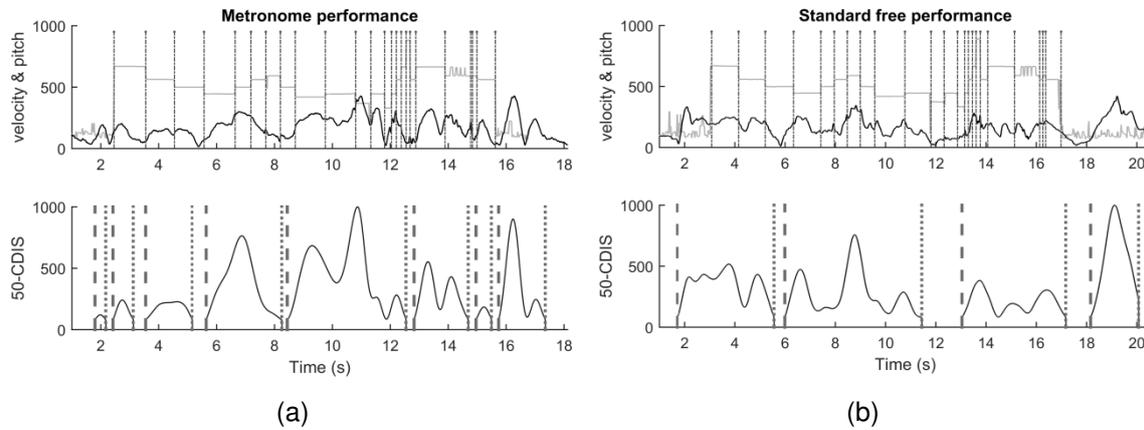


**Figure 9. Sample segmentation of three free expressive performances from a steady performer. Vertical lines represent movement onsets (dashed lines) and offsets (dotted lines). Continuous line represents normalized 50-CDIS as a function of time (s).**

Figure 10 compares gesture segments to note onsets and tangential velocity. Pitch information is also presented to provide a better understanding of the performance. Figure 10 (a) shows the results obtained for a performance constrained by the metronome. It is important to note the correlation between tangential velocity and cumulative dissimilarity. However, CDIS curve eliminate local information about small movements and outstands the extent of the gesture as related to bell displacement. Visual analysis of that Figure also suggests correlation between gesture onsets and note onsets. Usually, gesture starts close to a note onset, but end earlier. An amplitude envelope of the note could possibly show some correlation to the decay of the gesture. It is not our goal yet to analyze the correlation between the gesture amplitude and musical content, but this must be further investigated.

Figure 10 (b) shows the results obtained for an expressive performance. It is interesting to note that the gestures obtained by this segmentation procedure show correlation to some points of the music structure, especially to ascending and descending sequences. Usually, the gesture starts one note before the beginning of the sequence suggesting that such musical movements are being anticipated by the musician. The third gesture,

between seconds 13 and 17 is almost synchronized to the last *legato* ascending arpeggio that leads to the perfect cadence that closes the sentence.



**Figure 10. Gesture segmentation using 50-CDIS and 8% threshold compared to note onsets and tangential velocity. Vertical lines represent movement onsets (dashed lines) and offsets (dotted lines).**

All CDIS plots were normalized in order to give a better understanding of the segmentation process, including the thresholding procedure. However, gesture amplitude may play an important role in future gesture analyzes.

#### 4. Conclusion and Further Works

The main goal of this work was to provide a new way for automatic segmentation of body movement of musical performances into musical gestures. A method based on cumulative dissimilarity (CDIS) was presented. Sixty clarinet recordings were analyzed using 50-CDIS and thresholding of 8%. BPM based filtering allowed removal of meaningless short movements. The main contribution of our work is the application of a simple distance measure as a descriptor of musical gestures. Despite CDIS being a 1-dimensional descriptor, it retained some information about movement direction and shape in its value. Experimental results showed some correlation between gestures and some musical passages, suggesting that these gestures might contribute to convey performers' expressive intention.

However, segmentation results can be highly dependent on CDIS radius and threshold values. So, as a future work, we plan to investigate thoroughly the effect of these parameters in gesture segmentation and devise an automatic approach to parameter estimation. We also intend to further investigate the importance of gesture amplitude and its correlation to expressive musical content.

#### 5. Acknowledgements

The authors would like to thank the Brazilian research agencies CNPq, CAPES and FAPEMIG for funding this study.

#### 6. References

- Barbosa, A., Yehia, H., and Vatikiotis-Bateson, E. (2008). "Temporal characterization of auditory-visual coupling in speech". *Proceedings of Meetings on Acoustics*, 1, pp. 1–14.

- Campolina, T., Loureiro, M., and Mota, D. (2009). “Expan: a tool for musical expressiveness analysis”. In L. Nevada (Ed.), *Proceedings of the 2nd International Conference of Students of Systematic Musicology*. Ghent: IPEM – Institute for Psychoacoustics and Electronic Music. pp. 24–27.
- Camurri, A., Mazzarino, B., Ricchetti, M., Timmers, R., and Volpe, G. (2004). “Multimodal analysis of expressive gesture in music and dance performances”. In A. Camurri, & G. Volpe (Eds.), *Gesture-based communication in human-computer interaction*. Berlin: Springer. pp. 20–39.
- Caramiaux, B., Wanderley, M., and Bevilacqua, F. (2012). “Segmenting and parsing instrumentalists’ gestures”. *Journal of New Music Research*, 41 (1), pp. 13–29.
- De Poli, G., Canazza, S., Drioli, C., Rodà, A., and Vidolin, A. (2004). “Modeling and control of expressiveness in music performance”. *Proceedings of the IEEE*, 92 (4), pp. 686–701.
- Desmet, F., Nijs, L., Demey, M., Lesaffre, M., Martens, J., and Leman, M. (2012). “Assessing a clarinet player’s performer gestures in relation to locally intended musical targets”. *Journal of New Music Research*, 41 (1), pp. 31–48.
- Fenza, D., Mion, L., Canazza, S. and Rodà, A. (2005). “Physical movement and musical gestures: A multilevel mapping strategy”. *Proceedings of Sound and Music Computing*. pp. 1–6.
- Juslin, P. (2000). “Cue utilization in communication of emotion in music performance: Relating performance to perception”. *Journal of Experimental Psychology: Human Perception and Performance*, 26, pp. 1797–1813.
- Maestre, E. and Gómez, E. (2005). “Automatic characterization of dynamics and articulation of expressive monophonic recordings”. *Audio Engineering Society 118th Convention Paper 8*. New York: Audio Engineering Society (AES).
- Mitra, S., and Acharya, T. (2007). “Gesture recognition: A survey”. *IEEE Transactions on Systems, Man and Cybernetics*, 37 (3), pp. 311–324.
- Naveda, L., and Leman, M. (2009). “A cross-modal heuristic for periodic pattern analysis of samba music and dance”. *Journal of New Music Research*, 38 (3), pp. 255–283.
- Patrocínio Jr., Z. K. G., Guimarães, S. J. F. Silva, H. B. and Souza, K. J. F. (2010) “A unified transition detection based on bipartite graph matching approach”. In: BLOCH, I.; CESAR, R. M. (Ed.). *Iberoamerican Congress on Pattern Recognition*. [S.l.]: Springer. (Lecture Notes in Computer Science, v. 6419). ISBN 978-3-642-16686-0. pp. 184–192.
- Rasamimanana, N. (2012). “Towards a conceptual framework for exploring and modelling expressive musical gestures”. *Journal of New Music Research*, 41 (1), pp. 3–12.
- Rodger, M. W. M., O’Modhrain, S., and Craig, C. M. (2013). “Temporal guidance of musicians’ performance movement is an acquired skill”. *Experimental brain research*. 226 (2), pp. 221–30.

- Teixeira, E., Loureiro, M. and Yehia, H. (2010). “Methodological aspects of the research in musical expressiveness based on corporal movement information”. Unpublished Report. Retrieved from <http://hal.archives-ouvertes.fr/hal-00611660/en/>
- Teixeira, E. C.F., Loureiro, M. A., Wanderley, M. M. and Yehia, H. C. (2014). “Motion Analysis of Clarinet Performers”. *Journal of New Music Research*, 44 DOI: 10.1080/09298215.2014.925939. pp. 97–111.
- Teixeira, E. C. F., Yehia, H. C. and Loureiro, M. A. (2015). “Relating movement recurrence and expressive timing patterns in music performances”. *The Journal of the Acoustical Society of America*, 138, EL212-EL216, DOI:10.1121/1.4929621.
- Vatavu, R.-D., Grisoni, L., and Pentiu, S.-G. (2009). “Gesture Recognition Based on Elastic Deformation Energies”. In *Gesture-Based Human-Computer Interaction and Simulation*, Miguel Sales Dias, Sylvie Gibet, Marcelo M. Wanderley, and Rafael Bastos (Eds.). *Lecture Notes In Artificial Intelligence*, Vol. 5085. Springer-Verlag, Berlin, Heidelberg DOI: 10.1007/978-3-540-92865-2\_1. pp. 1–12.
- Vines, B., Dalca, I. and Wanderley, M. (2006). “Variation in expressive physical gestures of clarinetists”. Baroni, M., Addessi, A.R., Caterina, R. & Costa, M. eds. *Proceedings of the 9th International Conference on Music Perception and Cognition*. New York: The Society for Music Perception & Cognition (SMPC) and Jyväskylä: European Society for the Cognitive Sciences of Music (ESCOM). pp. 1721–1722.
- Wanderley, M. (1999). “Non-obvious performer gestures in instrumental music”. *Gesture-based Communication in Human–Computer Interaction*, 1. pp. 37–48.
- Wanderley, M. (2002). “Quantitative analysis of non-obvious performer gestures”. In I. Wachsmuth, & T. Sowa (Eds.), *Gesture and sign language in human-computer interaction*. Berlin: Springer. pp. 241–253.
- Wanderley, M., Vines, B., Middleton, N., McKay, C., and Hatch, W. (2005). “The musical significance of clarinetists’ ancillary gestures: An exploration of the field”. *Journal of New Music Research*, 34 (1), pp. 97–113.
- Winter, D. A. (1995). “Human balance and posture control during standing and walking”, *Gait & Posture*, 3 (4). ISSN 0966-6362, DOI: 10.1016/0966-6362(96)82849-9. pp. 193–214.

# Detecção de Refrão em Sinais de Áudio usando Extração de Características de Intensidade do Som

Renato Celso Santos Rodrigues, Geber Ramalho, Giordano Cabral

Centro de Informática – Universidade Federal de Pernambuco (UFPE)  
Caixa Postal 7851 – 50.732-970 – Recife – PE – Brasil

{rcsr, glr, grec}@cin.ufpe.br

**Abstract.** *This paper proposes a paradigm shift for chorus detection, based on the exploitation of the time domain rather than the frequency domain. State of the Art methods generally segment the signal by exploiting the presence of musical notes to measure the similarity between the music sections with Euclidean distance between Chroma and MFCC vectors. The proposed method eliminates the segmentation step and extracts the envelope of the signal, measuring the similarity between the envelopes of the parts of the song by correlation function, rendering the method independent of the presence of musical notes in the signal. It was carried out a comparative study of current approaches and the proposed method, highlighting their differences, advantages and disadvantages. The results indicate that the use of correlation function for the envelope signal achieves hit rate and performance at the same magnitude of current methods.*

**Resumo.** *Este trabalho propõe uma mudança de paradigma para a solução do problema da detecção de refrão, baseada na exploração do domínio do tempo em lugar do domínio da frequência. Métodos do Estado da Arte geralmente segmentam o sinal, explorando a presença de notas musicais para medir a similaridade entre os trechos da música com distância euclidiana entre vetores Chroma e MFCC. O método proposto elimina a etapa de segmentação e extrai a envoltória do sinal, medindo a similaridade entre as envoltórias das partes da música por função de correlação, tornando o método independente da presença de notas musicais no sinal. Foi realizado estudo comparativo entre as abordagens atuais e o método proposto, destacando suas diferenças, vantagens e desvantagens. Os resultados indicam que a utilização de função de correlação sobre a envoltória do sinal indica obtém taxa de acertos e desempenho na mesma ordem de grandeza dos métodos atuais.*

## 1. Introdução

Segundo Goto (2006), o refrão (*chorus* ou *refrain*, na literatura) é a parte mais repetida e memorável de uma música. É o trecho mais proeminente e representativo de uma canção, mais facilmente reconhecível e memorizável pelos ouvintes. Desta forma, o refrão é capaz de representar a música em um contexto de rápida visualização de álbuns musicais, como em aplicações de navegação ou recuperação de música.

Uma função de Preview oferece ao ouvinte uma pré-visualização de um álbum com pequenos trechos de suas músicas. É necessário oferecer ao ouvinte o trecho mais representativo de cada música, sendo muito útil para isso a detecção de refrão. Neste contexto, o objetivo de um sistema de detecção de refrão é identificar os pontos

extremos do refrão no eixo temporal ao longo da música: o instante no qual ele começa; e o instante no qual ele termina.

Entretanto, detecção de refrão não é simples. Embora seja intuitivo ao ser humano identificar o trecho mais representativo de uma música, não é fácil definir precisamente onde ele começa e onde termina. Para um computador, esta imprecisão na marcação dos extremos do refrão pode causar uma diminuição da taxa de acertos, sobretudo quando se utiliza como métrica de sucesso uma medida baseada na comparação entre o refrão retornado pela aplicação com o refrão anotado, previamente definido em uma marcação manual.

Outra dificuldade é que a detecção de refrão é baseada na identificação de partes similares entre si ao longo da música, mas uma ocorrência do refrão pode apresentar diferenças em relação às demais. Dependendo da forma como esta similaridade é medida, as diferenças entre as ocorrências dos refrãos, como a presença ou ausência de instrumentos específicos, ou a inclusão de arranjos musicais distintos, ou ainda diferenças na intensidade do som, podem reduzir o grau de similaridade entre um par de ocorrências do refrão na música.

Uma solução para o problema da detecção de refrão deve ter baixa taxa de erros e alto desempenho. Deve ser também robusta, com uma taxa de acertos e desempenho que não variem em função de características peculiares ao estilo musical, como o ritmo, por exemplo.

Neste contexto, o objetivo deste trabalho é propor uma abordagem alternativa para a detecção de refrão, testando-a experimentalmente e realizando um estudo comparativo, destacando suas diferenças, vantagens e desvantagens em relação às soluções do Estado da Arte. Esta mudança de paradigma é realizada pela substituição de alguns dos passos dos métodos de Goto e Eronen por outros que explorem formas diferentes de executar a mesma tarefa.

## 2. Estado da Arte

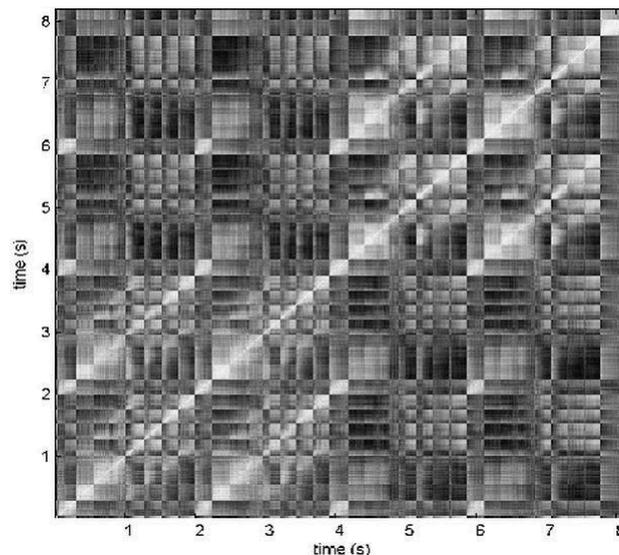
As duas principais abordagens para solução do problema da detecção de refrão são os trabalhos de Goto (2006) e Eronen (2007). Ambas começam com a segmentação do sinal, dividindo-o em fragmentos que serão comparados entre si, medindo-se o grau de similaridade entre cada par destes segmentos. Este processo realiza uma busca pelas batidas da música, mais evidentes em função dos instrumentos percussivos, detectadas conforme trabalho de Seppänen (2006), onde cada segmento da música é o trecho entre duas batidas consecutivas. Esta técnica obtém o menor número possível de fragmentos do sinal que ainda preserva a precisão de uma solução baseada na extração da frequência dos sons, reduzindo o seu custo nas etapas seguintes. Porém, o desempenho da solução torna-se uma função da frequência das batidas da música, porque músicas mais aceleradas serão divididas em um número maior de segmentos.

Realiza-se então a extração de características musicais de cada segmento, úteis no cálculo da similaridade entre cada dois segmentos do sinal. As características adotadas foram o Chroma Vector para ambos os métodos e ainda o vetor MFCC para o método de Eronen, ambos baseados na frequência do sinal em cada segmento. O Chroma Vector, construído conforme o trabalho de Goto (2006), possui 12 posições, uma para cada Pitch Class, que contém a soma das magnitudes das frequências de cada

Pitch Class em 6 oitavas, no espectro obtido por FFT em uma janela deslizante ao longo do segmento.

Depois, usa-se distância euclidiana para medir as similaridades entre estes vetores, onde uma menor distância significa uma maior similaridade. Uma desvantagem neste passo é a necessidade de tratar as modulações (mudanças na tonalidade da música em tempo de execução), porque elas anulam a similaridade entre refrãos executados em tonalidades diferentes. Este tratamento acrescenta um pouco de complexidade e custo ao algoritmo da solução, embora não seja difícil de ser realizado.

Após medir as distâncias, Eronen (2007) armazena-as em uma matriz de similaridade, que terá em cada posição  $A_{ij}$  da matriz o valor da similaridade entre os segmentos  $i$  e  $j$  do sinal. Goto (2006) utiliza um triângulo similar, que armazena a similaridade entre dois trechos em função também da distância entre eles no tempo da música. Uma diagonal de alta similaridade nesta matriz (ou uma linha de alta similaridade no triângulo) identifica uma repetição entre dois trechos da música. Esta técnica permite a utilização de filtros de processamento de imagens para refinar a detecção destas diagonais (ou linhas) pelo realce de seus valores em relação aos vizinhos, ou pela eliminação de valores espúrios. A Figura 1 traz um exemplo de matriz de similaridade.



**Figura 1. Exemplo de matriz de similaridade.**

No final de ambos os métodos são identificadas as repetições através da dicotomização da matriz (ou triângulo) com a anulação de valores abaixo de um limiar. Depois, sobre cada repetição são aplicadas heurísticas que calculam a probabilidade dela ser uma correspondência entre dois refrãos da música. A repetição com maior probabilidade terá um de seus trechos apontado como o refrão da música.

As heurísticas variam de abordagem para abordagem, e dependem de observações realizadas pelo autor do método sobre os refrãos das canções. Eronen (2007) adotou quatro heurísticas: uma repetição que contém refrãos provavelmente encontra-se próximo de um quarto e três quartos do tempo da música, possui alta similaridade média, alta intensidade média, e um dos trechos presente em outras repetições. Goto (2006) adotou três heurísticas: uma repetição que contém refrãos

provavelmente tem entre 8 e 40 segundos de duração, é inclusa em outra repetição maior e contém duas outras repetições menores.

Ambas as abordagens operam com uma taxa de acerto entre 80% e 85% em bases de dados específicas a cada autor. A base usada por Goto é constituída por 100 músicas populares, enquanto a usada por Eronen tem 206 músicas populares e rock. Os métodos de ambos trabalham retornando o refrão em 10 a 15 segundos para uma música de aproximadamente 3 a 4 minutos, para um computador de configuração mediana, acessível a um usuário comum.

### 3. O Método

A principal mudança proposta no método em relação às abordagens de Goto e Eronen é a substituição da extração de características do Chroma Vector e do vetor MFCC pela extração da envoltória do sinal, que é a curva que descreve a variação de energia do sinal ao longo do tempo. Esta mudança torna o método independente das frequências das notas musicais do sinal, dispensando o tratamento de modulações.

O sinal de áudio musical tem sua envoltória modulada por ondas na faixa de frequência audível pelo ouvido humano, resultante das frequências de suas notas. Cada vez que um trecho da música é executado, não apenas as frequências das notas musicais estarão presentes, mas também a contribuição de cada nota musical para a variação de amplitude do sinal. Assim, a envoltória do sinal também conserva a similaridade existente entre os trechos que se repetem, independente da frequência que a modulou.

Para medir a similaridade entre os trechos da música, o método utiliza o máximo da função de Correlação (Equação 1), uma vez que a filtragem das frequências de modulação do sinal impossibilita a utilização da distância euclidiana entre vetores Chroma e MFCC. Esta função tem máximos quando, para um dado lag  $\tau$ , o sinal  $x$  é similar ao sinal  $y$ .

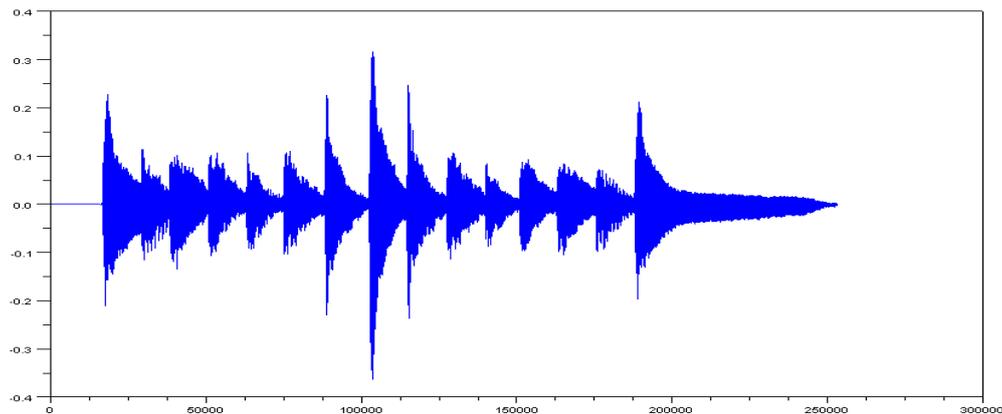
$$R_x(\tau) = \int_{t'}^{t'+T} x(t + \tau)y(t)dt, \tau \text{ constante} \quad (1)$$

Outra mudança proposta é a eliminação da etapa de segmentação, que simplifica o método, remove o custo empregado na detecção de batidas, torna o seu tempo de resposta independente da frequência destas batidas, e possibilita uma maior precisão no corte dos extremos do refrão, que não estará mais limitado às batidas do sinal.

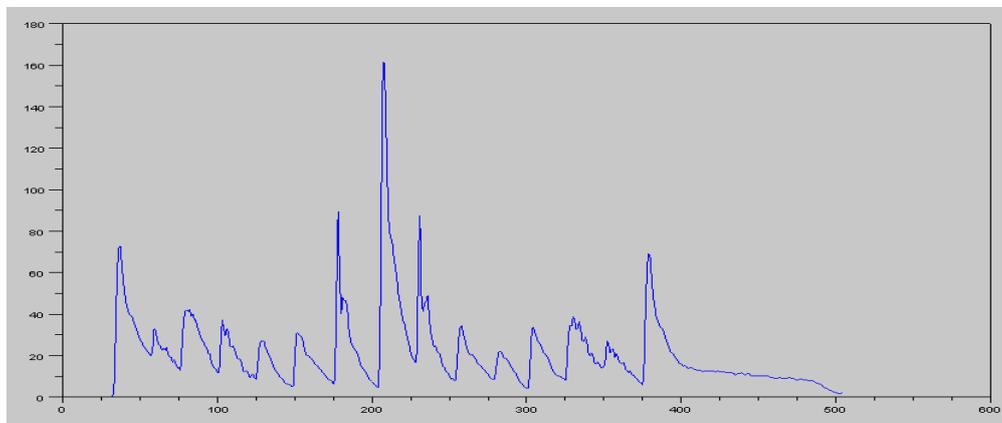
O restante desta seção descreve a sequência de passos do método proposto.

#### 3.1. Extração da Envoltória do Sinal de Áudio

A extração da Envoltória do sinal utiliza uma janela deslizante que se desloca ao longo do sinal recuperando o valor máximo deste frame em cada iteração. A sequência destes valores é armazenada em outro sinal que, com o devido ajuste do tamanho da janela e do valor do deslocamento, conterà a envoltória do sinal de áudio, que é a descrição aproximada da variação de sua energia ao longo do tempo, mas com um número de amostras menor. O tamanho do sinal resultante será aproximadamente o valor da razão entre o tamanho do sinal de áudio pelo deslocamento da janela adotado, ambos medidos em números de amostras digitais do sinal original. A Figura 2 possui um gráfico com 8 notas consecutivas tocadas por um violão no domínio do tempo. O sinal de envoltória obtido sobre este sinal encontra-se no gráfico da Figura 3.



**Figura 2. Sinal de áudio com 15 notas de um violão.**



**Figura 3. Envoltória do sinal de áudio do Gráfico 2.**

Observando o eixo horizontal de ambos os gráficos nota-se uma redução no tamanho do sinal de aproximadamente três ordens de grandeza, o que permite uma significativa redução de custos nas etapas seguintes deste método.

Quanto maior for o valor adotado para o deslocamento da janela deslizante, menor será o sinal de envoltória obtido, e quanto menor for o sinal de envoltória obtido, menor será o custo do processamento realizado sobre ele nas etapas seguintes do método. Entretanto, um deslocamento muito grande pode resultar em perda de informação da similaridade presente na envoltória, o que é indesejável. O valor do deslocamento adotado na abordagem foi 50 milissegundos, obtido empiricamente observando-se um bom compromisso entre a redução do tamanho e a preservação da similaridade.

Reduzir o tamanho da janela deslizante reduz o custo da construção do sinal de envoltória, mas aumenta a frequência de corte na obtenção da envoltória, permitindo que frequências mais baixas de notas musicais (mais graves) insiram ruídos na envoltória obtida. Mas uma janela muito grande perde em precisão, podendo incluir muitos padrões de variação de amplitude (ataque, decaimento, sustentação e relaxamento) consecutivos em uma só posição do sinal resultante, perdendo informação. Diante deste problema, o valor adotado foi 10 milissegundos.

### 3.2. Construção da Matriz de Similaridade

A matriz de similaridade é construída como no método de Eronen (2007), mas o cálculo de seus valores é diferente. Em cada posição da matriz de similaridade é armazenado, como métrica de similaridade, o máximo do sinal de correlação obtido tendo como entrada duas janelas deslizantes, que se deslocam ao longo do sinal de envoltória. O tamanho da janela e do deslocamento adotados foram 10 segundos e 1 segundo respectivamente (medidos em amostras do sinal de envoltória), obtidos empiricamente preservando um bom compromisso entre precisão, custo e taxa de acertos.

Para reduzir o custo do método nesta etapa é preenchido apenas do triângulo superior da matriz de similaridade. Uma vez que ela é simétrica, apenas um dos triângulos é necessário e suficiente para os processamentos posteriores, que somente utilizarão esta metade da matriz. A matriz à esquerda da Figura 3 é resultante deste processo sobre um sinal de áudio musical de aproximadamente 4 minutos.

### 3.3. Extração das Diagonais de Similaridade

As técnicas utilizadas neste passo são praticamente as mesmas utilizadas na abordagem de Eronen. A primeira é a filtragem de nitidez, que utiliza um kernel quadrado de 25 posições que se desloca por todo o triângulo superior da matriz e aumenta o valor da posição central do kernel em relação às demais se o seu valor de similaridade máxima estiver na sua diagonal principal, e diminui o valor deste elemento central em relação aos demais em caso contrário. Ao fim deste processo, espera-se que as diagonais de mais alta similaridade estejam mais nítidas em relação aos demais valores da matriz. Na Figura 4, a matriz de similaridade à direita é resultante desta filtragem de nitidez sobre a matriz à esquerda.

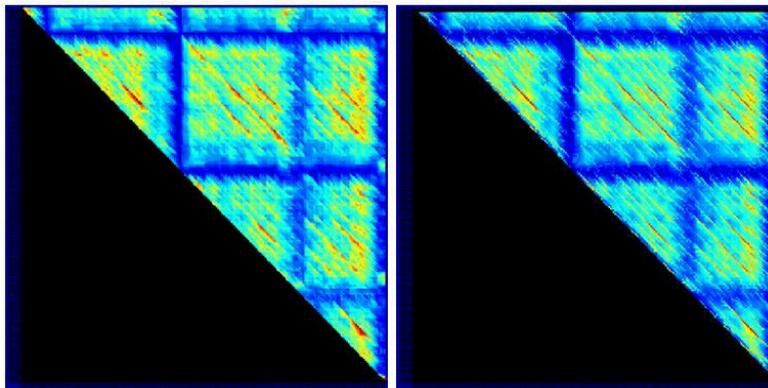


Figura 4. Matrizes de similaridade antes (E) e depois (D) da filtragem de nitidez.

A seguir calcula-se a similaridade média de cada diagonal do triângulo superior da matriz, e estes valores são armazenados. Em seguida, as 10 diagonais de maior média são selecionadas, e as demais são cortadas da matriz pela anulação de todos os seus valores. Após este corte, é calculado um limiar que permita a dicotomia de todos os valores não nulos onde exatamente 20% destes valores permaneçam acima deste limiar e o restante seja anulado. Isto é obtido pela concatenação das 10 diagonais não nulas em uma só sequência de valores, seguida pela ordenação decrescente destes valores (sort), escolhendo-se como limiar o valor correspondente à posição que tem como índice o tamanho da sequência dividido por cinco. Obtido este limiar, todo valor abaixo dele é também anulado. Ao fim deste processo espera-se que somente os trechos de mais alta

similaridade nas diagonais de maior similaridade média estejam não nulos na matriz. A escolha de 20% para este limiar de corte foi empírica, e realizada no trabalho de Eronen. A matriz da esquerda da Figura 5 é resultante deste processo de corte quando aplicado sobre a matriz à direita na Figura 4.

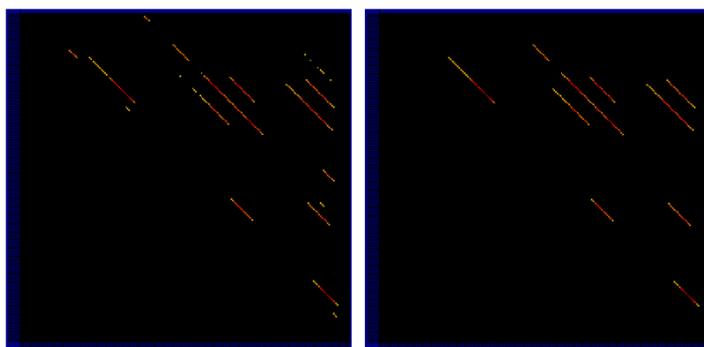
A última filtragem sobre a Matriz de similaridade é a remoção de gaps nulos entre diagonais não nulas, provenientes de curtas diferenças existentes entre os refrãos. São removidos os gaps inseridos em um intervalo não nulo de pelo menos 12 segundos, e que não ultrapassem o limite máximo de 4 segundos. Estes valores foram obtidos a partir dos valores sugeridos no trabalho de Eronen, originalmente medidos em batidas da música, sendo ajustados empiricamente em testes.

No final do processo descrito nesta seção espera-se ter as diagonais de repetições bem definidas na matriz de similaridade, cada uma contendo a correspondência entre dois trechos de música similares entre si.

### 3.4. Seleção da repetição candidata a refrão

Neste passo, as repetições encontradas na matriz de similaridade na etapa anterior são listadas pelo armazenamento de seus extremos (instante inicial e instante final). O número típico de repetições listadas está entre 10 e 15. Dentre estas repetições, que podem incluir tanto um refrão como outra parte da música que também é repetida, somente uma deve ser retornada pelo método como refrão obtido. Tal como nas soluções Goto e Eronen, esta seleção é realizada por heurísticas.

A primeira heurística adotada nesta solução é a do range de tamanho de refrãos. O range adotado com base em observações exaustivas sobre músicas de diversos estilos musicais é o intervalo de 5% a 25% da música, onde repetições fora deste range são descartadas, pois provavelmente possuem muitos trechos que não fazem parte do refrão. Na Figura 5, a matriz à direita é resultante deste processo sobre a matriz à esquerda, juntamente com o processo de remoção de gaps nulos descrito na seção 3.3. Esta heurística é a única que elimina uma repetição dentre as candidatas a refrão.



**Figura 5. (E) Matriz após corte de 80% dos valores; (D) Matriz após remoção de gaps nulos e repetições fora do range.**

As heurísticas descritas a seguir foram implementadas na forma de funções que retornam um valor normalizado (entre zero e um) interpretado como a probabilidade da repetição avaliada conter um refrão.

A segunda heurística testada foi o somatório do grau de correspondência com as outras repetições listadas, onde a repetição que tiver maior soma do grau de correspondência com todas as demais repetições é uma forte candidata a refrão. A

correspondência entre duas repetições é calculada usando F-measure entre os seus respectivos segmentos, tanto no eixo horizontal como no eixo vertical da matriz. A F-Measure, cujo cálculo é descrito na seção 4, retorna um valor entre zero (interseção nula) e um (intervalos idênticos) para os dois segmentos de entrada, identificados pelos seus extremos no eixo do tempo.

Uma matriz de correspondência é criada, equivalente à matriz de similaridade, e cada posição armazenará a correspondência de cada par de diagonais entre si, onde cada triângulo da matriz armazena as F-measures em relação a um dos eixos. Assim, a medida desta heurística, para cada repetição, é a soma dos valores contidos na linha com os valores contidos na coluna desta posição, relativas às correspondências tanto no eixo horizontal como no eixo vertical. Este valor, no fim, é normalizado, para que seja retornado um valor entre zero a um. Esta heurística considera o número de repetições de um trecho do sinal ao longo da música, porque um segmento com maior número de repetições tem maior chance de ser um refrão, pela própria definição de refrão.

A terceira heurística é a Similaridade Média, onde a diagonal cuja similaridade média na matriz de similaridade for maior tem maior probabilidade de ser o refrão. Desta forma, as similaridades médias de cada diagonal são calculadas e depois os valores são também normalizados.

A quarta heurística considerada é a do último segmento. Observando diversas músicas de diversos estilos musicais, facilmente se nota que é comum a música terminar com uma sequência de ao menos dois ou três refrãos consecutivos. Desta forma, uma repetição que contém um segmento localizado na região final da música tem grandes chances de conter um refrão. Neste caso, a probabilidade do segmento conter um refrão é inversamente proporcional à distância de seu ponto médio ao final da música.

A quinta heurística considerada é a da localização da repetição, onde a repetição que seja combinação do trecho mais próximo a um quarto combinado com o trecho mais próximo a três quartos do sinal de música mais provavelmente tem um refrão. A probabilidade aqui é calculada de maneira similar à quarta heurística, mas a distância considerada aqui é a do ponto médio do segmento até o ponto equivalente a  $\frac{1}{4}$  e  $\frac{3}{4}$  da música no eixo do tempo.

Nos testes realizados, combinações destas heurísticas foram também testadas. A seção a seguir descreve o procedimento de testes, bem como os resultados alcançados e uma discussão sobre estes resultados.

#### 4. Experimento

A base de dados utilizada possui 50 canções da música gospel brasileira, que recebe influências como a da música americana e de diversos ritmos nacionais e regionais, incorporando estilos como o soul, o rock em diversas vertentes, o pop, o romântico, o samba, o forró, conferindo uma variedade razoável. As músicas foram ouvidas e os refrãos foram marcados manualmente associando-se a cada canção um identificador e os pares de extremos (instante inicial e instante final) de cada refrão encontrado. As músicas contemplam ampla variedade de instrumentos musicais, tendo também solos em voz masculina, feminina, músicas com harmonia de vozes, ou ainda cantadas juntamente com as pessoas da plateia.

A métrica de sucesso é a mesma do Estado da Arte, a F-Measure, definida como a média harmônica entre a taxa de Recall (R) e Precisão (P). Seja  $a$  o refrão anotado manualmente e  $b$  o refrão retornado pela solução. Primeiro se mede a interseção entre estes dois intervalos  $Intersect(a,b)$ , e depois é calculado o valor de R, razão entre  $Intersect(a,b)$  e o tamanho de  $a$  (Equação 2), e o valor de P, razão entre  $Intersect(a,b)$  e o tamanho de  $b$  (Equação 3). Depois se calcula a média harmônica de R e P, razão entre o produto de R e P e a soma de R e P (Equação 4), que é a taxa de acerto.

$$R = \frac{Intersect(a,b)}{length(a)} \quad (2)$$

$$P = \frac{Intersect(a,b)}{length(b)} \quad (3)$$

$$F\_measure = \frac{2RP}{R + P} \quad (4)$$

A F-Measure retorna um valor entre zero e um que representa a correspondência no tempo entre  $a$  e  $b$ , onde zero significa correspondência nula (interseção inexistente) e um indica correspondência máxima, caso onde  $a$  possui exatamente os mesmos extremos (início e fim) que  $b$ .

Para cada música da base, o F-Measure foi calculado, para cada heurística ou combinação de heurísticas que foi testada, e os resultados obtidos foram registrados para análise posterior. Ao fim do processamento é calculada a F-Measure média para cada heurística ou combinação de heurísticas, e esta será a taxa de acertos final. As taxas de acerto obtidas foram comparadas com o caso ótimo, que assume que as heurísticas sempre acertam o segmento ótimo que maximiza o valor da F-Measure para uma música, e com o pior caso, que é a escolha aleatória de um segmento dentre os listados imediatamente antes da etapa de seleção de refrão.

Considerando as heurísticas descritas na seção 4.4, que são “Grau de Correspondência”, “Similaridade Média”, “Último Segmento”, “Localização da Repetição”, representadas respectivamente por h1, h2, h3 e h4, os testes apresentaram os resultados para o F-Measure descritos na Tabela 1.

**Tabela 1. Resultados obtidos com F-Measure média.**

Heurística	F-Measure Média	Heurísticas	F-Measure Média
Caso Ótimo	89,92%	h1 e h2	82,49%
Randômico	67,35%	h1 e h3	79,29%
h1	82,09%	h1 e h4	81,89%
h2	82,09%	h2 e h3	71,15%
h3	68,21%	h2 e h4	71,15%
h4	68,54%	h3 e h4	71,15%

Durante os testes de custo computacional, uma música de 4 minutos e 3 segundos foi escolhida para testar o tempo de resposta da aplicação, obtendo-se o tempo de resposta de 13 segundos.

#### 4.1. Avaliação

Algumas heurísticas ou combinação de heurísticas atingiram uma taxa de acertos entre 80% a 85%. O custo computacional do método proposto é função apenas do tamanho da

entrada, pois a razão entre o tamanho da música e o tempo de resposta é sempre mantida. Esta regra não é observada no Estado da Arte, onde o desempenho é função do número de batidas da música e, portanto, de sua velocidade. Como o desempenho obtido no Estado da Arte foi de cerca de 10 segundos para músicas de 3 a 4 minutos, foi comprovado que o custo computacional é da mesma ordem de grandeza dos métodos atuais, considerando um computador com uma configuração média.

A heurística que obteve resultado mais próximo do ótimo nos experimentos foi a h1 (Grau de Correspondência). A Figura 6 apresenta a mesma matriz à direita da Figura 5 com a repetição que foi selecionada destacada em azul, onde esta heurística obteve para esta música uma F-Measure de 92,3%.

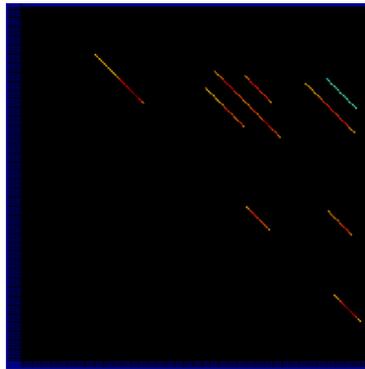


Figura 6. Matriz com repetição selecionada destacada.

## 5. Conclusão

Os resultados comprovaram que a utilização da função de correlação sobre o sinal de envoltória do sinal para estimativa da similaridade entre trechos da música também é eficaz para a detecção de refrão.

A abordagem deste trabalho realiza a detecção de refrão de uma forma diferente, com suas vantagens e desvantagens específicas. Uma vantagem é a independência da presença de notas musicais, dispensando a exploração do domínio da frequência, sujeita a dificuldades como a modulação, que exigem uma lógica de tratamento adicional. Além disso, espera-se que soluções dependentes de notas musicais não funcionem bem na estimativa de partes que se repetem ao longo do tempo em sinais de áudio que não tenham notas musicais, o que em tese não seria problema para o método proposto.

Explorar o domínio da frequência não é necessariamente pior ou mais complexo que explorar a intensidade do som em uma solução de detecção de refrão. Abordagens diferentes possuem vantagens e desvantagens diferentes que poderão ser mais adequadas a cada tipo particular de aplicação.

Trabalhos futuros incluem encontrar dinamicamente o limiar de dicotomização da matriz de similaridade, atualmente fixo em 20%, e a exploração mais profunda das heurísticas empregadas na etapa de seleção de refrão, refinando as que já existem ou encontrando novas formas de se eleger uma repetição candidata.

## Referências

- Bartsch, M. A. and Wakefield, G. H. (2005) “Audio Thumbnailing of Popular Music Using Chroma-Based Representation”, In: IEEE Trans. on Multimedia, vol. 7, no. 1, Feb. 2005, pp. 96-104.
- Cooper, M. and Foote, J. (2003) “Summarizing Popular Music Via Structural Similarity Analysis,” In: Proc. of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, WASPAA 2003, October 19-22, 2003, New Paltz, NY.
- Ellis, D. (2006) “Beat Tracking with Dynamic Programming”, MIREX 2006 Audio Beat Tracking Contest system description, Sep 2006, available at <http://www.ee.columbia.edu/~dpwe/pubs/Ellis06-beattrack.pdf>
- Eronen, A. (2007) “Chorus Detection with Combined use of MFCC and Chroma Features and Image Processing Filters”, in Proc. of the 10th International Conference on Digital Audio Effects (DAFx-07), Bordeaux, France, September 10-15, 2007.
- Goto, M. (2006) “A Chorus Section Detection Method for Musical Audio Signals and Its Application to a Music Listening Station”, In: IEEE Trans. on Audio, Speech, and Language Processing, vol. 14, no. 5, Sept. 2006 pp. 1783 – 1794.
- Levy, M., Sandler, M. and Casey, M. (2006) “Extraction of High-Level Musical Structure From Audio Data and Its Application to Thumbnail Generation”, In: Proc. IEEE ICASSP 2006, vol. V, pp. 13-16.
- Marolt, M. (2006) “A Mid-level Melody-based Representation for Calculating Audio Similarity”, In: Proc. of the 7th International Conference on Music Information Retrieval, ISMIR 2006, Victoria, Canada, 8 - 12 October 2006.
- Paulus, J. and Klapuri, A. (2006) “Music Structure Analysis by Finding Repeated Parts”, In: Proc. of the 1st Audio and Music Computing for Multimedia Workshop (AMCMM2006), Santa Barbara, California, USA, October 27, 2006, pp. 59-68.
- Seppänen, J., Eronen, A. and Hiipakka, J. (2006) “Joint Beat & Tatum Tracking from Music Signals”, In: Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR), Victoria (BC), Canada, October 8-12, (2006).
- Shiu, Y., Jeong, H. and Kuo, C. (2006) “Similarity Matrix Processing for Music Structure Analysis”, In: Proc. of the 1st Audio and Music Computing for Multimedia Workshop (AMCMM2006), October 27, 2006, Santa Barbara, California, USA.

# Análise do Vibrato e Bending na Guitarra Elétrica

Tairone N. Magalhães<sup>1</sup>, Davi A. Mota<sup>1</sup>, Aluizio B. Oliveira<sup>1</sup>, Mauricio A. Loureiro<sup>1</sup>

<sup>1</sup>Centro de Estudos do Gesto e Expressão Musical – Escola de Música da UFMG  
Av. Antônio Carlos, 6627 – Cep: 31270-010 - Belo Horizonte - MG

{tairone, aluizioneto, davimota}@ufmg.br, mauricio@musica.ufmg.br

**Abstract.** *On the electric guitar, vibrato and bending are techniques that, according to common sense, play a key role in a guitarist's playing signature. This work aims at investigating features of vibratos and bendings that differentiate one guitarist from another. Moreover, we propose a model for the categorization of performances composed by four stages: (1) pitch extraction; (2) note segmentation; (3) extraction of descriptors for bending and vibrato; (4) analysis and categorization of performances based on the proposed parameterization. In order to validate the model, we analyzed a set of several performances of two short musical excerpts, performed by 8 guitarists of distinct expertise.*

**Resumo.** *Na guitarra elétrica, o vibrato e o bending são técnicas que, de acordo com o senso comum, tem um papel importante na assinatura do guitarrista. Esse trabalho visa investigar características na execução do bending e do vibrato que diferenciam um guitarrista do outro. Propomos um modelo para a categorização das performances composto de quatro estágios: (1) extração de altura; (2) segmentação de notas; (3) extração de descritores de bending e vibrato; (4) análise e categorização de performances baseada na parametrização proposta. Para validar o modelo, analisamos um conjunto de performances de dois trechos, executadas por oito guitarristas de expertise distinta.*

## 1. Introdução

Na guitarra elétrica, os músicos desenvolvem, em grande parte, uma linguagem musical muito pessoal e característica. De acordo com [Méndez, 2012], ouvintes são capazes de distinguir claramente as manipulações de propriedades sonoras realizadas por guitarristas, e criam preferências baseadas nessas diferenças. É possível até mesmo reconhecer um guitarrista ouvindo apenas alguns compassos de uma performance. De acordo com o senso comum, o *vibrato* e o *bending* exercem um papel importante na assinatura do guitarrista, principalmente em estilos como o rock e o blues, em que tais técnicas são amplamente utilizadas. Em ambos recursos o instrumentista possui controle direto e contínuo da altura do som produzido, como em um instrumento de corda sem trastes ou na voz. Segundo [Grimes, 2014], o timbre e o vibrato de um guitarrista podem ser fortemente intrínsecos ao instrumentista, o que tornaria sua sonoridade tão particular quanto o timbre vocal. Diante de tal panorama, identificou-se a possibilidade de se promover um estudo sistemático do emprego de tais técnicas em performances na guitarra, com o objetivo de avaliar se é mesmo possível detectar uma assinatura ou categorizar performances baseado nos perfis de bending e vibrato empregados pelo músico.

### 1.1. O Vibrato

O vibrato é um tipo de modulação periódica muito utilizada como recurso expressivo em instrumentos musicais e no canto. Em uma nota com vibrato, o som pode variar de três

diferentes formas [Fletcher, 2001]: amplitude, frequência e timbre<sup>1</sup>. São utilizados três parâmetros para representar a modulação [Seashore, 1917]: taxa, extensão e forma. De acordo com [Desain et al., 1999], a taxa de vibrato geralmente gira entre 6 e 7 Hz, mas ela pode variar de 4 a até 12 Hz. A extensão varia de 0,2 a 0,35 semitons em instrumentos de corda, e 0,6 a 2 semitons no canto [Timmers and Desain, 2000]. O vibrato é muito utilizado para dar ao som mais “profundidade” e “sustentação”, e para fazê-lo se destacar do restante do espaço sonoro [Jarvelainen, 2002]. Na guitarra o vibrato é produzido alterando-se ciclicamente a tensão de corda, por meio do deslocamento da corda na direção transversal ou longitudinal<sup>2</sup>.

## 1.2. O Bending

Utiliza-se o termo *bending*, *pitch bending* ou simplesmente *bend* para se referir a essa técnica. De acordo com [Chen et al., 2009], o bending se refere ao ajuste da altura de uma nota musical. Geralmente representa uma variação suave em altura, e pode incluir portamento ou glissando, que se refere a uma variação de altura de uma nota para outra. Similar ao vibrato, o bending é realizado por meio de um deslocamento da corda de sua posição de equilíbrio, na direção transversal à mesma, utilizando o dedo que pressiona a corda entre os trastes. O uso da técnica possibilita ao músico incorporar características microtonais em sua performance [Grimes, 2014].

## 2. Metodologia

O estudo proposto neste trabalho engloba as seguintes etapas:



Figura 1: Diagrama Metodologia

A análise das gravações, segmentação e extração de descritores foram realizadas utilizando a ferramenta Expan [Campolina et al., 2009], que foi ampliada e reestruturada durante a execução deste trabalho. Foram necessárias algumas correções manuais nas segmentações realizadas pela ferramenta. Tais ajustes foram feitos utilizando o software Sonic Visualizer [Cannam et al., 2010], que facilita a navegação pelos arquivos de áudio e marcação dos pontos da segmentação.

### 2.1. Experimento

O experimento contou com a participação de oito guitarristas, tocando dois excertos de duas composições distintas. O primeiro excerto, extraído de *Still Got The Blues*, de Gary Moore (figura 2), foi escolhido por possuir notas executadas com vibratos e bendings de diferentes durações, compreendendo intervalos distintos. O excerto possui ainda ocorrências de notas finalizadas no meio do bending (com a corda ainda esticada) enquanto outras só terminam quando a corda retorna à sua posição inicial (e consequentemente à altura inicial). O segundo excerto (figura 3), composto pelo autor TNM, contém cinco notas de diferentes durações com bendings de um e meio tom, e três notas com vibrato.

<sup>1</sup>Em teoria, as três formas de vibrato citadas acima podem existir isoladamente ou combinadas, mas na prática o que se observa, em geral, é que elas ocorrem simultaneamente, cada uma delas em menor ou maior quantidade, dependendo do contexto e do instrumento.

<sup>2</sup>O vibrato longitudinal é mais comumente utilizado no violão clássico.

Os guitarristas gravaram quatro vezes cada um dos excertos. As partituras foram enviadas com antecedência aos sujeitos para o seu estudo. Todos os músicos tocaram no mesmo instrumento e utilizaram a mesma configuração para captura e monitoração da performance. Foi utilizado um *direct box*, que possibilita o envio do sinal direto da guitarra para a interface de áudio, ao mesmo tempo que ele é enviado para um pedal de *overdrive*, e em seguida para o amplificador de guitarra. Assim, capturamos o som limpo da guitarra enquanto o músico ouvia o som processado pelo pedal. A decisão de utilizar o pedal de *overdrive* foi tomada por este ser um efeito muito utilizado no rock e no blues.

Figura 2: Excerto 1 - *Still Got The Blues*, de Gary Moore

Figura 3: Excerto 2 - composto pelo autor TNM

## 2.2. Detecção da frequência fundamental e dos harmônicos

A detecção de frequências foi feita utilizando uma FFT (*Fast Fourier Transform*) de 65536 pontos ( $2^{16}$ ) utilizando uma janela de *hamming* de tamanho 2048 aplicada ao sinal de áudio com um salto de 512 amostras. Todos os sinais de áudio analisados foram gravados à taxa de amostragem de 44100 Hz, o que resulta em uma resolução de frequência de aproximadamente 0.67 Hz.

Para cada quadro de análise  $k$ , o algoritmo de detecção de frequência fundamental busca os picos proeminentes no espectro do sinal, que são escolhidos como candidatos à frequência fundamental. Para cada candidato, são calculados e localizados os seus prováveis harmônicos dentro de uma faixa de tolerância de 6%. O candidato que possuir

a maior energia harmônica  $E_H$  (equação 1) é escolhido como o candidato mais provável à frequência fundamental do quadro.

$$E_H(k) = \sum_{h=1}^N a_h(k)^2 \quad (1)$$

onde  $a_h$  é a magnitude da  $h$ -ésima componente harmônica (para um total de  $N$  harmônicos) e  $Re_i$  a magnitude da  $i$ -ésima componente espectral do quadro (para uma FFT de tamanho  $L$ ).

Calcula-se em seguida, utilizando o candidato escolhido, a ruidosidade  $R(k)$  do quadro (tradução do termo *noisiness*, do inglês) [Peeters et al., 2011] e realiza-se uma última verificação:  $R(k)$  deve estar abaixo de um limiar estabelecido empiricamente para que o candidato seja selecionado como a frequência fundamental. Caso ele não cumpra esse requisito, considera-se que o quadro não possui uma frequência fundamental definida. O descritor de ruidosidade  $R(k)$  é calculado, para o  $k$ -ésimo quadro de análise através da equação:

$$R(k) = \frac{E_N(k)}{E_T(k)} \quad (2)$$

$$E_N(k) = E_T(k) - E_H(k) \quad (3)$$

$$E_T = \sum_{i=1}^L Re_i^2 \quad (4)$$

onde,  $E_N(k)$  é a energia do ruído existente e  $E_T(k)$  a energia total para o mesmo quadro de análise. A energia total  $E_T$  é a soma das energias de todas as componentes da FFT. A energia do ruído  $E_N$  é a energia das componentes não-harmônicas do sinal, ou seja, a energia total da FFT menos a energia das componentes harmônicas  $E_H$ .

### 2.3. Segmentação

Para a análise pretendida do vibrato e do bending, as gravações realizadas no experimento foram segmentadas a partir da detecção dos instantes de início (*onset*) e fim (*offset*) de cada nota, assim como dos períodos de estabilidade da frequência (sustentação da nota).

#### 2.3.1. Início e fim de nota

Para detecção dos inícios de nota, foi desenvolvido um algoritmo de segmentação utilizando valores de *ruidosidade* e *energia RMS* do sinal. A curva  $R$  do descritor de ruidosidade (equação 2) tende a ficar próxima de 1 quando o conteúdo harmônico do sinal é baixo, como por exemplo, em instantes de silêncio ou de transição entre notas. Ao longo da sustentação da nota, o valor da ruidosidade do sinal deverá diminuir, devido à energia harmônica mais significativa nesta região. Portanto, a curva  $R$  é eficaz para detectar os transientes do sinal, desde que os trechos de baixa energia, correspondentes às pausas, sejam descartados. Para isso, utilizamos a curva de RMS:

$$RMS(k) = \sqrt{\frac{1}{L} \sum_{j=1}^L x(n)^2} \quad (5)$$

Ao invés de trabalhar com um limiar fixo na curva de RMS, utilizamos o próprio valor de RMS para modular o descritor de ruidosidade, obtendo assim a curva  $S$ :

$$S(k) = RMS(k) \times R(k) \quad (6)$$

Os candidatos prováveis de instantes de início de nota são estimados pelos picos da curva da derivada de  $S$ . Para a detecção dos instantes de fim de nota, utilizamos um algoritmo simples aplicado à informação de energia RMS do sinal.

### 2.3.2. Região de sustentação da nota

No início de uma nota existe um curto período de transientes (ataque) em que há um alto grau de componentes não-periódicas. Nessa região, a curva da frequência fundamental tende a ser ruidosa e imprecisa. O mesmo tende a ocorrer próximo ao final da nota, principalmente quando se executam notas ligadas (muitas vezes devido a notas consecutivas apresentarem uma leve superposição, por serem tocadas em cordas diferentes). Para contornar este problema, convencionamos que o início da região de sustentação é o instante em que a frequência fundamental se estabiliza, após o início da nota. Similarmente, o final da sustentação se dá quando a curva de frequência fundamental se torna instável, antes do final da nota. Verificamos que, nos períodos de instabilidade da curva de frequência fundamental, geralmente frequências bem mais baixas do que a da fundamental são detectadas. Assim, definimos um limiar de desvio de frequência para garantir que o algoritmo não descartasse regiões de bending. Os instantes de início e final da região de sustentação foram estimados pelo primeiro e o último ponto da curva de frequência fundamental em que o desvio em relação à frequência da nota tocada  $f_n$  é menor do que 3 semitons. Neste estudo, as análises dos dados obtidos das curvas de frequência fundamental consideram apenas as regiões de sustentação da nota.

### 2.4. Descritor de vibrato

Para calcular a extensão e taxa do vibrato, foi desenvolvido um algoritmo baseado na detecção de regularidade dos picos da curva de frequência fundamental de uma nota. O período de um ciclo de vibrato é o intervalo de tempo entre dois picos locais consecutivos na curva, e a extensão a diferença de altura entre pontos de mínimo e máximo consecutivos. A existência de vibrato foi considerada quando a taxa estiver entre 2 e 10 Hz e a extensão acima de um limiar de 0.5% do mínimo local de frequência. O valor de 0.5% foi escolhido por aproximação à curva do limiar de discriminação de modulação em frequência, que na região de médios gira entre 0.4% e 0.6%, de acordo com [Sek and Moore, 1995].

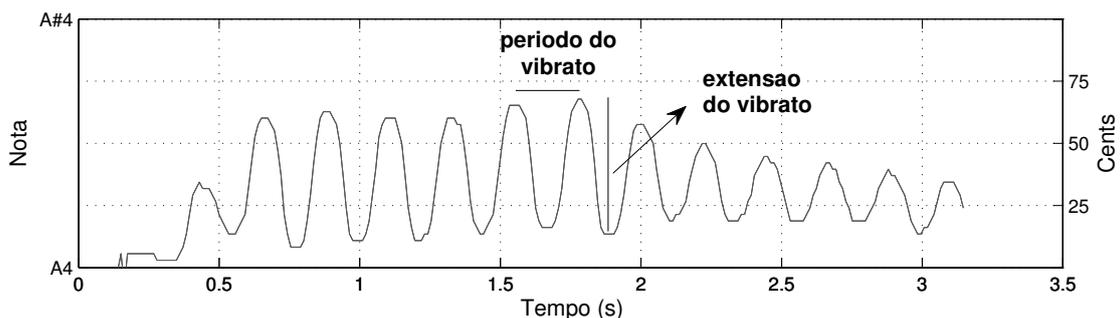


Figura 4: Curva de frequência fundamental para uma nota com vibrato tocada na guitarra.

Para caracterizar o vibrato foi definido um descritor multidimensional de 12 dimensões, estimadas como se segue:

1. regularidade da taxa do vibrato;
2. regularidade da extensão do vibrato.
3. média da taxa do vibrato;
4. média da extensão do vibrato.
5. máximo da taxa do vibrato;
6. mínimo da taxa do vibrato;
7. máximo da extensão do vibrato;
8. mínimo da extensão do vibrato;
9. instante em que a extensão do vibrato atinge seu máximo<sup>3</sup>;
10. tempo até o início do vibrato<sup>3</sup>;
11. tempo até o final do vibrato<sup>3</sup>;
12. perfil da extensão do vibrato.

#### 2.4.1. Regularidade do vibrato

A regularidade do vibrato representa o quanto a extensão e a taxa de oscilação se mantêm regulares ao longo da nota. Sejam as curvas  $r_v(l)$  e  $d_v(l)$  que representam a taxa e extensão do vibrato, respectivamente, para cada ciclo de vibrato  $l$ , sendo  $1 \leq l \leq C$ , sendo  $C$  o número de ciclos do vibrato. A regularidade da taxa e da extensão do vibrato foram estimadas pelas equações (7) e (8) respectivamente.

$$R_d = 1 - \frac{\sum_{l=2}^C |d_v(l) - d_v(l-1)|}{\sum_{l=1}^C d_v(l)} \quad (7)$$

$$R_r = 1 - \frac{\sum_{l=2}^C |r_v(l) - r_v(l-1)|}{\sum_{l=1}^C r_v(l)} \quad (8)$$

Os valores de  $R_d$  e  $R_r$  tendem a 1 para vibratos regulares em extensão e taxa, respectivamente, enquanto que para vibratos irregulares, estes valores tendem a zero.

#### 2.4.2. Valores médio, máximo e mínimo do vibrato

Os valores médio, máximo e mínimo da taxa e extensão do vibrato foram estimados nos trechos onde a existência de vibrato foi detectada, assim como o instante em que a extensão do vibrato atinge seu valor máximo.

#### 2.4.3. Tempo até o início e o final do vibrato

Os tempos até o início e o fim do vibrato foram definidos como os intervalos de tempo entre o início da nota e o início e o final do vibrato, respectivamente.

#### 2.4.4. Perfil da extensão do vibrato ao longo da nota

O perfil da extensão do vibrato definido como o coeficiente angular  $\alpha$  da regressão linear da curva da extensão do vibrato, descreve a evolução temporal da extensão do vibrato. Valores positivos correspondem a vibratos com extensão crescente e negativos para extensão decrescente.

<sup>3</sup>Normalizados em relação à duração do período de sustentação da nota.

## 2.5. Descritor de bending

Para extrairmos informações referentes ao bending, é necessário detectar quando um bending é realizado em uma nota, e determinar o seu início e fim. Seja  $\mathcal{P}(k)$  o valor da frequência fundamental do quadro  $k$ . Primeiramente, arredonda-se cada valor  $\mathcal{P}$  para a altura temperada mais próxima, obtendo-se a curva  $\mathcal{P}_r$ . A partir dessa curva arredondada, detecta-se o ponto  $k_s$  em que ocorre a primeira transição de altura no sentido ascendente ( $\mathcal{P}_r(k) > \mathcal{P}_r(k-1)$ ) e o ponto  $k_d$  da última transição no sentido descendente ( $\mathcal{P}_r(k) < \mathcal{P}_r(k-1)$ ). Aplica-se então à curva  $\mathcal{P}$  um filtro passa-baixas com frequência de corte em 20 Hz, e obtém-se a curva  $\mathcal{P}_f$ . Para detectar os instantes de início e final da subida do bending, localiza-se o primeiro ponto à esquerda e à direita de  $k_s$  em que a derivada de  $\mathcal{P}_f$  inverte de sinal. A mesma abordagem é adotada para a descida do bending, partindo do ponto  $k_d$ .

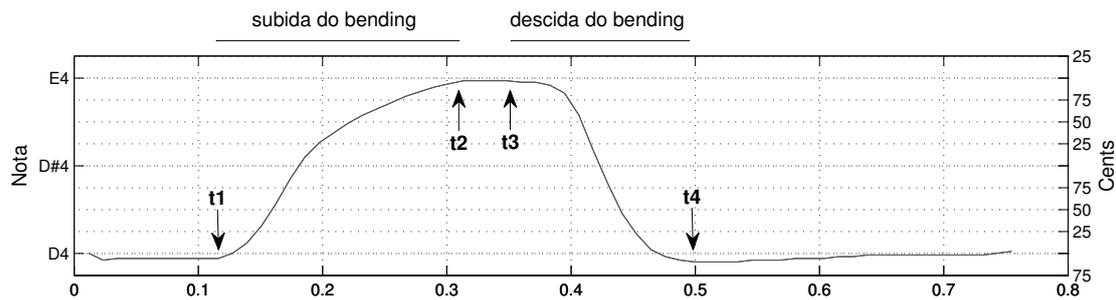


Figura 5: Curva de frequência fundamental para uma nota tocada com bending na guitarra.

Para caracterizar o bending foi também definido um descritor multidimensional de 12 dimensões, estimadas como se segue:

1. velocidade média do bending na subida;
2. velocidade média do bending na descida;
3. valor da velocidade instantânea máxima do bending na subida;
4. instante da velocidade instantânea máxima do bending na subida<sup>4</sup>;
5. valor da velocidade instantânea máxima do bending na descida;
6. instante da velocidade instantânea máxima do bending na descida<sup>4</sup>;
7. centróide temporal da altura na subida do bending;
8. centróide temporal da altura na descida do bending;
9. instante de início da curva de subida do bending<sup>4</sup>;
10. instante de fim da curva de subida do bending<sup>4</sup>;
11. instante de início da curva de descida do bending<sup>4</sup>;
12. instante de fim da curva de descida do bending<sup>4</sup>.

### 2.5.1. Velocidade média do bending e velocidade instantânea máxima do bending

A velocidade média do bending pode ser calculada tanto para a subida quanto para a descida da curva de frequência fundamental através da equação:

$$v = \frac{\Delta f}{\Delta t} \quad (9)$$

Para o cálculo da velocidade instantânea máxima do bending primeiramente aplicamos um filtro passa-baixas a 20 Hz na curva de frequência fundamental  $f(t)$ , onde  $t$

<sup>4</sup>Normalizados em relação à duração do período de sustentação da nota.

representa o número da amostra. Calculamos então a derivada para obtermos a taxa de variação instantânea da frequência fundamental  $i(t)$ .

$$i(t) = \frac{df(t)}{dt} \quad (10)$$

O pico da curva  $i(t)$  é o ponto de velocidade máxima do bending na sua subida. O vale da mesma curva é o ponto de velocidade máxima na descida. O valor da velocidade máxima na descida é o módulo da taxa encontrada para este ponto.

### 2.5.2. Centróide temporal da altura no bending

O centróide temporal da energia de uma nota é um descritor bastante utilizado em pesquisas de timbre. Neste trabalho, propomos a utilização de um centróide de altura, que é o centro de gravidade da curva de altura da nota. Seja  $p(t)$  uma curva da frequência fundamental ao longo do tempo  $t$ . O centróide temporal da altura  $c$  calculado para tal curva entre os instantes  $t_1$  e  $t_2$  será:

$$c = \frac{\sum_{t=t_1}^{t_2} p(t)t}{\sum_{t=t_1}^{t_2} p(t)}, \quad (11)$$

Para calcular o centróide temporal da altura para os trechos de subida e descida do bending, basta utilizar a equação 11 duas vezes: uma limitada pelos pontos de início e fim da subida e outra pelos pontos de início e fim da descida do bending.

### 2.5.3. Extensão do bending

A extensão do bending é calculada a partir da diferença de altura entre o ponto que o bending termina e o ponto em que ele se inicia.

## 2.6. Análise e categorização dos dados

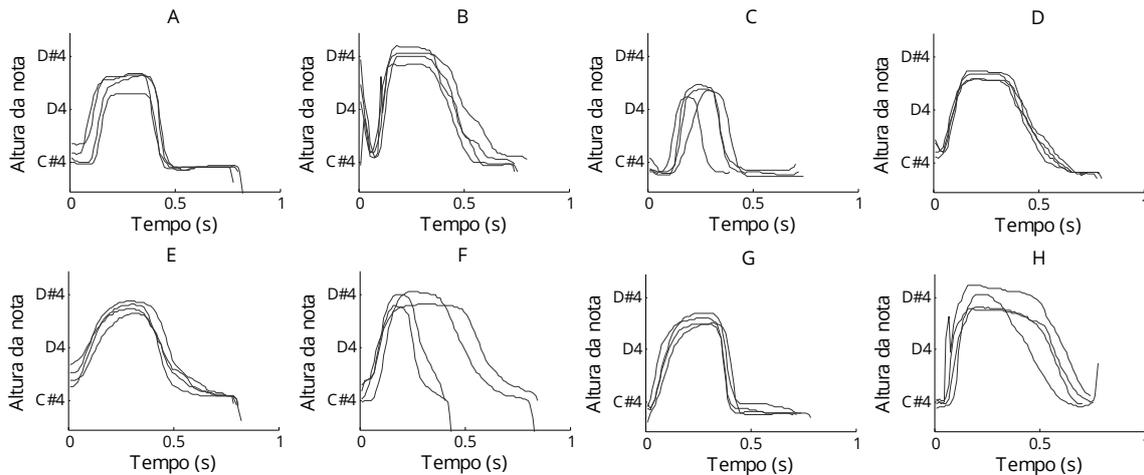
Mapas Auto-Organizativos de Kohonen (SOM - Self-Organizing Maps) são um tipo de redes neurais não-supervisionadas capazes de realizar um mapeamento de dados de entrada de grandes dimensões em espaços de baixa dimensão, preservando as relações topológicas dos dados originais. Assim, dados de entrada com maior grau de similaridade são exibidos próximos uns aos outros no mapa. Por ser um método de classificação não-supervisionado, pode ser utilizado para classificar e agrupar dados de entrada desconhecidos [Kohonen, 1995]. Utilizamos neste trabalho uma implementação do SOM na plataforma Matlab chamada *SOM Toolbox* [Vesanto et al., 1999], desenvolvida no Laboratório de Computação e Ciência da Informação (CIS) da Universidade de Tecnologia de Helsinki, na Finlândia.

Para realizar a análise e categorização dos dados, foram gerados Mapas Auto-Organizativos a partir dos parâmetros extraídos das performances gravadas. O objetivo foi criar um mapa que represente a similaridade na forma de se realizar bending e vibrato nas diferentes performances.

## 3. Resultados

A figura 6 mostra as curvas de frequência fundamental das quatro execuções da nota 18 do excerto 1, realizadas por cada um dos oito guitarristas gravados (cada letra de A a

H representa um músico). As curvas revelam a consistência na maneira de realizar o bending, sugerindo a existência de uma assinatura associada ao perfil do bending de cada guitarrista.



**Figura 6: Curva de altura da nota 18 do excerto 1 tocada por oito guitarristas.**

Aplicamos o descritor de bending proposto à nota 18 do excerto 1, e geramos em seguida um mapa (SOM) com o objetivo de avaliar como o sistema categoriza as diferentes execuções de tal nota pelos sujeitos gravados. Os resultados são mostrados na figura 7. Como se pode ver, as notas executadas por um mesmo sujeito tendem a se posicionar na mesma região do mapa. Os sujeitos G e E por exemplo, tiveram suas execuções agrupadas em uma única célula do mapa.

Para avaliar a categorização do vibrato, aplicamos o descritor de vibrato proposto à nota 22 do mesmo excerto. A figura 8 mostra o mapa gerado para a nota 22. Os sujeitos C, D e E tiveram seus vibratos agrupados em uma mesma unidade do mapa, enquanto os sujeitos B, F e G apresentaram um espalhamento significativo. Os sujeitos A e H tiveram três notas agrupadas em regiões próximas, e um valor atípico cada (A1 e H3).

Tais gráficos sugerem que o sistema é capaz de representar razoavelmente as similaridades entre bendings e vibratos de notas tocadas em um mesmo contexto musical. Pelo fato de tais gráficos terem sido gerados para notas isoladas de uma performance, é de se esperar que hajam valores atípicos e que os agrupamentos não representem exatamente a assinatura musical dos indivíduos. Afinal, uma nota isolada pode não ser suficiente para se identificar a assinatura de um sujeito.

Processamos então todas as notas que possuem vibrato ou bending nos excertos 1 e 2, e após extrair os descritores referentes a cada técnica, geramos um mapa com as suas informações combinadas. Portanto, o mapa gerado para cada excerto representa a caracterização das performances gravadas com base nos descritores de bending e vibrato, conjuntamente. Como já se esperava, a integração de ambos os descritores e a aplicação dos mesmos a várias notas de uma performance, tendem a gerar representações muito mais razoáveis das assinaturas dos sujeitos do que se tinha com notas isoladas (figuras 9 e 10).

#### 4. Conclusão

Os resultados deste trabalho reforçam a hipótese de que a forma de se realizar o bending e o vibrato exercem um papel fundamental na assinatura de um guitarrista. Afinal, o sistema desenvolvido se mostrou capaz de agrupar performances com perfis similares de execução dessas técnicas a partir dos parâmetros extraídos, e em alguns casos, de agrupar

performances do mesmo sujeito em uma única unidade dos mapas gerados. Os próximos passos deste trabalho serão: (1) avaliar a possibilidade de aplicar outros métodos de agrupamento e classificação aos dados; (2) investigar quais as dimensões dos descritores são mais importantes na separação de indivíduos; (3) avaliar a robustez dos descritores desenvolvidos e o quanto eles são generalizáveis para diferentes contextos musicais; (4) avaliar se os mapas de gerados possuem relação com a percepção de similaridade entre as diferentes performances; e (5) melhorar o método de detecção de frequência fundamental com o objetivo de reduzir os valores atípicos encontrados.

## 5. Agradecimentos

Agradecemos às agências brasileiras de fomento à pesquisa, CAPES, CNPq e FAPEMIG, e também ao Programa de Pós-Graduação em Música da UFMG pelo apoio.

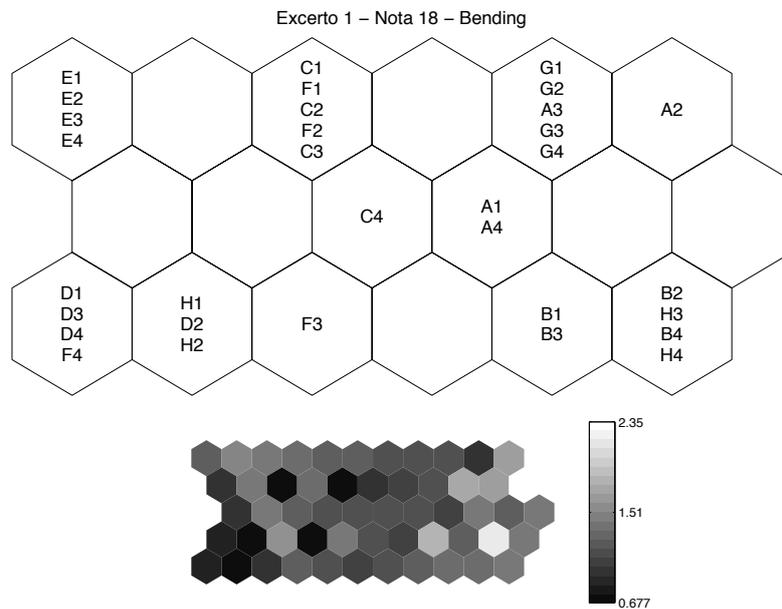


Figura 7: Mapa das diferentes execuções da nota 18 do excerto 1 (nota com bending).

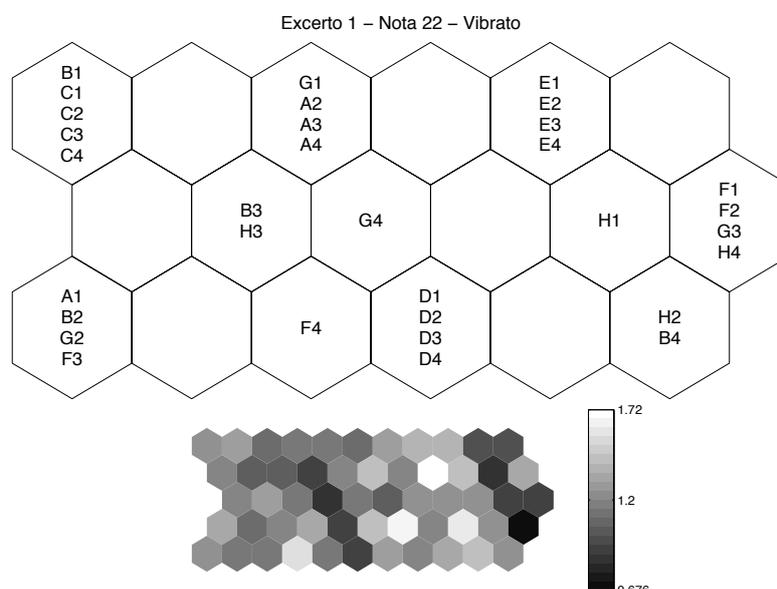
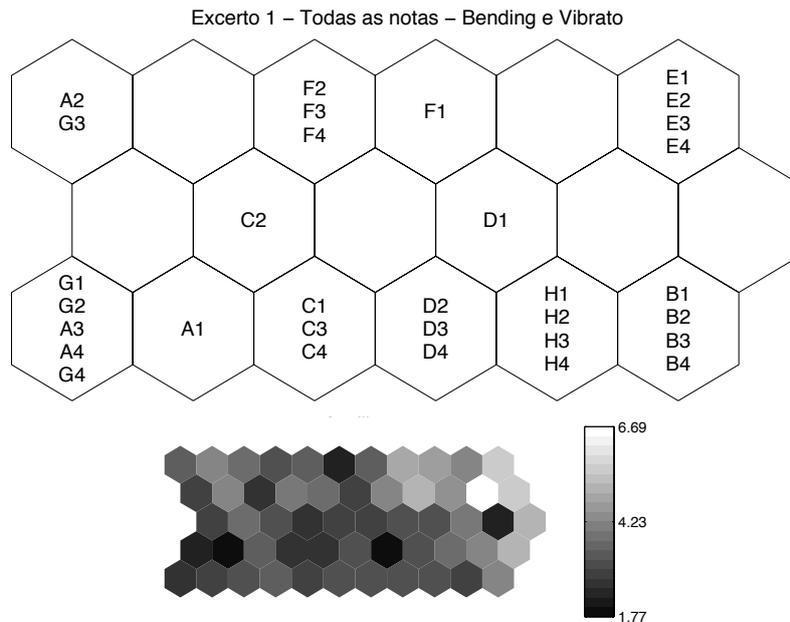
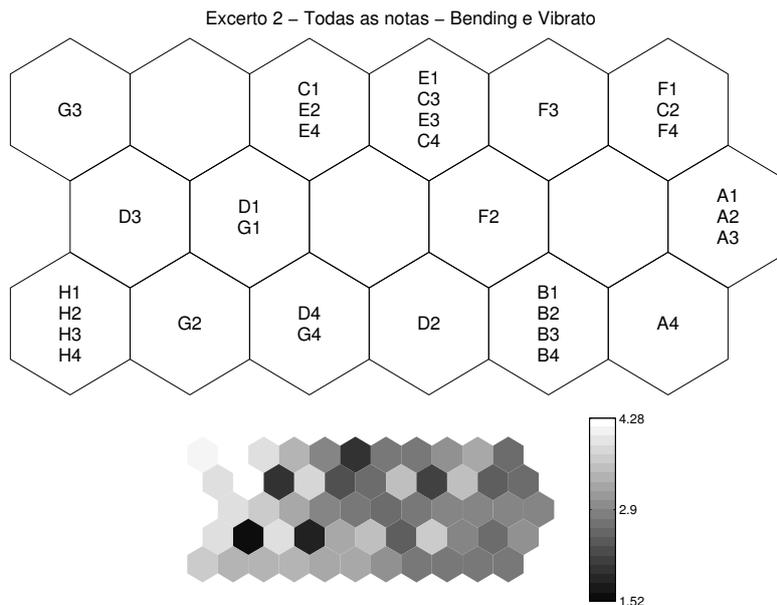


Figura 8: Mapa das diferentes execuções da nota 22 do excerto 1 (nota com vibrato).



**Figura 9:** Mapa gerado para as performances do excerto 1 utilizando todas notas com bending (4, 5, 11, 12, 17 e 18) e todas com vibrato (7, 13, 19 e 22).



**Figura 10:** Mapa gerado para as performances do excerto 2 utilizando todas as notas com bending (4, 8, 9, 17 e 19) e todas com vibrato (5, 18 e 20).

## Referências

- Campolina, T., Mota, D., and Loureiro, M. (2009). Expan: a tool for musical expressiveness analysis. In Naveda, L., editor, *Proceedings of the 2nd International Conference of Students of Systematic Musicology*, pages 24–27. IPEM.
- Cannam, C., Landone, C., and Sandler, M. (2010). Sonic visualiser an open source application for viewing, analysing, and annotating music audio files. In *Proceedings of the ACM Multimedia 2010 International Conference*, pages 1467–1468.
- Chen, J.-M., Smith, J., and Wolfe, J. (2009). Pitch bending and glissandi on the clarinet: roles of the vocal tract and partial tone hole closure. *The Journal of the Acoustical Society of America*, 126(3):1511–1520.

- Desain, P., Henkjan, H., Rinus, A., and Renee, T. (1999). Rhythmic aspects of vibrato. In Desain, P. and Windsor, L., editors, *Rhythm Perception and Production*, number 2000, pages 203–216. Swets & Zeitlinger.
- Fletcher, N. (2001). Vibrato in music. *Acoustics Australia*.
- Grimes, D. R. (2014). String theory - the physics of string-bending and other electric guitar techniques. *PloS one*, 9(7):e102088.
- Jarvelainen, H. (2002). Perception-based control of vibrato parameters in string instrument synthesis. *Proc. International Computer Music Conference*, (September):287–294.
- Kohonen, T. (1995). *Self-Organizing Maps - Springer Series in Information Sciences*. Springer Verlag.
- Méndez, S. I. G. (2012). *Modeling Embellishment , Duration and Energy Expressive Transformations in Jazz Guitar*. PhD thesis.
- Peeters, G., Giordano, B., Susini, P., Misdariis, N., and McAdams, S. (2011). The timbre toolbox: extracting audio descriptors from musical signals. 130(5).
- Seashore, C. (1917). The natural history of the vibrato. 3:141–147.
- Sek, A. and Moore, B. C. (1995). Frequency discrimination as a function of frequency, measured in several ways. *The Journal of the Acoustical Society of America*, 97(4):2479–2486.
- Timmers, R. and Desain, P. (2000). Vibrato: questions and answers from musicians and science. In *Proceedings of the Sixth International Conference on Music Perception and Cognition*, volume 2.
- Vesanto, J., Himberg, J., Alhoniemi, E., and Parhankangas, J. (1999). Self-organizing map in matlab: The som toolbox. *Proceedings of the Matlab DSP conference*, 99.

# Copista - Sistema de OMR para a recuperação de acervo histórico musical

Avner Maximiliano de Paulo<sup>1</sup>, Flávio Luiz Schiavoni<sup>1</sup>,  
Marcos Antônio de Matos Laia<sup>1</sup>, Daniel Luiz Alves Madeira<sup>1</sup>

<sup>1</sup>Departamento de Computação  
Universidade Federal de São João Del Rei  
São João Del Rei – MG – Brasil

fls@ufsj.edu.br, avnerpaulo.mg@gmail.com

marcoslaia@gmail.com, dmadeira@ufsj.edu.br

**Abstract.** *Optical Music Recognition (OMR) is a Computer Science field applied to Music and has opening problems to recognize handwritten scores. This paper presents a project called “Copista” that is investigating techniques to develop a software to recognize handwritten scores.*

**Resumo.** *Recuperação Musical Ótica (OMR) é uma área da Computação aplicada à Música com problemas em aberto para reconhecimento de partituras manuscritas. O presente artigo apresenta um projeto chamado “Copista” que está investigando técnicas com o intuito de desenvolver uma ferramenta computacional para o reconhecimento de partituras manuscritas.*

## 1. Introdução

Os municípios de São João del-Rei, Tiradentes e Prados possuem, juntas, uma história musical com raízes no início do século XVIII, e são depositárias de um dos mais relevantes conjuntos documentais, ao lado de Ouro Preto e Mariana, da história musical de Minas Gerais e do Brasil. A riqueza de gêneros vai da música sacra, passando pela ópera e música de concerto, ao repertório das bandas de música. Cronologicamente, possuem cópias musicais da segunda metade do século XVIII aos dias atuais, fruto do labor de uma centena de nomes de compositores, profissional este, sempre presente e atuante na história daquelas sociedades.

Neste cenário, o Departamento de Música (DMUSI) da Universidade Federal de São João Del Rei (UFSJ) está lançando o Programa Memória Viva, que tem como principal objetivo atuar junto a arquivos musicais em São João del-Rei, Tiradentes, Prados e São Tiago, visando o desenvolvimento e implementação de metodologias para descrição de documentos musicais (partituras e partes), que melhor atendam à peculiaridade de cada instituição. Muitas destas partituras são documentos históricos de valor inestimável e a sua disponibilização ao público depende de publicação de cópias físicas ou digitais. As cópias digitais idênticas da música original, de grande valor para a área da Musicologia, trazem em si uma dificuldade maior de leitura devido ao seu estado de conservação, conforme ilustra a Figura 1.

A criação de uma nova partitura reescrita a partir da partitura original traz a possibilidade de este acervo ser mais explorado, estudado, executado e difundido. Uma vez que

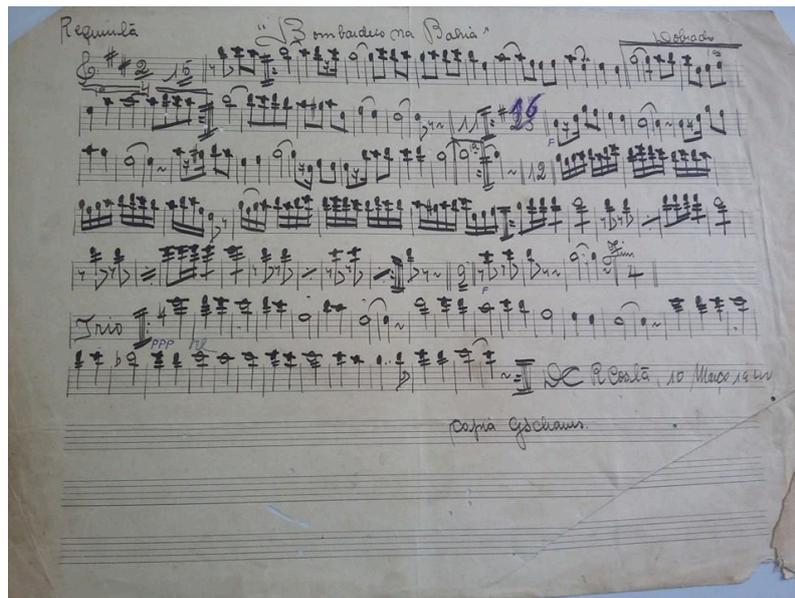


Figura 1. Exemplo de partitura presente no acervo

as peças musicais estejam catalogadas, pelo projeto Memória Viva, o próximo passo seria a conversão de tais partituras para um formato musical simbólico para a disponibilização do acervo ao público.

Como fruto de uma parceria entre o Departamento de Ciência da Computação (DCOMP) e o DMUSI, o presente projeto tem o intuito de prover uma ferramenta de amparo tecnológico para o Programa Memória Viva para a conversão destas partituras em formatos de música digital. Esta ferramenta é chamada **Copista**.

## 2. O Copista

O processo de conversão das partituras para um formato musical digital simbólico é também conhecido por OMR (Optical Music Recognition)[Rebello et al. 2012][Bainbridge and Bell 2001]. Apesar de existirem ferramentas que façam o reconhecimento de partituras a partir de uma imagem (OMR), a maior parte destas ferramentas a) não trabalha com reconhecimento de partituras manuscritas[Bainbridge and Bell 2001], b) possuem um alto custo de aquisição e c) não possuem código aberto de forma que a adaptação da mesma ao projeto em questão é impossível. Tais considerações foram levadas em conta para o início do desenvolvimento do projeto Copista.

O processo de OMR no Copista foi dividido em 3 partes distintas: o pré-processamento e restauro eletrônico das partituras originais, o reconhecimento dos elementos musicais por visão computacional e a representação digital dos elementos musicais reconhecidos, como ilustrado na Figura 2.

A intenção do projeto é que, a partir desta representação digital, será possível reescrever as partituras originais em programas de notação musical modernos de maneira a permitir a reedição destas partituras e a análise musical apoiada por computador das obras.

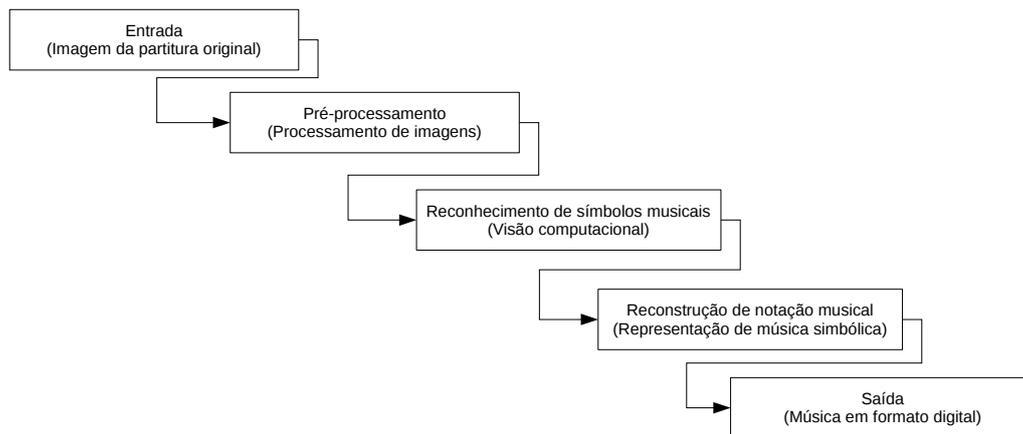


Figura 2. Estrutura básica de sistemas para OMR

### 2.1. Entrada

A entrada do Copista é a cópia digital das partituras que compõem o acervo regional. Nestes acervos, é comum que as partituras, muitas delas centenárias, tenham sido utilizadas em missas e procissões, e possuam dobras, marcas de cera de vela, amassados, rasgos e outros tipos de avarias, conforme pode ser visto na Figura 2.1:

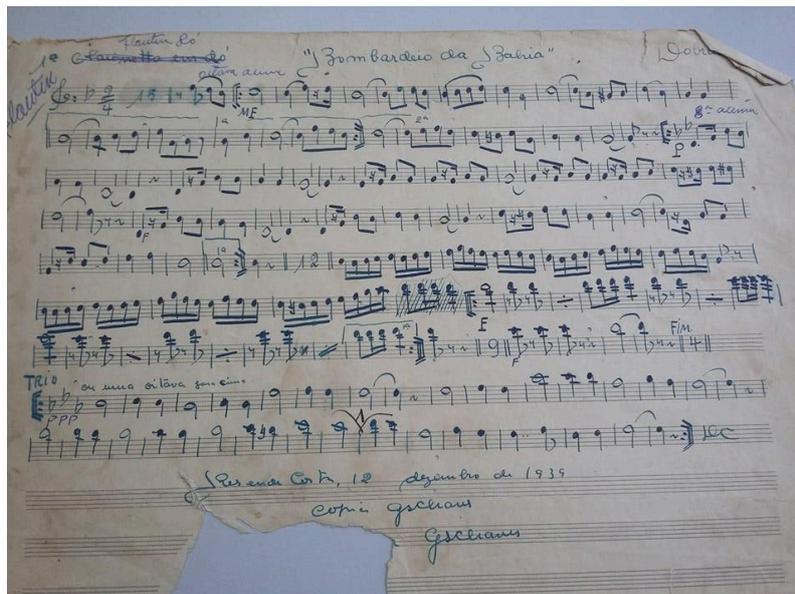


Figura 3. Partitura manuscrita e com avarias

Por esta razão, o Programa de Extensão deverá, primeiramente, tentar realizar o restauro físico das partituras dos acervos. Uma vez que este restauro for executado, a partitura deverá ser digitalizada para ser processado pelo Copista.

### 2.2. Pré-processamento

É comum que documentos muito antigos e manuscritos sofram de degradação ao longo do tempo. O manuseio constante pode fazer com que a tinta clareie, criando diferenças de tonalidades, ou pode criar diversas marcas. No exemplo das partituras, pingos de cera de

vela e marcas de suor de mãos ocorrem em vários casos. Além disso, o armazenamento inadequado provocou dobras, amassados, rasgos e até que traças fizessem buracos nas partituras. Todos esses artefatos não são relevantes à partitura e precisam ser retirados para, ao final do processo, obter a partitura o mais limpa possível, contendo somente os elementos necessários para o reconhecimento da mesma.

Não há uma técnica universal de pré-processamento, já que cada conjunto específico de documentos pode exigir tratamento diferenciado. Porém, dois passos podem ser destacados como básicos no processo de pré-processamento:

1. remoção de artefatos
2. limiarização

A primeira etapa envolve remoção de todo artefato não importante ao processo de reconhecimento da partitura. Estes artefatos, que se tornam ruído na imagem digitalizada, englobam as manchas, rasgos, furos e todas marcas que não são parte da partitura. O fundo do papel pode também ser considerado ruído devido ao fato de não ser importante para o reconhecimento da partitura em si. Para essa etapa, utiliza-se a filtragem da imagem original [Fujinaga 2004, Szwoch 2007]. Ao final deste processo, é necessária uma correção dos tons de cinza para ampliação do contraste, facilitando o próximo passo.

Com o ruído devidamente removido, a partitura deve ser convertida para preto-e-branco, através do processo de binarização (limiarização em dois tons de cinza). Este processo visa simplificar a representação da partitura, ao eliminar toda a variação de cores existente, e pode ser classificado em duas categorias: limiarização global ou local. Os métodos globais são classificados de acordo com um único limiar global [Otsu 1975, Pinto et al. 2011, Seixas et al. 2008], válido para toda a imagem. Ao utilizar um só limiar, os métodos tendem a ser mais simples e computacionalmente mais baratos. Porém ruídos que ocorram em somente parte da imagem irão influenciar na tomada de decisão do algoritmo, o que pode gerar resultados indesejáveis. Para contornar este problema, os métodos de limiarização local trabalham com subconjuntos da imagem de entrada, calculando o limiar ideal por região [Sauvola and Pietikäinen 2000, Bernsen 1986]. Nesses métodos há uma adaptatividade maior do método, ao permitir que a decisão sobre uma região dependa somente dela, independente das regiões vizinhas. Com isso, obtêm-se resultados melhores, mas também com custo computacional proporcionalmente maior.

Como nas partituras alvo do projeto temos artefatos como marcas de suor das mãos e pingos de vela, que não ocorrem em toda a área, os métodos locais tendem a ser mais indicados para o Copista. Nesta etapa então, o conjunto de técnicas de filtragem para remoção de diferentes ruídos e para binarização de imagens deverão ser avaliados. A avaliação dos resultados pode ser realizada através de partituras padrão, cujo conteúdo já é conhecido, em conjunto com a próxima etapa do Copista.

### **2.3. Reconhecimento de Símbolos Musicais**

O Reconhecimento de Símbolos Musicais fará uso de técnicas de visão computacional determinadas em etapas específicas:

1. Segmentação de linhas e notações;
2. Localização das notações
3. Definição de descritores para cada notação

4. Reconhecimento de padrões das notações
5. Mapeamento das notas nas partituras
6. Identificação de propriedades específicas de notações musicais

A etapa de segmentação [Gonzalez et al. 2004] permite separar elementos como linhas e as demais notações para serem treinadas. As linhas podem ainda ser utilizadas para definir a localização de uma notação. Por exemplo, a altura da nota de acordo com seu posicionamento em relação às linhas, separar diferentes símbolos sobrepostos [Bainbridge and Bell 1997] e de diferentes tamanhos ou posições rotacionadas [Mundy et al. 1992].

Cada notação, por sua vez, pode ser descrita por um conjunto de *features* [Koendrik 1992] (características). Cada *feature* pode representar algo da imagem a ser reconhecida como bordas, curvaturas, bolhas, cumes e pontos de interesse. As *features* extraídas das imagens são utilizadas posteriormente em um processo de reconhecimento de padrões [Fukunaga 2013, Ripley 1996], após serem qualificadas e quantizadas para uma análise estatística por um filtro para diminuir incertezas entre as alturas de notas ou presença de artefatos nas imagens.

Nessa etapa pode ser utilizado um filtro de Kalman [Laia 2013] que permitirá a correção dos dados gerados pela extração de *features*. Ao combinar técnicas de visão computacional em OMR, haverá um ganho maior sobre como gerar esses dados e garantindo a integridade e fidelidade ao que está presente no documento.

Além disto, técnicas de visão computacional utilizadas para outras aplicações como reconhecimento de caracteres [Dori et al. 1996], reconhecimento de escrita à mão [Xu et al. 1992], realidade aumentada com marcadores em baixa resolução [Furht 2011] também serão utilizadas para completar esta etapa do processo.

## 2.4. Reconstrução de Notação Musical

No processo de OMR, a etapa de reconstrução da representação simbólica deve receber dados da visão computacional e mapeá-los para um alfabeto de símbolos musicais. Este mapeamento pode incluir uma validação de um determinado símbolo de modo a auxiliar a etapa de reconhecimento quanto à corretude de um determinado elemento gráfico com uma análise a partir do modelo notacional [Good 2001] ou baseada no contexto musical [Medina et al. 2003]. A validação pode ocorrer pela criação de um conjunto de regras léxicas, sintáticas e/ou semânticas que definirão o formato de representação simbólica.

Uma questão principal de definir uma representação musical simbólica é encontrar uma representação genérica suficiente, bastante flexível mas ao mesmo tempo restrita em relação à suas regras para permitir uma validação da estrutura musical como um todo [Selfridge-Field 1997].

A maior parte dos modelos existentes parte de uma estrutura musical hierárquica [Buxton et al. 1978] onde há uma visão geral da música, dividida em várias pautas (linhas), que são divididas em compassos e estes compassos em tempo e tempos em notas. Para este projeto, será adicionado ao modelo uma hierarquia ainda mais profunda onde serão incluídas informações sobre as pautas e a página da partitura. Uma possibilidade computacional de alcançar tal representação é utilizar um modelo orientado

a objetos [Travis Pope 1996], de maneira a definir a representação por um conjunto de objetos com atributos valorados.

Tais atributos valorados deverão armazenar a notação musical de um símbolo assim como registrar informações do símbolo dentro da imagem. Por esta razão, dividimos a representação simbólica musical para OMR em duas partes, uma que registra a informação musical e a outra que registra a informação da imagem.

Os dados valorados da imagem original em que foi encontrado um símbolo musical são necessários para permitir uma reavaliação de dados reconhecidos erroneamente. Isto permitiria requisitar a visão computacional que a validação de um determinado símbolo seja refeita para conferência, de maneira automática.

Outros dados da imagem original que poderão ser armazenados dizem respeito aos processamentos iniciais feitos na imagem. Armazenar informações como brilho, contraste, cor, rotação, translação, histograma e quais passos foram realizados para retirada dos artefatos torna-se necessário para que, se houver dificuldade na identificação ou caso o software identifique incorretamente um símbolo, confundindo com uma rasura ou furo na página e assim causando erro léxico, o pré-processamento possa ser reajustado através da alteração destes parâmetros, na tentativa de melhorar a qualidade na leitura da página.

Assim, temos uma cadeia de processamentos que poderão ser alteradas por um classificador baseando-se em acertos e erros de leitura. Ao digitalizar uma partitura de ponta-cabeça por exemplo, e identificar que uma transformação de rotação se faz necessário, o algoritmo irá armazenar esta operação como um atributo do elemento página e reprocessá-la hierarquicamente caso seja necessário reprocessar algum elemento.

Além disto, deverá ser possível converter os dados valorados armazenados pela ferramenta em algum formato de música simbólica existente.

## 2.5. Saída

Esta última etapa, a definição do formato de representação simbólica, é uma tarefa crítica no contexto do desenvolvimento desta ferramenta. Esta definição irá influenciar o desenvolvimento da ferramenta pois a validação dos símbolos reconhecidos dentro do modelo de representação pode auxiliar no algoritmo de aprendizado da etapa de visão computacional e com isto diminuir a necessidade de intervenção humana no processo de transcrição das partituras digitalizadas.

A definição da representação também é uma tarefa crítica pois a mesma irá prover um modelo para a geração da saída da ferramenta. A saída da ferramenta deverá ser o mais interoperável possível de maneira a permitir que haja possibilidade de edição e intervenção humana na correção de uma partitura gerada, caso isto seja necessário. A correção humana realizada em uma partitura com problemas de identificação pode servir como uma nova entrada no sistema pois permitiria uma nova etapa de aprendizagem para os algoritmos propostos.

Para a avaliação de adaptação será levado em conta a) o levantamento de todos os símbolos utilizados nestas partituras b) a representação computacional hierárquica deste conjunto de símbolos, c) as regras léxicas, sintáticas e semânticas do formato para permitir a correção de partituras em seu formato simbólico e d) a conversão deste conjunto de símbolos em formatos comumente utilizados em aplicações musicais.

Entre os formatos de música simbólica para a ferramenta estão:

- **ABC**[Oppenheim et al. 2010]: É uma linguagem para notação de músicas que utiliza caracteres no formato ASCII. É uma linguagem que pode ser lida e interpretada, mesmo sem a utilização de software, bastando a pessoa ter conhecimento musical para fazê-lo. Apesar de conseguir representar os símbolos presente em uma partitura, não é adequada para este projeto, pois é crucial que a linguagem para notação musical também represente a posição em que a partitura foi escaneada e ao invés de representar a nota que vai ser tocada, é necessário que seja armazenada a posição (X, Y) de cada símbolo na partitura digitalizada.
- **MusicXML**[Good 2001]: É uma linguagem para notação musical que utiliza formato XML. Diferente do formato ABC, MusicXML não é tão intuitiva. É necessário o estudo e prática da mesma, para que seja possível ser feita a leitura de uma musica escrita em MusicXML por uma pessoa. Consegue fazer uma ótima representação de uma partitura, com detalhes minuciosos. Porém também é inadequada para este projeto, pois apesar de fazer ótima representação de uma partitura, também não armazena a posição (X, Y) dos símbolos e nem a posição que a partitura foi digitalizada.
- **Lilypond**[Nienhuys and Nieuwenhuizen 2003]: Também em formato ASCII. Muito intuitiva para leitura, mas exige estudo para escrita a mão. Consegue representar uma partitura tão bem quanto o MusicXML, porém utilizando o nome das notas para tal representação, diferente do MusicXML. Por utilizar o nome das notas para representação e nenhum meio para definir sua posição (X, Y) na partitura, esse tipo de notação também é inadequado para o projeto.
- **Music21**[Ariza and Cuthbert 2011][Cuthbert and Ariza 2010]: Formato também baseado em ASCII. Esta notação oferece um modelo de objeto em linguagem Python. Esse kit de ferramenta foi desenvolvido para auxiliar na criação e manipulação de dados de música simbólica. Consegue armazenar e ordenar a posição e a duração dos elementos. As notas são criadas a partir de comandos. A nota em si, que é escrita como cifra, é passada como parâmetro. Contudo, esta linguagem de notação musical também é inadequada para o projeto, por não ter um meio para o armazenamento das características como, cor, brilho, contraste nem histograma da partitura.
- **GUIDO**[Hoos et al. 1998]: Também em ASCII. Representa utilizando a cifra das notas, seguido da duração da mesma, e utiliza um método de localização das notas por oitava, onde é colocado o respectivo número ao lado da nota. Também é inadequado ao projeto. Apesar de dispor de um método de localização, não é o método necessário para o projeto. Neste projeto é necessário que exista um meio de localizar a nota por meio de uma posição (X, Y) a partir da página. Além disso, este formato também não dispõe de um meio para definir a posição em que foi digitalizada a partitura.

Também foram pesquisados os formatos de música simbólica MIDI[Association et al. 1996] e NIFF (Notation Interchange File Format)[Grande 1997] mas tais formatos são binários e por isto considerados inadequados para este projeto. Quanto ao formato MuseData[Hewlett 1997], não foi encontrado referência suficiente sobre o mesmo.

### 3. Resultados obtidos

O processo de reconhecimento de partituras musicais é feito por meio de etapas que abrangem o pré-processamento da imagem (retirada de possíveis ruídos e artefatos), segmentação (separação de elementos nas imagens), detecção, classificação e reconhecimento dos elementos musicais.

Na primeira etapa, o pré-processamento irá preparar a imagem para o processo de visão de computacional. Para esta etapa, podem ser utilizados filtros que transformam a imagem colorida em escala de cinza, para posterior conversão em preto-e-branco, filtragem de ruídos (que podem ocorrer devido à manipulação da partitura ou do próprio processo de digitalização da mesma), detecção e retirada de artefatos, ou seja, defeitos ou presença de objetos que não pertencem originalmente às partituras. Nesta etapa, também é feita a correção da orientação das pautas como também a equalização da luz sobre a pauta. Concluindo esta etapa, é feita a binarização da imagem, onde há a possibilidade de presença apenas dos elementos da partitura.

Os testes preliminares do processamento da visão computacional utilizou partituras geradas digitalmente para uma série de testes de implementações de algoritmos. Por esta razão, nestes testes não foi necessário realizar o pré-processamento das imagens.



Figura 4. Partitura original digitalizada

Na segunda etapa, é feita a segmentação de elementos para serem detectados e posteriormente reconhecidos nas próximas etapas. A imagem binarizada (apresentada na Figura 4), oriunda da etapa de pré-processamento, recebe um processo de segmentação onde é retirada as linhas das partituras para que cada elemento se torne desconectado, ou seja, não apresente pixels que possam conectar um elemento ao outro.



Figura 5. Partitura após o processo de dilatação

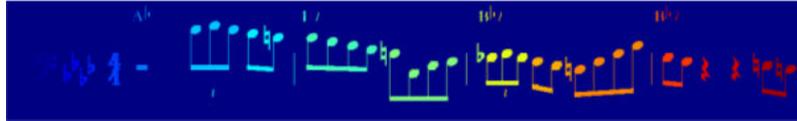
Para este processo, utilizou-se técnicas de operações morfológicas, primeiro aplicando a dilatação com o objetivo de se eliminar as linhas. A dilatação é um processo onde a parte clara da imagem é expandida, comprimindo a parte escura. Pode-se ver o resultado dessa operação na Figura 5.



Figura 6. Partitura após o processo erosão

Com a dilatação, as notas se apresentaram mais delgadas, devido a perda de pixels escuros. Para retornar o estado anterior a esse processo, pode-se utilizar outra operação

morfológica chamada erosão. A erosão é um processo onde a parte clara sofre uma diminuição enquanto a parte escura aumenta, em um processo inverso à dilatação. O resultado pode ser visto na Figura 6. Após a retirada das linhas, chega-se na etapa de detecção dos elementos e etiquetagem. Cada grupo de pixels conectados recebe um único valor, que o diferencia dos demais.



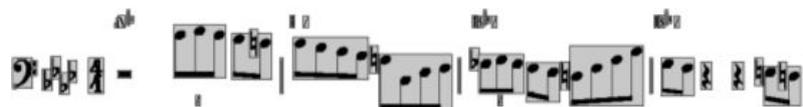
**Figura 7. Partitura com os elementos etiquetados**

Nesta etapa, os elementos podem ser selecionados um a um pelo valor ao qual é identificado. O fundo da imagem é apresentado com o menor valor (no caso da Figura 7, o valor é igual a 0) e o primeiro elemento com o valor consequente. Assim, se a imagem apresentar 200 elementos, o último elemento terá pixels no valor de 200.



**Figura 8. Elementos encontrados na imagem da partitura: (a) primeiro, (b) décimo sexto, (c) décimo nono e (d) trigésimo quinto**

Na etapa de separação dos elementos detectados, cada elemento é cortado da imagem e seus pixels normalizados para 1 (branco) para objeto e 0 (preto) para fundo, como se pode ver na Figura 8. O fundo é deixado em preto, pois para essa etapa são utilizados os momentos invariantes de Hu, onde é feita a descrição da imagem de cada elemento. Momentos de Hu se baseiam em momentos invariantes (invariação à escala, translação e rotação). O primeiro momento, por exemplo, fornece o centro do elemento. Logo, o elemento pode estar em escalas diferentes (apresenta-se em tamanhos maiores ou menores) ou posições diferentes, rotacionado ou espelhado na imagem. Na etapa de descrição ou identificação dos elementos, é criado um objeto. Esse objeto apresenta uma imagem recortada do elemento, seus momentos de Hu, posição do seu centro dentro da imagem e a área que ele ocupa. A partir dessa etapa, pode-se reconstruir a partitura novamente usando os dados disponíveis e futuramente usá-los para a etapa de classificação e reconhecimento, conforme ilustrado na Figura 9. Para averiguar a coerência dos dados, uma imagem contendo os elementos detectados foi reconstruída a partir dos dados da descrição.



**Figura 9. Imagem reconstruída com os dados da etapa de descrição**

No futuro, após a etapa de classificação, os objetos não conterão mais imagens, apenas sua posição na imagem original, seu tamanho, seus momentos invariantes e um

código identificador quanto a sua classificação: se é uma clave de fá ou sol, notas conectadas ou não, etc.

#### 4. Conclusão

Este projeto desencadeou a pesquisa conjunta de pesquisadores de diferentes áreas da Ciência da Computação como Visão computacional, Processamento de Imagens, Computação Musical, Inteligência Artificial, Compiladores, Redes complexas e Recuperação de Informação. A união destas áreas deverá auxiliar o desenvolvimento da ferramenta pretendida pelo projeto trazendo também ganhos pela interdisciplinaridade da pesquisa dentro da grande área da Ciência da Computação. Além de colaborar como pesquisa interdisciplinar na área da Ciência, o projeto deverá também auxiliar a área da música gerando uma ferramenta open-source para reconhecimento e reescrita de partituras.

Os primeiros passos deste projeto envolveram a pesquisa por técnicas e ferramentas computacionais a serem utilizadas em cada um dos passos do processamento do Copista. O levantamento destes algoritmos permitiu testes preliminares na área de Visão computacional com bons resultados iniciais. Os próximos passos do projeto deverão mesclar as técnicas levantadas e os códigos implementados nos passos individuais desta pesquisa em um primeiro protótipo funcional. Possivelmente, este primeiro protótipo deverá trabalhar ainda com partituras digitais e não-manuscritas para o treinamento do reconhecimento de uma rede neural que deverá ser utilizada para a tomada de decisão em relação à correção de um símbolo identificado.

Outro passo que deverá ser dado em breve será a integração da representação com a Visão Computacional e a verificação dos elementos identificados por meio de um compilador de música simbólica. Tal passo deverá auxiliar também no treinamento da ferramenta sendo mais um passo em busca de um melhor resultado para atender o objetivo proposto.

##### 4.1. Agradecimentos

Agradecemos a Universidade Federal de São João Del Rei e o Programa de Iniciação Científica que está apoiando este projeto.

#### Referências

- Ariza, C. and Cuthbert, M. (2011). *The music21 stream: A new object model for representing, filtering, and transforming symbolic musical structures*. Ann Arbor, MI: MPublishing, University of Michigan Library.
- Association, M. M. et al. (1996). *The complete MIDI 1.0 detailed specification: incorporating all recommended practices*. MIDI Manufacturers Association.
- Bainbridge, D. and Bell, T. (1997). Dealing with superimposed objects in optical music recognition.
- Bainbridge, D. and Bell, T. (2001). The challenge of optical music recognition. *Computers and the Humanities*, 35(2):95–121.
- Bernsen, J. (1986). Dynamic thresholding of gray-level images. In *International Conference on Pattern Recognition*.

- Buxton, W., Reeves, W., Baecker, R., and Mezei, L. (1978). The use of hierarchy and instance in a data structure for computer music. *Computer Music Journal*, pages 10–20.
- Cuthbert, M. S. and Ariza, C. (2010). music21: A toolkit for computer-aided musicology and symbolic music data.
- Dori, D., Doerman, D., Shin, C., Haralick, R., Phillips, I., Buchman, M., and Ross, D. (1996). Handbook on optical character recognition and document image analysis, chapter the representation of document structure: a generic object-process analysis.
- Fujinaga, I. (2004). Staff detection and removal. *Visual perception of music notation: on-line and off-line recognition*, pages 1–39.
- Fukunaga, K. (2013). *Introduction to statistical pattern recognition*. Academic press.
- Furht, B. (2011). *Handbook of augmented reality*. Springer Science & Business Media.
- Gonzalez, R. C., Woods, R. E., and Eddins, S. L. (2004). *Digital image processing using MATLAB*. Pearson Education India.
- Good, M. (2001). Musicxml for notation and analysis. *The virtual score: representation, retrieval, restoration*, 12:113–124.
- Grande, C. (1997). The notation interchange file format: A windows-compliant approach. In *Beyond MIDI*, pages 491–512. MIT Press.
- Hewlett, W. B. (1997). Beyond midi. chapter MuseData: Multipurpose Representation, pages 402–447. MIT Press, Cambridge, MA, USA.
- Hoos, H. H., Hamel, K. A., Renz, K., and Kilian, J. (1998). The guido notation format – a novel approach for adequately representing score-level music.
- Koendrik, J. J. (1992). Computational vision (book). *Ecological Psychology*, 4(2):121–128.
- Laia, M. A. d. M. (2013). *Filtragem de Kalman não linear com redes neurais embarcada em uma arquitetura reconfigurável para uso na tomografia de Raios-X para amostras da física de solos*. PhD thesis, Universidade de São Paulo.
- Medina, R. A., Smith, L. A., and Wagner, D. R. (2003). Content-based indexing of musical scores. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '03*, pages 18–26, Washington, DC, USA. IEEE Computer Society.
- Mundy, J. L., Zisserman, A., et al. (1992). *Geometric invariance in computer vision*, volume 92. MIT press Cambridge.
- Nienhuys, H.-W. and Nieuwenhuizen, J. (2003). Lilypond, a system for automated music engraving. In *Proceedings of the XIV Colloquium on Musical Informatics (XIV CIM 2003)*, volume 1. Citeseer.
- Oppenheim, I., Walshaw, C., and Atchley, J. (2010). The abc standard 2.0.
- Otsu, N. (1975). A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27.

- Pinto, T., Rebelo, A., Giraldi, G., and Cardoso, J. S. (2011). Music score binarization based on domain knowledge. In *pattern recognition and image analysis*, pages 700–708. Springer.
- Rebelo, A., Fujinaga, I., Paszkiewicz, F., Marcal, A., Guedes, C., and Cardoso, J. (2012). Optical music recognition: state-of-the-art and open issues. *International Journal of Multimedia Information Retrieval*, 1(3):173–190.
- Ripley, B. D. (1996). *Pattern recognition and neural networks*. Cambridge university press.
- Sauvola, J. and Pietikäinen, M. (2000). Adaptive document image binarization. *PATTERN RECOGNITION*, 33:225–236.
- Seixas, F. L., Martins, A., Stilben, A. R., Madeira, D., Assumpção, R., Mansur, S., Victor, S. M., Mendes, V. B., and Conci, A. (2008). Avaliação dos métodos para a segmentação automática dos tecidos do encéfalo em ressonância magnética. *Simpósio de Pesquisa Operacional e Logística da Marinha SPOLM*.
- Selfridge-Field, E. (1997). Beyond codes: issues in musical representation. In *Beyond MIDI*, pages 565–572. MIT Press.
- Szwoch, M. (2007). Guido: A musical score recognition system. In *icdar*, pages 809–813.
- Travis Pope, S. (1996). Object-oriented music representation. *Organised Sound*, 1(01):56–68.
- Xu, L., Krzyzak, A., and Suen, C. (1992). Methods of combining multiple classifiers and their applications to handwriting recognition. *Systems, Man and Cybernetics, IEEE Transactions on*, 22(3):418–435.

# Utilização de Software Livre no ensino de Música da UFSJ

Luan Luiz Gonçalves<sup>1</sup>, Flávio Luiz Schiavoni<sup>1</sup>

<sup>1</sup>Departamento de Computação  
Universidade Federal de São João Del Rei  
São João Del Rei – MG – Brasil

luanlg.cco@gmail.com, fls@ufsj.edu.br

**Abstract.** *This paper presents a survey of software needs of the Music Department (DMUSI) of UFSJ, preferentially, FLOSS tools (Free Libre Open Source Software). The survey aims to provide both specific tools for music education and support tools for the professional musician. After the survey, there will be assessments of FLOSS applications together with the Department of the Federal University of Music São João Del Rei and these applications will be available in the DMUSI computer lab.*

**Resumo.** *Este artigo apresenta um levantamento das necessidades de programas de computador do Departamento de Musica (DMUSI) da UFSJ buscando, preferencialmente, ferramentas FLOSS (Free Libre Open Source software). O levantamento pretende cobrir tanto ferramentas específicas para o ensino de música quanto para o suporte a profissional desta área. Após o levantamento, será realizado avaliações das aplicações FLOSS, juntamente ao Departamento de Música da Universidade Federal de São João Del Rei e essas aplicações ficarão disponíveis no laboratório de informática do DMUSI.*

## 1. Introdução

A informática, devido a sua transversalidade, vem sendo utilizada como ferramenta de/para ensino em diversas áreas do conhecimento humano. Há ferramentas computacionais desenvolvidas especialmente para o contexto de ensino e também ferramentas que são utilizadas no contexto de produção por profissionais e que, para preparar o aluno para o mercado de trabalho, são utilizadas no contexto do ensino. Independentemente de uma ferramenta ser voltada ao ensino ou à prática profissional, a escolha de ferramentas para a adoção em um curso superior pode ser influenciada por alguns fatores como, a disponibilidade do software e a possibilidade de alteração do mesmo para o seu aperfeiçoamento ou entendimento. A questão da disponibilidade inclui, por exemplo, o valor da licença do software e sua disponibilidade de forma a permitir ao aluno recém formado adquirir a ferramenta utilizada em seu período de graduação no início de sua vida profissional. Entre as opções de ferramentas disponíveis para tal tipo de atividade estão as ferramentas Software Livre.

A opção pelo software livre (também chamado de FLOSS - *Free Libre Open Source Software*) para as atividades acadêmicas se dá por vários outros fatores que vão além da questão de sua gratuidade [Wheeler 2005]. Entre estes fatores, está também a disponibilização do código-fonte da ferramenta, o que permite que o software possa ser estudado, alterado e adaptado por seus usuários. Com isto, a ferramenta tecnológica deixa

de ser uma “caixa-preta” e passa a ser passível de estudo e verificação, contribuindo para a desmitificação da informática e para o entendimento e melhoramento das mesmas.

A escolha de FLOSS para um currículo acadêmico depende de diversos fatores, como a existência de ferramentas, a simplicidade de sua utilização, a similaridade com ferramentas comerciais e a existência de manuais, tutoriais e documentação sobre o mesmo. Entre as áreas onde há uma quantidade significativa de ferramentas FLOSS para o auxílio do currículo acadêmico está a Música.

### 1.1. O que é Software Livre?

Software Livre, conforme a definição criada pela Free Software Foundation, é o software que pode ser usado, copiado, estudado, modificado e redistribuído sem restrição. A forma usual de um software ser distribuído livremente é sendo acompanhado por uma licença de software livre (como a GPL ou a BSD), e com a disponibilização do seu código-fonte [Campos 2006].

Software Livre se refere à existência simultânea de quatro tipos de liberdade para os usuários do software, definidas pela Free Software Foundation. As 4 liberdades básicas associadas ao software livre são [Campos 2006]:

1. A liberdade de executar o programa, para qualquer propósito;
2. A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades. Acesso ao código-fonte é um pré-requisito para esta liberdade;
3. A liberdade de redistribuir cópias de modo que você possa ajudar ao seu próximo;
4. A liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie (liberdade nº 3). Acesso ao código-fonte é um pré-requisito para esta liberdade.

### 1.2. Ferramentas FLOSS e práticas musicais

Várias práticas musicais podem ser auxiliadas por ferramentas computacionais. Entre estas práticas, podemos citar a composição, arranjo, performance, ensaio, gravação, distribuição, entre outras. Algumas ferramentas atendem exatamente a uma prática musical enquanto outras ferramentas atendem partes de uma prática. Em alguns casos, é possível combinar mais de uma ferramenta para atender uma determinada prática enquanto em outros casos uma mesma ferramenta pode atender mais de uma prática musical. Alguns exemplos de ferramentas são:

- Servidores de som [Letz et al. 2004]
- Gravadores multi pista [Davis et al. 2012]
- Editores e efeitos de áudio [Furse 2000]
- Editores de partitura [Pinhati and Siqueira 2012]
- Instrumentos virtuais [Bullock and Coccioli 2006]
- Ferramentas para música em rede [Schiavoni et al. 2011]
- Ambientes de análise musical [Cannam et al. 2006]
- Ferramentas para DJ [Andersen 2003]
- Linguagens de programação Musical [Lazzarini and Walsh 2007, Wilson et al. 2011, Puckette et al. 1996]

### 1.3. Motivação

Grande parte dos alunos e profissionais da música utilizam ferramentas computacionais para exercerem suas atividades específicas na música. Porém, apesar de haver iniciativas no intuito de mapear ferramentas para o apoio de práticas musicais [Amorim 2014], não há levantamento de ferramentas FLOSS que possam cobrir as atividades dos músicos para auxiliá-los na escolha destas ferramentas. Boa parte dos músicos opta então por ferramentas proprietárias, as quais dependem de um investimento muitas vezes alto para sua aquisição. Sabendo que os hardwares específicos para atividade do músico também dependem de um alto investimento (instrumentos, monitores de som, microfones, interfaces, controladores), há casos em que o profissional tem que escolher entre adquirir o hardware ou o software. Tal escolha não seria necessária se os músicos tivessem indicações de qual ferramenta FLOSS pode ser incluída em seu *metièr* tecnológico.

Com o intuito de identificar estes fatores, este projeto está realizando o levantamento e avaliação de aplicações FLOSS para a área de música, juntamente ao Departamento de Música da Universidade Federal de São João Del Rei.

## 2. Metodologia

A pesquisa iniciou-se com uma reunião do Departamento de Música (DMUSI) da Universidade Federal de São João Del Rei (UFSJ), solicitando a anuência do corpo docente para a sua realização. Com o devido esclarecimento e concordância, o DMUSI disponibilizou uma lista com o nome e e-mail de todos os professores.

Após este passo, foi feito um levantamento das disciplinas comuns aos bacharelados do Departamento de Música da UFSJ. O curso de Música na UFSJ tem duas qualificações, habilitação em Educação Musical e em Instrumento/Canto. O professor de cada disciplina foi entrevistado com objetivo de discutir quais as ferramentas utilizadas, tanto em sala de aula quanto na preparação de suas aulas. A entrevista incluiu alunos que fazem uso de ferramentas computacionais para atender as exigências das disciplinas do curso de Música da UFSJ.

Além de entrevistas e questionários diretos, a pesquisa também aconteceu em sala de aula. No primeiro semestre de 2015 foi oferecido uma matéria optativa, *Edição de Partituras e Tecnologias Sonoras aplicadas à Análise e à Educação Musical*, ministrada pelo professor Marcos Edson Cardoso Filho onde o autor deste artigo trabalhou como monitor. Os alunos da matéria optativa tiveram o primeiro contato com algumas ferramentas FLOSS, como o *Muscore*, software de edição de partitura e o *Sonic Visualiser*, para análise de áudio. O *Muscore* é equivalente ao *Finale* e o *Sibelius*, duas ferramentas pagas e as mais usadas no DMUSI pelos professores e alunos. Os softwares FLOSS apresentados foram bem avaliados pelos alunos e atendem as necessidades do DMUSI. É importante frisar que esta disciplina optativa envolveu alunos de qualificações e instrumentos diversos. Além disto, esta monitoria fez com que o laboratório de computadores ficasse aberto durante o horário de aulas, permitindo a interação do pesquisador com outros alunos do DMUSI.

No mesmo período, houve uma parceria entre os Departamentos de Música e de Ciência da Computação em um programa de extensão do DMUSI chamado *Vivências*

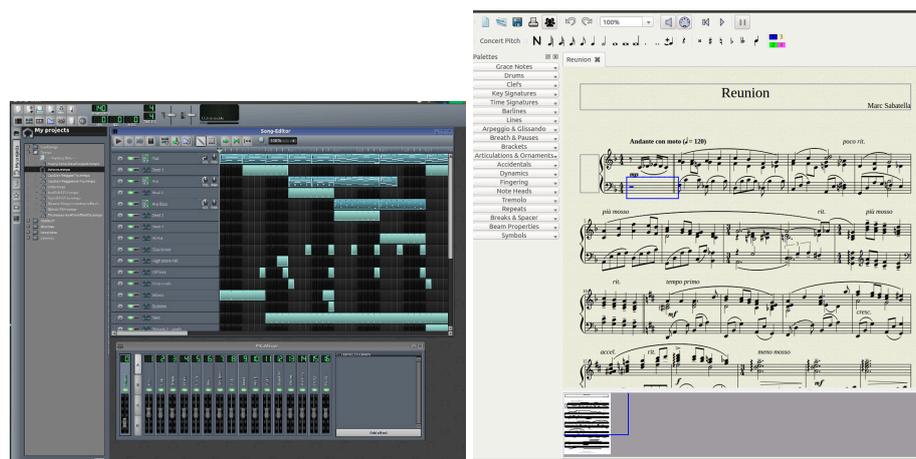


Figura 1. Tela dos programas utilizados no programa de extensão

*Musicais*<sup>1</sup>. Este programa de extensão, que até então só tinha participação dos alunos e professores da Música, passou a oferecer o curso de *Música e Tecnologia*, ministrado por alunos do curso de Ciência da Computação. O autor deste artigo foi ministrante deste curso, no qual os alunos aprenderam a usar uma ferramenta de edição de partitura (*Muscore*) e uma estação de trabalho de áudio digital (*LMMS*), ambas ferramentas FLOSS, apresentadas na Figura 1. Foi possível observar como os alunos lidavam com os softwares e notar a facilidade de uso dos mesmos.

Com isto, tivemos a oportunidade de discutir e utilizar algumas ferramentas FLOSS para práticas musicais em diferentes contextos: com alunos e professores do DMUSI, com alunos interessados em uma disciplina optativa de tecnologia e com membros da comunidade por meio de um projeto de extensão.

### 3. Levantamentos realizados

O levantamento está sendo realizado através de entrevistas, questionários online e impressos. Algumas ferramentas já foram levantadas, mas este processo ainda não foi encerrado. É importante o levantamento ser realizado pensando nos tipos de ferramentas, ao invés de pensar diretamente nos softwares, porque, sendo assim, esse projeto pode ser aproveitado por outras faculdades.

Das ferramentas levantadas, algumas já foram inicialmente testadas nas aulas de *Edição de Partituras e Tecnologias Sonoras* e no curso de *Música e Tecnologia*. Segue os tipos de ferramentas levantadas:

1. Acompanhamento: auxílio em disciplinas de Improvisação;
2. Análise de Áudio: suporte em edição e gravação de áudio e estudos sobre áudio;
3. Edição e Gravação de Áudio: atividades de composição;
4. Edição de Partitura: para disciplinas de Arranjos e Transcrições, Composição e demais disciplinas que faz o uso de partituras;
5. Educação Musical: disciplinas que estudam a teoria musical, como a disciplina de Harmonia;

<sup>1</sup><http://www.ufsj.edu.br/vivenciasmusicais/>

6. Recursos Multimídia: nas disciplinas do curso de música é comum o uso desses recursos, como players de vídeo e áudio;
7. Sequenciamento: disciplina de composição;
8. Servidores de Som: para gerenciar o uso dos dispositivos de áudio;
9. Treinamento de Percepção Musical: aulas e treinamento de percepção.

Todos os tipos de ferramentas citadas anteriormente foram utilizadas na disciplina optativa *Edição de Partituras e Tecnologias Sonoras*. Essa disciplina introduziu o estudo e a prática dos principais recursos tecnológicos utilizado no curso de música, o que contribuiu na realização do levantamento das ferramentas.

Além dos softwares relacionados a música, é importante ter programas para edição de texto, planilha e slides (os quais entram na categoria “Escritórios”). São ferramentas muito utilizadas por alunos e professores em atividades para a faculdade, como o desenvolvimento de monografia e artigos.

O Sistema Operacional escolhido para trabalhar com as ferramentas é o *Linux Mint*, que também é uma ferramenta FLOSS. O sistema é leve, elegante e apresenta uma organização parecida com a do *Windows*, o que não impacta negativamente para os usuários que não estão acostumado com o *Linux*. O *Linux Mint* é fácil de usar, possui a versão e suporte em português do Brasil e atende a necessidade de drivers para o hardware e periférico existente no laboratório de computação do DMUSI.

Tabela 1. Aplicativos associados à categoria

<b>Categorias</b>	<b>Ferramentas FLOSS</b>	<b>Sistema operacional</b>
Acompanhamento	LinuxBand <sup>2</sup>	Linux
Análise de Áudio	Sonic Visualiser <sup>3</sup>	Linux, Windows, MacOS
Escritório	Evince <sup>4</sup>	Linux, Windows
	LibreOffice <sup>5</sup>	Linux, Windows, MacOS
	PDF-Shuffler <sup>6</sup>	Linux
	Texmaker <sup>7</sup>	Linux, Windows, MacOS
Áudio	Ardour <sup>8</sup>	Linux, Windows, MacOS
	Audacity <sup>9</sup>	Linux, Windows, MacOS
Edição de Partitura	MuseScore <sup>10</sup>	Linux, Windows, MacOS
Educação Musical / Percepção	GNU Solfège <sup>11</sup>	Linux, Windows, MacOS
Sequenciamento	LMMS <sup>12</sup>	Linux, Windows, MacOS
Multimídia	Audacious <sup>13</sup>	Linux, Windows
	Brasero <sup>14</sup>	Linux
	VLC Media Player <sup>15</sup>	Linux, Windows, MacOS
Servidores de Som	JACK Audio Connection Kit <sup>16</sup>	Linux, Windows, MacOS
	ALSA <sup>17</sup>	Linux
	PulseAudio <sup>18</sup>	Linux, Windows, MacOS

#### 4. Resultados obtidos

Como resultado parcial desta pesquisa, chegamos a uma lista de aplicativos FLOSS que poderão ser utilizados nos cursos do departamento de música da UFSJ. Na Tabela 1 os aplicativos estão associados às categorias levantadas.

Alguns dos softwares listados já foram utilizados e bem avaliados por alunos e/ou professores do DMUSI, os mesmo estão listados a seguir:

- Audacity;
- GNU Solfege;
- JACK Audio Connection Kit
- LMMS
- MuseScore;
- Sonic Visualiser;
- VLC Media Player.

Os aplicativos candidatos a suprir as demais categorias foram escolhidos com base na utilização dos mesmos no curso de Ciência da Computação. Os servidores de áudio, recursos multimídias e editores de textos são populares no DCOMP e isso faz com que as avaliações dos softwares tendem a serem positivas.

As ferramentas escolhidas são sugestões e poderão ser substituídas por outras após as avaliações que serão realizadas.

A opção pela utilização de FLOSS por estas Universidades vai além da questão de seu valor e incluem a maior interoperabilidade entre as aplicações, maior capacidade de crescimento e adaptação das ferramentas e a possibilidade de interação com os desenvolvedores dos aplicativos, que permite uma colaboração direta dos usuários com a manutenção das ferramentas.

Não foi identificado a falta de alguma ferramenta para atender o DMUSI, mas esse ponto continuará sendo observado na continuação desse projeto.

#### 5. Conclusão

Este trabalho trouxe a proposta de utilização de ferramentas FLOSS para atender as necessidades do Departamento de Música da UFSJ. As ferramentas serão utilizadas no sistema

<sup>2</sup>Disponível em <http://linuxband.org/>

<sup>3</sup>Disponível em <http://www.sonicvisualiser.org/>

<sup>4</sup>Disponível em <https://wiki.gnome.org/Apps/Evince>

<sup>5</sup>Disponível em <https://pt-br.libreoffice.org/>

<sup>6</sup>Disponível em <http://sourceforge.net/projects/pdfshuffler/>

<sup>7</sup>Disponível em <http://www.xmlmath.net/texmaker/>

<sup>8</sup>Disponível em <http://ardour.org/>

<sup>9</sup>Disponível em <http://audacityteam.org/?lang=pt-BR>

<sup>10</sup>Disponível em <https://musescore.org/pt-br>

<sup>11</sup>Disponível em <https://www.gnu.org/software/solfege/>

<sup>12</sup>Disponível em <https://lmms.io/>

<sup>13</sup>Disponível em <http://audacious-media-player.org/>

<sup>14</sup>Disponível em <http://wiki.ubuntu-br.org/Brasero>

<sup>15</sup>Disponível em [www.videolan.org/vlc/](http://www.videolan.org/vlc/)

<sup>16</sup>Disponível em <http://www.jackaudio.org/>

<sup>17</sup>Disponível em <http://www.alsa-project.org/>

<sup>18</sup>Disponível em <http://www.freedesktop.org/wiki/Software/PulseAudio/>

operacional Linux Mint, mas é importante destacar que o termo FLOSS é para qualquer sistema. Os resultados apresentados são parciais e após o término dessa pesquisa é esperado uma lista de ferramentas que poderão ser utilizados nos cursos do DMUSI. O levantamento poderá também apresentar uma lista de aplicativos que poderiam ser utilizados nos cursos de música, mas que ainda não existem como software livre.

Alguns professores já sabiam dos valores das ferramentas FLOSS, mas os alunos não tinham esse conhecimento. O interesse original dos alunos pelas ferramentas FLOSS se deu apenas pela gratuidade das ferramentas, porém, com a introdução e explicação dos conceitos de FLOSS, eles começaram a entender os benefícios de ter acesso ao código fonte.

Por meio deste projeto, é possível que o DMUSI amplie a utilização das ferramentas FLOSS em suas atividades e que isto venha a fortalecer ainda mais a parceria entre este departamento e o DCOMP. O fato do código ser aberto permite que o FLOSS atenda qualquer necessidade, desde que as mesmas sejam identificadas e implementadas. Com os dois departamentos trabalhando juntos será possível identificar as melhorias que precisam ser feitas nos softwares existentes e identificar o que ainda precisa ser desenvolvido para que as necessidades do DMUSI sejam atendidas apenas com ferramentas FLOSS. Ambos departamentos deverão sair desta parceria mais fortalecidos, o DMUSI por ter acesso a ferramentas para seus alunos e o DCOMP com projetos de desenvolvimento de software para a área de Computação e música.

## Referências

- Amorim, J. N. d. (2014). O ensino do contrabaixo elétrico e as novas ferramentas tecnológicas: um estudo de caso na escola de música de Brasília.
- Andersen, T. H. (2003). Mixxxx: Towards novel dj interfaces. In *Proceedings of the 2003 conference on New interfaces for musical expression*, pages 30–35. National University of Singapore.
- Bullock, J. and Coccioli, L. (2006). Modernising musical works involving yamaha dx-based synthesis: a case study. *Organised sound*, 11(03):221–227.
- Campos, A. (2006). O que é software livre. *BR-Linux. Florianópolis, março de*.
- Cannam, C., Landone, C., Sandler, M. B., and Bello, J. P. (2006). The sonic visualiser: A visualisation platform for semantic descriptors from musical signals. In *ISMIR*, pages 324–327.
- Davis, P. et al. (2012). Ardour digital audio workstation.
- Furse, R. (2000). Linux audio developer's simple plugin api (ladspa).
- Lazzarini, V. and Walsh, R. (2007). Developing ladspa plugins with csound. In *Proceedings of Linux Audio Conference*, pages 60–63. Citeseer.
- Letz, S., Foher, D., Orlarey, Y., and Davis, P. (2004). Jack audio server: MacOS x port and multi-processor version. *Sound and Music Computing '04*, pages 20–22.
- Pinhati, F. and Siqueira, S. (2012). Utilização de ferramentas colaborativas na aprendizagem musical de alunos leigos. In *Anais do Workshop de Informática na Escola*, volume 1.

- Puckette, M. et al. (1996). Pure data: another integrated computer music environment. *Proceedings of the Second Intercollege Computer Music Concerts*, pages 37–41.
- Schiavoni, F. L., Queiroz, M., and Iazzetta, F. (2011). Medusa-a distributed sound environment. In *Proceedings of the Linux Audio Conference*, pages 149–156.
- Wheeler, D. A. (2005). Why open source software/free software (oss/fs, floss, or foss)? look at the numbers!
- Wilson, S., Cottle, D., and Collins, N. (2011). *The SuperCollider Book*. The MIT Press.

# A Study Design About the Influence of Spectrum Content in the Perception of Consonant and Dissonant Chords

Guilherme Feulo do Espírito Santo<sup>1</sup>, Micael Antunes da Silva<sup>2</sup>,  
Antonio José Homsí Goulart<sup>1 \*</sup>

<sup>1</sup>Grupo de Computação Musical– Instituto de Matemática e Estatística da USP  
Rua do Matão, 1010, Bl. C – 05508-090 São Paulo, SP

<sup>2</sup>FAAM - Faculdade de Artes Alcântara Machado  
Rua Taguá, 150 - Prédio 1 – 01508-010 São Paulo, SP

feulo@ime.usp.br

**Abstract.** *This paper tackles the basis for the musical consonance analysis and its relationship with the spectral content of chords present in the tonal system and, based on this, propose an experiment that explores through electroencephalography a relation between the harmonic nature of musical instruments and the brain response to the auditory stimulus.*

## 1. Introduction

Harmony is the basis of western music history. Consonance X Dissonance is the most important concept of harmony and it is perceived in the structure of tonal music or even in the emancipation of dissonance produced by the 20th century music.

In this study, we propose an experiment based on the idea of psychoacoustic dissonance [Helmholtz, 1954] [Plomp and Levelt, 1965], analysing the degree of dissonance in the spectra synthesized with different waveforms and implement an experiment similar to Pierce's [Pierce, 1966] on the construction of the octatonic spectrum.

During the auditory stimulus, it will be measured the EEG activity of the participants to search for correlations between the consonant sounds and the brain activity as showed in Park et al. [Park et al., 2011] and Maslennikova et al. [Maslennikova et al., 2013] looking for influence of the number of partials in the consonance perception.

## 2. Brief theory review

### 2.1. Sensory or psychoacoustic dissonance

Dissonance is a multidimensional attribute of sound and it can be approached in many ways, for instance by considering its cultural aspects, volunteers' level of musical practice or even the physical properties of the sound. Tenney [Tenney, 1988] divide these approaches to the study of dissonance in five different categories: melodic, polyphonic, functional, counterpointistic and psychoacoustic. In this paper we will see the idea of psychoacoustic dissonance, which then “reduces itself to one scientific, psychophysical aspect, disconnected to cultural and aesthetic factors, for example, which matches to complementary dimensions” [Porres, 2012].

---

\*The authors would like to thank CAPES and the Sonology Research Center at USP.

With the work of Plomp and Levelt [Plomp and Levelt, 1965] we have a review of the Helmholtz theory [Helmholtz, 1954], starting with the concept of critical bandwidth, defined as the smallest band of frequencies that activates the same region of the basilar membrane. With information obtained via an experiment with volunteers without musical training, they checked the degree of consonance between two sinusoidal sounds and realized that the roughness sensation appears only to intervals that are inside the same critical bandwidth, with the highest roughness sensation happening inside a quarter of the critical bandwidths [Sethares, 2005].

Considering that, starting from the decomposition of a spectrum in sinusoidal waves, it is possible to create the dissonance curves of complex tones, adding the dissonance relations between their partials. Figure 1 shows the consonance curve of a complex tone with six harmonic partials, demonstrating the highest consonance on intervals with simple ratios, confirming the practice of the traditional study of western harmony.

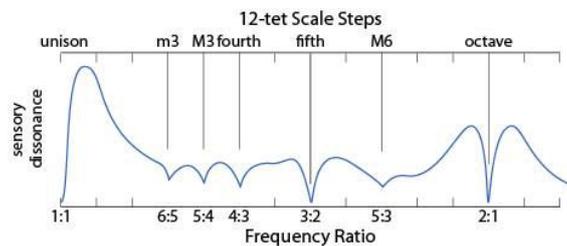


Figure 1: 6-partial sinusoidal dissonance curve

Based on Plomp and Levelt work [Plomp and Levelt, 1965], Pierce [Pierce, 1966] speculated the possibility of synthesis of inharmonic spectrum to obtain consonance on arbitrary scales. Starting with the octatonic scale (a octave divided in eight parts) we have  $r = \sqrt[8]{2} = 1.09051$ , and according to Sethares [Sethares, 2005], the octatonic spectrum partials are obtained by multiplying a base frequency by  $1, r^{10}, r^{16}, r^{20}, r^{22}$  and  $r^{24}$ .

Symmetrically dividing an octave in 4 parts, Sethares makes a comparison between the 12-tone scale and 8-tone scale based on the dissonance curve generated by the octatonic spectrum. We can see in Fig. 2 that there is a coincidence of maximum consonance between the 3th, 6th, 9th and 12th divisions of the 12-tone scale and 2th, 4th, 6th and 8th divisions of the 8-tone scale. So, using this spectrum, the diminished triads sound consonant and the perfect triads sound dissonant.

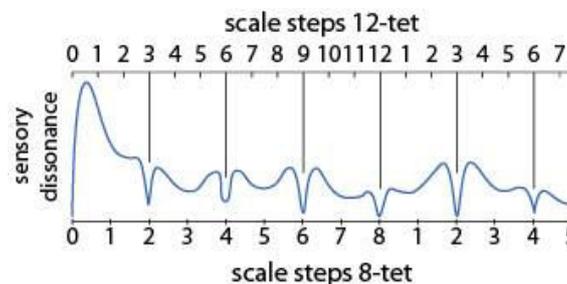


Figure 2: Octatonic spectrum dissonance curve

## 2.2. Partial and sound synthesis

We were curious about the influence of a different number of partials in the sounds used for the experiments. So we decided to run the same test using different timbres each round. The waveforms used were the classic sinusoidal, triangular, square and sawtooth waves. These waveforms were extensively used in the analog subtractive synthesis context [Dodge and Jerse, 1997]. The waveforms harmonic contents are summarized in Table 1.

Waveform	Partials present	Amplitude decay
Sinusoid	only the fundamental	–
Triangular	fundamental + odd partials	$1/p^2$
Square	fundamental + odd partials	$1/p$
Sawtooth	fundamental + all partials	$1/p$

**Table 1: Harmonic content of classic waveforms.  $p$  is the partial index**

Although just intonation is the more appropriate form to eliminate the roughness on perfect chords and thus achieve the highest level of consonance, in our experiment we apply the tempered system. Some discrepancies between the classical waveforms spectra and the equal temperament do not affect the perception of consonance on major and minor triads. We can see that on dissonances curves on [Porres et al., 2006].

In order to experiment with the octal spectrum, the additive synthesis technique was used to prepare the synthesis model. In this paradigm, each partial of the desired signal is created as an instance of a sinusoidal oscillator with its own amplitude, frequency and phase parameters [Dodge and Jerse, 1997]. Also, a piano-like sound, the default SuperCollider<sup>1</sup> instrument will be used in the tests.

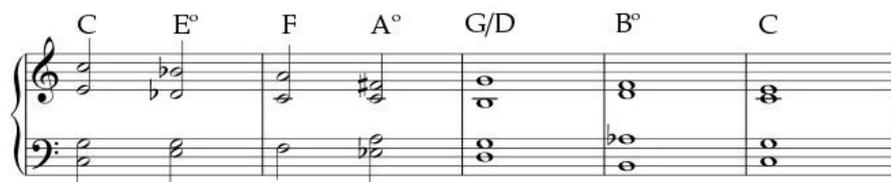
### 3. Experiment details

The proposed experiment consists of recording the EEG activity during the auditory stimulus with specific musical excerpts, which will be synthesized with different waveforms containing different spectral content, so we can check the influence of the partials on the consonance. We will test the classic waveforms of table 1, another waveform with the octal spectrum and a piano timbre.

The participants will be divided in 4 groups: popular music inclined musicians, academic trained musicians, music lovers lacking musical education, and people who do not care about music. In such a way we can investigate the influence of musical knowledge in the results, as pointed by [Park et al., 2011] [Maslennikova et al., 2013].

#### 3.1. Musical excerpts

For the construction of musical excerpts we relied on octatonic spectra cited on this paper. Therefore, the chords sequence have only perfect and diminished triads. In Figure 3 we can see the sequence I, IV, V, I and between them their respective diminished dominants, creating a tension and resolution movement.



**Figure 3: The musical excerpt used in the experiments**

We prepared our musical excerpts with SuperCollider, although any language could be used. The SuperCollider code and audio excerpts can be downloaded<sup>2</sup>, and we encourage their use and modification.

<sup>1</sup><http://supercollider.sourceforge.net/>

<sup>2</sup><http://www.ime.usp.br/~ag/dl/audio-exp.zip>

### 3.2. Analysis parameters and expected results

From the recorded EEG data we will look at both the different frequency bands RMS (Root Mean Squared) power between the chord changes and the ERPs related with the chord changes. This results will be grouped by the spectral content and the subjects musical knowledge. We are expecting three kinds of results:

- confirmation that Gamma activity will increase during the consonant chords and decrease during the dissonant chords as shown in [Park et al., 2011];
- checking if as the spectral content enriches the levels of the Gamma Band record higher values in the consonant parts and lower values in the dissonant parts, in comparison with the tests with less partials;
- check if there is a inversely proportional relation between the number of harmonic partials and the levels registered in the Gamma band for the dissonant parts;

### 4. Conclusion and ongoing work

Based on what was exposed in this paper we believe to have enough background to proceed with the proposed experiment and achieve significant results.

In order to help us foreseeing possible problems that can cost us time on the session, and also to help improving or tuning our methodology we want to share our ideas with the academic community, as we will have only limited access to a EEG system in a laboratory at our university, and we depend on a lot people that we will ask to contribute to this work as volunteers.

We hope the results obtained with this experiment can be used as basis for other experiments in the future for a deeper investigation of sound cognition and for the development of new EEG based Brain-Computer Interfaces

### References

- Dodge, C. and Jerse, T. (1997). *Computer Music: Synthesis, composition and performance*. Schirmer Books, New York, NY, USA, segunda edition.
- Helmholtz, H. L. F. (1954). *On the sensation of tone as a psychological basis for the theory of harmony*. Dover publications.
- Maslennikova, A. V., Varlamov, A. A., and Strelets, V. B. (2013). Evoked changes in eeg band power on perception of consonant and dissonant chords. *Neuroscience and Behavioral Physiology*, 43(6):670–673.
- Park, J. Y., Park, H., Kim, J., and Park, H. (2011). Consonant chords stimulate higher eeg gamma activity than dissonant chords. *Neuroscience Letters*, 488:101–105.
- Pierce, J. R. (1966). Attaining consonance in arbitrary scales. *Journal of acoustical society*.
- Plomp, R. and Levelt, W. J. M. (1965). Tonal consonance and critical bandwidth. *Journal of the acoustical society of america*, 38:548–568.
- Porres, A., Furlanete, F., and Manzolli, J. (2006). Análise de dissonância sensorial de espectros sonoros. In *Anais da ANPPON*, Brasília.
- Porres, A. T. (2012). *Modelos psicoacústicos de dissonância para eletrônica ao vivo*. PhD thesis, Universidade de São Paulo - Escola de Comunicação e Artes.
- Sethares, W. A. (2005). *Tuning, timbre, spectrum, scale*. Springer-Verlag, 2 edition.
- Tenney, J. (1988). *A history of consonance and dissonance*. Routledge, 4 edition.

# A Low Cost Computing Interface to Speed Up Braille Music Notation

Antonio Fernando da Cunha Penteadó<sup>1</sup>, Vilson Zattera<sup>2</sup>, José Fornari<sup>3</sup>

<sup>1</sup>Music Department - Arts Institute

<sup>2</sup>Accessibility Laboratory - Central Library

<sup>3</sup>Núcleo Interdisciplinar de Comunicação Sonora (NICS) - COCEN

Universidade Estadual de Campinas (Unicamp) - Campinas, SP - Brazil

nando@nandopenteadó.com, vilson.zattera@gmail.com,  
tutifornari@gmail.com

***Abstract** – Braille music notation is used by visually impaired musicians to engrave music. As normally performed, this is a difficult and time consuming task. This paper presents a study, a work in progress, on the design of a simple and affordable hardware interface that turns the process of braille music notation in computers easier and faster. Braille uses the same set of 6 dots to make all representations of letters, numbers, symbols and music notation according a context. The hardware presented here has two handles with 3 pushbuttons, each one related to one braille dot. An Arduino board reads the state of these 6 pushbuttons, translate them into a bitmask and send it to a computer. At the computer side there is a driver to make it possible to convert pushbuttons actions into image and sound.*

## 1. Introduction

Music notation using computers is both a necessity and a problem for the visually impaired or low vision musicians. Benefits conveyed by this method of music notation are many, such as “corrections without blot, printing with high quality and speed, instant separation of parts in a grand staff, transposition, replication and distribution, automatic digitalization without the use of a scanner, sound generation, etc.” [Penteadó and Fornari], but the drawback is that mainstream softwares for music notation make extensive use of GUI (Graphical User Interface) which depends on the computer mouse. It is self-evident that visually impaired (blind) people can't operate a mouse in a graphic-oriented system. This is the case of proprietary softwares, such as: Sibelius (<http://www.sibelius.com>), Finale (<http://www.finalemusic.com/>), Encore (<http://www.passportmusic.com/>), as well as free softwares, such as: Musescore (<https://musescore.org/>), Rosegarden (<http://www.rosegardenmusic.com/>), Tux Guitar (<http://www.tuxguitar.com.ar>), among others.

Blind musicians may use auditory information to replace the visual one, and it is exactly what a class of softwares called “screen readers” does. Screen readers convert text on the screen to audio (language spoken by a voice synthesizer), but it is impossible

to convert images into audio, thus they can't help with graphical interfaces. The most famous screen readers softwares are the proprietary one JAWS (<http://www.freedomscientific.com/>) and the open source NVDA (<http://www.nvaccess.org/>), both run only in Windows Operating System. For Linux, there is ORCA (<https://wiki.gnome.org/Projects/Orca>).

Another way to replace visual information is through the use of tactile computer interfaces, and for this, there is a class of devices called “braille displays” that can convert text displayed on the screen into braille, by an array of braille cells, each one formed by a combination of six dots (sometimes 8 dots) that can toggle in low or high relief to provide a tactile surface for finger reading. However this solution, like screen readers, is also inefficient to display images from a graphical software, and this interface is very expensive, ranging from US\$3,500 to US\$15,000 according to American Foundation for the Blind (<http://www.afb.org>).

It's remarkable that, in spite of the problems above mentioned, there is yet another issue regarding the way of inputting musical data in a computer. Music is notated mainly using symbols and drawings and the principal device to input data in a computer is the keyboard, which was developed for inputting text and only text, not graphics, images or drawings.

An alternative to overcome these problems is to modify the traditional music notation paradigm and, rather than representing music with drawings, use text to do this task. There is a growing community of developers related to free software that are investing a great amount of effort in this concept of using plain text for music notation, being Lilypond (<http://www.lilypond.org/>) one of the major projects. Lilypond uses a syntax similar to Latex (<http://www.latex-project.org/>) but to represent music.

This paper shows a research on the use of a new interface to make it easier to input braille music notation into a computer. This interface was created by the main author of this work and it is applied here to write music as text by a visually impaired person in a faster and easier way, when compared to traditional music notation softwares that are controlled through the mouse and the keyboard.

This is a work in progress, a prototype subject to further developments and implementations that are discussed in this theoretical paper.

## **2. The interface: Braille handles**

This experience used two handles made of PVC pipes (40mm in diameter and 120mm in length) with three pushbuttons in each one.

Pushbuttons were connected independently to a breadboard using network 4-pair wires, each one with one terminal grounded and another connected through a 10k pull-up resistor to +5v and also directly to a digital input of an Arduino board, using then six digital inputs. No extra electronic components were used, only 6 groups of pushbuttons with the same schematic showed in the figure (Fig. 2), schematics for the Arduino board can be found in the link <https://goo.gl/2X0FD8>.

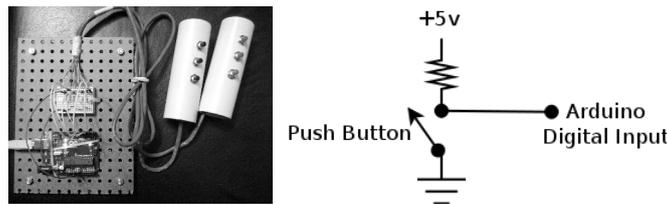


Fig. 2. The “Braille handle” device and pushbutton connection.

Arduino detects the pushbuttons states and convert them in to a 6 bits number, in fact a so called “bitmask” where each bit represent the state of a certain pushbutton; this number ranges from 0 up to 63 and is sent to the serial port of the Arduino board. The computer reads the detected bitmask through a USB port and processes it using an algorithm written as PHP script that takes care of making the conversion between that bitmask to the alphanumeric characters and/or some symbols.

Later on, characters can be interpreted following different grammars for different tasks. For instance, they can be interpreted as regular braille characters, as dedicated structures to accomplish rules of braille music notation or even as characters that can be treated with a specific grammar, which was actually the present case. It was used here a grammar of a language called Mustaq ([www.mustaq.org](http://www.mustaq.org)). This language is dedicated to represent music as text, taking advantage of stenographic notation.

### 3. Discussion

It was possible to input text and music with the proposed solution but many problems were faced and thus some future developments, modifications, tweaks and tests are still needed, as much as further suggested improvements discussed here. Bellow are an itemized discussion on each separated items to make it easier to individually emphasize some relevant points of this prototype.

Handle diameter: it was used a PVC pipe measuring 40mm in diameter; increasing and decreasing this number may lead to a more comfortable device.

Handle material: for the sake of affordability and simplicity, not mentioning resistance and durability at a very light weight, PVC was the option of choice. It is easy to mould, perforate and cut, besides being also washable.

Handle shape: PVC pipes are cylindrical (with parallel walls), maybe it would be useful to try other more ergonomic shapes, something like a pistol grip.

Pushbutton type: the pushbuttons used in this implementation were a bit little, causing the need of more finger pressure to operate them then if they were larger.

Input alternatives: there are still many other alternatives to get gestures from the musician as an input. It could be used a hall effect sensor as related by [Manzolli 1993] in his doctoral thesis. It is also possible to use a webcam to capture musician gestural like the one related by [Fornari 2012]. But pushbuttons give more tactile and pleasant feedback. Voice commands could also be further employed.

Number of pushbuttons: maybe another good improvement to this proposed hardware could be to enlarge the amount of pushbuttons, maybe one or two more for

each thumb or little finger, which could be used as a shift key like SHIFT, ALT, ALTGR and CONTROL keys commonly present in regular keyboards.

Feedback: two free and open source softwares were used to bring audible feedback. The first one was Audacity (<http://audacityteam.org/>) for recording and editing a wave file containing speech of musical terms. Then Sox ([sox.sourceforge.net](http://sox.sourceforge.net)), to play those pieces of recorded audio, Sox can produce musical notes in many different timbres like sine wave, plucked instruments, etc, providing sound as tones and speech.

#### 4. Conclusions

In this paper we presented a research on the use of a physical interface to input texts into computers. In this approach the focus was on speeding up music notation using a direct way to write 6-dot braille cells.

One of the main advantages of this proposed hardware is the fact that it doesn't interfere with screen readers, once it is a dedicated hardware. All inputs are beforehand filtered with a dedicated software that can interpret incoming information in different ways and translate them before the screen reader takes control. Braille cells have different meanings according to the context where they are used. A same cell can be a letter "D" or a number "4" or maybe a musical note (C as a quarter note), just to mention some ambiguity, but there many more specially for representing music.

The use of two separated handles provided freedom for hands, arms and shoulders. The musician has all the keys under the fingers, making writing easier.

Some softwares such as *Braille Fácil* (<http://intervox.nce.ufjf.br/brfacil/>) or *Delius* (<http://sourceforge.net/projects/delius/>) that use a computer keyboard to input braille texts reported hardware conflicts or no operation depending on the brand of the keyboard. These softwares try to use a regular keyboard to represent each dot of a braille cell with a key, thus sometimes keys must be pressed together and they are not recognized correctly. The hardware proposed here has zero conflicts once Arduino runs smoothly on Windows, Mac and Linux operating systems.

Whilst in this experiment the syntax was very specific for music, this interface can be used to directly introduce braille text or braille music in a very cheap way.

Further and detailed informations on the building of this hardware can be found at <http://www.nandopenteado.com/posts/braille-handles/>

#### References

- Penteado, A. F. C. and Fornari, J.. Stenographic Patterns for Speed Music Notation. XI SIMCAM. Simpósio Internacional de Cognição e Artes Musicais. Pirenópolis, GO. 26 a 29 de maio de 2015.
- Manzolini, J. "Non-linear Dynamics and Fractal Geometry as a Model for Sound Synthesis and Real-time Composition". Tese de Doutorado, Music Department, University of Nottingham, 1993
- Fornari, José. "Designing Bodiless Musical Instruments". anais da AES Brasil 2012. 8 - 10 maio. Expo Center Norte. São Paulo. Brasil.

## Árvore das Memórias: Instalação Multimídia Interativa

Ayer Ribeiro Netto, Luan Casthologe, Amanda Oliosi, Ariane Mateus, Leandro Costalonga, Daniel Coura

Departamento de Computação e Eletrônica – Universidade Federal do Espírito Santo  
BR 101 Norte, Km 60, Bairro Litorâneo, CEP: 29932540 – São Mateus – ES - Brasil

ayerrsn2@gmail.com, luancpp@gmail.com, amandasoliosi@gmail.com, ariane.ccomp@gmail.com, leandro.costalonga@ufes.br, daniel.coura@ufes.br

**Abstract.** *Interactive Multimedia Installation is a rich and multidisciplinary field that aims to merge technical and artistic views of the world into a tangible artifact. This paper presents a multimedia installation developed as an educational project. The artistic theme takes into account the passive role of a tree in social interactions despite the fact that it is often present on the spot where it takes place, especially in tropical countries such as Brazil. Therefore, it makes sense to provide the tree with a computational system that allows it to (voice) interact with the surrounding audience. On the technical side, Android and Arduino devices were used. A detailed technical specification and the results are presented.*

**Resumo.** Instalações Multimídias Interativas é um rico campo multidisciplinar que coloca ao serviço da arte tecnologias e conhecimentos técnico-científicos oriundos das engenharias e da computação. Esse artigo apresenta uma instalação multimídia interativa desenvolvida como projeto de ensino. A temática artística da instalação, denominada Árvore de Memórias, circundava o fato da árvore ser espectadora passiva de momentos sociais mas, uma vez dotada de recursos eletrônicos, poderia se tornar elemento ativo nessa interação – um repositório de memórias dos eventos que a circundam. Os aspectos técnicos da instalação envolvem desenvolvimento Android, manipulação e chaveamento de sinal de áudio por Arduino. A descrição técnica da instalação é detalhada e seus resultados são apresentados.

### 1. Introdução

O emprego dos recursos multimídia como impulsionador no processo de ensino/aprendizagem é senso comum entre educadores e altamente discutido na literatura (Goodwin, K., 2008; Iskander et. al 1995). Da mesma forma, é fato notório que a oportunidade da aplicação do conhecimento em projetos práticos, como por exemplo, em uma metodologia construtivista, solidificam o processo de aprendizagem. Um exemplo disso é o uso da robótica em projeto de ensino como descrita por Alimisis et. al (2007). Um outro exemplo de aplicação direta dos conhecimentos acadêmicos em projetos prá-

ticos foi descrita por Kim et. al (2007), porém esse último adota uma abordagem interdisciplinar envolvendo arte e tecnologia. Esse será o foco desse artigo.

Instalações Multimídias Interativas é um rico campo multidisciplinar que coloca ao serviço da arte tecnologias e conhecimentos técnico-científicos oriundos das engenharias e da computação. Tais instalações (peças/obras de arte) se caracterizam por sua forma dinâmica que reagem e se transformam mediante a interação com o público; Esse, por sua vez, também reage a mudanças da obra, que volta a reagir em um ciclo permanente (circular causal) de mudanças. Tal conceito, oriundo da cibernética, tem estado presente desde a década de 1930 em um movimento artístico conhecido com *Machine Art* (Penny, 2013).

As Instalações Multimídia Interativas são artefatos oriundos do choque entre visões de mundo muitos diferentes. As Engenharias normalmente tem seu foco na funcionalidade, otimização, e desempenho; Já as Artes, podem extrapolar sua visão para a estética, dialogo, e bem estar. O diálogo das artes com as engenharias propiciam um relacionamento simbiótico com influencia mútua. Tanto engenheiros se permitem abordar soluções mais criativas e menos ortodoxas, como artistas tendem a ser menos prolixos na concepção e descrição de suas obras. Claramente, as Artes podem se favorecer incorporando elementos tecnológicos nas suas obras de forma a possibilitar a exploração do comportamento interativo da audiência; Da mesma forma, as engenharias e computação se beneficiam ao incorporar a forma de dialogo típico das artes nos dispositivos que interagem com humanos. Claramente, há um viés educacional forte nessa interação.

A incorporação das novas tecnologias nas artes propicia a criação de obras multimídias que vão além do aspecto visual, sonoro, e tátil: as novas tecnologias trazem vida às obras e ao seu entorno, favorecendo a interação social local da comunidade acadêmica. Do ponto de vista social, as instalações são especialmente importantes nos campi do interior do Brasil, onde atividades sociais e culturais podem ser escassas. Do ponto de vista educacional, as instalações multimídia interativas proporcionam a aplicação direta do conhecimento na construção de algo tangível e que ficará exposto no campus da universidade.

Esse artigo descreve uma instalação desenvolvida como um projeto de ensino, intitulada *Árvore de Memórias* como parte de um programa de extensão em artes digitais (Costalonga, 2013). Os detalhes do projeto são apresentados na seção 2. Na seção subsequentes, abordam-se a exibição e principais resultados observados. Na seção 4 tratamos na versão 2015 da instalação descrevendo os principais elementos agregados ao projeto, seguida de uma breve conclusão.

## 2. *Árvore de Memórias (MemoryTree)*

Há um corrente debate sobre os impactos sociais das tecnologias modernas, em especial das redes sociais (Bauman, 2013). Enquanto alguns defendem que essas tecnologias aproximam pessoas que estão longe, outros criticam o fato de limitarem as comunicações presenciais possivelmente afastando aqueles que estão pertos em detrimento da praticidade e conveniência do uso dessas tecnologias. As instalações interativas do pro-

jeto Campus Vivo (Costalonga, 2013) são pensadas de modo a favorecerem a interação social local e maximizar a experiência universitária e cultural, em especial, nos campi do interior do Brasil.

A Árvore de Memórias foi a primeira instalação do projeto Campus Vivo, e o comprometimento observado nos alunos atuantes no projeto foi notável. A sensação de apropriação da comunidade a obra exposta motivou os membros discentes a trabalharem no projeto de forma raramente vista até então em nosso centro acadêmico. Esse fato suscitou a importância pedagógica de adotar tais práticas educacionais com maior frequência.

O projeto consistia em dotar uma árvore no Campus da UFES de São Mateus, ponto de encontro de muitos alunos devido sua vasta sombra, com um sistema que permitisse os alunos a interagir com a árvore. A temática artística circundava o fato da árvore ser espectadora passiva de momentos sociais mas, uma vez dotava de tal sistema, ser tornaria elemento ativo nessa interação – um repositório de memórias dos eventos que a circundam. Considerando a época do ano, optou-se por usar uma decoração natalina para disfarçar o equipamento desenvolvido.

Na parte técnica, o projeto consistia no desenvolvimento de um sistema que interagisse visualmente através da iluminação (mangueiras de LED) e auditivamente através micro-caixas de som instaladas dentro de bolas de natal. As bolas natalinas eram afixadas na extremidade da mangueira de LED de modo que, quando uma mensagem sonora soasse, a mangueira LED se ascenderia. As mensagens sonoras eram provenientes de mensagens de voz enviadas pelo aplicativo de mensagens *WhatsApp*. Ainda sobre a parte sonora, vale ressaltar que era uma preocupação do grupo que a instalação não perturbasse o ambiente com sons altos, logo, os mesmos eram mantidos em volume suficiente para a audição daqueles que encostavam o ouvido nas bolas natalinas.

Inicialmente a instalação projetada para possuir 8 saídas (LED + Áudio), o projeto precisou ser simplificado para 4 saídas por questões de custo. O orçamento da instalação era de aproximadamente um salário mínimo (R\$ 724,00). Devido a essa simplificação, somente uma mensagem por vez era tocada. A Figura 1 ilustra a esquemática do sistema proposto.

As instalações do projeto Campus Vivo privilegiam o uso de sucata, favorecendo a reciclagem de componentes eletrônicos. Nesse sentido, utilizou-se no projeto: a) um antigo smartphone Android Motorola Defy; b) uma fonte de computador de 300W; c) Coolers/resfriadores extraídos de fontes de computadores sucateadas; d) um arduino UNO; e) Fios de cabos de Ethernet sucateados; f) Amplificador de áudio de 5 W extraído de caixas de som de computadores pessoais; g) mini-caixa de som de 5W-8Ohms. Foi preciso adquirir: a) 30 metros de mangueira LED; b) Caixa elétrica para ambientes externos; c) Enfeites natalinos diversos. Foi construído somente as placas de circuito impresso (uma por saída) para realizar o chaveamento de áudio e de energia (mangueira de LED);

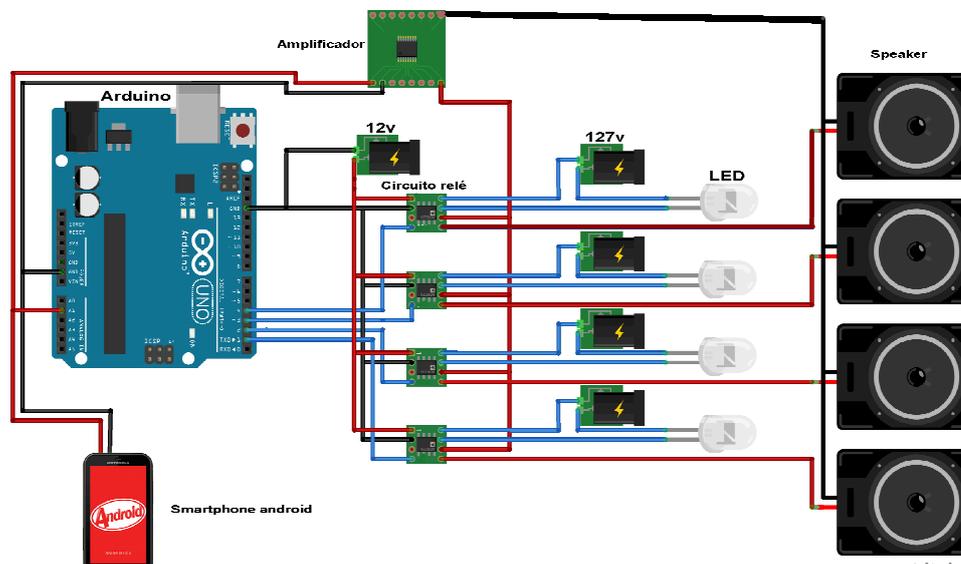


Figura1 – Arquitetura interna da instalação.

Sob o aspecto da inteligência computacional, desenvolveu-se dois programas independentes. O primeiro rodava no smartphone Android e funcionava como um tocador customizado das mensagens de áudio. Esse aplicativo buscava na pasta de gravação de arquivos de áudio do *Whatsapp* (configurado para baixar automaticamente arquivos de áudio) as mensagens que deveriam tocar, gerenciando uma espécie de fila circular dessas mensagens. A fila era criada de modo randômico, mas atribuía um peso proporcional ao tempo cronológico do envio das mensagens – quanto mais nova, com maior frequência a mensagem era tocada. A equalização e normalização do áudio foram configuradas usando recursos do próprio dispositivo. Um modo de depuração também foi implementado garantindo que fosse possível reiniciar o sistema remotamente, remover mensagens inapropriadas e verificar se todas as saídas estavam funcionando adequadamente.

O sinal de áudio oriundo do smartphone é chaveado para uma das saídas possíveis. Esse chaveamento é feito por um programa embarcado no Arduino. Ele busca o momento de silêncio entre as mensagens no sinal de áudio e, quando o encontra, o programa troca a seleção das placas de relés indicando uma nova saída. Por questões estéticas, antes de enviar a mensagem para uma saída, um efeito visual que simulava o funcionamento de uma "roleta" era empregado. Esse efeito fazia com que os LEDs piscassem rapidamente de modo alternado, ficando mais lento a cada ciclo, até finalmente permanecer em uma saída que era usada para tocar a mensagem. Esse efeito maximizava o comportamento interativo com a audiência, que ia de uma mangueira LED para outra de modo a conseguir ouvir a mensagem.

### 3. Exibição Natalina 2014: MemoryTree

A exibição da instalação ocorreu duas semanas antes do natal, sendo desmontada no início no recesso letivo (24/12). O local escolhido foi uma árvore de grande porte próxima a secretaria estudantil e biblioteca, logo, um local já bastante frequentado. Postagem na rede social do grupo de pesquisa e um e-mail de divulgação foi enviado a comunidade acadêmica do centro universitário com o seguinte texto:

*“A presente instalação, denominada Árvore da Memória, remete a um observador imóvel que ouve fragmentos de conversas, confissões e declarações, seja de transeuntes do seu entorno, seja dos que gozam de sua sombra. Imponente, mas discreta, a árvore sempre vocalmente passiva, nesse Natal ganhou um presente! Hoje, ela pode contar tudo ouve... para cada um, uma mensagem de alguém que um dia conversou com ela. Interaja e deixe também sua mensagem.”*

No local, um cartaz (Figura 2) com instruções básicas foi afixado.



**Figura2: Guia de Interatividade com Audiência**

Após a montagem, ocorreram alguns dias de chuvas muito fortes. A caixa elétrica, mesmo tendo características para resistir as intempéries climáticas, não conseguiu vedar de modo satisfatório, assim, foi necessária a troca de alguns componentes. Notou-se também durante essa manutenção que alguns insetos adentraram na caixa elétrica pelos orifícios de resfriamento. Para resolver este problema instalou-se uma tela protetora. Nesse processo grande parte das mensagens recebidas se perdeu. E após estes reparos o dispositivo funcionou adequadamente e ininterruptamente pelos dias subsequentes.

Conforme já mencionado, priorizou-se o uso de materiais reciclados e isso teve um preço na qualidade do áudio. Foram utilizados partes de cabos Ethernet descartados para conectar o amplificador as bolas de natal com os mini-falantes. Notou-se que os cabos não eram apropriados a esse fim, surgindo problemas de interferência e ruídos. Outro problema notado foi a proximidade da fonte de alimentação com o amplificador, promovendo interferência e ruídos no sinal de áudio e superaquecimento do conjunto.

Em pouco mais de uma semana de uso interrupto, a árvore recebeu aproximadamente 120 mensagens, inclusive de países da Europa. Não foi necessário remover nenhuma mensagem de conteúdo inapropriado. Um vídeo explicando em detalhes o processo de construção e principais resultados, incluindo entrevistas com espectadores, está disponível em <https://youtu.be/4Bd2OXaKcjc>. Um outro vídeo mais sintetizado mostrando todo o processo de montagem e a reação da audiência está disponível em [https://youtu.be/B\\_uEOHLupy4](https://youtu.be/B_uEOHLupy4). A Figura 3 mostra fotos da instalação em exibição.



Figura 3: Fotos da Instalação em Exibição

#### 4. Instalação Natalina 2015: Memories Trees (Memories Forest)

Originalmente a instalação teria uma única exibição, no entanto a comunidade se mostrou receptiva e houve solicitações para uma nova instalação em 2015. Aproveitando essa comoção criou-se uma disciplina optativa aos cursos de Engenharia e Ciência da Computação para trabalhar os conceitos de programação sônica, eletrônica, sistemas embarcados e programação Android. A opção de criar uma disciplina específica permite abordar estratégias de ensino do conteúdo necessário à construção da instalação com o tempo adequado.

Os resultados iniciais foram apresentados ao grupo de pesquisa em Música Ubíqua (g-UbiMus) em reunião de pesquisa na Universidade de Maynooth (Irlanda) e novamente em junho no VI UbiMus, realizado em Vaxjo, Suécia. Grupos de pesquisa parceiros demonstraram interesse em replicar o projeto em suas respectivas instituições de ensino. Para tanto, acordou-se uma série de funcionalidades básicas e uma arquitetura de hardware comum de modo que todas as árvores pudessem se comunicar. Dentre as principais funcionalidades que estão em desenvolvimento para versão 2015 da instalação, estão:

**Embodiment:** todas as instalações devem ser capazes de emular uma outra instalação geograficamente remota. Por exemplo, a instalação localizada em São Mateus(ES) deve ser capaz de emular a instalação localizada em Maynooth na Irlanda, não somente recriando sua estética (controle de iluminação) como também seus algoritmos e mensagens. Isso implica no uso de fitas LED multicoloridas (RGB), a adoção do RaspberryPi rodando Pure Data e armazenamento remoto de mensagens (*cloudstorage*);

**Virtual MemoriesTree App:** Um aplicativo está previsto para dispositivos Android de modo que possa emular o funcionamento da instalação. Essa “árvore virtual” pode então ser replicada ou emular o comportamento de uma instalação física (monitoramento);

**Multimodal communications:** O aplicativo *WhatsApp* continua sendo o método preferencial de envio de mensagens uma vez que seu uso demonstrou ser muito efetivo, principalmente porque desobriga aos usuários a instalar aplicativos específicos para interagirem com a instalação. No entanto, não deseja-se estar limitado a mensagens de voz. Para 2016, tanto mensagens de voz como mensagens de texto serão suportadas através de algoritmo de síntese de voz - *TextTo Speech* (TTS); Imagens, vídeo e outras mídias estão em planos futuros.

**Audio file preprocess mechanism:** Considerando que os arquivos deverão estar armazenados na nuvem, um conjunto de processamentos off-line devem garantir tanto o aumento da qualidade do áudio como a remoção de mensagens com conteúdo inapropriadas antes dos arquivos serem disponibilizados para *download* das instalações físicas.

Dentre as principais modificações na arquitetura do hardware estão: a) construção de amplificadores de áudio; b) controle independente de fitas RGB LED; c) nova configuração dos componentes internos na caixa protetora a fim de minimizar interferências, ruídos e aquecimento; d) Substituição do dispositivo Android por uma placa com um processador com maior capacidade de processamento; e) 4 saídas independentes de áudio, permitindo que se tenha a execução simultânea de 4 mensagens.

## 5. Conclusão

A engenharia, com sua visão funcional, e a arte, com sua estética e bem estar, se fundem em projetos de ensino que abordam a criação de instalações multimídia interativas. A instalação intitulada *Árvore de Memórias* foi a primeira em exibição no Campus da UFES em São Mateus e tinha como objetivo aproximar a comunidade acadêmica. Bauman (2013) defende que o uso intenso das redes sociais tendem a deixar relações interpessoais mais superficiais e, conseqüentemente, propiciar o afastamento daqueles que estão perto em detrimento de aproximar os que estão longe. Logo, integrar as redes sociais em uma instalação que aproximasse as pessoas que estão geograficamente próximas foi premissa básica na concepção dessa instalação.

Este artigo apresentou a descrição técnica, concepção artística e principais resultados de uma instalação multimídia interativa denominada *Árvore de Memórias* desenvolvida no contexto de um projeto de ensino e vinculado a um projeto de extensão. Um vídeo com maiores detalhes do desenvolvimento, montagem e reação da comunidade está disponível <https://youtu.be/4Bd2OXaKcjc>.

## 6. Referências

Alimisis, D., Moro, M., Arlegui, J., Pina, A., Frangou, S., & Papanikolaou, K. (2007, August). Robotics & constructivism in education: The TERECoP project. In *EuroLogo* (Vol. 40, pp. 19-24).

- Bauman, Z. (2013). *Liquid modernity*. John Wiley & Sons.
- Costalonga, L. ; Aguiar, E. ; Coura, D. ; Neves, M. V. M. (2013). *Campus Vivo Instalações Artísticas e Artefatos Culturais Computadorizados*. In: *Workshop on Ubiquitous Music (IV UbiMus)*, 2013, Porto Alegre. *Proceedings of the IV Workshop on Ubiquitous Music*, 2013.
- Goodwin, K. (2008). *The impact of interactive multimedia on kindergarten students' representation of fractions*. *Issues in Educational Research*, 18(2), 103-117.
- Iskander, M. F., Catten, J. C., Jones, A., Jameson, R., & Balcells, A. (1995, November). *Interactive multimedia lessons for education*. In *Frontiers in Education Conference*, 1995. *Proceedings.*, 1995 (Vol. 1, pp. 3a2-1). IEEE.
- Kim, H. J., Coluntino, D., Martin, F. G., Silka, L., & Yanco, H. A. (2007, August). *Artbotics: community-based collaborative art and technology education*. In *ACM SIGGRAPH 2007 educators program* (p. 6). ACM.
- Penny, S. (2013). *Art and robotics: sixty years of situated machines*. *AI & society*, 28(2), 147-156.

# Reflexões sobre o Ensino de Computação Musical: Como é a disciplina de Introdução a Computação & Música no INF/UFRGS

Marcelo Johann, Marcelo Pimenta

Laboratório de Computação Musical (LCM: [www.inf.ufrgs.br/lcm](http://www.inf.ufrgs.br/lcm))

Instituto de Informática, UFRGS - Porto Alegre, RS – Brasil

{johann, mpimenta}@inf.ufrgs.br

***Abstract.** This paper describes the course on Introduction to Computer Music given at the Instituto de Informática, UFRGS, with seven editions offered between 2005 and 2014. Emphasis is given to the selected contents, specific resources used in class, the integration of students from different levels and curricula, and evaluation methodology, including theoretical, practical and artistic assignments. The course provides an opportunity for students to gain a common base on audio and music concepts, and also to become aware of the large variety of related topics and applications with which they might work in the future. Finally, a set of questions about teaching Computer Music in Computer Science courses is proposed.*

***Resumo.** Este artigo descreve a disciplina de Introdução a Computação e Música, no Instituto de Informática da UFRGS, com sete edições já ocorridas entre 2005 e 2014. É dado enfoque ao conteúdo selecionado, ao uso de recursos específicos durante as aulas, integração de alunos de diferentes níveis e cursos e na forma de avaliação, que inclui trabalhos teórico, prático e artístico. A disciplina oferece oportunidade para os estudantes formarem uma base de conhecimentos comuns de áudio e música e perceberem a grande diversidade de assuntos e aplicações com as quais podem trabalhar no futuro. Finalmente, alguns temas são propostos para reflexão sobre ensino de Computação e Música nos cursos de computação.*

## 1. Introdução

Computação Musical (CM) é uma área inerentemente interdisciplinar e que engloba problemas computacionais interessantes e relevantes, como por exemplo, de representação de conhecimento musical, de controle em tempo-real de dispositivos, de interação com usuários possivelmente leigos em informática, para enumerar apenas alguns, subjacentes ao universo musical.

No entanto, normalmente é uma área não abordada de forma adequada nos currículos de computação e engenharia. Alguns dos temas envolvidos até aparecem em diversas disciplinas de cursos relacionados, mas normalmente estão fracionados, e a teoria não é facilmente associada à prática pelos alunos. O que se espera de uma disciplina de introdução à CM é: (i) apresentar os fundamentos teórico-práticos do uso de sistemas e tecnologias computacionais para composição, síntese, performance de sons e música, assim como para educação musical com auxílio do computador; e (ii)

permitir que os alunos entrem em contato com múltiplas (quantidade) e diversas (variedade) tecnologias envolvidas em CM.

Claramente, uma das dificuldades é a carência de livros-texto em Português. Atualmente, alguns dos conceitos necessários incluídos em disciplinas de CM relacionadas a programas de graduação e pós-graduação poderiam fazer parte do currículo de referência da SBC (Sociedade Brasileira de Computação) (SBC 2015). Apesar da grande diferença dentro dos planos de ensino de disciplinas relacionadas à área de CM (diversas ementas, conteúdos programáticos e nomenclaturas das disciplinas), existe aparentemente um consenso: a falta de livros-texto adequados em português (e mesmo em inglês) que contemplem as características de multidisciplinariedade e diversidade inerentes à área de CM. Como em geral a disciplina de CM ocorre em cursos em que esta perspectiva multidisciplinar não é usual, cabe aos professores assumir a grande responsabilidade de estimular e propiciar o contato dos alunos com ideias, conceitos, técnicas e práticas destas múltiplas áreas. Além do mais, alunos de CM devem lidar frequentemente com uma multiplicidade (saudável a nosso ver) de linguagens, técnicas, ferramentas, ambientes e plataformas, com algumas dificuldades para integração e portabilidade.

No Brasil, existem disciplinas de CM (com este título ou não) em cursos de vários níveis (graduação, pós-graduação) e com diferentes enfoques e abordagens. Neste artigo, apresentamos um relato sobre as várias ocorrências da disciplina CM no Instituto de Informática da UFRGS. O objetivo deste artigo é primeiramente relatar a abordagem particular na implementação da disciplina na UFRGS, compartilhando a experiência no processo de concepção e realização das suas várias edições, e resultados observados, com os demais pesquisadores e professores da área. Em segundo lugar, buscamos também levantar a partir desta experiência alguns questionamentos que, do ponto de vista dos autores, são fundamentais na discussão sobre o Ensino de CM.

O artigo está estruturado como segue. A seção 2 descreve de forma geral a abrangência e organização dos assuntos da disciplina. A seção 3 mostra o público-alvo e os alunos que efetivamente a cursam, bem como os colaboradores e convidados que atuam para apoiar e enriquecer o conteúdo. A seção 4 apresenta de forma mais detalhada a metodologia e os recursos didáticos empregados, com exemplos das aulas mais importantes, que caracterizam essa abordagem, e são pontos importantes de referência e reflexão. A seção 5 descreve brevemente os variados mecanismos de avaliação, e finalmente a seção 6 apresenta as nossas conclusões e discussões propostas.

## **2. Abrangência e Organização da Disciplina**

Nesta seção, é apresentado um resumo da disciplina de CM, principalmente pela lista de assuntos e pelo enfoque adotado. Em linhas gerais, pode-se observar a divisão das aulas e o tratamento construtivo dos conceitos abordados, partindo dos mais fundamentais, inicialmente conceitos físicos e matemáticos, até os mais complexos e aplicados, com maior ênfase em pesquisa na parte final.

A disciplina de Introdução à Computação e Música é oferecida no Instituto de Informática como disciplina eletiva, com o nome “Tópicos Especiais em Computação: Introdução à Computação e Música” e é oferecida neste formato tanto para a graduação quanto para a pós-graduação. Disciplinas de “tópicos especiais” são uma forma flexível que a Instituição possui para incluir no currículo básico alguns

temas atuais ou não convencionais e que não sejam parte de disciplinas regulares (periodicamente oferecidas). É nossa intenção que CM torne-se uma disciplina eletiva regular, principalmente nos cursos de graduação Bacharelado em Ciências da Computação (CIC) e Engenharia da Computação (ECP) e também do Programa de Pós-graduação em Computação (PPGC).

Para os 3 cursos – CIC, ECP e PPGC – a disciplina CM não exige pré-requisitos específicos. Entretanto, existe a exigência de que o aluno tenha integralizado 100 créditos obrigatórios do seu currículo, de forma a garantir que tenha adquirido bem os fundamentos e tecnologias essenciais, e cursá-la mais para o final de seu bacharelado. Apesar de não exigir pré-requisitos, é notória a ausência de alunos de outros cursos, sobretudo do curso de Música, talvez por uma falta de divulgação maior ou ainda por esta disciplina não constar do currículo perene do curso, sendo oferecida como uma disciplina de tópicos especiais. O oferecimento da disciplina para os alunos de CIC, ECP e PPGC ocorre simultaneamente, ou seja, as 3 turmas são integradas em uma só, com horário e sala unificados, como ocorre eventualmente em outras disciplinas de interesse eventual ou de pesquisa, oferecidas nesse formato. Isto foi deliberado para estimular a convivência entre alunos com diferentes tipos de maturidade, conhecimento e experiências e também para facilitar a operacionalização das aulas. Embora integradas, as turmas de graduação e pós-graduação são avaliadas diferentemente, como será detalhado adiante.

Em todas as sete ocorrências da disciplina (em 2005, 2006, 2009, 2010, 2012, 2013 e 2014), o conteúdo e o roteiro (ordem em que os conteúdos são vistos) em sua essência foram pouco alterados. As variações de uma ocorrência a outra são relativas à forma de organizar as aulas em cada parte do roteiro, à profundidade dedicada a um ou outro tópico e ao tipo de avaliação efetuado. Apresentamos abaixo os objetivos e o conteúdo programático em formato semanal, conforme registrado no sistema de dados da Universidade:

**Objetivos:** “A disciplina tem por objetivo geral introduzir os conceitos básicos sobre Computação e Música. O objetivo específico é permitir aos participantes conhecer, compreender e utilizar os conceitos principais, equipamentos, técnicas, modelos, ferramentas e linguagens de Computação e Música.”

**Tabela 1. Conteúdo Programático**

Semana	Título	Conteúdo
1	Introdução e Histórico	Introdução, motivação e perspectivas; História dos instrumentos, métodos, descobertas e invenções para síntese de som e música.
2 a 3	Conceitos de som e música	Conceitos básicos de som e música, vibrações, ondas, sons e sinais elétricos, domínio tempo e frequência, harmônicas, consonância e dissonância, timbre, intervalos, notas e acordes, evolução das escalas musicais e temperamento.
4 a 5	Síntese de Som	Métodos e equipamentos de síntese: síntese, osciladores, síntese aditiva, síntese subtrativa, filtros e envelopes, modulação em frequência, modelagem física, síntese granular, outras formas de síntese.
6	Representação de som e música	Representação da informação sonora: formatos de áudio digital e MIDI. Gravação, processamento e reprodução de áudio digital.
7	Definição de Trabalhos	Apresentação e discussão de temas para trabalhos teóricos, práticos e artísticos
8	Verificação Escrita	Verificação Escrita (PROVA)

9	Educação musical	Educação musical: aspectos principais, exemplos e utilização.
10	Ferramentas de <i>Software</i>	<i>Software</i> musical para acompanhamento, edição de partituras, seqüenciamento, gravação de áudio, estações de audio digital e programação de <i>plug-ins</i> .
11	Música em Jogos de Computador	Música em Jogos de Computador: Histórico, circuitos de síntese e características; Formatos e padrões atuais; Compositores; Eventos de Música de Jogos;
12	Inteligência Artificial e Interfaces	Inteligência artificial e música, interação homem-computador e música, novas interfaces musicais;
13	Tendências e Tópicos	Tendências em Software, Hardware, Pesquisa e Tópicos Atuais
14 a 15	Apresentações	Apresentação de Trabalhos Práticos

Tomando como referência a edição de 2013, apresentamos abaixo o programa detalhado de assuntos tratados em cada aula. O programa é adaptado a cada semestre em função dos cronogramas dos professores e das oportunidades de participação de convidados. Junto com o programa, aparecem tecnologias computacionais indicadas, e as referências principais de audições escutadas no final de cada aula.

**Tabela 2. Programa de cada aula em 2013**

Data	Aula	Assuntos Principais	Audições Principais	Linguagem
09.Ago	01	Introdução, Histórico	Paganini, Demo ATC-1	
16.Ago	02	Escalas, Notas	Jean Jacques Perrey	Pure Data
23.Ago	03	Timbre, Síntese aditiva	Hammond players	Impromptu
30.Ago	04	Síntese subtrativa, filtros, controle	Walter Carlos - Swtched on Bach	
06.Set	05	Componentes e equipamentos	Gravações OSP/ULBRA	swarm.cc, osc
13.Set	06	Áudio digital, FM e Sampling	ELP, Robert Moog, EMS synthi A	JavaSound, Falstad
20.Set	-	Feriado - Farroupilha		
27.Set	07	MIDI, representação musical, osc	Animusic - Pictures at an Exhibition	JavaSound's MIDI
04.Out	08	Verificação Escrita		
11.Out	09	Discussão de Trabalhos, Artigos	Tomita - Quadros de uma Exposição	Csound
18.Out	10	VideoGame Music, pesquisa Áudio	Música de Jogos em Orquestra	OpenAudioL
25.Out	-	Semana Acadêmica		
01.Nov	11	Tópicos atuais em síntese, SW, HW	Jingles produzidos em Porto Alegre	[gibber, chuck]
08.Nov	12	Sequenciamento, DAW, plugins	Eloy Fritsch, Miranda, SBCM, ICMC	VST
15.Nov	-	Feriado - República		
22.Nov	13	Educação Musical e Pesquisa LCM	Kraftwerk, Vangelis, Kitaro	MusicXML
29.Nov	14	Tópicos atuais em C&M, interfaces	Pet Shop Boys, Jean Michel Jarre	
06.Dez	-	IV Workshop UBIMUS		
13.Dez	15	Apresentação trabalhos práticos		
20.Dez		Recuperação		

### 3. Alunos, Colaboradores e Convidados

Como visto, o público da disciplina é formado principalmente por alunos de graduação, alunos de pós-graduação (mestrado, doutorado) regularmente matriculados nos cursos de computação (CIC e ECP). Em nível de pós-graduação, também são aceitos alunos especiais (alunos que não são alunos regulares de cursos de UFRGS mas que podem se submeter a um processo de inscrição e seleção para frequentar a disciplina e, se aprovados, requerer os créditos em matrículas futuras) e alunos ouvintes (que não estão oficialmente matriculados e solicitam informalmente aos professores permissão para assistir as aulas). Eventualmente, devido aos programas de intercâmbio que o INF possui com outras universidades do exterior (notadamente da França, Alemanha e Peru), já tivemos alunos de diferentes nacionalidades convivendo na turma.

A tabela abaixo apresenta o número de alunos que participaram em todas as edições realizadas até o ano de 2014, incluindo a edição de 2011 com o formato de disciplina de projetos.

**Tabela 3. Participação Discente**

Edição	CIC	ECP	Mús	EE	Inter	PPGC	Micro	Espec	Ouv	Total
2005-2	15	9			2	11			1	38
2006-2	24	5	1			3	1	5	2	41
2009-2	23	3				9			1	36
2010-2	11	4		1					1	17
2011-2	13								1	14
2012-2	20					12	1	1		34
2013-2	14					1		4	3	22
2014-2	10									10
<b>totais</b>	<b>130</b>	<b>21</b>	<b>1</b>	<b>1</b>	<b>2</b>	<b>36</b>	<b>2</b>	<b>10</b>	<b>9</b>	<b>212</b>

Além dos estudantes, a disciplina CM tenta promover a convergência de uma heterogeneidade de conteúdos e formações também entre seus colaboradores, que ajudam os professores a conduzir as aulas e a apresentar tópicos com os quais têm bastante familiaridade e experiência. Os colaboradores com quem já tivemos a oportunidade de contar foram principalmente:

- pesquisadores do LCM (tipicamente alunos de doutorado e mestrado), como por exemplo Evandro Miletto, Luciano Flores, Leandro Costalonga, na época em que eram estudantes;
- pesquisadores ou professores do CME/UFRGS (tipicamente professores do Departamento de Música), como por exemplo Eloy Fristsch, James Correa, Rafael Oliveira;
- pesquisadores de outras instituições com quem o LCM tem cooperação estabelecida, como por exemplo Eduardo Miranda (Univ. Plymouth), Damián Keller (UFAC);
- músicos, como por exemplo Yanto Laitano (que demonstrou ao vivo seu Theremin na disciplina), Marcelo Birck (integrante da banda Graforreia Xilarmônica), Hique Gomes (ator, diretor, produtor, violinista);
- profissionais da área de produção musical, que em sua atividade profissional utilizam intensivamente aplicações musicais (p.ex. para fazer *jingles*), como Eduard Follmann, da Produtora TechnologicaWeb, em Porto Alegre;
- Contamos também com a grata colaboração de outros interessados com conhecimento e experiência específicas. Um exemplo é a ex-aluna do curso de Ciência da Computação da UFRGS, Sílvia Regina Vargas Gomes, que possui um enorme e completo acervo sobre músicas de *videogames*, em áudio, MIDI emuladores, e colaborou para organizar e ministrar uma aula sobre a história do áudio e música em *videogames*, contemplando todas as gerações de jogos desde os mais primitivos, até os atuais eventos de *Video Game Live*.

#### 4. Metodologia e Recursos didáticos

A disciplina oferecida, com apenas dois créditos (2 horas-aula por semana, em 15 semanas), é bastante densa em termos de conteúdo. Assim, percebe-se desde o início o desafio de desenvolvê-la da forma mais eficiente. Apesar de ser realmente uma disciplina com caráter mais informativo, e não um extenso treinamento com foco limitado, e desenvolvimento de habilidades bem caracterizadas, ela não pode se tornar apenas uma longa listagem de tópicos existentes. Ao contrário, deve fornecer experiências de aprendizagem marcantes e concretas.

Por essa razão, tanto a lista de conteúdos como a forma de abordar as aulas, ou seja, a metodologia de ensino, foram cuidadosamente selecionadas em termos de profundidade e material de suporte.

A lista de tópicos abordados já define a sequência fundamental de conteúdos. Além disso, uma série de tecnologias devem ser apresentadas, pelo menos sucintamente. São ferramentas de *hardware* e *software*, como as linguagens de programação orientadas a música, ou sistemas e padrões musicais, essenciais para que os estudantes trabalhem e manipulem áudio e música. Igualmente importante são as referências estéticas e artísticas, que mostram o contexto, as aplicações e o impacto que essas tecnologias tiveram. Na nossa abordagem, procuramos evitar que tais tópicos fossem concentrados em algumas aulas, e principalmente que viessem a ser introduzidos muito tarde, após os conceitos mais fundamentais.

Assim, chegamos a dois aspectos importantes da abordagem dessa disciplina. Primeiro, que a maioria das aulas, com pouquíssimas exceções, seria composta por um tema principal associado a um conceito fundamental (como escalas, síntese aditiva, interfaces), pela apresentação de ao menos uma diferente ferramenta computacional (como PureData, Impromptu, VST) e uma parte final de audição, dedicada ao aspecto artístico, lembrando e usando como exemplo algum dos grandes artistas/compositores da área (como Wendy Carlos, Vangelis, Kraftwerk). O segundo aspecto é que procuraríamos fazer um esforço para trazer as experiências mais concretas e autênticas possíveis. Em vez de mostrar apenas uma figura ou slide, damos preferência a trazer um equipamento e um instrumento e permitir que os alunos observem e ouçam o resultado da maneira bem realista, mesmo com tempo e recursos limitados, em aulas eminentemente expositivas. Acredita-se que esse contato com o objeto real de estudo, apesar de demandar esforço e um pouco mais do tempo de aula, seja essencial para que os alunos mais rapidamente compreendam os conceitos, se aproximem das questões e soluções reais.

De fato, pelos relatos que tivemos, a experiência de cursar essa disciplina parece ter sido muito significativa. Por exemplo, um conceito fundamental como espectro, Teorema de Fourier, normalmente é matéria obrigatória em um curso de engenharia. Mas não só ele é em grande parte dissociado da música, do conceito de síntese aditiva, como não são oferecidas experiências onde o aluno possa ver, ouvir, e calcular sons reais e sintéticos na aula onde isso é apresentado. Normalmente se deixa para que em outra oportunidade o aluno chegue a ter essa experiência, apresentando somente a teoria. Algumas pessoas, entretanto nunca chegam a ter a oportunidade de experimentar aquilo, o que dificulta muito sua compreensão, faz com que não conectem o conceito com o mundo ao seu redor, ou o esqueçam com facilidade. Outro exemplo é toda a complexa matemática envolvida na definição das notas e escalas. Virtualmente as pessoas ignoram isso completamente, e valem-se majoritariamente de regras ou dogmas para aprender quais são as notas, quais são as escalas existentes,

sem entender ou experimentar por quê, quais são os problemas e razões de sua existência.

Nesse sentido, as ferramentas computacionais (e eletrônicas em geral) são excelentes para oferecer experiências variadas que demonstram rapidamente esses fenômenos. Com sintetizadores e programas musicais, é fácil perceber a análise ou composição de timbres com harmônicas, e é também fácil demonstrar diferentes afinações, consonâncias e dissonâncias básicas com notas e frequências, usando simultaneamente o som, a imagem das ondas, e a matemática, com as medidas de frequência. Para melhor demonstrar essa metodologia, citamos os recursos empregados em algumas dessas aulas iniciais, como exemplo.

Para demonstrar as escalas, usamos a abordagem apresentada por (FERENCE 1978), que toma a escala “verdadeira” ou Ptolemaica como uma referência principal, para demonstrar a grande quantidade de consonâncias perfeitas obtidas com frações simples ideais. Essa experiência é oferecida com um *patch* em PureData que reproduz duas ondas dente-de-serra (mais fáceis de identificar do que apenas um seno), um osciloscópio digital de duplo canal, um frequencímetro, uma mesa de som externa que permite rotar e isolar os dois canais, e o quadro onde anotamos as frequências percebidas como consonantes. Os alunos são solicitados a identificarem quais frequências melhor combinam, e começam a descobrir as frações básicas  $2/1$ ,  $3/2$ ,  $5/4$ ,  $4/3$ , pontos centrais para definição de escalas. Então é apresentado o problema de como definir todas as notas intermediárias, ou dividir os intervalos de tom maior e tom menor ( $10/9$  e  $9/8$ ) em intervalos menores, de forma a obter notas que combinem com as já encontradas, problema que sabe-se não ter uma solução completa. Ao apresentarmos a escala igualmente temperada, predominante na música ocidental moderna, usamos um sintetizador com as duas afinações para mostrar claramente como os acordes soam em cada um dos sistemas, evidenciando os problemas e vantagens de cada uma. Os alunos ficam impressionados em perceberem pela primeira vez a relativa “desafinação”, em relação a acordes com frações perfeitas, da escala moderna, mas entendem facilmente que não é possível afinar um acorde perfeitamente como em uma escala Justa pois isso implicaria em deixar muito mais dissonantes outros acordes.

Como segundo exemplo de metodologia e procedimentos, aparecem as aulas de síntese aditiva e subtrativa. Elas também são executadas com auxílio de osciloscópio, analisador FFT e espectrograma, permitindo aos alunos ouvirem e verem as formas de onda e os efeitos de adição e subtração de harmônicas. Acrescentamos um elemento marcante nessas aulas que é a presença de sintetizadores analógicos, de forma a oferecer a experiência sonora mais autêntica possível. Para a síntese aditiva, usa-se equipamentos apropriados (em geral um Korg CX-3 modelo antigo, analógico, ligado a pré-amplificador valvulado MesaBoogie V-twin e simulador de caixas Leslie Dynacord CLS-222). Isso permite-nos executar facilmente desde as experiências mais simples de síntese com duas ou três harmônicas, quando eles percebem que as ondas senoidais são facilmente identificadas separadamente mas fundem-se na nossa percepção quando passam a acompanhar uma a outra quando disparadas pelas mesmas teclas, até o ápice da composição de um som famoso como o obtido por um órgão Hammond com o efeito sonoro característico das caixas Leslie.

Já para síntese subtrativa, trazemos para a aula um pequeno sintetizador puramente analógico, em geral um ARP AXXE. Novamente, ele permite as experiências fundamentais que demonstram os conceitos e a operação dos osciladores,

filtros, envelopes e modulação, com audição e visualização no espectrograma, e também a riqueza e abrangência de tais timbres assim como produzidos pelos circuitos eletrônicos clássicos no domínio analógico. Em geral acreditamos que a presença desses instrumentos originais dá maior sentido para a compreensão da capacidade plena desses métodos, sem contar com sua importância histórica e pelo caráter de curiosidade que desperta nos estudantes. Somente no final da aula passa-se a apresentar as emulações, recriações virtuais, ou mesmo modelos computacionais mais simples que também empregam tais métodos de síntese, mas então com a vantagem de estarem acessíveis a qualquer um em qualquer lugar.

## 5. Mecanismos de Avaliação

Como já citado, mesmo considerando a disciplina como mais informativa, pela enorme abrangência e interdisciplinaridade, e pela diversidade de formação prévia dos alunos, sempre entendemos que os conhecimentos mais fundamentais devem ser sólidos e precisos. Os procedimentos didáticos são selecionados justamente para isso. Desta forma, mesmo com o pouco tempo, propusemos desde o início que o aproveitamento e desempenho dos alunos fosse avaliando com uma prova escrita e com a realização de trabalhos de implementação e escrita de textos. A escrita de um texto toma a forma de um artigo, ou “*survey*” sobre um tema de escolha dos alunos, sendo que este artigo, com melhor fundamentação teórica, é um dos aspectos de avaliação que recebe maior importância e peso para as turmas de pós-graduação. O trabalho de implementação é em geral um trabalho maior, final, a ser apresentado para todo o grupo em uma ou duas das últimas aulas.

Após as primeiras edições, percebeu-se que o aspecto artístico também deveria ser abordado, não somente pela participação, ainda que pequena, de alunos de outros cursos (artes, música, outros), como pela importância do aspecto artístico nesse tema. Assim, passamos a propor a realização de mais um trabalho artístico. Esse consiste na produção de um trecho de áudio com entre 30 segundos e 1 minuto, feita utilizando-se algum recurso computacional, preferencialmente algum método relacionado à computação, com algoritmo, função, linguagem, e não apenas com o uso de um programa pronto. Ou seja, estimulamos os alunos a pensarem computacionalmente, criativamente, sobre a música, e não a serem usuários de algum produto já existente, embora essa interface não seja sempre clara.

Assim, a avaliação consiste finalmente nesses quatro componentes, com os seguintes pesos para os alunos de graduação e pós-graduação:

**Tabela 4. Pontuação na Avaliação**

Nível	Prova	Trab. Artístico	Trab. Teórico	Trab. Prático
graduação	3	2	2	3
pós-graduação	3	2	3	2

Os trabalhos teóricos em geral apresentam em nossa opinião um nível médio de qualidade e profundidade. De fato, na maioria dos casos eles demonstram apenas o aprendizado dos alunos, que descrevem determinados temas como foram compreendidos por eles, e enriquecidos com outras referências ou exemplos. Em diversos casos, entretanto, os alunos procuram também temas de seu interesse que não

foram cobertos nas aulas da disciplina, parte desses com o objetivo de servirem como suporte conceitual e estudo para a implementação de algo que pretendem como projeto de trabalho prático, o que lhes é proposto e incentivado. Eventualmente alguns trabalhos abordam temas inesperados, inovadores, que não haviam sido sequer considerados pelos professores, o que consideramos uma boa contribuição de originalidade e criatividade dos alunos.

Entretanto, é nos trabalhos práticos finais e nos trabalhos artísticos que os alunos mais se destacam e mostram sua capacidade, empenho e criatividade. Nos trabalhos finais, geralmente na forma de implementações em *software* ou em *hardware*, os alunos tem demonstrado ao longo das várias edições uma variedade enorme de ideias, algumas implementadas com grande sucesso. Uma lista completa não pode ser apresentada neste artigo, mas os trabalhos incluem programas de síntese, análise, variados jogos musicais, diversas interfaces, algumas envolvendo dispositivos como aparelhos celulares, câmeras, controles de videogame, sistemas de hardware com acelerômetros, sensores ópticos, etc. Já foram apresentados jogos envolvendo realidade virtual, síntese de imagem a partir do som e síntese de som a partir de imagens, a maioria dos sistemas com gráficos em 3D, jogos musicais, baseados em MIDI, entre outros. Um aspecto que merece destaque é o fato de que mesmo com a maioria dos alunos sendo de cursos de Ciências da Computação, diversos protótipos de hardware foram apresentados funcionando perfeitamente, incluindo tanto *hardware* digital com microprocessador, com circuitos discretos, ou em FPGA, quanto circuitos analógicos, implementados por alunos de Computação.

Finalmente, a realização do trabalho artístico proposto é um dos fatores que mais nos chamou a atenção, e mais demonstra o potencial criativo dos alunos. Os trabalhos são reunidos previamente, junto com um slide de informações essenciais sobre cada um, como um formulário, e são apresentados em uma sessão de audição específica, que pode ser em parte de uma aula, ou em um horário específico, por exemplo, na Semana Acadêmica da Universidade. Ao serem incentivados a produzirem áudio sem ter exigências técnicas ou estéticas, os estudantes parecem explorar mais livremente as suas ideias. Eles sabem que não necessariamente precisam produzir uma música “convencional”, ou algo fácil de ser ouvido, mas algum som que tenha uma forma computacional, como se estivessem desenvolvendo um recurso, uma ferramenta, respondendo à alguma pergunta, testando alguma hipótese. Trabalhos apresentados incluem diferentes fórmulas ou modelos de geração de áudio ou sequências MIDI, traduções ou processamentos nos dois domínios, uma enorme variedade de recursos digitais, amostras obtidas ou criadas por eles, e principalmente, como é o objetivo do trabalho, propostas e resultados estéticos diversos e interessantes. Embora alguns trabalhos isolados sejam compostos apenas por alguns ruídos (na linha de “*noise music*” e “*glitch music*”), todos percebem na sessão de audição que o importante é como se produziu estes sons e não se são esteticamente aprovados ou não: assim, os trabalhos artísticos no conjunto ganham sentido, como exploração de recursos em uma grande e diversa paleta de cores. A experiência de audição sempre tem sido vista por todos como um momento lúdico, descontraído, surpreendente, e carregado de aprendizado, inclusive pela oportunidade de percepção e conhecimento entre os alunos, pelo que eles produzem ou o que o método de sua escolha produziu. A proposta desse trabalho não poderia ter sido mais acertada, e superou em muito as nossas expectativas.

## 6. Discussão e Considerações Finais

As seguintes reflexões emergiram a partir das experiências descritas. Muitas delas levam a questões fundamentais dentro da problemática que é o Ensino de CM e acreditamos que devem ser discutidas de uma forma ou de outra por vários docentes de CM no Brasil. As observações estão reunidas em três grupos básicos associados às três seguintes perguntas:

Reflexão 1 - Por quê e quando estudar CM? Quais são os pré-requisitos? É necessária alguma maturidade por parte dos alunos ou isto não é relevante?

Reflexão 2 - O que estudar dentro da área de CM? Quais tópicos privilegiar? Como conciliar o conhecimento teórico com a aplicabilidade prática? Como desenvolver exercícios práticos de uso das várias tecnologias (*hardware* e *software*) no prazo limitado de um semestre letivo?

Reflexão 3 - Qual(ais) o(s) enfoque(s) mais adequados para ensino de CM no contexto dos cursos do Brasil?

Estas questões podem servir como um roteiro para futuras discussões sobre o assunto. Espera-se que em nosso fórum da comunidade (SBCM) sempre tenhamos espaço para este tipo de discussão, e neste artigo procuramos registrar a abordagem realizada como forma de contribuir para esse processo.

### ***Reflexao 1:***

Constatou-se nas várias ocorrências da disciplina de CM que a grande maioria dos alunos matriculados e ouvintes são os que possuem algum interesse (geralmente em caráter amador) em atividades musicais: pessoas que tocam algum instrumento, ou que tiveram alguma formação musical no passado ou ainda que sejam ouvintes intensos, frequentemente com conhecimento sobre um determinado tema e estilo que supera o conhecimento dos professores. Ou seja, o público tem um interesse e um potencial invejável. A nossa experiência nesse caso foi bastante promissora, mesmo tendo em uma mesma turma alunos de graduação, pós graduação, computação, engenharia elétrica e música.

### ***Reflexao 2:***

Nas várias ocorrências da disciplina CM temos seguido o objetivo fundamental de conciliar teoria e prática, Estamos acostumados em outras disciplinas da Informática a pensar que frequentemente o que precisa ser alcançado é a aquisição de conceitos, com as técnicas e ferramentas sendo somente tratadas a nível introdutório ou ilustrativo. No entanto, esta separação entre teoria e prática não nos parece apropriada em CM. Se é verdade que muitos antigos e consolidados conceitos de CM continuam sendo importantes – tais como as noções de síntese aditiva, síntese subtrativa e síntese FM – também é verdade que é preciso saber “botar a mão na massa” para ter assimiladas as características e dificuldades de cada forma de síntese e também para reconhecer “no ouvido” as limitações de cada uma.

Além disto, optamos por não privilegiar *uma* tecnologia (linguagem, ferramenta, técnica, ambiente) em detrimento de outras. Esta posição pode ser controversa pois muitos cursos no Brasil focam a disciplina de CM basicamente em atividades usando PureData (PD). De fato, o livro de Miller Puckette [3] é muito bom, muito bem escrito, muito completo e consegue mostrar o poder de expressão e a

versatilidade de PD. Se os alunos percorrerem uma boa parte do livro entendendo e executando os *patches* certamente terão uma boa ideia do campo da CM.

No entanto, acreditamos que na CM convive-se e deve-se conviver com uma multiplicidade (quantidade) e uma diversidade (variedade) de tecnologias. Embora possa parecer criticável a primeira vista, temos preferido possibilitar acesso dos alunos – mesmo que superficialmente em alguns casos – às várias tecnologias existentes e isto só é viável se – mesmo sem ter condições ou tempo para utilizar todas – os alunos conhecerem os fundamentos subjacentes a estas tecnologias.

Em relação à falta de um livro-texto em português, os autores acreditam que a elaboração de livros didáticos ou artigos tipo "*survey*" sobre alguns tópicos desta área é uma meta indispensável à melhoria do conteúdo da disciplina. Algumas iniciativas isoladas de textos em português existem, como a apostila do curso "Introdução à CM" (Miletto et all 2004) e a tradução do livro de Miller Puckette (Puckette 2003) efetuada por Alexandre Porres (SP) mas ambas contemplam apenas parcialmente o conteúdo da nossa disciplina. Ambos podem ser solicitados diretamente aos autores ou encontrados na Internet. Uma ideia interessante para atenuar esta falta seria a criação de um repositório de material (copyleft de preferência) – algo como CompMusLib(re), a biblioteca "livre" da compmus - que pudesse armazenar e disponibilizar para a comunidade brasileira os materiais criados por membros desta mesma comunidade.

Além disto, é muito importante que os alunos tenham acesso a bibliografia (artigos em conferências como ICMC, ISMIR, NIME e journals como Leonardo, Organised Sound, Journal of New Music Research, Computer Music Journal; e capítulos de livro, etc) na sua versão original (tipicamente em inglês) e que a sua leitura seja estimulada pelo(s) professor(es). Propiciar o acesso direto dos alunos às fontes de informação sem necessidade da intermediação do professor (como acontece nas aulas expositivas) é uma forma de incentivar a autonomia da turma e de inculcar nos alunos o salutar hábito de leitura.

### ***Reflexão 3:***

Um enfoque muito baseado em ferramentas tem como obstáculo eventual a dificuldade de obtenção de ferramentas. Uma maneira de obter ferramentas, além de desenvolvê-las, é a adoção de ferramentas open source disponíveis.

Em relação aos trabalhos práticos, os alunos encontram grandes dificuldades para realizar trabalhos que sejam ao mesmo tempo complexos o suficiente para permitir o aprofundamento desejado de conhecimentos e delimitados o suficiente para serem realizados no período letivo de um semestre (tipicamente 4 meses). Trabalhos grandes demais ou pequenos demais têm uma função didática questionável. Pode-se pensar como alternativa em uma disciplina adicional de CM (p.ex. Projeto de Aplicações CM, como implementamos no semestre 2011-2), onde um trabalho pode iniciar em uma disciplina e terminar em outra. Outra opção é tolerar que os trabalhos de CM sejam aproveitados no contexto de alguma outra disciplina (p.ex. IHC, Computação Gráfica, etc) e assim os alunos implementam, estudam e exploram conceitos das 2 disciplinas complementarmente em um trabalho só.

Em relação a participação dos alunos, constata-se em geral uma melhoria substancial na compreensão dos conceitos quando se utilizou aulas participativas. Existe uma correlação grande entre o rendimento da disciplina, a personalidade dos

estudantes e a agilidade do(s) professor(es) para orientar a discussão para o rumo desejado. Isto pressupõe alunos que perguntam e perguntas que sejam adequadas para permitirem avançar no tópico tratado e um professor que as utilize como "gancho" para os conceitos que pretende apresentar.

Em suma, neste artigo, a experiência dos autores na condução da disciplina CM no INF/UFRGS foi descrita e serviu como ponto de partida para coletar uma série de reflexões sobre questões usualmente relacionadas a problemática do ensino de CM. Os autores pretendem que o relato e suas reflexões sirvam como ponto de partida de um maior fórum de discussões a respeito pela comunidade de CM no Brasil.

## Referências

- SBC (2015). SBC – Sociedade Brasileira de Computação. “Currículo de Referência”. Disponível em: <[http://www.sbc.org.br/en/index.php?option=com\\_jdownloads&Itemid=195&task=view.download&catid=36&cid=52](http://www.sbc.org.br/en/index.php?option=com_jdownloads&Itemid=195&task=view.download&catid=36&cid=52)>
- Miletto et all (2004). E. M. Miletto, L. L. Costalonga, L. V. Flores, E. F. Fritsch, M. S. Pimenta e R. M. Vicari. 2004. “Introdução à Computação Musical”. Congresso Brasileiro de Ciência da Computação, Itajaí, 2004, pag. 883 – 902. Itajaí, SC – Brasil, ISSN 1677-2822. Disponível em: <<http://www.inf.ufrgs.br/lcm>>
- Puckette (2003). Puckette, Miller. “Theory and Techniques of Electronic Music”, 2003. Disponível em: <<http://msp.ucsd.edu/techniques/v0.02/book-html/>>
- Ference (1978). Ference Jr, M.; Lemon, H. B.; Stephenson, R. J. “Física: cursos colegial e vestibular. v.3 - Ondas (Som e Luz)”. - Editora Edgard Blücher, 1978.
- LCM (2015). “Laboratório de Computação e Música”. Instituto de Informática/UFRGS. Disponível em: <<http://www.inf.ufrgs.br/lcm>>
- CME (2015). “Centro de Música Eletrônica da UFRGS”. Instituto de Artes/UFRGS, Disponível em: <[www.ufrgs.br/musicaeletronica](http://www.ufrgs.br/musicaeletronica)>
- Johann e Pimenta (2015). Johann, Pimenta. “Tópicos Especiais em Computação: Computação e Música”. Disponível em: <[http://www.inf.ufrgs.br/lcm/site\\_arquivos/topespcm.html](http://www.inf.ufrgs.br/lcm/site_arquivos/topespcm.html)>

# Symbolic regression as a computer-aided music tool for analysis and composition

José Henrique Padovani<sup>1,2\*</sup>, Jônatas Manzolli<sup>1</sup>

<sup>1</sup> mus<sup>3</sup> - Musicology, Sonology and Computing – Music Department – UFPB  
Federal University of Paraíba, Campus I - Castelo Branco, João Pessoa - PB, 58051-900

<sup>2</sup>NICS - Interdisciplinary Nucleus of Sound Communication – UNICAMP - University of Campinas  
Rua da Reitoria, 165 - Cidade Universitária "Zeferino Vaz", Campinas - SP, 13083-872

josehenriquepadovani@pq.cnpq.br, jonatas@nics.unicamp.br

**Abstract.** *The paper presents a new OpenMusic library that implements a Genetic Programming method of Symbolic Regression on sets of input data-points and seeks for a Common Lisp function (S-expression) that can be used either to create mathematical models that could potentially help to understand the mathematical behavior of the input data or to generate parameters in computer-aided composition. Stressing that a number of issues must still be addressed to improve the proposed library, the paper presents some of the strategies to do this and to make Symbolic Regression a practical tool in computer-assisted music composition and analysis.*

## 1. Introduction

Despite the established distinction between algorithmic music and computer-aided composition (CAC) and the related differentiation regarding the role and prevalence of algorithms in the creative processes [12, 29, 5], music composition and analysis with algorithmic tools demand, to a greater or lesser extent, the formalization and modeling [38, 7] of musical materials and processes. While algorithmic and mathematical *models* of musical procedures and materials are usually developed by composers, analysts or researchers, *genetic programming* (GP) [20, 18, 28] techniques can be explored to achieve optimal models that could be employed in different analytical and compositional computer-aided music (CAM) contexts.

Our work addresses the question of automatically generating rough *algorithmic models* in CAM environments by using GP techniques such as *Symbolic Regression* (SR). Currently, the process is implemented as library for the CAM environment *OpenMusic* and as a external application in SBCL. While the project is still in its first stages, the proposed approach provides a new compositional and analytical tool to create *mathematical functions* and *symbolic expressions* that could optimally describe or outline data-sets such as two or more interrelated musical parameters (pitch, onset, duration, audio features, etc). Used in iterative processes, those expressions can be employed to generate new values for the selected parameters or as a starting point to develop more accurate models.

## 2. Algorithmic models for composition and analysis

In computer-aided, algorithmic and generative music, as well as in other specific musical practices that directly use computational algorithms in music composition and analysis,

---

\*Supported by CNPq.

musical materials, procedures and processes are to be described as data structures and *algorithms* or *computational methods* [27, p. 4]. That is to say that they need to be declared in any computer language in way that both the relevant properties of music materials and the steps/methods of musical procedures and processes become unambiguously and formally stated so that they can be automatically created, processed, managed or operated by a computer.

While in the field of music composition this formalization requirement can be identified already in seminal works on composition with algorithmic tools [19, 44] as well as in more recent texts of composers dealing with computer-aided music (CAM) environments [1, 2], in the field of musicology the use of these same tools has lead to new approaches in music analysis, specially *analysis by modelling* or *formalized analysis* [38, 39]. The formalized statement of musical materials and processes results in *algorithmic models* that can be used to generate new musical ideas or to reconstitute given music excerpts (*local models*) or entire pieces (*global models*) [38, p. 91].

Given the fact that an *algorithmic model* does not describe music as a score [42, p. 1-6], being rather a formal description of the structures, procedures and processes that can generate music notation or synthesized sounds, it becomes an invaluable tool both in composition and in musicology. In composition, these tools provide means to prototype a musical piece or a passage with different input parameters and simulate the possible results of a formalized compositional process. In computational musicology, on other hand, algorithmic modelling makes it possible not only to better understand the intrinsic features of materials, processes and procedures of a given piece or compositional technique but also offers a glimpse of other possible results that the *algorithmic model* of the analyzed piece could yield. Whereas this approach to music analysis has lead to a number of *algorithmic models* of different pieces and styles [35, 11, 26, 14], also the modeling of counterpoint and other compositional techniques [3] used in many algorithmic and computer-aided compositions since the *Illiac Suite* (1957) [19] actually apply *formalized analysis* in creative processes.

In the context of creating *algorithmic models* either to compose or to analyze music, the process of *formalizing* structures and processes to develop compositional procedures is not just a particular stylistic approach. Indeed, it becomes, rather, “a necessity and, even, an imperative” [32, p. 234] to the development of what may be called a “*sofêge* of models”: the ability to control and “to master both the musical result of a given generative model and the relationship between the graphic and/or textual representations of some music software and the music outcome” [31, p. 142]. However, to compose or to analyze music pieces or excerpts by using algorithmic tools does not imply that all the compositional procedures and materials are readily reducible to *algorithmic models* [39, p. 66]. Indeed, even if the “rule-based nature” [17, p. 108] of some musical ideas, compositional techniques and specific music pieces makes them to be specially suitable for being formally described by *algorithmic models*, the very process of coding them with computer languages is often extraneous to the original creative intuition or, in the case of *formalized analysis*, to the composer techniques. Further, there are cases in which the behavior of given musical parameters (that may be empirically taken from an improvisation or from a musical piece, for example) do not seem to be generated or easily described by any known rule or formalized process: the model, if exists, is unknown.

Bearing in mind that *algorithmic models* for composition and analysis do not necessary reflect the way a composer thought or created a given piece or excerpt [39, p. 65] it is possible to hypothesize that, even music pieces or excerpts that apparently are not based on any rule system or methodical procedures, there are *models* to be found that

could optimally generate similar outcomes for interrelated variables such as pitch, onset, etc.

### 3. Genetic Algorithms, Genetic Programming and computer-aided music

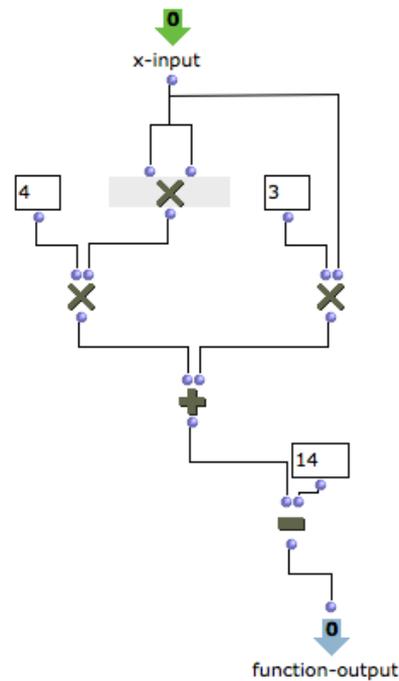
Among many artificial intelligence approaches to this problem, evolutionary computing provide tools that may be used in the heuristic search for optimal models. Briefly, *genetic algorithms* (GA) [20, 18] and *genetic programming* (GP) [28] processes rely on data-structures and algorithms that are based on evolutionary theories and on the concept of *natural selection*. As the *genome* or digital representation of given *individuals* (an algorithm or a data-structure) is transformed by a series of bio-inspired processes such as *mutation* and *cross-over*, *individuals* are selected to “survive” through a series of successive iterations (*generations*) according to their *adaptation* to any sort of evolutionary pressure – usually given by a *fitness function*. While in GA the *individuals* are usually any sort of digitally represented data-structure, in GP they generally are *algorithms*, *symbolic expressions* or *computational instructions* that are automatically generated and selected according to its *fitness rate* in solving a given problem.

The use of GA and GP techniques in computer music is not new. Since the 1990s, a number of artists and researchers have applied evolutionary computation techniques in music composition and in computer music research. Many of these works use these techniques as a means to generate music [23, 41, 6, 24, 10, 33, 36, 25] or as an heuristic tool to solve specific problems – such as finding optimal FM synthesis parameters [22, 21], to automatically extract descriptors from audio signals [45] or to create mimetic orchestrations[13] from an input audio target, for instance. However, despite being more frequently employed in generative algorithmic music and as tools to solve specific problems in comparison with its use in composition/analysis contexts with CAM environments, the non-deterministic behavior and the heuristic potential of evolutionary computational techniques makes them appealing to the interactive work-flow that typically characterizes computer-aided composition and formalized analysis paradigms. Particularly, we envisage potential benefits for the composer and for the musicologist in using CAM tools not only to model musical materials and procedures but, also, as a heuristic instrument in the search for unknown models in the form of *mathematical functions* and *symbolic expressions* that may be used in iterative computations.

### 4. S-expressions and symbolic regression

In visual CAM environments such as *OpenMusic* (OM) [7, 8] and *PWGL*[30], both based on *Common Lisp* (CL), algorithms are visually represented in the form of *patches*. “Boxes” may represent functions, data-types (lists, integers, floating-point numbers, strings, etc.), methods or classes that are interconnected by “chords” to create more complex computational procedures. Internally, though, as in any *Common Lisp* code, algorithms are formed by *symbolic expressions* or *s-expressions*: trees of lists that combine atoms (numbers, strings, symbols and some other non parenthesized structures) and lists that may contain simple data-types or that may call functions and macros that are sequentially evaluated to yield their outputs to the respective parent-lists. Hence, more complex algorithms and computer applications can be thought as a tree of lists that either contain data (terminals) or functions.

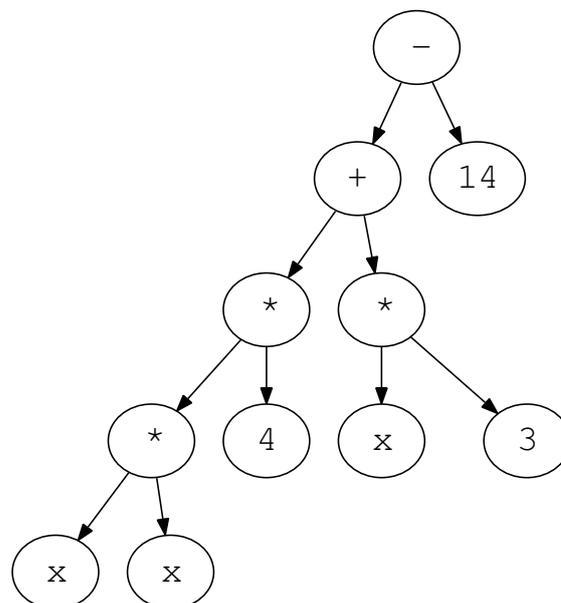
Since that in *Common Lisp* algorithms can be written as *s-expressions* and that *s-expressions* themselves are represented as lists and trees, algorithms and iterative computation processes can be generated automatically by creating well-formed trees, sub-trees



**Figure 1: An *OpenMusic* patch representing the polynomial  $4x^2 + 3x - 14$ . In raw *Common Lisp*, the function would be written as the symbolic expression `(- (+ (* 4 (* x x)) (* 3 x)) 14)`.**

and lists from a given set of functions and terminals and specific instructions regarding each function's number of arguments and respective data-types. If the tree has one or more mathematical functions, one of the terminals is an independent variable  $x$  and numerical constants are created within a range of random numbers, the generated *s-expression* can be interpreted as a mathematical function  $f(x)$  that may be used in iterative processes, as an algorithm, to generate outputs from given inputs.

While the generation of blind random *s-expressions* would not be an efficient method to cover the search space of all possible *s-expressions* for a given set of func-



**Figure 2: A tree representation of the same polynomial**

tions, range/type of terminals, and tree depth, GP techniques are far more effective in the heuristic search for expressions that optimally describe the behavior of the variables. Particularly, *Symbolic Regression* (SR) proves to be handy to seeking models (represented as a *s-expressions*) that best fulfill the requirement of optimally describe a set of two or more variables.

SR is a GP method of *sequence induction* that

*involves finding a mathematical expression, in symbolic form, that provides a good, best, or perfect fit between a given finite sampling of values of the independent variables and the associated values of the dependent variables. That is, symbolic regression involves finding a model that fits a given sample of data.* [28, p. 11-12].

SR has been used in a number of different applied fields such as economics, chemistry and physics to retrieve mathematical models from experimental data [34, 16, 40, 4]. Unlike other regression analysis methods used as statistical process for estimating relationships among variables (linear, quadratic, polynomial or logistic regressions, for instance) [15, 37], in symbolic regression there is no assumption about the kind of function that will model the input data: generally, only the mathematical operators and the type and range of random constants to be generated are pre-specified.

Furthermore, SR processes not necessarily need to use only usual mathematical operators as functions: in the same way that it is possible to use sum, subtraction and trigonometric operations as computational functions to create *s-expressions*, it is also possible to use custom defined functions or macros to generate compound computer algorithms written as *s-expressions*. Indeed, this flexibility is ordinarily used to avoid errors during the fitness evaluation process: the subtraction operation, for example, is usually replaced by a custom-defined “protected division” function “%”, that typically yields 1 when the divisor is 0. Similarly, it is also possible to structure symbolic regression processes that, instead of dealing with individual numbers and with functions that expect their arguments to be individual numbers, work with other data-types such as arrays or matrices, for example.

## 5. The OMGP library

The *OpenMusic Genetic Programming* (OMGP) is a new library for the *OpenMusic* (OM) CAM environment that implements *Symbolic Regression* by means of GP techniques. The library is based on the original proposition by John Koza [28, p. 237] and, in its current version (0.1), the library use works in three steps: (1) parametric configuration, (2) search, and (3) best function retrieve/application.

In the first stage, a series of parameters must be configured to set-up the target data-points (to be used in the fitness evaluation of each generated *s-expression* and the SR algorithm properties). This is done with the functions:

- `omgp-functions_and_terminals_vars`,
- `omgp-symbolic_regression_vars`
- `omgp-settargetsarray`.

There are also three auxiliary functions that may be used to inspect or set the parameter values:

- `omgp-getvars`,
- `omgp-setfvars`

- `omgp-mapset fvars`

In practical situations, these functions may be useful, for example, to check specific values or to create custom configurations.

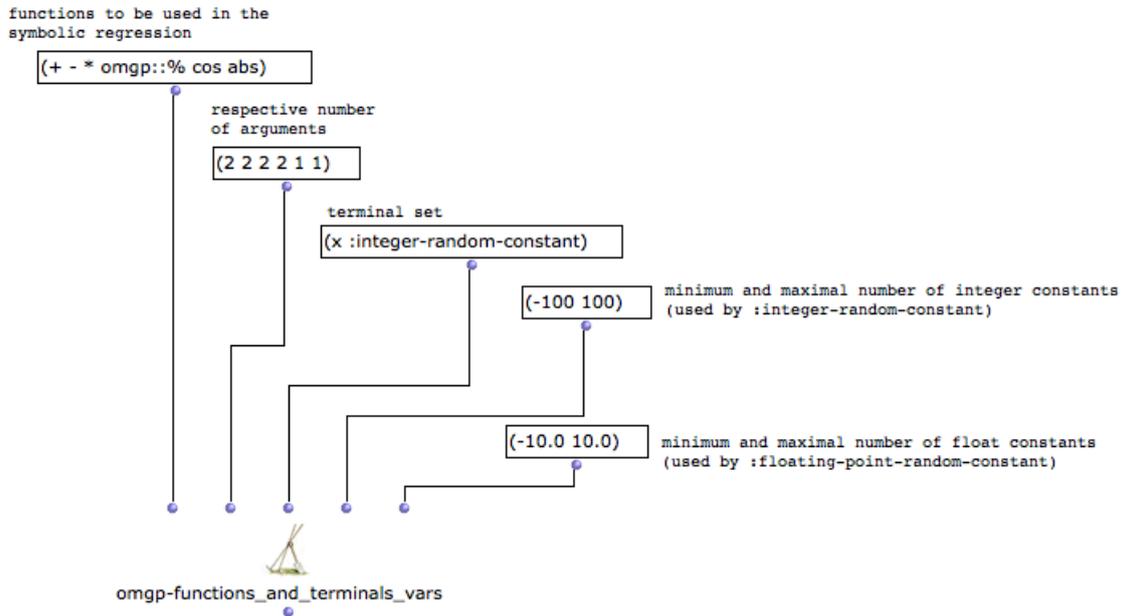


Figure 3: Function `omgp-functions_and_terminals_vars`

The function `omgp-functions_and_terminals_vars` is evaluated to indicate the functions repertoire or vocabulary to be used by the SR algorithm and the respective number of arguments that these functions expect to be given as input. Here the user also specifies the “terminals” of the *s-expression* tree, i.e., the kind of random constants to be generated and the independent variable symbol. Given that the user can choose between integer and floating-point numbers generators to create the constants, it is also in this function that one should specify the range of values to be used to the pseudo-random number generators. Here, it is also possible to define invariant constants (like  $e$  or  $\pi$ ) that could potentially be used in the final expression.

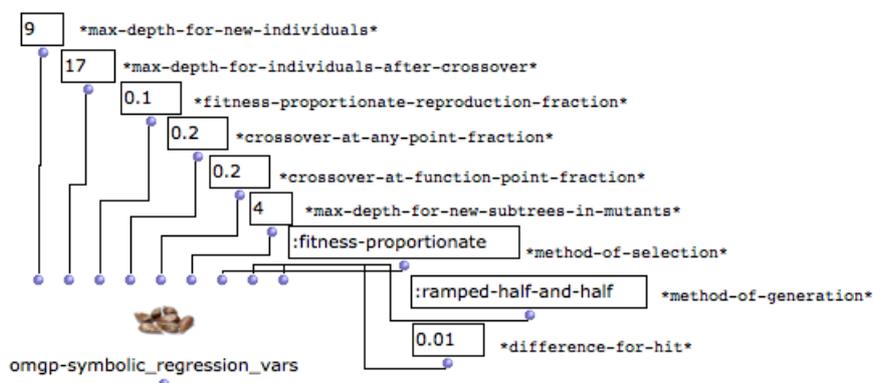
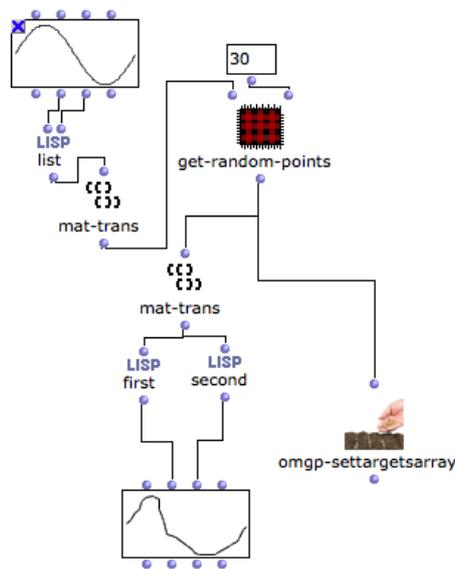


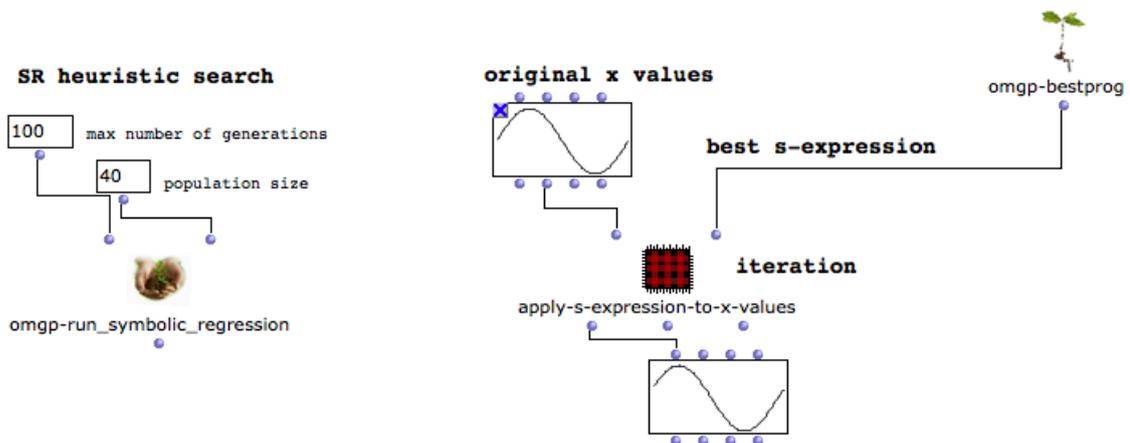
Figure 4: Function `omgp-symbolic_regression_vars`

The function `omgp-symbolic_regression_vars` sets parameters that are specific related to the SR algorithm and the genetic programming paradigm. For instance, one must specify the maximum depth of new individuals *s-expressions* trees, the maximum depth of individuals created through *crossover*, fractions related to the percentage

of the population that will experience *crossover* at any point or at the function points of the tree, the depth of new subtrees, the method of generation of new branches and other values.



**Figure 5: Getting random point of a sinusoidal function to use as target values using the function `omgp-settargetarray`**



**Figure 6: Functions `omgp-run_symbolic_regression` and `omgp-bestprog`, used to run the SR heuristic search and to retrieve/apply the best fitted *s-expression* to a series of independent variable values.**

The function `omgp-settargetarray` is used to store pairs of dependent and independent variables in an array that will be used to evaluate the *fitness* of a given *s-expression*. The fitness of the expression will be equal to its error to yield the expected dependent variable values when they are evaluated with the respective independent variable. Clearly, then more the *raw fitness* (sum of absolute values of the difference between obtained  $f(x)$  value for all target  $X$  independent variables) is closer to 0, then more the generated *s-expression* is adapted to solve the problem.

The second step in using the OM-gp library is to run the SR algorithm by using the function `omgp-run_symbolic_regression`. This functions expects two arguments: the numbers of generations to be evaluated and the size of the initial population.



could at same time be controlled and retrieve optimal results to OM through OSC commands, for example.

Additionally, there are a number of strategies that could potentially boost both the computation time and the accuracy of the results. Specially, we plan to rewrite the functions used to run SR process to make them more compact and efficient, using specific data-structures to represent and manipulate data, as well as declared types for arguments/functions (techniques that previous works have shown to be particularly effective to make CL applications run faster [43]).

As these issues are related to different characteristics of the system, we are currently rewriting the whole library so that it works, as well, as an external application in SBCL. The server will work on input data autonomously and receive commands and retrieve data by using network OSC commands and, when necessary, other streams (such as text files, for example). As the same code that will be used in this external app may be used inside OM (as a library), the idea is to benchmark the SR system being run both in OM/LispWorks and in the independent SBCL server, adapting the code to work more accurately and fast.

## 7. Conclusion

We have presented a new tool to be used in CAM environments that brings together modern analysis/composition techniques and powerful heuristic and artificial intelligence instruments by applying *Symbolic Regression* and *Genetic Programming* to the problem of formalize *algorithmic models* of musical processes and procedures. In this paper we have exposed the main concept and the implementation of a new OM library that, despite being in its initial stage and demanding further development, proves to be promising to address the proposed objective. In future works we intend to apply and test this new tool to specific analytical and compositional projects, potentially expanding the features and the effectiveness of OMGP and evaluating its performance in different situations.

## References

- [1] Agon, C., Assayag, G., and Bresson, J., editors (2006a). *The OM composer's book 1*, volume 1 of *Musique-sciences*. IRCAM-centre Pompidou ; Delatour, Paris : [Sampzon].
- [2] Agon, C., Assayag, G., and Bresson, J., editors (2006b). *The OM composer's book 2*, volume 2 of *Musique-sciences*. IRCAM-centre Pompidou ; Delatour, Paris : [Sampzon].
- [3] Agustín-Aquino, O. A., Junod, J., and Mazzola, G. (2015). *Computational Counterpoint Worlds: Mathematical Theory, Software, and Experiments*. Springer.
- [4] Alaa F. Sheta, Sara Elsir M. Ahmed, and Hossam Faris (2015). Evolving stock market prediction models using multi-gene symbolic regression genetic programming.
- [5] Ariza, C. (2005). *An open design for computer-aided algorithmic music composition: athenaCL*. Dissertation.com, Boca Raton, Fla.
- [6] Biles, J. (1994). GenJam: A Genetic Algorithm for Generating Jazz Solos. pages 131–137.
- [7] Bresson, J. (2007). *La synthèse sonore en composition musicale assistée par ordinateur : modélisation et écriture du son*. phdthesis, Université Pierre et Marie Curie - Paris VI.
- [8] Bresson, J., Agon, C., and Assayag, G. (2009). Visual Lisp/CLOS programming in OpenMusic. *Higher-Order and Symbolic Computation*, 22(1):81–111.

- [9] Bresson, J., Agon, C., and Assayag, G. (2011). OpenMusic: Visual programming environment for music composition, analysis and research. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, pages 743–746, New York, NY, USA. ACM.
- [10] Burton, A. R. and Vladimirova, T. (1999). Generation of Musical Sequences with Genetic Techniques. *Computer Music Journal*, 23(4):59–73.
- [11] Buteau, C. and Anagnostopoulou (2011). Motivic Topologies: Mathematical and Computational Modelling in Music Analysis. In *Mathematics and computation in music third international conference, MCM 2011, Paris, France, June 15-17, 2011: proceedings*. Springer, Berlin.
- [12] Buxton, W. (1977). A composer's introduction to computer music. *J. of New Music Res. Journal of New Music Research*, 6(2):57–71.
- [13] Carpentier, G., Tardieu, D., Assayag, G., Rodet, X., and Saint-James, E. (2007). An Evolutionary Approach to Computer-Aided Orchestration. In Giacobini, M., editor, *Applications of Evolutionary Computing*, number 4448 in Lecture Notes in Computer Science, pages 488–497. Springer Berlin Heidelberg.
- [14] Charles De Paiva Santana, J. M. (2015). Towards a Borgean Musical Space: An Experimental Interface for Exploring Musical Models.
- [15] Draper, N. R. and Smith, H. (1981). *Applied regression analysis*. Wiley, New York.
- [16] Duffy, J. and Engle-Warnick, J. (2002). Using symbolic regression to infer strategies from experimental data. In Chen, P. S.-H., editor, *Evolutionary Computation in Economics and Finance*, number 100 in Studies in Fuzziness and Soft Computing, pages 61–82. Physica-Verlag HD.
- [17] Essl, K. (2007). Algorithmic composition. In *The Cambridge companion to electronic music*, Cambridge companions to music, pages 107–125. Cambridge University Press, Cambridge ; New York.
- [18] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition.
- [19] Hiller, L. and Isaacson, L. M. (1959). *Experimental music; composition with an electronic computer*. McGraw-Hill, New York.
- [20] Holland, J. H. (1975). *Adaptation in natural and artificial systems an introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press, Ann Arbor.
- [21] Horner, A. (1996). Double-Modulator FM Matching of Instrument Tones. *Computer Music Journal*, 20(2):57–71.
- [22] Horner, A., Beauchamp, J., and Haken, L. (1993). Machine Tongues XVI: Genetic Algorithms and Their Application to FM Matching Synthesis. *Computer Music Journal*, 17(4):17–29.
- [23] Horner, A. and Goldberg, D. (1991). Genetic Algorithms and Computer-Assisted Music Composition. In *Proceedings of the 1991 International Conference on Genetic Algorithms and Their Applications*, pages 337–441.
- [24] Jacob, B. L. (1995). Composing with genetic algorithms. In *Proceedings of 1995 International Computer Music Conference*, Alberta. ICMA.
- [25] Jacobs, J. P. and Reggia, J. (2012). Evolving musical counterpoint: The chronopoint musical evolution system. *arXiv:1207.5560 [cs]*.
- [26] Keller, D. and Ferneyhough, B. (2004). Analysis by Modeling: Xenakis's ST/10-1 080262. *Journal of New Music Research*, 33(2):161–171.
- [27] Knuth, D. (1968). *The art of computer programming*. Addison-Wesley, Reading (Mass.) Menlo Park (Calif.) London etc.
- [28] Koza, J. R. (1992). *Genetic programming: on the programming of computers by means of natural selection*. MIT Press, Cambridge, Mass.

- [29] Laske, O. (1981). Composition Theory in Koenig's Project One and Project Two. *Computer Music Journal*, 5(4):54–65.
- [30] Laurson, M., Kuuskankare, M., and Norilo, V. (2009). An overview of PWGL, a visual programming environment for music. *Computer Music Journal*, 33(1):19–31.
- [31] Malt, M. (2000). *Les Mathématiques et la Composition Assistée par Ordinateur: concepts, outils et modèles*. Thèse de Doctorat, École des Hautes Études en Sciences Sociales, Paris.
- [32] Malt, M. (2006). Concepts et modèles, de l'imaginaire à l'écriture dans la composition assistée par ordinateur. *Musique, instruments, machines*, (19):213–234.
- [33] Manzolli, J., Moroni, A., Gudwin, R., and Zuben, F. v. (2000). Vox Populi: An Interactive Evolutionary System for Algorithmic Music Composition. *Leonardo Music Journal*, 10(1):49–54. <p>Volume 10, 2000</p>.
- [34] McKay, B., Willis, M., and Barton, G. (1997). Steady-state modelling of chemical process systems using genetic programming. *Computers & Chemical Engineering*, 21(9):981–996.
- [35] Mesnage, M. and Riotte, A. (1988). Un modèle informatique d'une pièce de Stravinsky. *Analyse Musicale*, 10:51–66.
- [36] Miranda, E. R. and Biles, A. (2007). *Evolutionary computer music*. Springer, London.
- [37] Rawlings, J. O., Pantula, S. G., and Dickey, D. A. (1998). *Applied regression analysis: a research tool*. Springer texts in statistics. Springer, New York, 2nd ed edition.
- [38] Riotte, A. (1996). Formalismes, modèles: un nouvel atout pour la composition et l'analyse. *Musurgia*, 3(3):90–105.
- [39] Riotte, A. (2003). Quelques réflexions sur l'analyse formalisée. *Musurgia*, 10(1):61–71.
- [40] Schmidt, M. and Lipson, H. (2009). Distilling free-form natural laws from experimental data. *Science (New York, N.Y.)*, 324(5923):81–85.
- [41] Spector, L. and Alpern, A. (1994). Criticism, Culture, and the Automatic Generation of Artworks. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (Vol. 1)*, AAAI '94, pages 3–8, Menlo Park, CA, USA. American Association for Artificial Intelligence.
- [42] Taube, H. (2004). *Notes from the metalevel : introduction to algorithmic music composition*. Taylor & Francis Group, London New York.
- [43] Verna, D. (2006). How to make lisp go faster than c. *IAENG International Journal of Computer Science*, 32(4):499–504.
- [44] Xenakis, I. (1963). *Musiques formelles: nouveaux principes formels de composition musicale*. Richard-Masse, Paris.
- [45] Zils, A. and Pachet, F. (2004). Automatic Extraction of Music Descriptors from Acoustic Signals Using EDS. Audio Engineering Society.

# Towards a Virtual Portable Radio Platform for Audio Transmission

Estêvão da Fontoura e Evandro Manara Miletto

<sup>1</sup>Instituto Federal do Rio Grande do Sul - Campus Porto Alegre  
Av Cel Vicente, 281 - Centro - 91.030-041 - Porto Alegre - RS - Brazil

estevao.fontoura@farroupilha.ifrs.edu.br, evandro.miletto@poa.ifrs.edu.br

**Abstract.** *Audio transmission in restricted area can bring different possibilities and be explored for education, arts and entertainment purposes. This paper presents an ongoing project that aims to develop an open-based technology device to broadcast audio to portable devices such as smart phones and tablets in a small range. Gambiarradio is the preliminary name coined to describe this project, which implements a dynamic local radio station to transmit audio for portable devices using wi-fi network. The goal of this project is to take advantage of low-cost equipment, based on free software philosophy and make it available to be used in education, promoting the inclusion of blind students and enabling live performances, for instance. The main features of the project, the first steps and the preliminary results are presented, as well as the prospective work.*

## 1. Introduction

New technologies are definitely transforming our lives and the way we interact with people in different environments. Specially in the last fifteen years, in terms of scientific research for the development of new technologies for musical expression <sup>1</sup> and artistic performance, researchers and musicians from all over the world have shared their knowledge and recent work on new performing interface design.

Inspired on that new possibilities, this project emerged from the scratch having in mind the challenge of creating a solution for a band to rehearsal at home without noise. One alternative was the idea of transmitting sound via FM radio waves instead of having loud speakers, with something similar to the idea of the ubiquitous music [Damian Keller 2014]. It was the birth of the project that would become a multimedia artistic performance entitled *Gambiarradio* (GR)<sup>2</sup>, presented for the first time at *Escola Caseira de Invenções* (Home-made School for Invention), part of the educational project of the 9<sup>th</sup> edition of *Mercosul Biennial*<sup>3</sup>, in 2013. In 2014, inside the prize winning *Projeto Casa Grande*<sup>4</sup>, in partnership with the maker group *Matehackers Hackerspace*<sup>5</sup>, we have established the GR, which got a new dimension with insertion of a tiny low-cost affordable computer, *Raspberry Pi*. Today new challenges arises as master degree project: turning GR into a multipurpose tool.

---

<sup>1</sup><http://www.nime.org/past-nimes/>

<sup>2</sup><http://www.estevaodafontoura.com>

<sup>3</sup><http://www.fundacaobienal.art.br/site/>

<sup>4</sup><http://www.projetocasagrande.org/>

<sup>5</sup><https://matehackers.org>

This paper aims to present this ongoing project called *Gambiarradio*, describing its main characteristics, some architecture details and the next steps to be achieved for a full experiment and evaluation test with real users in a real context of use. This manuscript is organized with this introduction, one section where we describe the GR project, and a final section with the prospective work.

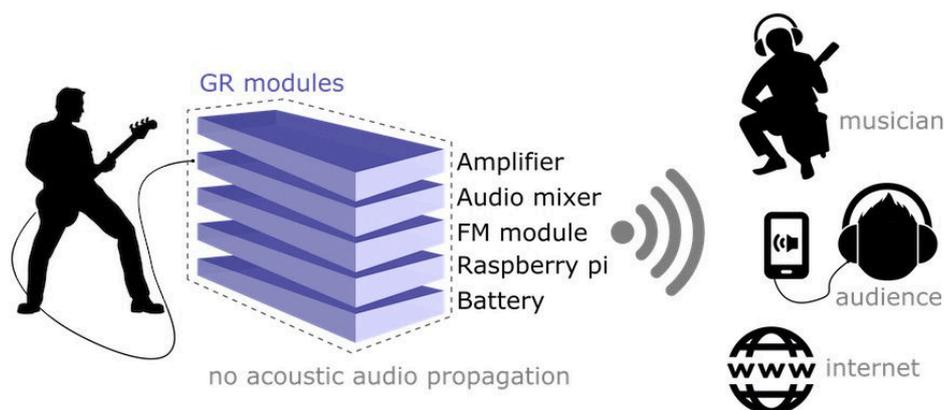
## 2. What is Gambiarradio?

GR was initially a set of different tools to transmit sound from an audio mixer via FM radio waves. The idea was to enable live performances that are audible only to the listeners. This is done by plugging only electric and electronic instruments into the audio mixer and spreading its signals wirelessly. The idea originated from a trio of amateur musicians who wanted to play music just for fun without worrying about if they were disturbing the neighbors.

After they started to use the FM transmitter they called the rehearsals / performances *Silent Gigs*. After performing the *Silent Gigs* for one year, they understood that it had the potential to be a platform for musicians to improvise in long jams without bothering anyone, giving the public the power to decide if they wanted to listen to the sound or not. In that way, they gave birth to GR: a democratic tool for live music performances. Therefore, what was first simply a home solution for the trio became a platform for long improvised gigs in which random musicians could play together and exchange instruments without knowing each other.

### 2.1. Preliminary architecture

The GR system is based on a portable free software/hardware philosophy. First versions were divided in too many parts, including external mixer and amplifier, so they weren't portable. The goal now is to achieve a modular and portable structure, with a base module composed by the battery and subsequent modules assembled by demand<sup>6</sup>. Figure 1 illustrates an overall architecture of the system.



**Figure 1.** This figure shows the actual idea for the modular design of the next version of Gambiarradio, and how it works in a performance.

<sup>6</sup>This modular logic for the equipment was proposed by Joel Grigolo, from *Matehackers*, which keep being partner in this project

GR allows to use different configuration, adding extra modules for more complex tasks. Battery, Mixer and FM radio transmitter modules, for instance, will enable the use for a simple FM transmission. If internet transmission is necessary or desired adding *Raspberry Pi* module is mandatory. Other possibilities are explored further.

After the group be invited to present it as a performance at the 9<sup>th</sup> edition of the *Mercosul Biennial*, it was noticed that more potency for bigger places was needed. In 2014, during the realization of the *Projeto Casa Grande*, the group got help from *Matehackers*, which suggested the use of *Raspberry Pi*, in order to find a better solution to improve the system with more potency and better quality.

*Raspberry Pi* was then chosen because it matches with the project philosophy, and the group took advantage of *Matehackers* previous experience with it and from other projects such as *Easy As Pi*, where free workshops at public libraries were held to promote computational thinking and developing skills needed to work with technology, proving the educational vocation of *Raspberry Pi* [Wing and Meyers 2014].

## 2.2. The GR first appearance

In 2007 an one-day performance event called *Musicircus* created by John Cage took place into the 7<sup>th</sup> *Mercosul Biennial*. First performed in November 1967 at the Stock Pavilion at the University of Illinois, the idea behind John Cage's *Musicircus* is "nothing more than an invitation to a number of musicians to perform simultaneously anything in any way they desire" [Rønningsgrind 2012]. Another description can be found at the review "Musicircus Rocks Stock Pavilion" by Bruce Zumstein. The Daily Illini, Nov.18, 1967.

Then, in 2013, GR was used in a performance at the *Mercosul Biennial*, as part of the *Escola Caseira de Invenções* (Home-made School for Invention) program. The theme was to present workshops and experimental performances focused on educational relations between art, science and technology. In some way *Musicircus* can be considered an influence for GR: there wasn't even the embryonic idea of GR, which appeared in 2011. Therefore, watching *Musicircus*, the first possibility of thinking about participating in the *Mercosul Biennial*, the idea emerged and GR was born later on.

## 2.3. Potential applications of GR

With the possibility of thinking about GR as a tool also for educational purposes, it was decided to develop it into only one equipment: a multimedia portable radio platform. However, in this new version of GR, which is under development, some other possibilities can be also explored, such as: a) temporary local FM radio transmission (School radio), b) Simultaneous translation, c) audio description, and d) experimental music performances.

- **a) GR as temporary local radio transmission.** The idea is to make available several possibilities for students, such as a) a school radio during the breaks and lunch time to listen to music or send some announcements, b) live transmission of a new song that a group of students just created, playing live instruments as guitars and vocals, c) groups of students having their own temporary radio station for social or educational purposes, d) to make a pronouncement by the director to all the campus community.
- **b) GR as a simultaneous translation device.** Simultaneous translation is a very expansive service for a small public school to afford. The possibility of having its own system of transmission would encourage small school communities

to organize international events, as meetings, seminars or conferences and even web conferences. GR can be used in this context by taking advantage of a low-cost solution. With this equipment any school would self-organize a simultaneous translation just needing a language teacher capable to translate.

- **c) GR as audio description.** Considering the inclusion of people with disabilities, which is supported by the Brazilian Federal law (9394/96, Art. 4), every school in Brazil has the possibility and responsibility of receiving special students, such as blind or deaf. In this situation, GR can be used as a tool that promotes inclusion, enabling the teacher to show a video, for example, to his or her class and transmit via FM radio waves the correspondent audio description of that video only for the blind student who could use a smart phone as receiver.
- **d) GR as experimental music performances.** The Brazilian Government has recently approved educational law with the commitment to teach music as a subject of art classes. GR can be an useful tool to provide different approach in this matter. Silent electronic performances, as the original purpose of GR, with students playing instruments such as keyboards and MIDI controllers; explore the portable radio transmission property, making live music performances in movement, such as a guitar player walking through school corridors playing his guitar noiselessly, would be some ways of using GR at art classes.

### 3. Prospective work

This paper presented an ongoing work that aims to produce a portable free device to broadcast live audio, which can be used for several purposes. Since the development of the new modules can be achieved and tested, the project is still at the begin, looking at the very first steps. Some experiences will be considered with the previous versions of GR, when it was focused as a platform for improvisation in music jams and for artistic performances.

Next steps will include studying and implementing the new modules, printing a case for this device as make some preliminary experiments to validate the project. Considering it will be all based on free hardware and software licensed under a Creative Commons, a free device accordingly, it has a very big potential to contribute in several fields, making audio and music more accessible and achievable. At last, the authors want to produce the GR kit to put it available to anyone interested in make it, inside the *Do It Yourself* philosophy.

### References

- Damian Keller, Victor Lazzarini, M. S. P. (2014). *Ubiquitous Music*, volume 1. Springer, Springer International Publishing AG, Gewerbstrasse 11 6330 Cham, Switzerland, 1 edition.
- Rønningsgrind, G. (2012). *Meaning, Presence, Process. The Aesthetic Challenge of John Cage's Musicircus*. PhD thesis, Norwegian University of Science and Technology, NTNU - Trondheim.
- Wing, D. and Meyers, E. (2014). Easy as pi: Designing a library program to support computational thinking in preteens. *BCLA Browser: Linking the Library Landscape*, 6(3).

# A modular platform for a subtractive synthesizer on Arduino Due

Rodolfo Pedó Pirotti<sup>1</sup>, Marcelo Johann<sup>1</sup>, Marcelo Pimenta<sup>1</sup>

<sup>1</sup>Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)  
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

rodolfo.p.pirotti@gmail.com, {johann, mpimenta}@inf.ufrgs.br

***Abstract.** In this paper we present an initial design to implement a platform of a subtractive synthesizer by using the Arduino Due board. We intend to develop a modular system, which can be used with a real musical keyboard or as a MIDI device, to play and generate musical notes by using subtractive synthesis, including different types of waveforms, envelope generators, filters and configurations. To achieve a high quality output signal, we intend to attach simple but effective external DACs and analog output stages. By reaching this goal, we will be able to provide a powerful low-cost platform for subtractive synthesis.*

## 1. Introduction

Subtractive synthesis is a method for sound generation that became very popular in the 60's and 70's, with the original implementation being done by using electronic analog circuits. Through the years, "Virtual Analog" became available on digital synthesizers to simulate the sound resulting from the old subtractive analog synthesizers, but most of these machines were and are still expensive, obtained by significant design efforts, making use of the most up-to-date technologies [Urban 2002] [Wiffen 1997].

There are some open and Do-It-Yourself (DIY) projects which implemented or investigated subtractive synthesis on low cost platforms, such as Mozzi [Barrass 2015], Scalable Polyphony-MIDI Synthesizer [Huovilainen 2010], AVRSynth [Biddulph et al 2015], and others. But we intend to combine a high quality audio output on a wide available and low cost popular platform (in this case, Arduino Due), which we did not find in the above examples.

The Arduino Due is a widely available low cost board with a 32-bit ARM processor running at 84 MHz, introduced in 2012. Its usage nowadays is becoming very popular for applications that require interaction with the real world and people, including audio applications [Johann 2015].

### 1.1 Objective

With the goal to create a flexible, modular and powerful platform for subtractive synthesis, achieving a high quality output signal on a low cost and wide available platform, we started the design and development of a subtractive synthesizer using the Arduino Due board.

In this paper, we present the initial design and development, and our goals for the finished platform. This is an ongoing project for which only initial tests and code is already completed.

## 2. Subtractive Synthesis

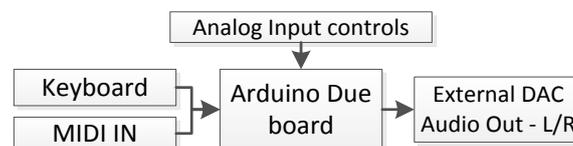
Subtractive synthesis is a type of synthesis based on an input signal containing a lot of harmonic content. Examples are the square and sawtooth waves. On this signal, a filter is applied, which selectively removes some of the harmonic content. And so, we have the name of “subtractive synthesis”.

On a subtractive synthesizer, the first stage is an oscillator which generates the waveform at the desired frequency. Usually, this oscillator is called VCO (voltage controlled oscillator). On the output of the oscillator, a filter is connected. This can be a low-pass filter, high-pass filter, band-pass filter or a band-stop filter. The filter is usually called VCF (voltage controlled filter). A low-frequency oscillator (LFO), connected to VCO, VCF or Amplifier blocks can also be used with the goal to create amplitude and frequency modulation on the output signal [Urban 2002].

## 3. Hardware Architecture

To implement a powerful platform for subtractive synthesis, special care is needed on all hardware blocks. Real time input controls are needed to modify the sound parameters on real time. In the output stage, a high quality digital to analog converter (DAC) was chosen, to achieve a high quality audio output signal.

For musical note input, the hardware can be used with two different options: with a musical keyboard, or as a MIDI host. Figure 1 shows a block diagram of the hardware prototype.



**Figure 1. Hardware prototype blocks**

The Audio Out block is implemented with an external DAC to achieve a higher quality than using Arduino’s internal DAC. Like in [Johann 2015] a TDA1543 16-bit DAC is connected to Arduino board SAM3X8E processor, which has an I2C interface.

## 4. Software modules and design

The software design must use good software engineering approaches, such as low coupling, high cohesion and well defined interfaces, in order to achieve the goal of a modular platform, which includes the possibility of easily change some blocks of the synthesizer, such as the filter type, effect, etc., but keeping the code fast enough to run properly on Arduino Due processor.

### 4.1 Waveform sample and oscillator

This block is responsible for the generation of the desired audio signal, at the desired frequency, on the digital domain. It uses a numeric controller oscillator (NCO) and a phase to amplitude converter (PAC) to generate the output data.

The NCO uses a phase accumulator to increment the phase of a waveform during time. The phase value is then used to index a second block, the PAC, which is implemented with a lookup table containing the waveform sample. This approach

allows us to easily change the waveform sample, by changing the lookup table. A fixed point approach was chosen instead of a floating point approach for faster processing.

#### 4.2 Envelope generator

The envelope generator module is responsible for generating an ADSR curve (attack-decay-sustain-release), which can be applied to any stage of the synthesizer. As an independent module, the envelope generator (EG) produces an output value between 0 and 1 (using fixed point format), which should be multiplied by the output of the block. For an example, if we want to apply EG on the output of the filter signal, the output value of EG is multiplied by the filter output data. As a separate module, more than one EG can be instantiated and applied to different software blocks.

#### 4.3 Filter

The filter block consists of a low-pass filter, with cutoff frequency and resonance as input parameters, which can be changed in real time using input analog controls.

#### 4.4 Effects

On the final stage of the synthesizer, an effect block is implemented. The first prototype has only a delay with feedback. Other types of effects will be analyzed in the future.

#### 4.5 Other features

A low frequency oscillator (LFO) is implemented, which can be connected to modulate the frequency of the main oscillator (adding a vibrato); a soft pitch slide when changing notes (portamento) is implemented and can be turned on, off and its time configured; the selection of the note priority (last, highest, lowest) and trigger mode, to behave like old classic analog synthesizers [Reid 2000], is also available.

#### 4.6 Module connections

All blocks are implemented as independent modules. This allows us to continuously develop different modules for each stage, changing and adding different characteristics to our synthesizer. Figure 2 shows a basic module block diagram.

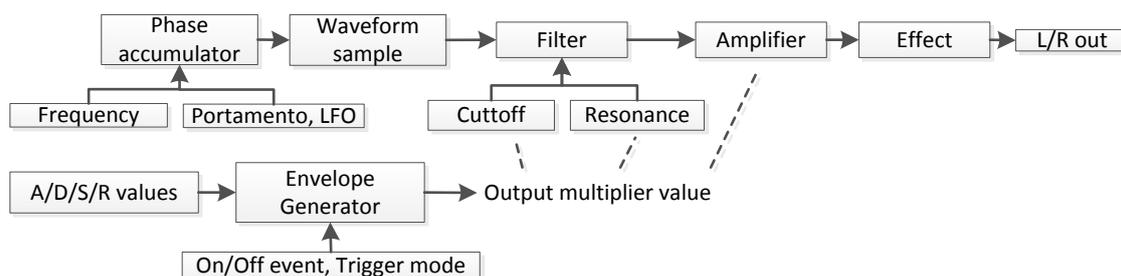


Figure 2. Software module blocks

### 5. Next Steps

As of current development, an aliasing is easily noticeable on higher frequency notes, requiring us to implement an antialiasing method for waveform generation, such as on [Pekonen et al 2008] and [Valimaki et al 2007]. Faster algorithm for the note priority selection feature (sort and find algorithm) and the low pass filter is also required.

A higher order low-pass filter with a higher resonance is desired. We intend to research mathematical models of low pass filters used on classic analog synthesizers

and implement on our platform. To improve control capabilities, when used as a MIDI host it is possible to implement the decoding of some MIDI messages to change internal parameters, as an option to increase real time control with no need to add external electronic components to the board.

As of sound quality, a deep analysis of audio output using instruments such as oscilloscope and spectrum analyzer, plus some ABX testing is also planned [Johann et al 2013].

## 6. Conclusions

In this paper we presented an initial design and prototype of a modular, powerful and low-cost subtractive synthesizer using an Arduino Due. Initial prototype tests have demonstrated that it is feasible to implement this type of synthesizer using a low-cost Arduino Due board.

We intend to continue the development of this platform, including other features, such as: different waveform sample selection, different filter selection, more effect types, and also the possibility for different block connections between modules, just like old analog synthesizer used to work.

## 7. References

- Barrass, Tim (2015) “Mozzi sound synthesis library for Arduino”.  
 <<http://sensorium.github.io/Mozzi/>>
- Biddulph, L., Ziembicki, J. (2015) “AVRSynth – monophonic virtual analogue synth”.  
 <<http://www.elby-designs.com/contents/en-us/d5.html>>
- Huovilainen, Antti (2010) “Design of a Scalable Polyphony-MIDI Synthesizer for a Low Cost DSP”. Master’s Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology. Aalto University School of Science and Technology. Espoo, Finland.
- Johann, M., Folle, L., Gomes, W., Cordoni, R. (2013) “Estabelecendo uma Metodologia de Avaliação Subjetiva de Qualidade de Áudio”. IV Ubimus. Porto Alegre, Brazil.
- Johann, Marcelo. (2015). “An additive synthesis organ with full polyphony on Arduino Due”. VI Ubimus. Växjö, Linnaeus University, Sweden.
- Pekonen, J., Valimaki, V. (2008) “Filter-based alias reduction for digital classical waveform synthesis”. IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV. 2008.
- Reid, Gordon (2000). “Synth Secrets, Part 18: Priorities & Triggers”.  
 <<http://www.soundonsound.com/sos/oct00/articles/synthsec.htm>>
- Urban, Ondrej. (2002). “Short overview of methods of sound synthesis.” The 32nd International Acoustical Conference - EAA Symposium "Acoustics Banská Štiavnica 2002". September, Slovakia.
- Wiffen, Paul. (1997) “Synth School, Part 1: Analogue Oscillators, Filters & LFOs”,  
 <[http://www.soundonsound.com/sos/1997\\_articles/jun97/synthschool1.html](http://www.soundonsound.com/sos/1997_articles/jun97/synthschool1.html)>
- Valimaki, V., Huovilainen, A. (2007) “Antialiasing Oscillators in Subtractive Synthesis”. Signal Processing Magazine, IEEE (Volume:24 , Issue: 2 ).

# Composição Algorítmica em Redes Complexas

Vitor Rolla<sup>1</sup>, Luiz Velho<sup>1</sup>

<sup>1</sup>Laboratório VISGRAF – IMPA  
Instituto Nacional de Matemática Pura e Aplicada  
Estrada Dona Castorina, 110  
Jardim Botânico, Rio de Janeiro - RJ, 22460-320

{vitorgr, lvelho}@impa.br

**Abstract.** *A relevant aspect of algorithmic composition is the ability to materialize music. Creating musical pieces is a complex endeavour that comprises both, technical knowledge and creativity. This paper proposes an algorithm to compose music based on complex network analysis. The computer represents the music of a composer within a network, and re-utilizes such knowledge to compose new music.*

**Resumo.** *Um aspecto relevante da composição algorítmica é a capacidade de materializar música. Criar peças musicais é um esforço complexo que compreende tanto conhecimento técnico, quanto criativo. Este trabalho propõe um algoritmo para compor músicas através da análise de redes complexas. O computador representa a música/obra de um compositor através de uma rede, que então é re-utilizada para compor novas músicas.*

## 1. Introdução

A geração automática de música pode ser alcançada através de um modelo que permita o computador aprender a música ou obra de um compositor, para então re-utilizar esse conhecimento e compor novas músicas. Este trabalho mostra um método para construir uma rede que representa uma (ou mais de uma) peça musical. Tal rede é posteriormente explorada por um algoritmo para criar novas composições.

A Seção 2 apresenta um modelo e um algoritmo para se criar uma rede a partir de uma música. A Seção 3 apresenta um algoritmo para composição automática de novas músicas a partir das redes que foram criadas. Os trabalhos futuros estão na última seção do documento.

O presente trabalho de pesquisa é a continuação do trabalho apresentado em [Rolla 2015]. Algumas das músicas de compositores clássicos utilizadas para criar as redes deste experimento, assim como uma seleção das melhores composições algorítmicas extraídas deste experimento, estão disponíveis em: <http://eden.dei.uc.pt/~vitorgr/MS.html>.

## 2. Música em Forma de Rede

Uma peça musical pode ser vista como um sistema dinâmico com valores discretos, ou seja, um conjunto finito de notas musicais em evolução no tempo [Cruz-Alcázar and Vidal 2008]. Para construir uma rede para uma determinada peça de música é necessário definir os nós e as arestas da rede.

No modelo apresentado nesse artigo, os nós podem ser notas individuais ou acordes (um conjunto de notas tocadas simultaneamente). Assim como em [Liu et al. 2010], as arestas são definidas cronologicamente, através das conexões entre notas enquanto a música é tocada. No modelo aqui proposto, as arestas representam e contém os dados de ritmo/tempo utilizados pelo compositor entre duas notas/acordes. Portanto, é possível que dois nós estejam ligados por diversas arestas diferentes. Isto é, através de diferentes tempos (ritmos).

Arquivo MIDI			
Tempo	Evento	Nota	Velocidade
0	Note On	60	99
0	Note On	64	98
120	Note Off	60	0
120	Note Off	64	0
360	Note On	65	56
480	Note On	69	67
540	Note Off	65	0
⋮	⋮	⋮	⋮

**Tab: 1. Simplificação de um arquivo MIDI.**

Basicamente, um arquivo em formato MIDI possui a informação apresentada na Tabela 1. Obviamente, as informações mais importantes para se criar uma rede a partir de um arquivo de música MIDI são o tempo ( $t$ ) e a nota, respectivamente as colunas um e três da referida tabela. Quando o arquivo é "tocado", as informações são extraídas conforme a evolução do tempo. Duas notas com o mesmo tempo formam acordes. A diferença de tempo entre notas/acordes (no caso, os nós da rede) é definido por  $\Delta_t$  (linha 11 do Algoritmo 1). Sempre que uma determinada aresta for utilizada pelo compositor da música, o peso ( $\psi$ ) dessa aresta é incrementado em 1 unidade (linha 17 do Algoritmo 1).

**Algoritmo 1: Construindo uma rede através de uma música (arquivo MIDI).**

<b>Entrada:</b> arquivo MIDI.	
<b>Saída:</b> grafo NetworkX.	
1	<b>inicio</b>
2	<b>para</b> cada linha do arquivo.mid <b>fazer</b>
3	tempo, nota, vel = map(linha());
4	<b>se</b> (últimoTempo == tempo) <b>então</b>
5	#O próximo nó do grafo é um acorde.
6	acorde.adicionar(nota);
7	últimaNota = nota;
8	continue; #Salte para a próxima iteração do loop "para"
9	<b>fim</b>
10	<b>se</b> (últimoTempo > tempo) <b>então</b>
11	$\Delta_t = \text{tempo} - \text{últimoTempo}$ ;
12	#Adicione aresta entre última nota (ou acorde) e nota atual.
13	grafo.adicionarAresta(últimaNota, nota, $\Delta_t$ );
14	#Ou grafo.adicionarAresta(acorde, nota, $\Delta_t$ );
15	<b>se</b> aresta já existente <b>então</b>
16	#Incrementar o peso da aresta
17	pesoAresta++;
18	<b>fim</b>
19	últimaNota = nota;
20	últimoTempo = tempo;
21	<b>fim</b>
22	<b>fin</b>
23	<b>fin</b>

O Algoritmo 1 apresenta o pseudocódigo do processo de criação de uma rede a partir de um arquivo MIDI. A ferramenta utilizada para criar as redes musicais deste

trabalho de pesquisa chama-se NetworkX [Hagberg et al. 2008]. A informação de velocidade mostrada na Tabela 1 representa a força com que uma nota individual foi tocada pelo instrumentista, isto é, o músico que estava tocando a peça musical no momento da geração do arquivo MIDI. Essa informação também é introduzida como um metadado do nó na rede criada com a ferramenta NetworkX. Tal informação pode ser posteriormente utilizada para enriquecer o processo de composição com base na rede musical.

### 3. Composição Algorítmica em Redes Complexas

A presente seção descreve um algoritmo para gerar novas músicas utilizando as redes que foram criadas. Dado que essas redes apresentam estrutura hierárquica não trivial, pode-se supor que novas composições baseadas nessas redes possam ter valor estético musical, ou mesmo representarem um estilo musical, ou quem sabe o estilo de um compositor. Depois de construir as redes musicais foi possível computar uma série de parâmetros. São eles:

$\varepsilon$	→	conjunto de arestas da rede.
$\varepsilon_i$	→	conjunto de arestas, onde $i$ é o nó origem.
$\varepsilon_{ij}$	→	conjunto de arestas, onde $i$ é o nó de origem e $j$ é o nó sucessor.
$N\varepsilon_i$	→	é o número de arestas em $\varepsilon_i$ .
$N\varepsilon_{ij}$	→	é o número de arestas em $\varepsilon_{ij}$ .
$\psi$	→	é o peso de uma aresta em particular.
$\Psi$	→	é a soma dos pesos de todas as arestas em $\varepsilon_{ij}$ .

A interpretação mais básica de uma composição musical é uma melodia, ou seja, uma sequência (ou fluxo) de notas musicais. A composição da melodia pode ser efetuada através de um procedimento de caminhada aleatória na rede. Por exemplo, o Algoritmo 2 começa em um nó da rede (nota ou acorde) selecionado aleatoriamente e move-se para um outro nó através de uma das arestas de saída, isto é, o nó seguinte é o nó que termina na aresta selecionada.

#### Algoritmo 2: Caminhada aleatória em uma rede musical.

```

Entrada: grafo NetworkX.
Saída: arquivo tipo MIDI.
1  inicio
2      #Selecionar um nó aleatoriamente.
3      nóAtual = random(grafo.NósdoGrafo));
4      #Imprimir dados no novo arquivo MIDI.
5      #Início de uma nova música. Portanto, tempo = 0.
6      tempo = 0;
7      imprimir(tempo, nóAtual);
8      para um dado número de iterações hacer
9          para cada aresta de saída do nó atual hacer
10             #Calcule as probabilidades de cada aresta de saída ser a escolhida.
11             #Calcule, conforme as equações 1 e 2.
12             p[] = grafo.CalcularProbabilidades(arestaSaída);
13         fin
14         #Seleção aleatória com base nas probabilidades p[].
15         arestaEscolhida = randomChoice(p[]);
16         #Atualiza o próximo nó (ou nota).
17         nóAtual = arestaEscolhida(nóDestino);
18         #Imprimir dados no novo arquivo MIDI.
19         tempo = tempo + arestaEscolhida( $\Delta_t$ );
20         imprimir(tempo, nóAtual);
21     fin
22 fin

```

A probabilidade ( $p$ ) de cada aresta de saída ser a aresta selecionada é proporcional a: (i) o número de vezes que o compositor utilizou a nota  $j$  logo após a nota (ou acorde)

$i$ , e (ii) ao seu próprio peso  $\psi$ . O peso da aresta representa o número de vezes que o compositor da peça musical utilizou o mesmo ritmo (tempo) entre dois nós (notas/acordes) da rede. A Equação 1 descreve o cálculo de  $p$  para cada aresta de saída. O primeiro termo na multiplicação é a probabilidade da nota  $j$  ser selecionada logo após a nota  $i$ , que é dada pelo número de arestas em  $\varepsilon_{ij}$  dividido pelo número de arestas em  $\varepsilon_i$ .

$$\forall (i, j) \in \varepsilon_i : \quad p = \left( \frac{N\varepsilon_{ij}}{N\varepsilon_i} \right) * \left( \frac{\psi}{\Psi} \right) \quad (1)$$

O segundo termo da multiplicação representa o quão importante é o tempo  $\Delta_t$  entre um par de notas  $(i, j)$ . Portanto, a probabilidade  $p$  também é proporcional ao peso ( $\psi$ ) específico de uma aresta dividido pela soma dos pesos ( $\Psi$ ) de todas as arestas do conjunto  $\varepsilon_{ij}$ . O cálculo de  $\Psi$  é dado pela Equação 2.

$$\Psi = \sum_{(i,j) \in \varepsilon_{ij}} \psi \quad (2)$$

#### 4. Trabalhos Futuros

Os trabalhos futuros para estender esta pesquisa são: (i) criação e análise de redes utilizando diferentes gêneros musicais, por exemplo: pop, rock, trance, tango, etc; (ii) melhoramento do Algoritmo 2 para levar em consideração aspectos rítmicos mais complexos, como por exemplo o conceito de motivo (*motif*).

Na Seção 2 foi apresentado um modelo para construção de redes que representam música. A seção 3 descreve um algoritmo baseado em caminhada aleatória para compor música artificialmente com base nas redes musicais. Composições algorítmicas extraídas deste experimento, estão disponíveis em: <http://eden.dei.uc.pt/~vitorgr/MS.html>.

#### Referências

- [Cruz-Alcázar and Vidal 2008] Cruz-Alcázar, P. P. and Vidal, E. (2008). Two grammatical inference applications in music processing. *Applied Artificial Intelligence*, 22(1-2):53–76.
- [Hagberg et al. 2008] Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., and Millman, J., editors, *Proceedings of the 7th Python in Science Conference*, pages 11 – 15, Pasadena, CA USA.
- [Liu et al. 2010] Liu, X. F., Tse, C. K., and Small, M. (2010). Complex network structure of musical compositions: Algorithmic generation of appealing music. *Physica A: Statistical Mechanics and its Applications*, 389(1):126 – 132.
- [Rolla 2015] Rolla, V. G. (2015). Knowledge representation and algorithmic composition with multidigraphs. In *8th International Workshop on Machine Learning and Music (MML2015) held in conjunction with the International Symposium on Electronic Art (ISEA2015)*, Vancouver, Canada.

# Compilador musical: Uso da tecnologia de compiladores para validação de música simbólica

Leandro E. F. Pessoa<sup>1</sup>, Flávio L. Schiavoni<sup>2</sup>

<sup>1</sup>Instituto de Ciências Exatas e Tecnológica  
Universidade Federal de Viçosa - Campus Florestal  
Rodovia LMG 818, km 06 – Cep: 35690-000 – Florestal – MG – Brasil

<sup>2</sup>Departamento de Ciência da Computação  
Universidade Federal de São João Del Rei (UFSJ)  
São João Del Rei – MG – Brasil

leandro.pessoa@ufv.br, fls@ufs.j.edu.br

**Abstract.** *The symbolic musical notation is characterized by the use of textual elements denoting as a musical figures and the relationship between these symbols. Such characteristics allude to programming languages that have an analytical methods condensed in a process known as compilation. These similarities inspire the application of the three constituent stages of compilation: lexical analysis, syntactic and semantic in the musical context as an aid to various applications. This paper presents an approach to the analysis of musical symbolic notation based on the current technology to developing compilers.*

**Resumo.** *A notação simbólica musical se caracteriza pelo uso de elementos textuais que denotam figuras musicais e a relação entre estes símbolos. Tais características fazem alusão a linguagens de programação, que possuem métodos de análise condensados em um processo conhecido como compilação. Estas similaridades inspiram a aplicação das três etapas constituintes da compilação: análise léxica, sintática e semântica no contexto musical como auxílio à diversas aplicações. Este trabalho apresenta uma abordagem para a análise da notação simbólica musical com base na atual tecnologia para desenvolvimento de compiladores.*

## 1. Introdução

A notação simbólica musical fornece para os sistemas computacionais uma abordagem formal que permite o estudo para resolução de desafios presentes na representação musical [Dannenberg, 1993]. Por meio de notações simbólicas, se torna possível e adequada atividades de análise que permitem validação de notação musical padrão.

Muitas das notações simbólicas possuem formato textual estruturado, se assemelhando com a morfologia de linguagens de programação de alto nível, onde a análise léxica, sintática e semântica se mostram necessárias para o processo de compilação presentes no *front end* [Aho et al., 2008], que como resultado final, apresenta um programa executável em uma linguagem de máquina capaz de ser interpretada em nível de hardware, produto final do *back end* do compilador.

O uso da tecnologia de compiladores para análise de uma peça musical representada em uma notação simbólica, se mostra útil devido a possibilidade de definição dos parâmetros necessários para uma especificação completa ou relevante de peças musicais

no escopo de sua notação. Se torna possível então a caracterização de sistema de escrita utilizado para representar uma peça musical em uma notação simbólica, onde existem símbolos a serem reconhecidos como lexemas pertencentes a gramática durante análise léxica. A gramática é constituída de figuras musicais em formato textual. O correto uso das figuras musicais se caracterizam pela sintaxe, que é determinada pelas relações formais que interligam os constituintes da música, atribuindo assim uma estrutura.

## 2. Proposta de implementação

Para a elaboração de um compilador para uma linguagem de programação de alto nível, devemos implementar primeiramente as três etapas que compõem o processo de análise da compilação. Existem ferramentas que são capazes de atuar na busca de elementos léxicos baseadas em padrões determinados através de expressões regulares que representam os padrões léxicos da linguagem.

$$[A-Za-z] [A-Za-z0-9]^*$$

**Figura 1: Expressão regular capaz de reconhecer identificadores.**

Durante a análise léxica, uma tabela de símbolos é gerada com os respectivos *tokens* que denotam cada lexema encontrado no código de entrada, e seus respectivos atributos.

Através da análise dos *tokens* que foram instalados na tabela de símbolos, a análise sintática se torna possível em etapa posterior. Durante a análise semântica, duas etapas principais devem ser executadas: a análise de contexto e é possível a geração de código intermediário.

Todas estas etapas podem ser realizadas com o uso de duas ferramentas conhecidas como Lex [Lesk and Schimidt, 2001] e Yacc [Johnson, 2001].

Diante das similaridades entre linguagens de programação e linguagens de notação simbólica musical, o uso de ferramentas para construção de compiladores se torna intuitivo e requer poucas adaptações. .

O ponto de partida deste trabalho são as linguagens simbólicas existentes como Lilypond [Nienhuys and Nieuwenhuizen, 2003], music21 [Ariza and Cuthbert, 2011, Cuthbert and Ariza, 2010], ABC [Oppenheim et al., 2010], MuseData [Hewlett, 1997], MIDI [Association et al., 1996], MusicXML [Good, 2001] entre outras. Cada formato um destes formatos de música simbólica trará um conjunto de lexemas e uma estrutura musical hierárquica para a aplicação das técnicas de compilador.

## 3. Compilando música simbólica

Assim como no processo de compilação de linguagens de programação, a estrutura básica do compilador é respeitada, onde existem duas seções evidentes: análise e síntese. Se a análise detectar que o programa de entrada está sintaticamente mal formado ou semanticamente incorreto, devem ser emitidas mensagens de erro de maneira que o usuário consiga corrigir estes erros. A parte de síntese constrói o programa objeto desejado a partir da representação intermediária e das informações na tabela de símbolos gerada pela análise. Respectivamente chamamos estas duas partes do compilador de **front-end** e **back-end**.

### 3.1. Front-end

As análises que o front-end executa para a compilação de música simbólica são semelhantes às empregadas nas linguagens de programação. Para reconhecimento de padrões da notação são especificadas expressões regulares. Como as análises do front-end são dependentes da linguagem de programação, usaremos como exemplo nesta seção a linguagem de notação simbólica Abc.

#### Análise Léxica

Temos então como lexemas a serem identificados na primeira etapa, informações da música como seu título e compositor. Temos também informações sobre a métrica, tempo e tonalidade que serão utilizadas para definir um contexto em posterior análise sintática e semântica. Temos ainda como lexemas, notas musicais e símbolos associados de acordo com a construção da notação Abc que representam sua duração, possíveis acidentes, dinâmica, ornamentos, adornos e articulação. Há também lexemas que representam linhas de compasso simples ou dupla, compasso tracejado e barra final, funcionam como lexemas delimitadores. E ainda lexemas que denotam repetições, marcas de interrupção e marcas de pedal.

```
X:0
K:C
T:Test Song
C:Hal 9000
L:1/4
M:4/4
Q:"Without afraid" 1/4=92
cegb|cega|
```

**Figura 2: Código de exemplo na linguagem Abc.**

O código apresentado na Figura 2 ao passar pelas etapas do front-end, em sua etapa inicial, lança e instala tokens correspondentes aos campos que definem o contexto para análise sintática e semântica, além de tokens que denotam título, compositor e andamento. O campo L:1/4 e o campo M:4/4, definem respectivamente a unidade de uma semínima como menor duração de uma nota no compasso e a quantidade máxima e mínima de 4 semínimas para completo preenchimento do compasso. O campo K: define a armadura de clave. Os demais campos são irrelevantes para análises no front-end.

Os demarcadores de compasso representados pelo carácter “|”, definem se a regra de compasso será ou não aplicada naquele compasso. Convenciona-se então que apenas o primeiro compasso da peça musical não precisa ter seu início delimitado pelo símbolo de compasso, permitindo assim anacruse e compassos acéfalos.

Como erros léxicos, temos a identificação de lexemas que não pertencem a gramática da notação simbólica. Se um lexema encontrado não se encaixa nos padrões definidos pelas expressões regulares, este é rejeitado e uma mensagem de erro é emitida.

### 4. Resultados

Apesar de ser de fácil utilização para a escrita, a linguagem ABC se mostrou bastante permissiva com relação a notação musical padrão. A verificação que é realizada pelos softwares existentes que processam esta linguagem fazem apenas verificações que são

inerentes a estrutura da linguagem simbólica, deixando a cargo do compositor /copista a verificação de regras da notação musical.

Após estudo da documentação da linguagem, foram elaboradas expressões regulares básicas para reconhecimento de elementos léxicos e também foi definido a coleção de erros sintáticos apresentado neste trabalho.

## 5. Conclusão

O uso da tecnologia de compiladores para a validação da notação musical padrão em linguagens simbólicas, se mostra atraente devido a similaridade entre a estruturação destas linguagens e as linguagens de programação. Algumas linguagens simbólicas se mostram bastante permissivas, e os compiladores destas linguagens apenas fazem verificação da notação simbólica, deixando a cargo do compositor respeitar a notação musical padrão. A abordagem exposta permite uma automatização na validação da notação musical padrão em processos de transcrição musical auxiliados por computador.

A abordagem aqui exposta permite a verificação desta representação simbólica em busca de erros ocasionados pela leitura ótica, que pode reconhecer erroneamente defeitos físicos como elementos de notação, se mostrando uma forte ferramenta para verificação em recuperação de partituras através de **OMR - (Optical Music Recognition)**.

### 5.1. Trabalhos futuros

O próximo passo desta pesquisa é a elaboração das demais etapas pertencentes ao front-end e back-end, viabilizando assim a integração de outras linguagens simbólicas para executar tarefas de conversão entre linguagens.

## Referências

- Aho, A. V., Lam, M. S., Sethi, R., and Ullman, J. D. (2008). *Compilers: Principles, techniques, and tools*, volume 2 ed.
- Ariza, C. and Cuthbert, M. (2011). *The music21 stream: A new object model for representing, filtering, and transforming symbolic musical structures*. Ann Arbor, MI: MPublishing, University of Michigan Library.
- Association, M. M. et al. (1996). *The complete MIDI 1.0 detailed specification: incorporating all recommended practices*. MIDI Manufacturers Association.
- Cuthbert, M. S. and Ariza, C. (2010). music21: A toolkit for computer-aided musicology and symbolic music data.
- Dannenberg, R. B. (1993). Music representation issues, techniques, and systems. *Computer Music Journal*, 17(3):pp. 20–30.
- Good, M. (2001). Musicxml for notation and analysis. *The virtual score: representation, retrieval, restoration*, 12:113–124.
- Hewlett, W. B. (1997). Beyond midi. chapter MuseData: Multipurpose Representation, pages 402–447. MIT Press, Cambridge, MA, USA.
- Johnson, S. C. (2001). Yacc: Yet another compiler-compiler.
- Lesk, M. E. and Schmidt, E. (2001). Lex - a lexical analyzer generator.
- Nienhuys, H.-W. and Nieuwenhuizen, J. (2003). Lilypond, a system for automated music engraving. In *Proceedings of the XIV Colloquium on Musical Informatics (XIV CIM 2003)*, volume 1. Citeseer.
- Oppenheim, I., Walshaw, C., and Atchley, J. (2010). The abc standard 2.0.

# Entre Notas e Alaridos: Perspectivas na Bioacústica de Mamíferos Terrestres

Regis Rossi Alves Faria<sup>1</sup>, Aline D. Carneiro Gasco<sup>1</sup>, Patrícia Ferreira Monticelli<sup>1\*</sup>

<sup>1</sup>Faculdade de Filosofia Ciências e Letras de Ribeirão Preto (FFCLRP)  
Universidade de São Paulo (USP)  
CEP 14.040-901 Ribeirão Preto, SP, Brazil

regis@usp.br, pmonticelli@ffclrp.usp.br

**Abstract.** *In the last decades considerable advances in monitoring and analysis technologies have improved the knowledge about the organization and role of animal sounds. This paper focus on some current issues in the study of mammalian species, showing some of the latest and promising technologies, trends in bioacoustics research and its applications.*

**Resumo.** *Nas últimas décadas consideráveis avanços nas tecnologias de monitoramento e análise ampliaram a compreensão sobre a organização e papel das emissões sonoras animais. Este artigo foca em algumas das questões atuais no estudo dos mamíferos terrestres, mostrando algumas das tecnologias mais recentes e promissoras, perspectivas na pesquisa em bioacústica e suas aplicações.*

## 1. Introdução

Neste artigo oferecemos à comunidade de computação musical uma abordagem introdutória aos conceitos, técnicas e recursos atuais usados no monitoramento passivo (automático), ativo (manual) e na análise de vocalizações de mamíferos terrestres. A bioacústica orienta-se ao estudo dos sons produzidos pelos animais para investigar processos mentais e aspectos ecológico-funcionais do comportamento geral e de comunicação, e tem-se mostrado uma ferramenta poderosa no monitoramento de populações naturais e na prática de conservação das espécies [Rocha, 2011]. Buscamos mostrar a importância crescente da bioacústica no mundo atual e sua aproximação com problemas e tecnologias da área de computação musical.

## 2. Estratégias e Sistemas para Estudo e Monitoramento de Mamíferos Terrestres.

Os sistemas de gravação adotados hoje seguem duas estratégias: monitoramento acústico passivo (PAM, *Passive Acoustic Monitoring*) que dispensam a presença do pesquisador e minimizam interferências sobre o comportamento animal; e o tradicional método de registro ativo pelo pesquisador, que pode ajustar em tempo real seu comportamento e o equipamento, melhorando a qualidade do registro sonoro e relatando o contexto de emissão de interesse. Os PAMs permitem a obtenção de um enorme conjunto de dados, impossível de ser obtido manualmente. Correspondem a gravadores instalados em caixas resistentes aos intempéries e que tem acoplados 2 ou 4 microfones. São programáveis

\* Apoio: Universidade de São Paulo, FAPESP, CNPq e CAPES.

para gravar a intervalos predefinidos, têm alta capacidade de armazenamento de dados e alimentação de longa duração. Com isso, pode-se coletar dados sem voltar ao campo, de forma contínua ou espaçadas no tempo (sazonalmente ou em janelas de interesse) e no espaço (por uma área de interesse amostral).

Sistemas mais sofisticados podem incluir ainda meios de comunicação remota (transmissão por radiofrequência acionados/monitorados à distância) e dispositivos GPS para rastreamento de posição. O monitoramento de espécies de interesse é feito através do rastreamento de suas vocalizações previamente conhecidas nas gravações, que acusam a sua presença no local e fornecem estimativas populacionais e outras informações sobre a dinâmica intra/entre espécies. A empresa Wildlife Acoustics produz gravadores autônomos (com microfones inclusive à prova d'água) para captação e análise sonora.

O monitoramento passivo não permite ainda, contudo, a gravação seletiva e contextualizada de sons de uma espécie, com a identificação da situação de emissão que levará a hipóteses sobre sua função comunicativa. Também ainda não se alcançou um nível de qualidade de registro que permita seu uso na descrição física e sirva como testemunho da espécie, muito menos para uso em estudos comparativos entre indivíduos ou espécies que levariam a inferências sobre o nível de socialidade e de complexidade de seus processos mentais, ou de relações filogenéticas, respectivamente. Esse tipo de registro ainda é obtido pessoalmente, com um gravador portátil profissional acoplado a um microfone direcional e um fone de ouvido.

Diversos modelos de gravadores profissionais portáteis são usados atualmente, como Marantz PMD660/661/671, Tascam DR100, Zoom H4N, Sony PCM-D100, Roland R26 ou Sound Device 722/788-T, acoplados a microfones supercardióides condensadores (*shotguns*). O levantamento e a descrição contextualizada do repertório de vocalizações de mamíferos terrestres da nossa fauna tem sido feito no Laboratório de Etologia e Bioacústica (EBAC) desde 2011. Requer a obtenção de vocalizações em uma variedade de contextos e a posterior categorização dos sinais segundo sua morfologia. Em um segundo nível de análise, os “tipos” de sinais são categorizados quanto ao “contexto de emissão”, como a defesa de território (o aulido do lobo-guará [Rocha et al., 2015]), o alerta e intimidação de predadores (o latido de quatis [Gasco, 2013]), o cortejo (o *purr* do macho preá [Monticelli and Ades, 2011]) e o reencontro entre membros do grupo, incluindo entre mães e filhotes (assobios de capivaras [Suzuki, 2015]). Os avanços na nossa compreensão sobre os processos biológicos e estados internos dos animais nos permitem hoje identificar parâmetros vocais que codificam inclusive suas emoções [Briefer et al., 2015].

Um problema clássico em bioacústica é detectar e isolar os sons produzidos simultaneamente pelas diversas espécies dentro da “paisagem acústica” registrada. A segregação e o isolamento das interferências do meio podem contar com técnicas de separação de fontes, matrizes de microfones e *beamforming* (separação baseada na localização). Ao estudo de mamíferos terrestres impõe-se uma série de requisitos à escuta, ao registro, à reprodução fidedigna para fins de testes experimentais de função comunicativa (*playback*), e à descrição e análise para fins científicos. Por iniciarem-se em uma faixa de frequências abaixo de 100Hz e terem enfatizadas frequências dominantes ou *peak frequency* abaixo de 1kHz, esses sons sobrepõem-se ao ruído de fundo ambiental.

Uma vez registradas as amostras, a próxima tarefa é detectar as vocalizações e extraí-las para análise. Esse processo de identificação e segmentação é guiado observando-se níveis de sinal/ruído e as características temporais e espectrais dos sinais. Uma correta calibração de parâmetros como intensidade, tamanho, tipo e percentual de *overlap* das janelas de análise, profundidade de bits e resolução de tempo e frequência será determinante para o sucesso. Cada tipo de ambiente e espécie implicam em um conjunto de

parâmetros mais adequado para o processamento e análise dos sinais, sendo esta busca, no caso da detecção e análise manual, um dos processos mais demorados.

O pesquisador atualmente dispõe de pacotes e recursos computacionais projetados para detecção automática e análise de sons animais, como os *softwares* Raven, do Laboratório de Ornitologia da Cornell University, o Avisoft SASLab, e a plataforma Arbimon. Rocha e outros [Rocha et al., 2015] descreveram o uso do pacote de análise sonora XBAT (*Extensible Bioacoustics Tool*) para a plataforma MATLAB na detecção automática de vocalizações de lobos em gravações de longo prazo, através da busca por sons que se auto-correlacionem com modelos preestabelecidos e o rastreamento de eventos para os quais o valor de correlação exceda um determinado limiar. Preconizando uma arquitetura extensível, que permite que componentes compartilhem uma mesma infraestrutura, os usuários deste *software* podem acessar, visualizar, buscar, anotar e medir eventos de interesse nos sons.

Os formatos de codificação perceptuais voltados para voz humana, como o AMR (*Adaptive Multi-Rate*) e o 3GPP disponíveis em *smartphones*, não são adequados para um registro fidedigno de vocalização animal, para o que é usado o formato não comprimido WAV (PCM), com amostragem de 16-24bit@44-48kHz.

### 3. Monitoramento Mediado por Auralização.

Uma estratégia relativamente recente de estudo está relacionada à captura de emissões vocais dos animais induzidas artificialmente, por meio da auralização no ambiente que os estimulariam a dar uma resposta. Tal estratégia é usada desde a 2ª Guerra por ornitólogos e herpetólogos (estudiosos de anuros) para verificar a ocorrência (induzindo a aproximação) e um aumento na emissão de uma resposta, vocal ou não, que permita caracterizar comportamentos e inferir funções comunicativas. Mas, até hoje foi pouco usado com mamíferos. O realismo da auralização parece ser um requisito crítico de sucesso. A capacidade de cobertura espacial da projeção sonora dependerá de um sistema de auralização com alto nível de imersão e com resolução e qualidade suficientes para garantir uma comunicação efetiva.

### 4. Desafios e Perspectivas na Bioacústica de Mamíferos Terrestres.

Muitos são os aspectos importantes a considerar em um sistema de monitoramento bioacústico, como a diversidade de espécies, o uso de detectores de atividade para disparar eventos de gravação, variações climáticas, interferências acústicas, ruídos e separação de sons. Do ponto de vista funcional, os objetivos da análise irão determinar as ferramentas de processamento a serem utilizadas. Usualmente, o vocabulário de emissões vocálicas de mamíferos e outros animais são analisáveis a partir de uma segmentação em unidades sonoras, que podem se assemelhar às sílabas. Uma série de ferramentas utilizadas em *Musical Information Retrieval*, como analisadores espectrais, descritores sonoros, transformadas e segmentadores são utilizados na análise desses sons. Todavia há diferenças substanciais entre estes e os sinais musicais, particularmente na estrutura temporal e em parâmetros espectrais [Stowell and Plumbley, 2011]. Enquanto na música encontramos timbre e uma estrutura temporal bem definidos, em emissões animais encontramos sílabas de diferentes conteúdos soando com durações próximas a  $1/2$  segundo, e compondo estruturas ou frases que se repetem. A identificação do tipo de emissão e seu significado só será possível a partir da segmentação de toda a sequência (frase).

Na FFCLRP-USP, o EBAC em colaboração com o Laboratório de Acústica e Tecnologia Musical (LATM) tem buscado levantar requisitos e técnicas de processamento de sinais para melhor isolar as interferências ambientais, rumo a uma análise diferenciada

para sons de mamíferos terrestres. O grupo vem definindo metas em direção a um sistema para monitoramento, lançando mão de técnicas para auralização espacial em campo e para captura sonora espacial. O monitoramento acústico com múltiplos microfones é útil para localizar animais e rastrear seus movimentos em uma escala espacial maior [Blumstein et al., 2011]. Considerando a migração dos aplicativos para as plataformas móveis e a oferta de acessórios para este segmento, atualmente é praticável a captação de sons em boa resolução em tablets (ex: iPads). Ainda, os sistemas que utilizam áudio *wireless* expandem as possibilidades, principalmente se aliados a um dispositivo de recarregamento autossustentável de baterias.

Finalmente, um estudo interessante prospectivo é a avaliação da emoção e expressividade no registro sonoro de animais silvestres [Moura et al., 2008]. Assim como no caso do canto humano, correlações entre duração, tonalidade e intensidade de diversas vocalizações forneceriam subsídios para uma análise aprofundada do conteúdo da *cantoria*, identificando contornos melódicos e harmônicos em frases longas e caracterizando modulações. Avaliações de longo prazo do registro sonoro assim, extrapolariam aspectos puramente ecológicos do monitoramento tradicional, avançando sobre análises semânticas como hoje temos na área de música.

## Referências

- Blumstein, D. T., Mennill, D. J., Clemins, P., Girod, L., Yao, K., Patricelli, G., Deppe, J. L., Krakauer, A. H., Clark, C., Cortopassi, K. A., Hanser, S. F., McCowan, B., Ali, A. M., and Kirschel, A. N. G. (2011). Acoustic monitoring in terrestrial environments using microphone arrays applications, technological considerations and prospectus. *Journal of Applied Ecology*, 48(3):758–767.
- Briefer, E. F., Maigrot, A. L., Mandel, R., Freymond, S. B., Bachmann, I., and Hillmann, E. (2015). Segregation of information about emotional arousal and valence in horse whinnies. *Scientific Reports*, 9989(4):1–11.
- Gasco, A. D. C. (2013). *Repertório Acústico do Quati (Nasua nasua)*. PhD thesis, Universidade de São Paulo.
- Monticelli, P. F. and Ades, C. (2011). Bioacoustics of domestication: Alarm and courtship calls of wild and domestic cavies. *Bioacoustics*, 20:169–192.
- Moura, D. J., Silva, W. T., Naas, I. A., Tolón, Y. A., Lima, K. A. O., and Vale, M. M. (2008). Real time computer stress monitoring of piglets using vocalization analysis. *Computers and Electronics in Agriculture*, 1(64):11–18.
- Rocha, L. H. S., Ferreira, L. S., Paula, B. C., Rodrigues, F. H. G., and Sousa-Lima, R. S. (2015). An evaluation of manual and automated methods for detecting sounds of maned wolves (*chrysocyon brachyurus illiger 1815*). *Bioacoustics*, 24(2):185–198.
- Rocha, Y. T. (2011). Técnicas em estudos biogeográficos. *Raega O Espaço Geográfico em Análise*, 23:398–427.
- Stowell, D. and Plumbley, M. D. (2011). Birdsong and c4dm a survey of uk birdsong and machine recognition for music researchers. *Centre for Digital Music, Queen Mary, University of London, London, UK, Tech. Rep. C4DM-TR-09-12*, 181.
- Suzuki, C. T. (2015). *Estudo Naturalístico do Comportamento de Comunicação Vocal das Capivaras*. PhD thesis, Universidade de São Paulo.

## Protótipo Robótico de Baixo Custo para Performance Musical em Violão Acústico

Higor A. F. Camporez, Trajano S. R. Mota, Leandro Costalonga, Helder Rocha

CEUNES – Universidade Federal do Espírito Santo (UFES)  
Rodovia BR 101 Norte, Km. 60, Bairro Litorâneo, CEP 29932-540, São Mateus – ES

{higorcamporez, trajanoreuter}@gmail.com,  
{leandro.costalonga, helder.rocha}@ufes.br

**Abstract.** *The research field of electronics applied to music technology goes beyond the improvement of audio quality. Robotics have been largely used in educational projects but rarely in the computer musical context. This paper presents a robotic device capable of performing on an acoustic guitar. The project was developed as a part of an extension project that encourage undergrad students to build musical projects. The description of the device and its musical and technical limitation are presented. Regardless of the musical quality, a demonstration video can be seen at visualizada em <https://youtu.be/dS3c-ZuiV7g>.*

**Resumo.** *O emprego da eletrônica no desenvolvimento de tecnologia musical não se limita a questões de qualidade do áudio. O uso da robótica como estratégia de ensino é amplamente difundida mas ainda pouco explorada no contexto ensino computação musical. Este artigo apresenta um dispositivo robótico projetado para executar performances musicais em violão acústico. Tal dispositivo foi desenvolvido como parte de um projeto de graduação no contexto de um projeto de extensão que visa oferecer a oportunidade de aplicação do conhecimento em projetos musicais. A descrição do dispositivo robóticos e suas limitações musicais e técnicas são apresentadas nesse artigo. Obstante a qualidade musical, uma demonstração do protótipo pode ser visualizada em <https://youtu.be/dS3c-ZuiV7g>.*

### 1. Introdução

A relação entre a música e eletrônica não é nova (Hiller, L., & Beauchamp, J 1965). A criação de novos instrumentos eletrônicos propiciam explorar novas sonoridades assim com novas técnicas de expressividade musical (Miranda, E. R., & Wanderley, M. M. 2006). Com o desenvolvimento e popularização da computação e eletrônica, usuários atualmente podem prototipar seus próprios instrumentos musicais (Berdahl, E., Salazar, S., & Borins, M. 2013) visando experimentação sonora ou ampliação de técnicas de performances musicas devido a limitações inerentemente humanas ( Costalonga, 2009). Nesse contexto, robobres capazes de tocar um instrumentos musicais se tornam realidade (Singer et. al. 2004).

O uso da robótica como estratégia de ensino foi amplamente discutido por Alimisis et. al. (2007). Desde 2013, o emprego de estratégias de ensino construtivistas vem sendo

usada nos projetos de ensino ligados ao NESCoM, em especial no projeto Campus Vivo (Costalonga, 2013). Projetos dessa natureza oportunizam a consolidação do aprendizado através a aplicação direta do conhecimento em projetos multidisciplinares que, direta ou indiretamente, envolvem a computação musical. Exemplos dessa integração perpassa pelo ensino da programação através de uma linguagem sônica, programação de trilhas para vídeo game, desenvolvimento de instalações multimídia interativas e, como tratado nesse artigo, desenvolvimento de robôs visando a criação de um festival de música robótica nos moldes da Robocup (Kitano et. al., 1997)

O presente artigo descreve um protótipo de robô projetado para tocar violão acústico. O robô é composto de 3 partes que se comunicam: a) dispositivo deslizante de mão esquerda; b) dispositivo de deflexão de corda de mão esquerda; c) dispositivo de excitação de cordas de mão direita. Cada um desses módulos serão apresentados com maiores detalhes nas seções subsequentes. Na Seção 2 são apresentados os detalhes do hardware usado para simular a mão direita, mão esquerda e o dispositivo deslizante de mão esquerda. Na Seção 3 é apresentado o hardware usado para fazer sincronização entre os dispositivos de mão direita e mão esquerda. A Seção 4 descreve o algoritmo de sincronização implementado usando o micro-controlador Arduino. Resultados e dificuldades encontrados são comentados na Seção 5. Por fim, a Seção 6 apresenta as principais conclusões baseadas nos resultados obtidos.

## 2. Descrição do Hardware

A popularização de materiais eletrônicos como o Arduino e seus derivados vem auxiliando o desenvolvimento de protótipos com ótimo desempenho, baixo custo e fácil implementação, além de possibilitar um melhor aprendizado na parte prática de certas disciplinas (de Souza, A. R. al., 2011). Um dos pilares desse trabalho é favorecer o uso de materiais de baixo custo eletrônicos na confecção de um protótipo robótico para performance musical. Assim, nesta secção, são descritos os equipamentos eletrônicos usados para simular o comportamento de um violonista.

### 2.1. Mão direita (MD)

A mão direita do violonista (destro) desempenha uma função rítmica nas performances em violão (Reboursière, L. al., 2012). Deste modo, um dispositivo foi especialmente desenvolvido para esse fim, ou seja, tal dispositivo pode excitar uma corda específica (ou até todas), em um determinado ritmo pré-programado. No dispositivo de mão-direita são utilizados seis servomotores, estando cada um posicionado logo acima de sua específica corda. O servomotor usado pode ser visto na Figura 1.



Figura 1. Servomotor

Observe na Figura 1, que a haste do servomotor é muito curta, não sendo possível posicioná-los de modo que alcancem a corda. Logo, foi necessário ampliar o comprimento da haste. Como citado anteriormente, uma das premissas do projeto é o reuso de materiais eletrônicos descartados, favorecendo seu baixo custo. Assim sendo, utilizou-se uma capa rígida de fio de alimentação, encontrada em sucatas disponíveis no laboratório da própria universidade, para fazer estender a haste e agir como palheta/unha de um violonista.



**Figura 2. Servomotores com “palhetas”**

Para fixação dos motores, utilizou-se uma estrutura de madeira reciclada. Os motores foram fixados ao suporte, e o mesmo foi posicionado acima da cavidade do violão conhecida como boca. O suporte foi pensado de modo que sua altura pudesse ser regulada, permitindo ajustar mecanicamente a intensidade da excitação da corda pela haste do servomotor.

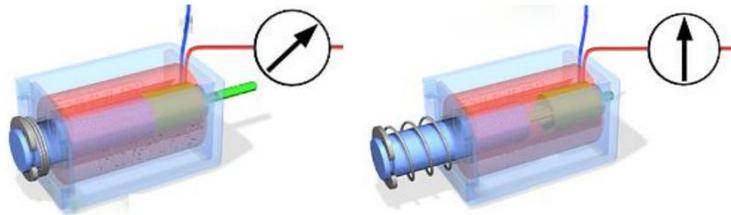
O sistema é alimentado por uma fonte independente de 5V. A necessidade da utilização de uma fonte independente se deve ao fato do alto consumo de corrente de cada servomotor, cerca de 250 mA. A fonte alimenta os motores e o Arduino, que também funciona com 5V. Cada servomotor é conectado a uma saída PWM (*Pulse-Width Modulation*) do Arduino, que permite o controle do pulso em cada motor, para então, o controle do ângulo de rotação do mesmo. Deste ponto em diante todo processo é feito por meio de programação. Não foi necessário implementar circuitos auxiliares, visto que o servomotor já contém seu próprio sistema de controle.

Com o auxílio da biblioteca `servo.h`<sup>1</sup> é possível mandar um valor específico em graus para o motor, onde o próprio interpreta o pulso PWM e rotaciona até a posição desejada. Ao ligar o sistema, devem-se colocar todos os servomotores em uma posição inicial, assim, sincronizando o software com o hardware. Para que isso seja possível, é dado um pulso nos seis motores para colocá-los todos na mesma posição. O próximo passo é identificar a corda que deve ser tocada, o que depende da música, logo, é aplicado um pulso no servomotor referente a corda requerida, girando 25° (ângulo adquirido empiricamente) no sentido de tocar a corda. A programação do arpejo utilizado em cada acorde é pré-definido utilizando a função `delay()`<sup>2</sup>. Caso seja necessário excitar simultaneamente todas as cordas envolvidas na performance do acorde, basta aplicar os pulsos nos servomotores sem o uso da função.

## 2.2. Mão esquerda (ME)

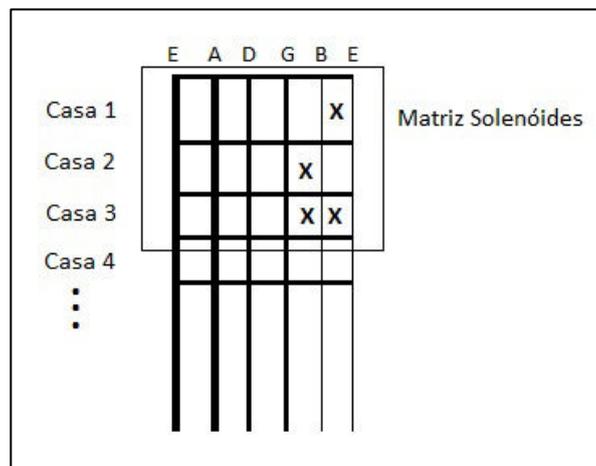
A formação de um desenho de acorde perpassa pela interrupção das cordas contra determinados trastes no braço do violão. Tal procedimento efetivamente encurta o comprimento da corda e, conseqüentemente, altera sua frequência de sua vibração (nota

musical). Essa interrupção mecânica é normalmente feita usando os dedos da mão esquerda do violonista (Costalonga 2009). No contexto desse trabalho, tal processo foi recriado eletromecanicamente através de uma matriz de solenóides que representa a mão esquerda do violonista. Uma solenoide é ilustrada na Figura 3.



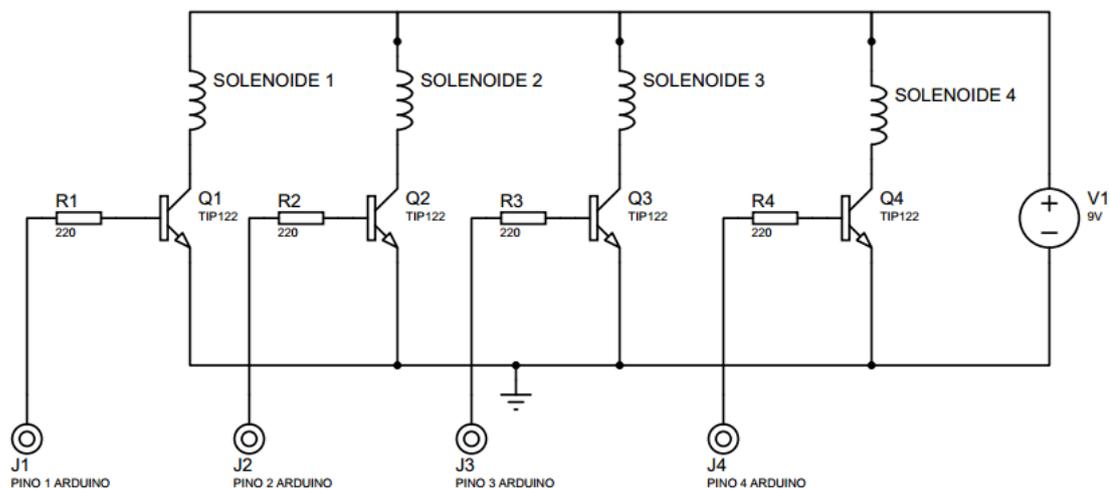
**Figura 3. Solenóide ativo/inativo**

Na solenoide, caso a mesma seja energizada, ocorre a criação de um campo magnético que induz uma haste de metal no sentido do campo. Existe uma mola que retorna a haste a sua posição inicial caso não esteja energizada. O deslocamento da haste efetua a função de apertar a corda contra o traste, função que normalmente seria feita pelo dedo do violonista. Cada solenoide utilizada consome em média 1A para que possa mudar de posição com a força necessária para pressionar a corda contra o traste - a tensão varia entre 5 e 9V. Considerando o pior cenário, onde 6 cordas seriam requisitadas simultaneamente, 6A de corrente serão necessários, assim, o circuito precisa de uma fonte externa que suporte alimentar tal sistema. Para a fixação dos solenóides criou-se um suporte de madeira. Visto que a matriz é uma estrutura rígida e há a necessidade de preencher 4 casas simultaneamente para formar os desenhos de acordes que deseja-se suportar nesse projeto, seriam necessários 24 solenóides dispostos em uma matriz 6X4. Como se trata apenas de um protótipo, foram utilizados apenas 4 solenóides, como mostra a Figura 4, para que os testes pudessem ser efetuados. Ainda, considerando que a extremidade da haste do solenoide é metálica e escorregadia, foram utilizados pedaços de borracha para simulação da digital do dedo humano.



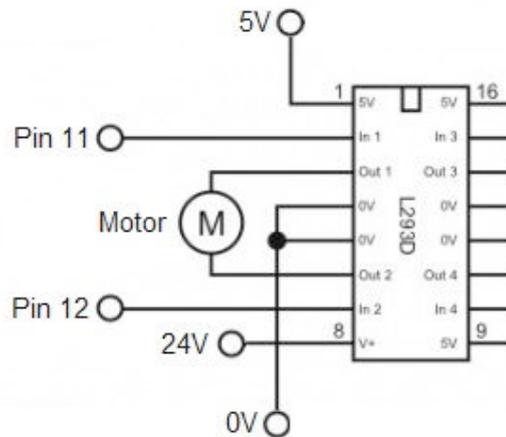
**Figura 4. Disposição dos solenóides na matriz**

Para o controle dos solenoides, foi criado um circuito de chaveamento feito com transistores TIP 122 e resistores, pois somente o Arduino não consegue fornecer a corrente necessária sem que seu circuito seja danificado, visto que a máxima corrente de cada porta digital é apenas 40mA. Cada pino digital transmite uma tensão de +5V caso esteja em estado *HIGH*, assim, para que um solenoide seja ligado, deve-se mudar o estado do pino. Para garantir que a corrente não ultrapasse o limite suportado pelas porta do Arduino, foram utilizados resistores de 220 ohms, assim,  $\frac{5 - 0,7}{220} \cong 19,54 \text{ mA}$ , que é suficiente para colocar o transistor em estado de condução, fazendo o seu chaveamento. Vale ressaltar que o valor 0,7V é proveniente do terminal de junção base-emissor do transistor. O controle de ativação/desativação de cada solenoide é implementado no Arduino alterando os estados dos pinos de *LOW* para *HIGH* e vice-versa. Ao ligar o sistema, todos os pinos conectados aos solenoides recebem o estado *LOW*, pois para que a matriz se movimente corretamente nenhum pino pode estar pressionado. A Figura 5 demonstra o circuito de chaveamento.



**Figura 5. Circuito de Chaveamento**

Além do movimento que permite pressionar as cordas, também há o movimento de translação da matriz pelo braço. O mecanismo que possibilita tal movimento veio de impressora descartada, onde aproveitou-se o suporte de cartucho e o sistema de deslizamento. O motor que faz o movimento da matriz, é do tipo DC de 24V. Caso o mesmo seja alimentando diretamente com os 24V que necessita, o movimento do braço é muito rápido, sendo difícil controlá-lo com precisão. Então, foram usadas portas do Arduino que possuem saída PWM, fazendo com que o motor continue com o mesmo torque e possibilitando o controle de sua velocidade. Para que fosse possível o movimento para ambos os lados, usou-se uma ponte H, implementado no circuito integrado L293D, como mostra a Figura 6.



**Figura 6. CI ponte-H**

Para garantir o correto alinhamento da matriz com os trastes, utilizou-se um sensor ultrassônico como mostrado na Figura 7. O sensor posicionado próximo a junção do corpo do violão com o braço emite uma onda ultra-sônica que rebate na matriz que foi usada como objeto de referência. O tempo de retorno da onda é usado para determinar a distancia entra as partes. Para que tudo isso seja possível, foi elaborado um programa no Arduino que possibilita o controlar com precisão a distância. No programa, há uma função em que recebe como paramento o valor da casa em que a matriz deve se posicionar para formação de um especifico acorde ou nota, valor esse obtido empiricamente. Com o valor recebido do sensor, um sinal é enviado para o motor girar para a esquerda ou direita, ajustando a posição da matriz no braço. Ao chegar à posição desejada, o motor fica parado, até que receba outra chamada com uma nova localização.



**Figura 7. Sensor Ultrassônico**

### 3. Sincronização do hardware

Um único Arduino Mega 2560 foi utilizado no projeto, ficando o mesmo também responsável pela sincronização do Hardware entre as 3 partes do mecanismo robótico: dispositivo deslizante de mão esquerda, dispositivo de deflexão de corda de mão esquerda, dispositivo de excitação de cordas de mão direita. O Arduino Mega 2560 é baseado no microcontrolador ATmega2560, possuindo 54 pinos de entradas e saídas digitais, onde 15 tem propriedade PWM. Possui ainda 16 entradas analógicas com resolução de 10 bits, 4 UARTs , cristal oscilador de 16 MHZ.

Dados os 6 servomotores foram necessárias 6 portas de saídas PWM, sendo utilizadas as portas de 2 a 7, refletindo nas cordas do violão 1 a 6, respectivamente. Para os solenoides precisou-se de apenas 4 portas digitais de saída ( 22, 24, 26 e 28). Já o sensor

ultrassônico necessitou de uma entrada digital e uma saída digital, assim, utilizou-se as portas 8 e 9 respectivamente. Por fim, o controle do motor DC foi feito com duas portas de saída PWM, sendo as escolhidas a 11 e 12. A Figura 8 ilustra a utilização das portas através do esquemático do protótipo.

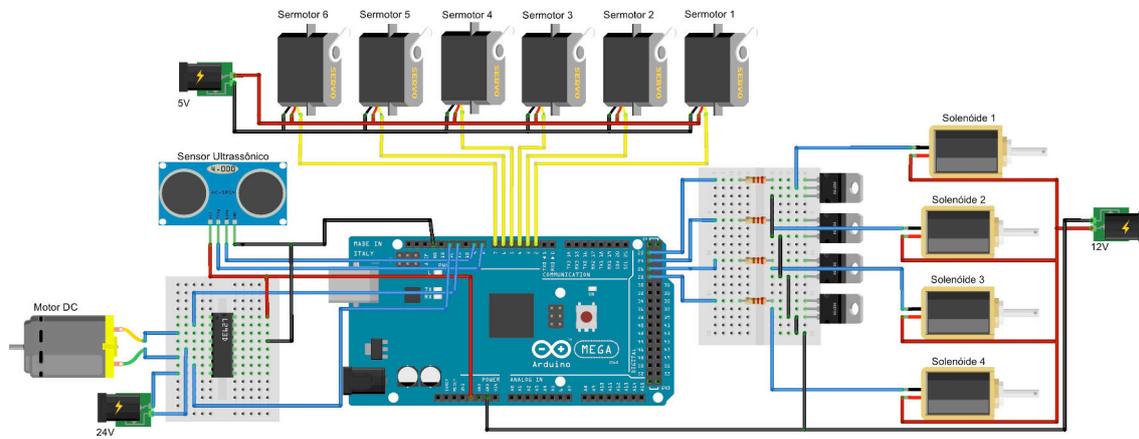


Figura 8. Esquemático de todo protótipo

#### 4. Software de controle

Após a construção de toda parte física do protótipo, foi preciso pensar a sincronização de todos os módulos visando a performance musical. O funcionamento do sistema é exemplificado no Fluxograma representado pela Figura 9.

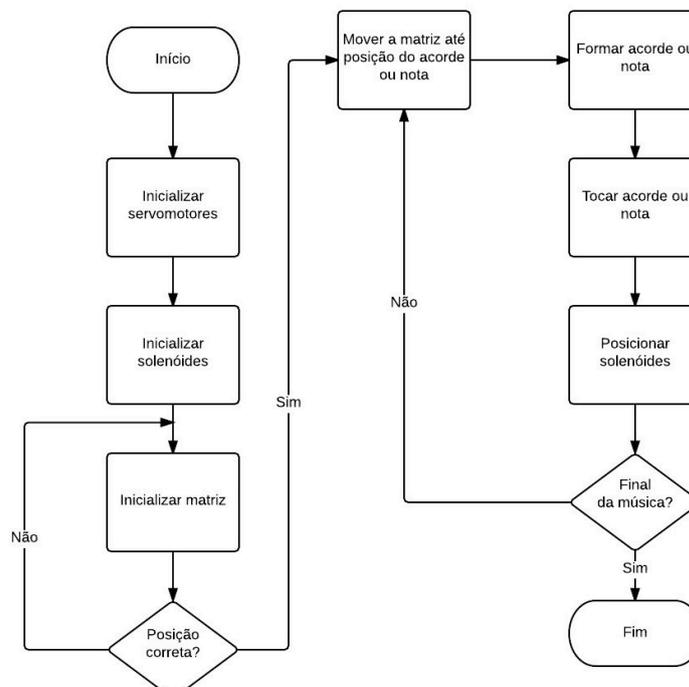


Figura 9. Fluxograma sobre o funcionamento do protótipo

A figura 10 mostra um trecho de código de controle do protótipo.

```
void loop()
{
  andarAte(2); // Move a matriz até a casa dois
  delay(500); // Espera 500 milisegundos
  ativarSolenóide(1,2); // Ativa as solenóides 1 e 2 ao mesmo tempo
  delay(100); // Espera 100 milisegundos
  tocar(3,2); // Toca as cordas 3 e 2 ao mesmo tempo
  delay(500); // Espera 500 milisegundos
  desativarSolenóide(1,2); // Desativa as solenóides 1 e 2 ao mesmo tempo

  andarAte(5); // Move a matriz até a casa cinco
  delay(500); // Espera 500 milisegundos
  ativarSolenóide(3,4); // Ativa as solenóides 3 e 4 ao mesmo tempo
  delay(100); // Espera 100 milisegundos
  tocar(3,4); // Toca as cordas 3 e 4 ao mesmo tempo
  delay(500); // Espera 500 milisegundos
  desativarSolenóide(3,4); // Desativa as solenóides 3 e 4 ao mesmo tempo
}
```

Figura 10. Trecho de código

## 5. Resultados

Uma demonstração do funcionamento do protótipo robótico projetado para performance musical está disponível no site: <https://youtu.be/dS3c-ZuiV7g>. A Figura 11 mostra o protótipo montado.



Figura 11. Protótipo montado

Ao longo da montagem do protótipo foram enfrentados alguns desafios que serão reportados a seguir.

### 5.1. Mão direita (MD):

Na mão direita (MD) houve problema no momento em que os servomotores tocam as cordas, como primeira experiência foram usadas palhetas presas com parafusos na haste dos servomotores, assim, ocorreram quatro problemas: o primeiro foi com a fixação da palheta pois, para que a mesma ficasse no sentido de tocar uma corda e em posição para ser fixada no servomotor, foi-se necessário dobra-la ao meio, como o seu material é de

plástico o mesmo não resistiu quando foi usado varias vezes, assim, a dobra quebrava. Uma saída encontrada foi usar uma capa de fio de alimentação (Figura 2). O segundo consistiu na altura das cordas, pois, cada uma atingia alturas diferentes em relação aos servomotores, dependendo da casa em que a mesma era pressionada. Quanto mais próximo ao final braço do violão, mais as cordas se distanciavam das “palhetas” de forma que ao tocar uma nota o som não saia, ou saia abafado. Uma possível solução para esse problema seria implementar um sistema com algum tipo de “elevador”, onde se possa controlar a altura necessária para alcançar as cordas. O terceiro problema foi o ruído característico do movimento do servomotor devido as suas engrenagens, problema esse que pode ser resolvido com a utilização de um captador e um filtro, assim, removendo a perturbação. Por fim, houve problema no controle de força com que as cordas são tocadas, pois no protótipo não há como fazer esse ajuste dinamicamente, visto que o servomotor não tem controle de velocidade e intensidade. Desta forma a dinâmica da música e, conseqüentemente a expressividade, foi prejudicada.

## 5.2. Mão esquerda (ME):

Na construção do dispositivo que simula a mão esquerda também foram encontrados dificuldades, tais como: dimensão, preço e força dos solenoides, variação no tamanho das casas do violão e o alargamento das cordas à medida que se aproxima da junção do braço. Os solenoides usados eram muito largos, assim, houve uma dificuldade em deixá-los nas posições corretas para a formação de pelo menos um acorde. O preço foi um fator complicador, pois no Brasil não se acha com facilidade solenoides com valores baixos. No projeto foram usados solenoides de baixo custo que tinham pouca força no deslocamento da haste; Para resolver esse problema foi preciso a aplicar uma tensão além do recomendado para obter uma força considerável. Porém, esse excesso de potência gera mais calor, ocasionando um superaquecimento dos solenoides, fazendo com que o sistema não pudesse operar por muito tempo. Outro fator foi a variação no tamanho das casas do violão, já que no seu início elas são mais largas e a medida que se aproxima da região mais aguda, as casas diminuem de tamanho, ilustrado na figura 12, nos pontos ‘c’ e ‘d’. Assim, como a matriz foi feita com madeira e tem formato fixo, não foi possível que se fizesse a propagação de uma mesmo desenho de acorde para outras regiões do violão, pois os solenoides acabavam pressionando locais errados. Da mesma forma, o alargamento da distancia entre as cordas a medida que a matriz se aproxima do fim do violão também ocasionou a má formação de acordes, ilustrado na figura 12, nos pontos ‘a’ e ‘b’. Logo, ao se chegar em certas regiões os acordes não eram formados corretamente.



Figura 12. Ilustração de pontos críticos

Na parte do movimento da matriz, foram observados três problemas: barulho causado pelo motor DC, sensor ultrassônico e a estrutura do suporte de madeira. Os movimentos mecânicos de um motor DC causam ruídos sonoros, este que pode interferir no som do violão, como foi o caso observado no protótipo. O motor utilizado foi retirado juntamente com a estrutura de uma impressora, assim, o projeto teve sons característicos da mesma. No caso do sensor ultrassônico, foi observado que o mesmo retornava valores inconsistentes em determinadas aferições, levando a um erro no posicionamento da matriz, assim, para resolução do problema foi implementado um algoritmo que realizava cinco medidas e a média das mesmas para reduzir o impacto de *outliers*. A etapa mais relevante dessa parte foi a construção da estrutura fixada ao violão, visto que, a mesma não poderia danificar o instrumento com pregos ou parafusos, assim, foi construída uma estrutura que se encaixava no seu braço, porém, a mesma acabava saindo da sua posição original caso ele fosse movimentado.

### 5.3. Tempo

O estratégia utilizada para controle do tempo da performance musical foi o atraso sucessivo dos eventos musicais, ou seja, não há utiliza-se a referencia de tempo inicial mas o tempo decorrido da ultimo evento (Figura 10). No futuro, isso pode gerar um problema, visto que objetiva-se a criação de outros robôs musicais que possam tocar juntos. Desta forma, o acumulo de atrasos pode causar problemas de sincronização entre os instrumentos. Uma possível solução implica em utilizar janelas de sincronização para que de tempos em tempos, todo o sistema novamente se sincronizasse emulando a capacidade humana de ouvir e predizer aspectos rítmicos, como no ato de bater palmas.

## 6. Conclusão

O presente artigo apresentou um dispositivo robótico para executar performances musicais em violão acústico. Uma demonstração do protótipo pode ser visualizada em <https://youtu.be/dS3c-ZuiV7g>.

O robô é composto de 3 partes que se comunicam: dispositivo deslizante de mão esquerda, dispositivo de deflexão de corda de mão esquerda, dispositivo de excitação de cordas de mão direita. O Arduino Mega 2560 é responsável pela sincronização do Hardware entre as 3 partes do mecanismo robótico. Foi desenvolvido um algoritmo de sincronização de todos os módulos visando a performance musical.

Dentre as limitações musicais cita-se: restritos desenhos de acordes possíveis, impossibilidade de controle dinâmico da força dos atuadores de mão direita (amplitude das notas), possível má formação de acordes e sua qualidade sonora devido características de construção do violão acústico, controle de tempo inapropriado para execução em conjunto, ruído sonoro oriundo dos motores e outros componentes eletromecânicos. As limitações técnicas perpassam por: tamanho dos solenoides, força aplicável dos solenoides, controle de velocidade dos servomotores, redimensionamento automático da matriz de solenoides (efeito sanfona), programação musical estática.

Os trabalhos futuros implicam em buscar solução para as limitações supracitadas, além da construção de novos instrumentos visando um festival de música robótica. Uma linguagem de programação específica para o controle dos robôs está em desenvolvimento.

## 7. Referências

- Alimisis, D., Moro, M., Arlegui, J., Pina, A., Frangou, S., & Papanikolaou, K. (2007, August). "Robotics & constructivism in education: The TERECOP project. In EuroLogo (Vol. 40, pp. 19-24)".
- Berdahl, E., Salazar, S., & Borins, M. (2013). "Embedded networking and hardware-accelerated graphics with satellite ccrma". In Proceedings of the International Conference on New Interfaces for Musical Expression (pp. 325-330).
- Costalonga, L. (2009). "Biomechanical Modeling of Musical Performace: A Case Study of the Guitar. PhD thesis, School of Computing, Communications and Electronics, University of Plymouth, 2009.
- Costalonga, L. ; Neves, M. V. M.(2013). Impacto Regional do SBCM: Caso de estudo em Vitória-ES. In: Simpósio Brasileiro de Computação Musical, 2013, Brasília. Anais do Simpósio Brasileiro de Computação Musical, 2013. v. 1
- de Souza, A. R., Paixao, A. C., Uzêda, D. D., Dias, M. A., Duarte, S., & de Amorim, H. S. (2011). "A placa Arduino: uma opção de baixo custo para experiências de física assistidas pelo PC". Revista Brasileira de Ensino de Física, 33.
- Hiller, L., & Beauchamp, J. (1965). "Research in music with electronics". Science,150(3693), 161-169.
- Kapur, A., Eigenfeldt, A., Bahn, C., & Schloss, W. A. (2009). "Collaborative composition for musical robots. Journal of Science and Technology of the Arts",1(1), 48-52.
- Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., & Osawa, E. (1997, February). "Robocup: The robot world cup initiative. In Proceedings of the first international conference on Autonomous agents (pp. 340-347)". ACM.
- Lund, H. H., & Ottesen, M. (2008). "RoboMusic: a behavior-based approach.Artificial Life and Robotics", 12(1-2), 18-23.
- Miranda, E. R., & Wanderley, M. M. (2006). "New digital musical instruments: control and interaction beyond the keyboard (Vol. 21)". AR Editions, Inc..
- Reboursière, L., Lähdeoja, O., Drugman, T., Dupont, S., Picard-Limpens, C., & Riche, N. (2012, May). "Left and right-hand guitar playing techniques detection". In Proc. of NIME.
- Singer, E., Feddersen, J., Redmon, C., & Bowen, B. (2004, June). "LEMUR's musical robots". In Proceedings of the 2004 conference on New interfaces for musical expression (pp. 181-184). National University of Singapore.

---

1 Disponível em: <<https://www.arduino.cc/en/Reference/Servo>>. Acesso em 6 jun. 2015

2 Disponível em: <<https://www.arduino.cc/en/Reference/Delay>>. Acesso em 6 jun. 2015

3 Disponível em: <<http://playground.arduino.cc/Main/UltrasonicSensor>>. Acesso em 15 jun. 2015

4 Disponível em: <<https://www.arduino.cc/en/Main/ArduinoBoardMega2560>>. Acesso em 1 set. 2015

# Representação de Conhecimento Musical e Programação Lógica Indutiva - Uma Revisão Sistemática

Clenio B. Gonçalves Junior<sup>1,3</sup>, Murillo R. P. Homem<sup>1,2</sup>

<sup>1</sup>Programa de Pós-Graduação em Ciência da Computação de Sorocaba  
Departamento de Computação (DComp) Sorocaba  
Universidade Federal de São Carlos (UFSCar) – Sorocaba, SP - Brasil

<sup>2</sup>Departamento de Computação (DC) São Carlos  
Universidade Federal de São Carlos (UFSCar) – São Carlos, SP - Brasil

<sup>3</sup>Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP)  
Campus São Roque – São Roque, SP - Brasil

clenio@ifsp.edu.br, murillo@ufscar.br

**Abstract.** Knowledge representation concept is an essential matter regarding Computer Music. Inductive Logic Programming is a research field which combines concepts of Logic Programming and Machine Learning. Due to its declarative feature, both acquired and produced knowledge can be presented to not-expert users in a naturally understandable way. This paper presents a systematic review to approaches that apply Inductive Logic Programming on musical knowledge representation process. Study selection was based on the relevant issues they aim to address and key components were identified for each implementation.

**Resumo.** O conceito de representação de conhecimento constitui um elemento essencial em Computação Musical. A Programação Lógica Indutiva é um campo de pesquisa que incorpora conceitos de Programação em Lógica e Aprendizado de Máquina. Seu caráter declarativo possibilita que o conhecimento musical, tanto adquirido como produzido, seja apresentado a usuários não especialistas de modo naturalmente compreensível. Este trabalho apresenta uma revisão sistemática sobre abordagens que aplicam a Programação Lógica Indutiva na Representação de Conhecimento Musical. Foram levantadas questões importantes que esses trabalhos procuram atender, bem como identificados aspectos característicos em cada implementação.

## 1. Introdução

Programação Lógica Indutiva (PLI) é um campo de pesquisa crescente que combina os conceitos de Programação em Lógica e Aprendizado de Máquina. PLI baseia-se na lógica de primeira ordem, o que lhe confere um caráter declarativo, isso permite que o resultado do processamento gerado por um sistema possa ser apresentado a usuários não especialistas de uma forma simples e intuitiva. Por meio da programação lógica indutiva, novo conhecimento musical pode ser gerado automaticamente a partir da derivação de estruturas e regras expressas na forma de cláusulas de Horn. Uma série de modelos têm sido propostos em trabalhos envolvendo conceitos como contraponto, expressividade performática, representação harmônica, inferência modal, entre outros. Adicionalmente, a PLI tem sido aplicada como um eficiente recurso para representação de conhecimento no que tange à caracterização de gêneros e estilos musicais. O presente trabalho apresenta um levantamento considerando abordagens que apliquem a PLI no processo de representação de conhecimento musical, investigando critérios e práticas adotados, bem como a utilização de recursos computacionais, além de conceitos envolvidos no campo da Musicologia.

## 2. Metodologia

A metodologia de pesquisa seguiu os critérios apresentados em [Kitchenham et al., 2009]. Foram levantadas duas questões para servir de diretriz ao trabalho: **1)** Qual arquitetura é proposta para representação do conhecimento musical? **2)** Como a abordagem aplica os recursos computacionais ligados à PLI? A Tabela 1 apresenta as abordagens selecionadas, cada uma recebeu uma identificação numérica e um nome formado pelas duas primeiras palavras do título.

**Tabela 1: Relação de trabalhos revisados**

ID	ABORDAGEM	TÍTULO	REFERÊNCIA
1	SymbolicRepresentation	Symbolic Representation of Chords for Rule-Based Evaluation of Tonal Progressions	Chong and Ding, 2014
2	ImprovingMusic	Improving Music Genre Classification Using Automatically Induced Harmony Rules	Anglade et al., 2010
3	ProbabilisticLogic	Probabilistic and Logic-Based Modelling of Harmony	Dixon et al., 2011
4	ApplicationILP	Application of ILP in a musical database: learning to compose the two-voice counterpoint	Pompe et al., 1996
5	LearningMusical	Learning Musical Rules	Morales and Morales, 1995
6	ExpressiveConcatenative	Expressive Concatenative Synthesis by Reusing Samples from Real Performances Recordings	Maestre et al., 2009
7	ConstructiveAdaptive	Constructive Adaptive User Interfaces - Composing Music Based on Human Feelings	Numao et al., 2002
8	ModelingMoods	Modeling Moods in Violin Performances	Perez et al., 2008
9	ModelingExpressive	Modeling Expressive Music Performance in Jazz	Ramirez and Hazan, 2005

### 3. Resultados e Análise

A seguir apresenta-se uma discussão sobre as abordagens. Considerando o inter-relacionamento entre os componentes levantados, alguns aspectos serão analisado juntamente com sua descrição.

#### 3.1. Qual arquitetura é proposta para representação do conhecimento musical?

Na representação musical, componentes como notação, formatos de arquivo, codificação e abstração de dados têm sido aplicados. SymbolicRepresentation [Chong and Ding, 2014] utiliza um sistema baseado em regras para incorporar conhecimento no processo de interpretação da informação musical. As regras são formuladas a partir de considerações como deslocamento da fundamental, movimentação do baixo e fatores ligados à condução da voz principal. A abordagem de ImprovingMusic [Anglade et al., 2010] é voltada para a identificação de estilos e gêneros musicas. Progressões harmônicas são tratadas utilizando a descrição sequencial de acordes. Realiza-se a combinação de descritores de áudio de baixo nível com uma ferramenta de aprendizado de máquina, o que possibilita a classificação baseada em cadências. Integrando-se ao mecanismo indutivo, elementos estruturais foram utilizados de modo a compor cada modelo de representação de conhecimento. A Tabela 2 apresenta esses elementos.

**Tabela 2: Elementos estruturais para representação de conhecimento**

	1	2	3	4	5	6	7	8	9
Utiliza o padrão MIDI			■			■	■		■
Realiza coleta analógica de dados								■	■
Realiza pré-processamento		■							■
Realiza síntese aditiva						■			
Estrutura básica: melodia									■
Estrutura básica: contraponto				■	■				
Estrutura básica: harmonia	■	■	■						
Técnica integrada: Algoritmos genéticos							■		
Técnica integrada: Mineração de dados									■
Técnica integrada: Processo estocástico			■						
Técnica integrada: Restrições lógicas	■				■				
Representação de textura polifônica						■			

ProbabilisticLogic [Dixon et al., 2011] apresenta duas abordagens voltadas à modelagem harmônica: probabilística e lógica. Com isso pode-se obter tanto a categoria de conhecimento musical, como também o raciocínio utilizado por um músico ao realizar tarefa análoga. Acordes são transcritos a partir de gravações de áudio. O sistema realiza a modelagem de contexto musical em alto nível. ApplicationILP [Pompe et al., 1996] desenvolveu uma ferramenta com a finalidade de realizar o processo de descoberta de conhecimento em uma base de dados com dezenas de milhares de instâncias. O sistema foi implementado de modo que a indução de uma hipótese é tratada como um problema de otimização. LearningMusical [Morales and Morales, 1995] utiliza a lógica de primeira ordem para expressar regras de contraponto, possibilitando que relações entre estados musicas sejam descritas de uma maneira compacta e compreensível. O modelo de ExpressiveConcatenative [Maestre et al., 2009] realiza a síntese expressiva por meio da aquisição de conhecimento obtido em gravações de áudio. ConstructiveAdaptive [Numao et al., 2002] lida com estruturas musicais capazes de causar sentimentos humanos característicos. Foi construído

um sistema para arranjo e composição de modo automático, que gera peças musicais com a capacidade de produzir sentimentos específicos em uma pessoa. A abordagem de ModelingMoods [Perez et al., 2008] é voltada ao aspecto instrumental de obras executadas no violino. O sistema armazena padrões expressivos que são adquiridos de modo automático. Em ModelingExpressive [Ramirez and Hazan, 2005] a abordagem tem o intuito de investigar a performance musical em melodias de *Jazz*. Utiliza técnicas de aprendizado de máquina para extrair movimentos regulares e padrões de desempenho.

### 3.2. Como a abordagem aplica os recursos computacionais ligados à PLI?

Os recursos computacionais utilizados envolvem mecanismos para indução lógica, ferramentas de pré-processamento, linguagens de programação, conceituação teórica para percepção e cognição melódica. O mecanismo de inferência lógica TILDE (*Top-down Induction of Logical Decision Trees*) fundamenta-se na lógica de primeira ordem e aplica a indução por meio de árvores de decisão. Sua funcionalidade é considerada uma extensão ao algoritmo C4.5 que em vez de testar valores de atributos em nós de uma árvore, testa predicados lógicos [Maestre et al., 2009]. Em ImprovingMusic, regras descrevendo padrões harmônicos de um dado gênero podem coexistir com regras de outros gêneros em uma mesma árvore. ProbabilisticLogic realiza a descrição de acordes em termos de sua nota fundamental, grau na escala e categorias de intervalos. ExpressiveConcatenative utiliza árvores proposicionais de decisão com técnicas de poda. ModelingMoods realiza a mineração de dados estruturados aplicando o algoritmo *top-down* em árvores de decisão. Os recursos e ferramentas computacionais utilizados em cada abordagem são apresentados na Tabela 3.

**Tabela 3: Recursos computacionais ligados à PLI**

	1	2	3	4	5	6	7	8	9
Aleph									
C4.5									
SFOIL									
Narmour									
SMSTools									
TILDE									
Pal									
JBoss Drools									
Indução a partir de Prolog									
Utiliza WEKA									
Utiliza base de treinamento adicional									

O sistema de programação lógica indutiva Aleph (*A Learning Engine for Proposing Hypotheses*) foi desenvolvido com o propósito de explorar ideias expressas por cláusulas de Horn com alta capacidade representativa. Escrito em Prolog, possibilita a descrição de expressões complexas, incorporando simultaneamente o conhecimento adquirido, com a capacidade de escolha da ordem de geração das regras, alteração nas funções de avaliação e busca. Em ProbabilisticLogic, esses conceitos são aplicados para se encontrar um conjunto mínimo de regras que seja capaz de descrever a totalidade das amostras positivas e um número mínimo de amostras negativas. ModelingExpressive utiliza seu algoritmo padrão para manipulação em conjuntos com a finalidade de construir hipóteses individuais. Narmour é uma teoria para percepção e cognição de melodias aplicada à análise musical. Auxilia na compreensão tanto do significado melódico como do conhecimento envolvido em sua criação. ExpressiveConcatenative utiliza o modelo de implicação/realização dessa teoria, onde cada nota pertence a uma estrutura Narmour. ModelingMoods utiliza o contexto Narmour, onde são definidos grupos específicos com os quais uma nota mantenha uma relação de pertinência. ModelingExpressive faz uso extensivo deste conceito, incorporando informações sobre notas prévias e sucessoras, além de propriedades intrínsecas a intervalos.

### 3.3. Análise

Com relação aos componentes de Musicologia tratados, destacam-se a caracterização de gênero e estilo, aplicados à performance expressiva, composição automática e instrumentação. No que diz respeito aos experimentos e testes, realizaram-se considerações sobre complexidade computacional, *benchmarks* e avaliação de acurácia, entre outros. A Tabela 4 apresenta os elementos abordados. Destacam-se as ferramentas Aleph e TILDE que realizam a indução lógica sobre predicados. Por meio da modelagem Narmour, melodias são expressas por listas de estruturas sobrepostas, possibilitando a representação conveniente de agrupamentos.

Tabela 4: Musicologia, Experimentos e Testes

	1	2	3	4	5	6	7	8	9
<b>Elementos da Musicologia</b>									
Caracterização de estilo									
Caracterização de gênero									
Análise em nível de áudio									
Análise em nível melódico/harmônico									
Funções tonais									
Graus diatônicos									
Representação de partitura									
Sistema Modal									
Processamento interativo									
Voltado à performance expressiva									
Voltado à composição automática									
Voltado à instrumentação									
Propósito educacional									
<b>Experimentos e Testes</b>									
Considerações sobre complexidade computacional									
Estudo por processos empíricos									
Avaliação por músicos e não músicos									
Realiza benchmark									
Realiza beta-teste									
Realiza testes completos									
Avaliação de acurácia									

## 4. Conclusão

Com relação à arquitetura para representação de conhecimento, observou-se a importância fundamental de 3 elementos: linguagem declarativa, mecanismo para implementação da indução lógica e técnica integrada para aquisição de conhecimento. Nesse sentido, destaca-se a linguagem Prolog, que aliada a um mecanismo de indução como Aleph ou TILDE, possibilita elevada capacidade representativa, além de simplicidade ao se apresentar tal conhecimento.

## Referências

- Anglade, A., Benetos, E., Mauch, M., and Dixon, S. (2010). Improving music genre classification using automatically induced harmony rules. *Journal of New Music Research*, 39(4):349–361.
- Chong, E. K. M. and Ding, Q. (2014). Symbolic representation of chords for rule-based evaluation of tonal progressions.
- Dixon, S., Mauch, M., and Anglade, A. (2011). Probabilistic and logic-based modelling of harmony. In *Exploring Music Contents*, pages 1–19. Springer.
- Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., and Linkman, S. (2009). Systematic literature reviews in software engineering—a systematic literature review. *Information and software technology*, 51(1):7–15.
- Maestre, E., Ramírez, R., Kersten, S., and Serra, X. (2009). Expressive concatenative synthesis by reusing samples from real performance recordings. *Computer Music Journal*, 33(4):23–42.
- Morales, E. and Morales, R. (1995). Learning musical rules. In *Proceedings of the International Joint Conference on Artificial Intelligence*. Citeseer.
- Numao, M., Takagi, S., and Nakamura, K. (2002). Constructive adaptive user interfaces—composing music based on human feelings. In *Proceedings of the National Conference on Artificial Intelligence*, pages 193–198. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.
- Perez, A., Ramirez, R., and Kersten, S. (2008). Modeling moods in violin performances. In *SMC 08: 5th Sound and Music Computing Conference: Sound in Space-Space in Sound, July 31st-August 3rd, 2008, Berlin, Germany: Proceedings*, page 30. Universitätsverlag der TU, Universitätsbibliothek.
- Pompe, U., Kononenko, I., and Makse, T. (1996). An application of ilp in a musical database: Learning to compose the two-voice counterpoint.
- Ramirez, R. and Hazan, A. (2005). Modeling expressive music performance in jazz. In *FLAIRS Conference*, pages 86–91.

# V-Conga: Um Estudo de Caso Para Avaliar as Capacidades de Captura de Nuances Gestuais do Leap Motion© em um Sistema Híbrido

Eduardo S. Silva<sup>1</sup>, Giordano R. E. Cabral<sup>1</sup>, Geber L. Ramalho<sup>1</sup>

<sup>1</sup>Centro de Informática – Universidade Federal de Pernambuco (UFPE) Caixa Postal 50.740-560- Recife – PE – Brasil.

{ess, grec, glr}@cin.ufpe.br

**Abstract.** *Gesture interfaces have been expanding the possibilities of creating new Digital Musical Instruments (DMIs). The Leap Motion© Controller offers a fine-grained hand tracking capabilities. In this study, the gesture recognition precision and the system's latency are evaluated by simulating a conga drum. In addition, we present an experiment done with a professional musician to investigate the sensor's capabilities within the instrument's context of use.*

**Resumo.** *Interfaces gestuais têm expandido as possibilidades de criação de novos Instrumentos Musicais Digitais (DMIs). O Leap Motion© Controller oferece um rastreamento de mãos com alta granularidade. Neste estudo, avalia-se a precisão do reconhecimento de gestos e latência do sistema ao simular uma conga. Em seguida, apresentamos um experimento feito com um músico profissional para investigar as capacidades do sensor dentro do contexto de uso do instrumento.*

## 1. Introdução

Nos últimos anos, o surgimento de dispositivos como smartphones, tablets e controles de videogame influenciou o rompimento dos paradigmas de Interação Humano-Computador (HCI) consolidados ao longo do tempo, afastando-se das interfaces baseadas no paradigma da máquina de escrever. Essas interfaces gestuais despertaram o interesse da comunidade científica pelas possibilidades de aplicações em diversas áreas de pesquisa, como é possível encontrar na literatura.

Na área de computação musical, esses avanços possibilitam novas interações com a música e a criação dos chamados Instrumentos Digitais Musicais (DMIs). Diversos instrumentos têm sido desenvolvidos utilizando diferentes controles de movimento [Jordà et al., 2005; Silva, 2012; Rosa-Pujazón et al., 2013].

Apesar do crescente número de pesquisas e demonstrações (formais e informais<sup>1</sup>), poucos DMIs estão sendo utilizados por músicos profissionais. Limitações de algumas tecnologias dificultam o uso profissional [Rosa-Pujazón et al., 2013; Todoroff, Leroy and Picard-Limpens, 2011]. Pesquisas indicam problemas relativos a latência excessiva, ausência de *feedback* visual/háptico e carência de nuances gestuais (problemas relativos à expressividade) [Silva, 2012].

Novas tecnologias mostram grande potencial para construção de DMIs expressivos. Dispositivos como o Leap Motion Controller© oferecem um rastreamento de gestos com uma alta precisão.

---

<sup>1</sup> Demonstrações disponibilizadas em websites de compartilhamento de vídeo, e.g. Youtube e Vimeo.

Avaliações preliminares do dispositivo [Han and Gold, 2014; Silva et al., 2013] observaram diversas limitações. Porém, mostram o potencial de captura de nuances gestuais. Requer-se uma análise aprofundada para avaliar a captura dessas nuances em instrumentos que demandam maior precisão espacial, principalmente após as mudanças no algoritmo de rastreamento do Leap Motion© (SDK v2.0). Outras pesquisas também utilizam simulação de instrumentos de percussão para avaliar tecnologias [Collicutt, Casciato, and Wanderley, 2009].

Neste estudo, foi desenvolvido um DMI (V-Conga) que simula um instrumento de percussão. Utilizou-se os princípios de integração de múltiplos sensores, baixo custo de implementação e um processo de *design* centrado no usuário.

## 2. Escolha do Instrumento

As congas foram escolhidas por possuírem um vasto repertório de gestos e nuances gestuais, por terem características físicas e de geração sonora similares à diversos outros instrumentos (bongôs, atabaques, etc.) e por serem e utilizadas em diversos ritmos musicais.

Os **Gestos Específicos** analisados produzem as notas: **Open Tone**, **Muffle Tone** (*Open Tone* abafado), **Slap Tone**, **Closed Slap Tone** (*Slap Tone* abafado), **Pressed Slap Tone** (*Slap Tone* abafado com a outra mão), **Bass Tone**, **Heel Tone**, **Tip Tone**, **Gliss** (ou *Glissando*) e algumas nuances como os **Flams** e uma técnica de mão fechada em punho (que foi denominado **Fist Note**). As diferenças nos gestos estão relacionadas com a posição da mão no instrumento, o formato da mão e o ângulo de ataque entre a mão e a pele do instrumento.

## 3. Sistema

O sistema (Figura 1) conta com uma superfície de duas camadas de madeira revestida de um emborrachado preto (para evitar reflexão no Leap Motion©), nela estão conectados dois sensores (situados entre as duas camadas). O Leap Motion© fica acima do sistema, em um pedestal comum e rastreando a cena (de cima para baixo). Mudanças no SDK permitiram uma maior a estabilidade em outras orientações de rastreamento.



Figura 1: Configuração do protótipo versão 1.0.

Os outros sensores integrados ao sistema foram em sensor piezoelétrico, ou piezo, para melhorar a latência do sistema (captura o ataque do gesto através da vibração na superfície) e um sensor de pressão FSR (*Force Sensing Resistor*), que funciona como um resistor variável, para captar nuances de abafamento das notas do instrumento (pela variação da resistência). Ambos os sensores foram integrados por meio de uma placa Arduino.

## 4. Resultados

Os aspectos focados foram a latência e o reconhecimento dos gestos específicos da conga. Além disso, foi feito um experimento com um músico profissional utilizando uma abordagem qualitativa [Barbosa et al., 2011]. Os valores apresentados representam a média de 30 medidas (para a latência) e 60 medidas (para cada gesto da conga).

Na **avaliação da latência** (Tabela 1), utilizou-se um microfone comum para capturar os sons do ataque na superfície e do *samples* disparados. A latência foi aprimorada ao longo da construção do artefato, escolhendo novas abordagens e implementando novos sensores.

**Tabela 1: Latência do artefato desenvolvido.**

Versão	Crystal Piano (Java)	0.1	0.2 <sup>2</sup>	1.0 <sup>3</sup>	V-Conga
Latência (ms)	150	71	17	7	7

A **avaliação dos gestos** (Tabela 2) ocorreu em duas etapas: com “**uma mão**” e depois com “**duas mãos**” dentro do campo de visão do sensor. Na segunda etapa, cada gesto foi analisado com cada uma das mãos. Erros são notas falso-negativas e disparo de *samples* incorretos.

**Tabela 2: Reconhecimento dos gestos da conga com apenas uma mão.**

Notas	Primeira Etapa (Percentagem Erro)		Segunda Etapa (Percentagem Erro)	
	Mão 1	Mão 2	Mão 1	Mão 2
Bass	6,67 %		100 %	100 %
Open	1,67 %		3,33 %	95 %
Muff	0 %		0 %	91,67 %
Slap	1,67 %		3,33 %	93,33 %
Closed Slap	0 %		8,33 %	86,67 %
Pressed Slap	-		0 %	91,67 %
Heel	0 %		0 %	95 %
Tip	3,33 %		0 %	100 %
Fist	0 %		0 %	98,33 %

Com uma mão, todos os gestos são capturados com poucos erros. Flutuações podem afetar o reconhecimento dos gestos (e.g. as notas *Tip* e *Bass*). Maior tolerância no rastreamento pode resolver o problema. Na segunda etapa, quase todos os gestos foram reconhecidos com estabilidade na “mão 1”, enquanto que a “mão 2” apresentou diversas falhas, por oclusão e perdas de rastreamento da segunda mão. A baixa robustez apresentada pelo Leap Motion®, na orientação “de cima para baixo” e rastreando as duas mãos, pode alterar os parâmetros de descrição da cena, causando erros. Apesar dos erros, o sistema foi capaz de detectar a maioria dos gestos e nuances do instrumento. Em contrapartida, a presença de erros, mesmo que poucos, pode tornar o instrumento imprevisível e inviável no contexto profissional.

O Leap Motion® não foi capaz de capturar algumas nuances de expressividade, como o abafamento das notas, necessitando que outro sensor fosse integrado. Entretanto, outros parâmetros podem ser utilizados para isso (mão aberta/fechada).

<sup>2</sup> Piezo integrado ao sistema desenvolvido em puredata.

<sup>3</sup> Sistema portado para C/C++. Driver ASIO integrado ao sistema.

Outras nuances não foram capturadas devido à arquitetura do sistema e às limitações do campo de visão do sensor (*Glissando* e abafamento com o cotovelo, respectivamente).

**O experimento com o músico profissional**, de avaliação qualitativa, envolve 3 etapas: exploração livre, realização de tarefas musicais (ritmos simples) e entrevista semiestruturada.

A instabilidade do sistema dificultou a execução das tarefas pelo usuário, durante a segunda etapa, salvo naquela realizada usando apenas uma mão. Segundo o músico o DMI possui pouca tolerância a erros de execução, o que não ocorre com o instrumento acústico. Há falhas se os gestos são executados com pouca precisão. Isso afeta seu raciocínio e gera a necessidade de ele reaprender o instrumento que já domina.

## 5. Conclusões e Trabalhos Futuros

Neste artigo, avaliou-se a capacidade do Leap Motion© de capturar nuances gestuais para construção de DMIs. O Leap Motion© foi essencial para o reconhecimento dos gestos e nuances das congas. O sistema conseguiu capturar a maioria das nuances, com baixa percentagem de erro, mostrando o potencial de captura desse dispositivo.

A instabilidade observada indica que novos avanços devem ser realizados para a construção de um DMI para músicos profissionais. Outras técnicas (e.g. visão computacional, aprendizagem de máquina) devem ser investigadas para solucionar o problema da segunda mão. Entretanto, a implementação de algoritmos mais complexos pode afetar a latência do sistema. Outra possibilidade é modificar a superfície de contato, para eliminar possíveis problemas de rastreamento. Ademais, pode-se usar dois sensores Leap Motion©. Outros experimentos podem ainda ser realizados com músicos profissionais.

## 6. Referências

- Barbosa, J., Calegario, F., Magalhães, F., Teichrieb, V., Ramalho, G. & Cabral, G. (2011). Towards an evaluation methodology for digital music instruments considering performer's view: a case study. In *Proceedings of 13th Brazilian Symposium on Computer Music*.
- Collicutt, M., Casciato, C., & Wanderley, M. M. (2009, June). From real to virtual: A comparison of input devices for percussion tasks. In *Proceedings of NIME* (pp. 4-6).
- Jordà, S., Kaltenbrunner, M., Geiger, G. and Bencina R. (2005). The reactable. *Proceedings of the international computer music conference (ICMC 2005)*, Barcelona, Spain, pages 579-582. CiteSeer.
- Rosa-Pujazón, A., Barbancho, I., Tardón, L. J., & Barbancho, A. M. (2013). Drum-hitting gesture recognition and prediction system using kinect. In *I simposio espanol de entrenamiento digital seed* (pp. 108-118).
- Silva<sup>1</sup>, E. S., de Abreu<sup>1</sup>, J. A. O., de Almeida<sup>1</sup>, J. H. P., Teichrieb, V., & Ramalho, G. L. (2013). A preliminary evaluation of the leap motion sensor as controller of new digital musical instruments.
- Silva, J. V. S., (2012). *Avaliando Interfaces Gestuais Para Prática de Instrumentos Virtuais de Percussão*. Dissertação de Mestrado. Universidade Federal de Pernambuco, Recife.

# Investigação sobre Combinações de Frases Percussivas através de Levantamento de Hipóteses e Análise Computacional

Luca Bezerra<sup>1</sup>, Geber Ramalho<sup>1</sup>, Giordano Cabral<sup>1</sup>

<sup>1</sup>Centro de Informática – Universidade Federal de Pernambuco (UFPE)  
CEP 50670-901 – Recife – PE – Brasil

lucabezerra@gmail.com, {glr,grec}@cin.ufpe.br

**Abstract.** *When studying computer music and music itself, a very important element receives the spotlight: the rhythm. Before trying to answer any questions about rhythm and its generation, we must first find out the answer to a more fundamental question: how to know if two rhythms match one another? We have interviewed specialists in the field to obtain their hypothesis about this question, we chose the one that seemed more suitable for being used in the research and we have achieved results that allowed us to refine the initial hypothesis and bring new ways of looking at the available data.*

**Resumo.** *No estudo que tangencia as áreas de computação musical e a própria música, um elemento essencial fica em evidência: o ritmo. Antes de responder quaisquer questionamentos sobre o ritmo e sua geração, é preciso chegar à resposta de uma pergunta mais fundamental: como saber se dois ritmos combinam? Entrevistamos especialistas na área para obter suas hipóteses sobre esta pergunta, escolhemos aquela que apresentava o maior potencial para ser utilizada e obtivemos resultados que nos permitiram refinar a hipótese inicial e trazer novas interpretações dos dados.*

## 1. Introdução

Uma problemática bem conhecida no mundo da música é a definição de “o que faz uma composição ser considerada boa ou agradável?”. Essa problemática é relevante porque, dentre outros motivos, define o grau de adequabilidade a critérios musicais que um conteúdo gerado automaticamente pode atingir. Apesar de ser possível definir regras e condições para que as composições sejam criadas, baseando-se em características extraídas de acervos pré-existentes, por exemplo, existem alguns detalhes mais profundos, muitas vezes implícitos (e até inconscientes) para quem toca, que regem a criação de uma boa música.

## 2. Problema

Dado esse contexto, é fácil perceber que descrever de forma precisa e objetiva as características que fazem uma composição ser considerada “boa” tem sido um dos grandes desafios da musicologia. Carlos Sandroni, por exemplo, se dedica a estudar as características do Samba em [Sandroni 2001].

### 2.1. Ritmo

Toussaint (2013) através de citação de Monahan et al (1985), diz que, dos muitos componentes da música, dois se destacam: ritmo e melodia, sendo o ritmo comumente associado com tempo e o eixo horizontal numa partitura ocidental típica. Na literatura é

possível encontrar diversas pesquisas que abordam o estudo dos componentes da música, como em Malgaonkar et al. E Wu et al. (2013, 2014). Enquanto cada pesquisa traz uma abordagem diferente, muitas delas possuem um aspecto em comum: o foco em harmonia e melodia [Alfonseca, et al. 2007; Gustafson 1988; Gillet et al. 2004]. Apesar destes serem elementos extremamente importantes, percebe-se uma deficiência de estudos sobre o ritmo, área esta talvez até mais essencial dentro do espectro musical [Sampaio et al. 2005; Sampaio 2006; Gouyon et al. 2005; Honing 2002]. Embora o ritmo seja tão essencial e onipresente, não há ainda uma teoria amplamente aceita sobre essa dimensão musical [Sampaio 2006], de acordo com Weyde e Honing (2001, 2002). Indo contra a tendência e adentrando então no estudo do ritmo, chegamos em uma de suas questões fundamentais: quais as características e os fatores que fazem com que uma determinada levada percussiva combine seja musical e ritmicamente com outra?

### 3. Estado da Arte

Em [Toussaint 2013], o autor transforma frases rítmicas em figuras geométricas e tenta analisar características que essa nova forma de visualização permite observar, como a noção de máxima regularidade dentro de uma frase, a imparidade rítmica, batidas fora do tempo, complexidade métrica, dentre outras. Pesquisamos métodos de representação de dados musicais que variavam desde notações clássicas, como partituras e tablaturas, até outras menos convencionais, como a notação cíclica [Toussaint 2004], a notação TEDAS [Gustafson 1988], e notações binárias, como a TUBS (*Time-Unit Box System*). Buscamos também trabalhos que utilizam as mais diversas técnicas para gerar composições rítmicas, desde a utilização de aprendizagem de máquina e modelos de similaridade para amostras de bateria, em [Pampalk et al. 2008], até algoritmos genéticos, como em [Alfonseca, et al. 2007]. Não sendo possível ou fácil aplicar aprendizagem de máquina, tomamos outro caminho, seguindo o método científico clássico de formulação e verificação de hipóteses por meio empírico. Realizamos entrevistas com especialistas na área (músicos e estudiosos) para elicitarem hipóteses e buscamos definir formas de representá-las computacionalmente, para que fosse possível testá-las de forma automatizada.

### 4. Formulação da Hipótese

A partir das entrevistas e da revisão da literatura, compilamos as características mais relevantes para esta pesquisa. A teoria que consideramos mais apta foi a do Equilíbrio de Coincidências (EC), mencionada tanto em [Sampaio 2006] como nas entrevistas. Levando em conta o casamento de duas levadas rítmicas, a medida de coincidências tenta encontrar o equilíbrio ideal entre batidas coincidentes e não-coincidentes, sendo a única teoria levantada que de fato se mostrou factível de ser utilizada como métrica.

Dada a hipótese do EC, definimos que seu valor seria calculado a partir da média aritmética das batidas dos instrumentos durante uma composição ou frase rítmica. Em outras palavras, a quantidade de instrumentos observada a cada momento em que algo foi tocado seria acumulada e então dividida pela quantidade de ocorrências do tipo, resultando no valor do Equilíbrio de Coincidências daquela amostra.

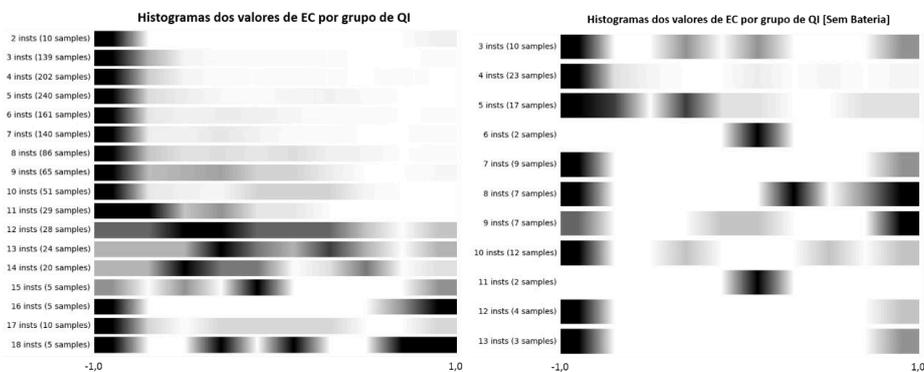
### 5. Representação e Processamento dos Dados

Reunimos uma base de amostras musicais e quantizamos seus dados, possibilitando assim dividi-la em grandes grupos, de acordo com a quantidade de instrumentos participantes (QI) de cada amostra (ou seja, colocamos as amostras com QI = 2 num grupo; as amostras com QI = 3 noutro grupo, etc). Calculamos então o valor do EC das amostras de cada

grupo de QI, normalizamos esses valores dentro de cada grupo para uma escala entre -1,0 e 1,0 e geramos os histogramas da distribuição dos valores de EC por grupo de QI. Os gráficos com uma análise mais aprofundada podem ser vistos em [Dias 2015].

## 6. Teste da Hipótese

Cada uma das linhas da Figura 6.1 representa a distribuição dos valores de EC normalizados, para um grupo de QI. No começo de cada linha é possível ver a QI das amostras daquela distribuição, assim como a quantidade de amostras que estão ali representadas. Cada gradiente varia de -1,0 a 1,0 e quanto mais escura for uma região, maior é a quantidade de amostras daquele grupo que possuem EC dentro daquele mesmo intervalo. Quanto maior for a concentração de áreas escuras do lado esquerdo dos histogramas, menores são os valores do EC das amostras em média, e quanto mais à direita se concentrarem, significa que as amostras, em média, tendem a apresentar maior número de coincidências entre batidas.



**Figura 6.1 - Histogramas com amostras de bateria e batucada (A) e (B).**

Na Figura 6.1 (A), é possível perceber uma grande concentração de amostras nos menores valores de EC (lado esquerdo) até por volta do grupo de QI = 10. Daí em diante, nota-se uma melhor distribuição das amostras ao longo do histograma, atingindo inclusive o extremo lado direito (valores maiores de EC), mas não é possível traçar um padrão de comportamento. Entende-se então que, para uma base que inclui amostras de bateria, mesmo aumentando a quantidade de instrumentos (QI) numa amostra, grandes quantidades de amostras apresentarão valores baixos de EC. Já na Figura 6.1 (B), que representa apenas as amostras de batucadas (sem bateria), apesar de também haver uma concentração de amostras do lado esquerdo dos histogramas (valores menores), é possível perceber, desde números menores de QI, que já há uma presença considerável de amostras nas faixas de valores de EC maiores. Isso indica que provavelmente as batucadas tendem a apresentar maiores valores de EC, em média, do que amostras de bateria, o que corrobora nossa premissa de que as amostras de batucada são mais adequadas para o propósito desta pesquisa.

## 7. Conclusão

Pudemos observar que, quanto maior é a QI de uma composição, maior é o valor geral de EC. Talvez haja, então, uma tendência maior de músicos tocando juntos se complementarem e se sobreporem durante a execução das músicas do que tocarem isoladamente “na sua vez”. Além disso, observamos que, nas amostras de percussão sem bateria, costuma ocorrer uma concentração de amostras com valores altos de EC em relação às outras. Isso pode significar que, nas batucadas, as coincidências de batidas

costumam ser mais comuns, talvez inclusive uma característica-chave no processo de construção das mesmas. A obtenção de uma base com amostras mais bem distribuídas entre as QI's ajudaria a esclarecer todas essas teorias. Como uma das contribuições deste trabalho, criamos um sistema que possibilita testes de hipóteses musicais numa base de amostras percussivas, servindo como ferramenta de avaliação. Este ferramental foi capaz de, dada uma hipótese (do E.C.), automatizá-la e validá-la com composições percussivas reais, permitindo que um conceito subjetivo e pouco definido fosse avaliado de forma objetiva e gerasse resultados concretos. Acreditamos que o produto deste trabalho permite que outras áreas da ciência, como a Musicologia, possam se utilizar desse potencial para chegar ainda mais longe nos seus estudos e ao mesmo tempo possam entender melhor como essa parceria é benéfica para ambas as áreas. Demonstramos também como a computação pode ser utilizada para representar e analisar informação musical num patamar mais empírico, fugindo das análises puramente teóricas.

## References

- Sandroni, C. (2001) "*Feitiço decente: transformações do samba no Rio de Janeiro, 1917-1933*", Editora UFRJ, Rio de Janeiro.
- Toussaint, G. (2013) "*The Geometry of Musical Rhythm: What Makes a "Good" Rhythm Good?*" s.l. : Chapman and Hall/CRC Press.
- Toussaint, G. (2004) "*A Comparison of Rhythmic Similarity Measures*". Technical Report SOCS-TR-2004.6, School of Computer Science, McGill University, Canadá.
- Monahan, C. B. e Carterette, E. C. (1985), "Pitch and Duration as determinants of musical space", *Music Perception: An Interdisciplinary Journal*.
- Malgaonkar, S., et al. (2013), "An AI Based Intelligent Music Composing Algorithm: Concord", *International Conference on Advances in Technology and Engineering*.
- Alfonseca, M., Cebrián, M. e Ortega, A. (2007), "A Simple Genetic Algorithm for Music Generation by means of Algorithmic Information Theory", *IEEE Congress on Evolutionary Computation*.
- Gustafson, K. (1988), "The Graphical Representation of Rhythm", *Progress Reports from Oxford Phonetics (PROPH)*. University of Oxford, Vol. 3.
- Gillet, O. e Richard, G. (2004), "Automatic Transcription of Drum Loops", *IEEE International Proceedings on Acoustics, Speech and Signal Processing*.
- Sampaio, P. (2006), "CInBalada - Um Laboratório Rítmico", UFPE.
- Sampaio, P., Tedesco, P. e Ramalho, G. (2005), "CInBalada: um Laboratório Multiagente de Geração de Ritmos de Percussão", *Proceedings of the X Brazilian Symposium on Computer Music (SBCM)*.
- Honing, H. (2001), "From time to time: The representation of timing and tempo", *Computer Music Journal*, Vol. 25
- Weyde, T. (2001), "Grouping, Similarity and the Recognition of Rhythmic Structure", *Proceedings of the International Computer Music Conference*.
- Wu, C., Liu, C. e Ting, C. (2014), "A Novel Genetic Algorithm Considering Measures and Phrases for Generating Melody", *IEEE Congress on Evolutionary Computation*.
- Pampalk, E., Herrera, P. e Goto, M. (2008), "Computational Models of Similarity for Drum Samples", *IEEE Transactions on Audio, Speech and Language Processing*, Vol. 16.
- Dias, L. (2015), "Levantamento, Representação e Análise, Computacional de Hipóteses Sobre Combinações de Frases Percussivas", *Dissertação de Mestrado, Centro de Informática, UFPE*.

## Um Estudo sobre as TICs como Ferramentas na Educação Musical de Alunos com Deficiência Visual

Alexandre Santos<sup>1</sup>, José Fornari<sup>2</sup>, Vilson Zattera<sup>3</sup> Adriana Mendes<sup>4</sup>

<sup>1</sup> Departamento de Música, Instituto de Artes (IA) – UNICAMP

<sup>2</sup> Núcleo Interdisciplinar de Comunicação Sonora (NICS) – UNICAMP

<sup>3</sup> Laboratório de Acessibilidade (LAB) – UNICAMP

<sup>4</sup> Departamento de Música, Instituto de Artes - UNICAMP

{alexjazzbass, tutifornari, vilson.zattera, aamende65}@gmail.com

**Abstract:** *this article presents a theoretical study on the use of ICTs in aid to the musical education of students with visual impairment. Here are described some of the most used technological tools to promote accessibility to music, such as screen reader softwares and tactile methodologies, using miniature of musical instruments (handmade or 3D printed) . This study aims to help bridge the gap of social inclusion of music students with visual disabilities through alternative technologies that promote their digital inclusion.*

**Resumo:** *este trabalho apresenta um estudo teórico sobre a utilização das TICs no auxílio à educação musical dos alunos com deficiência visual. Serão aqui descritas algumas das ferramentas mais utilizadas para promover a acessibilidade à música, tais como os softwares leitores de tela e as metodologias de abordagem táctil, utilizando miniaturas de instrumentos musicais (feitos à mão ou em impressoras 3D). Este estudo busca auxiliar a suprir a existente lacuna de inclusão social dos estudantes de música com deficiência visual, através de alternativas tecnológicas que promovam a sua inclusão digital.*

### 1. Introdução

A visão é um dos órgãos mais significativos para a aquisição de informações externas. Louro (2012) nos diz que aproximadamente 75% de nossa percepção se dá por meio deste sentido (LOURO, 2012, p. 247). Uma falha no sistema visual pode comprometer a qualidade de vida e conseqüentemente o aprendizado do indivíduo deficiente, de diferentes e inesperadas maneiras. No caso dos músicos Deficientes Visuais (DV), existe uma série de barreiras que dificultam seu desenvolvimento. Embora exista uma crença de que os músicos cegos têm a audição mais apurada, Louro (2012) nos diz que, ainda que as pessoas DV se valham mais do sentido da audição e assim desenvolvam mais esta percepção para os eventos do cotidiano, não significa necessariamente que o ouvido musical destes indivíduos seja automaticamente desenvolvido, ou seja, audição aguçada do DV para entender sonicamente a realidade que o cerca é um processo cognitivo distinto daquele utilizado pelo seu ouvido musical (LOURO, 2012, p. 263).

Em contrapartida, a sociedade contemporânea tem, em grande parte, suas ações cotidianas estabelecidas em processos que envolvem o uso de tecnologias de informação e comunicação digital – as chamadas TIC – para a realização de tarefas do dia-a-dia. Segundo Ramos (2008), estão neste contexto incluídos dispositivos como: o

computador, *smarthphones* (telefones celulares com acesso à internet e que rodam aplicativos), *tablets* (similares aos smartphones, porém com telas maiores e sensíveis ao toque; as *touchscreens*) e o próprio *cyberspace*, viabilizado pela internet. O uso destes dispositivos tem influenciado significativamente o processo de desenvolvimento do músico e do estudante de música, pois proporcionam ao músico e/ou estudante muitas possibilidades para o aprendizado através de uma série de ferramentas como *softwares* e *hardwares* com características específicas que permitem desde a manipulação e gravação de áudio em formato digital, editoração digital de partituras e estudos interativos de teoria musical e percepção.

Os músicos DV naturalmente já sofrem com os obstáculos advindos da própria deficiência quando estudam música de maneira tradicional, pois, ainda são escassos os materiais escritos em Braille, como livros e partituras. Um dos grandes obstáculos para a inclusão digital dos DV é o fato de que as TIC, em sua imensa maioria, não são projetadas levando em conta possíveis usuários com deficiência visual. As TIC foram naturalmente desenvolvidas para se valer da existência do sentido da visão e de sua grande capacidade de coletar informação. Imagine utilizar um computador, *smartphone* ou *tablet* de olhos cerrados. Toda a informação disposta na tela seria, em princípio, descartada. Como alternativa, o que (e quando) ocorre é a adaptação das TIC criadas para indivíduos videntes (com visão normal) que são utilizadas com algumas modificações para tornarem-nas acessíveis aos usuários DV. Partindo destes pressupostos, o presente estudo pretende apontar possíveis caminhos metodológicos através da utilização das TIC para a educação musical de alunos DV.

## 2. Propostas Metodológicas

As tecnologias usadas como ferramentas de auxílio aos DV, são também chamadas de tecnologias assistivas. Estas são ferramentas de *softwares* e *hardwares* que propiciam a superação da lacuna entre o que uma pessoa com deficiência quer fazer e o que existe de infraestrutura tecnológica, que a permite executar tal tarefa. Segundo Hasher, a tecnologia assistiva “é constituída por equipamentos, dispositivos e sistemas que podem ser usados para superar as barreiras sociais referentes à infraestrutura e outros obstáculos vividos pelas pessoas com deficiência e que impedem a sua participação plena e igualitária em todos os aspectos da sociedade”. (HASHER, 2008, p. 4). Mesmo com o grande avanço tecnológico, que reflete em inúmeras contribuições para o estudo da música para pessoas DV, as tecnologias disponíveis, em sua grande maioria, apresentam algumas dificuldades de manipulação para este público, devido ao fato de que os *softwares* de música possuem suas interfaces quase que exclusivamente gráficas (visuais) e que muitas vezes não são acessíveis aos *softwares* leitores de tela; ou seja, *softwares* capazes de transformar em linguagem (através de um módulo sintetizador de voz) trechos de arquivos de texto disposto na tela, de modo que o usuário DV pode ter acesso sonoro (na forma de linguagem falada) ao conteúdo textual (na forma de linguagem escrita). São exemplos de leitores de tela os *softwares*: Dosvox, Virtual Vision, Jaws, Orca e o NVDA (Non Visual Desktop Access) TUDISSAKI, 2015, p. 67). Para que as tecnologias assistivas de fato representem uma possibilidade de inclusão para os músicos e alunos DV aos recursos oferecidos pela informática, devem ser criadas metodologias específicas para a utilização das TIC na educação musical às pessoas DV, bem como levar em consideração as possibilidades de possíveis adaptações das TIC já existentes.

### 3. Metodologias de desenvolvimento

A presente pesquisa, apresentada aqui de forma teórica, pretende ter a sua parte empírica desenvolvida no Laboratório de Acessibilidade da Biblioteca Central Cesar Lattes da UNICAMP. Serão elaboradas, discutidas e mapeadas as experiências com os *softwares* e *hardwares* propostos. A partir dos dados coletados e analisados serão desenvolvidas possíveis soluções para os problemas estudados. Posterior a esta etapa, será proposto um projeto piloto que poderá ser oferecido e desenvolvido gratuitamente nos centros de atendimento às pessoas DV de Campinas e Limeira - SP. Vale lembrar que, se necessário, também será feita uma atualização em informática e acessibilidade para os alunos participantes, pois, conforme dito anteriormente, o acesso à informática pelos DV ainda é fortemente dependente da informação visual e, portanto, um grande desafio.

Os *softwares* a serem usados no projeto serão as ferramentas de código-livre, que não apenas deixam de gerar custos para a sua aquisição, como permitem ser modificados e estendidos por usuários mais avançados. Por enquanto, estão no âmbito deste estudo, experimentos com o uso dos *softwares* Sonic Pi ([www.sonicpi.net](http://www.sonicpi.net)), Audacity ([www.audacity.org](http://www.audacity.org)) e o Anvil Studio ([www.anvilstudio.com](http://www.anvilstudio.com)). O Sonic Pi é um software livre e foi desenvolvido pelo Dr. Sam Aaron na Universidade de Crambridge. Foi projetado especificamente para sala de aula para incentivar os alunos a aprenderem a programar. Os parâmetros sonoros são inseridos e manipulados no programa através de linhas de texto em tempo real. O *software* também pode ser usado em performances ao vivo. Funciona nas plataformas Windows, Mac e Linux. O Audacity também é um *software* livre e de código aberto. É um editor de áudio e gravador multipistas. Funciona nas plataformas Windows, Linux e Mac OS X. Já o Anvil Studio é *Freeware*, ou seja, é gratuito, mas não tem o código aberto, além do mais se o usuário quiser pode ter acesso à recursos adicionais mediante pagamento de taxas. No entanto, sua versão *free* é suficiente para atender a proposta da presente pesquisa. O critério para escolha destes *softwares*, além do fato de serem de operação e manipulação aberta, também estão relacionados ao fato da possibilidade de que alguns deles permitirem a manipulação por linhas de texto - o que é vantajoso para o usuário DV, uma vez que este não faz uso do mouse, e outros que mesmo tendo a linguagem gráfica, também permitem o acesso com o leitor de tela, como no caso do Audacity.

Também fazem parte da pesquisa a utilização de instrumentos em miniatura, provenientes de criação através de impressoras 3D ou adquiridos através de coleções prontas. Com estas miniaturas os DV podem entender através do tato a anatomia de um instrumento musical e assim construir uma imagem mental deste objeto. Experiências futuras serão conduzidas com usuários DV explorando a miniatura através do sentido do tato ao mesmo tempo que ouvem o som equivalente ao instrumento musical correspondente à miniatura, associando assim o som do mesmo com as particularidades de sua construção. Por exemplo, explorando a miniatura de um trombone ao mesmo tempo que ouve este som típico, o DV poderá compreender o modo como este som é produzido, tendo uma noção da arquitetura dos seus tubos, vara e o bocal.

### 4. Procedimentos Metodológicos

Em relação à metodologia que será usada para o tratamento dos dados coletados, o método a ser usado nesta etapa será principalmente a Análise de Conteúdos

de Lawrence Bardin (2011). Esta metodologia pode ser usada tanto para relacionar dados quantificáveis como respostas à questionários fechados, como também pode ser usado para análise de entrevistas e filmagens dos voluntários participantes da pesquisa. (BARDIN, 2011, p. 125). Já para a questão pedagógica o método a ser utilizado como base teórica, será uma a metodologia chamada de Conhecimento Tecnológico Pedagógico do Conteúdo – CTPC ou *Technological Pedagogical and Content Knowledge* – TPACK (BAUER, 2014, p. 12). O quadro teórico aqui proposto é uma extensão ao método chamado Conhecimento Pedagógico do Conteúdo, proposto anteriormente por Lee Shulman (BAUER, 2010, p. 12). Misha e Koeller inseriram o componente tecnológico ao modelo de Shulman e então chegaram ao TPACK.

## 5. Considerações Finais

Este trabalho pretende contribuir para o surgimento de metodologias tecnológicas que possibilitem o processo a inclusão dos alunos DV no ambiente digital contemporâneo, sobretudo com as aplicações para o ensino e estudo de música. As TIC estão ubiquamente permeadas da vida cotidiana contemporânea e assim torna-se cada vez mais relevante criar alternativas de uso destas ferramentas para melhorar a qualidade de vida e acesso à informação para os usuários DV. O estudo aqui apresentado e a ser futuramente desenvolvido não ambiciona, obviamente, esgotar o assunto, mas busca instigar o seu aprofundamento, contribuindo para a discussão sobre a inclusão de pessoas deficientes visuais em todos os âmbitos sociais, em especial na área musical. Como dito inicialmente, a inclusão digital dos indivíduos DV é acima de tudo um direito, e trazer este enfoque para a pesquisa acadêmica contribui para que esta cumpra o seu papel, que é, entre outros, o de proporcionar melhorias e possíveis soluções para os anseios sociais.

## 6. Referências

- Bardin, Lawrence. *Análise de Conteúdo*. Tradução: Luís Antero Reto e Augusto Pinheiro. São Paulo: Editora 70, 2011.
- Bauer, Willian I. *Music Learning Today: Digital Pedagogy, Performing and Responding Music*. New York: Oxford University Press, 2014.
- Hersh, Marion A. e Jhonson, Michael A. *Assistive Technology for Visually Impaired and Blind People*. London: Springer – Verlag London Limited, 2008.
- Louro, Viviane. *Fundamentos da aprendizagem musical da pessoa com deficiência*. 1ª ed. São Paulo: Editora Som, 2012.
- Ramos, Sérgio. *Introdução às TIC*. In: AFONSO, Adriano. *Manual de Tecnologias da Informação e Comunicação e OppenOffice.org*. 2ª ed. Lisboa, 2010. Disponível em: [http://www.adrianoafonso.net/files/manuais/manual\\_tic\\_2ed.pdf](http://www.adrianoafonso.net/files/manuais/manual_tic_2ed.pdf)
- Tudissaki, Shirlei Scobar. *Ensino de Música para Pessoas com Deficiência Visual*. São Paulo: Cultura Acadêmica, 2015.

# Termpot: criação e edição de funções no navegador em tempo de execução.

Guilherme Lunhani<sup>1</sup>, Flávio Luiz Schiavoni<sup>2</sup>

<sup>1</sup>Instituto de Artes e Design – Universidade Federal de Juiz de Fora  
Juiz de Fora, MG

gcravista@gmail.com

<sup>2</sup>Departamento de Computação – Universidade Federal de São João Del Rei  
São João Del Rei, MG

fls@ufsj.edu.br

**Abstract.** This panel reports the development of a sound synthesis program, *Termpot*. The Web Audio API technology was used in a reinterpretation of a [Mathews and Moore 1970]’s work, *GROOVE*. The resulting work is still in its infancy, but allows create and edit sound functions at runtime, in a web browser.

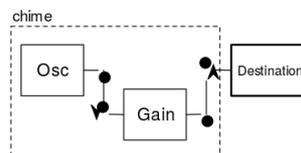
**Keywords:** *GROOVE*; Web Audio API; DSP.

**Resumo.** Este painel reporta o desenvolvimento de um programa de síntese sonora, *Termpot*. A tecnologia Web Audio API foi usada em uma reinterpretação de um trabalho de [Mathews and Moore 1970], *GROOVE*. O trabalho resultante ainda é incipiente, mas possibilita criar e editar funções no tempo de execução, em um navegador web.

**Palavras-chave:** *GROOVE*; Web Audio API; DSP.

## 1. Introdução

A síntese sonora em *web browsers* é sumarizada por [W3C 2012, Roberts et al. 2013, Wyse and Subramanian 2014]. [Srikumar 2013] exemplifica uma concatenação de *nós de áudio* em um grafo de DSP. Três nós diferentes são representados na figura 1. Existe um outro nó (figura 2) que possibilita definir formas de ondas customizadas. O *Termpot* explora este nó, e delega, como o *wavepot*<sup>1</sup>, as customizações para um(a) improvisador(a), que cria suas próprias *funções de transferência e interfaces de controle* [Mathews and Moore 1970].

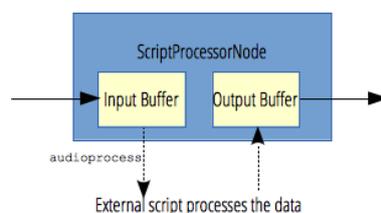


**Figura 1:** Estrutura básica de um sintetizador webaudio. *OscillatorNode*, *GainNode* e *DestinationNode* (alto-falante). **Fonte:** [Srikumar 2013]

## 2. Trabalho relacionado: GROOVE

Reinterpretamos um trabalho de [Mathews and Moore 1970], o sistema *GROOVE*, *Generated Real-time Operations On Voltage-controlled Equipment*. A compositora

<sup>1</sup>Disponível em <https://www.github.com/wavepot/wavepot>



**Figura 2:** “A interface do *ScriptProcessorNode* permite a geração, processamento ou análise de áudio usando JavaScript”. **Fonte:** [W3C 2012]

Laurie Spiegel (figura 3) sumariza suas características, durante a produção de *The Expanding Universe* (1975)<sup>2</sup>, entre as salas 2D-506 da Bell Labs (contendo o computador DDP-224) e a sala analógica 2D-562 (laboratório de Mathews):

Todas as músicas no GROOVE eram representadas na memória digital como funções abstratas do tempo, séries paralelas de dois pontos, cada ponto sendo um instante no tempo e um valor instantâneo. A taxa de amostragem para essas funções, usada principalmente como controle de voltagem, era cronometrada por um grande e antiquado oscilador analógico que era normalmente fixado em 100 Hertz, cada ciclo do oscilador pulsando à frente do código, o computador lia, em cada uma das funções, naquele ponto do tempo, todos dispositivos de entrada e executava todas amostras. (...) Tínhamos uma pequena caixa com 4 potenciômetros e quatro chaves (alternadores fixados onde você os colocava) e dois botões de disparo.<sup>3</sup>



**Figura 3:** Laurie Spiegel configurando a saída analógica do GROOVE, durante a produção de *The Expanding Universe*. **Fonte:** [Spiegel 1975].

<sup>2</sup>Disponível em <https://www.youtube.com/watch?v=dYUZmsfm4Ww>.

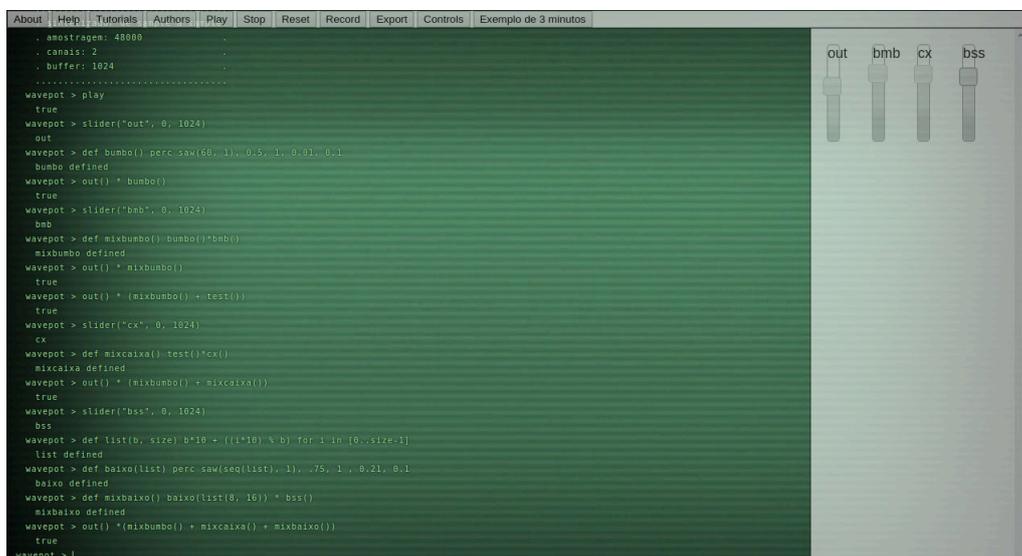
<sup>3</sup>Tradução de *All music in GROOVE was represented in digital memory as abstract functions of time, parallel series of point pairs, each point being an instant in time and an instantaneous value. The sampling rate for these functions, which would be used mostly as control voltages, was clocked by a big old-fashioned analog oscillator that was usually set to 100 Hertz, each cycle of the oscillator pulsing one run through the code, the computer reading all of the real time input devices and playing of all of the samples at that time point in each of the time functions. (...) We had a small box with 4 knobs, 4 set switches (toggles that stay where you put them) and 2 momentary-contact push buttons on it.*

### 3. Metodologia de desenvolvimento

1) Customização um emulador terminal *Ptty.js* (<http://code.patxipierce.com/jquery-plugin/ptty/>). 2) Definição de um ambiente interno, baseado no ambiente *Wavepot*, controlador do *ScriptProcessorNode*. 3) Definição de comandos deste ambiente interno: inspeção de funções, definição de novas funções, tocar, parar, pausar, gravar e download, criação de controles gráficos (*jQueryUI*) e gravação (<https://github.com/mattdiamond/Recorderjs/blob/master/recorderWorker.js>).

### 4. Resultados

O *Termpot* (ver figura 4)<sup>4</sup> é uma customização do *Wavepot*. Funções sonoras podem ser definidas em linguagem *coffeescript* [Burnham 2011], executadas e gravadas (Código 1).



```

About | Help | Tutorials | Authors | Play | Stop | Reset | Record | Export | Controls | Exemplo de 3 minutos
. amostras: 48000
. canais: 2
. buffer: 1024
.....
wavepot > play
true
wavepot > slider("out", 0, 1024)
out
wavepot > def bumbo() perc saw(60, 1), 0.5, 1, 0.01, 0.1
bumbo defined
wavepot > out() * bumbo()
true
wavepot > slider("bmb", 0, 1024)
bmb
wavepot > def mixbumbo() bumbo()*bmb()
mixbumbo defined
wavepot > out() * mixbumbo()
true
wavepot > out() * (mixbumbo() + test())
true
wavepot > slider("cx", 0, 1024)
cx
wavepot > def mixcaixa() test()*cx()
mixcaixa defined
wavepot > out() * (mixbumbo() + mixcaixa())
true
wavepot > slider("bss", 0, 1024)
bss
wavepot > def list(b, size) b*10 + ((i*10)%b) for i in [0..size-1]
list defined
wavepot > def baixo(list) perc saw(seq(list), 1), .75, 1, 0.21, 0.1
baixo defined
wavepot > def mixbaixo() baixo(list(8, 16)) * bss()
mixbaixo defined
wavepot > out() * (mixbumbo() + mixcaixa() + mixbaixo())
true
wavepot > |

```

Figura 4: Aplicativo *Termpot*. Fonte: autores.

### Problemas Técnicos

Existem *xruns*. Um *xrun* “(...) pode ser um estouro de buffer ou de uma saturação de um buffer. Um aplicativo de áudio ou não foi rápido o suficiente para transmitir dados (...) ou não rápido o suficiente para processar os dados”[User:Markc 2013]<sup>5</sup>. Existe uma hipótese, não verificada, que a possível fonte dos *xruns* é a biblioteca *jQuery*.

### 5. Conclusão

O *Termpot* está em estágio inicial de desenvolvimento. Com a reinterpretação de um caso histórico, desafios como síntese sonora, performance, e sintaxe simplificada em uma linguagem de programação, necessitaram de algum esforço. Por outro lado, existem problemas técnicos. Neste sentido, o desenvolvimento do *Termpot* aguarda contribuições da comunidade, com especial interesse em uma abordagem pedagógica para o ensino de processamento de sinais digitais de áudio na internet.

<sup>4</sup>Disponível em <https://jahpd.github.io/termpot>.

<sup>5</sup>Tradução nossa de *An "xrun" can be either a buffer underrun or a buffer overrun. In both cases an audio app was either not fast enough to deliver data (...) or not fast enough to process data.*

---

```

1 $ | (1)
2 $ wavepot 1024 (2)
3 .....
4 . sintetizador de sample a sample. (3)
5 . amostragem: 44100
6 . canais: 2
7 . buffer: 1024
8 .....
9 wavepot > play (4)
10 true
11 wavepot > def AM440(f, a) sin f, sin(440, a) (5)
12 AM440 defined
13 wavepot > inspect (6)
14 funcoes definidas
15 tau tmod mute stereo sin sin2 saw ramp ttri
16 tri sqr pulse noise perc test seq bpm nextevent
17 AM440
18 wavepot > slider "f", 1, 1025 (7)
19 true
20 wavepot > slider "a", 0, 1024
21 true
22 wavepot > record (8)
23 true
24 wavepot > AM440 f()*1000, a() (9)
25 true
26 wavepot > export (10)

```

---

**Código 1:** Pty aguardando dados de entrada do improvisador (1). *Boot* do ambiente *wavepot* com um buffer de 1024 amostras por ciclo de DSP (2). Informações diversas do sistema (3). Execução do DSP (4). Definição de uma função *AM440* (5). Informações sobre as funções disponíveis (6). Definição de GUIs (7). Gravação em um arquivo de áudio (8). Execução da função *AM440* com controles (9). Download da gravação (10)

## 5.1. Trabalhos Futuros

*i)* Criar um servidor; *ii)* otimizar o emulador, talvez substituindo o Pty ou propondo melhorias; *iii)* suporte para amostras pré-gravadas.

## 5.2. Agradecimentos

Guilherme Rafael Soares pelas sugestões, aos desenvolvedores do *Wavepot*, e a FAPEMIG por subsidiar a pesquisa.

## Referências

- Burnham, T. (2011). Coffeescript: Accelerated javascript development. *Pragmatic Bookshelf*.
- Mathews, M. V. and Moore, F. (1970). GROOVE a program to compose, store, and edit functions of time. *Bell Telephones Laboratories*, page 7.
- Roberts, C., Wakefield, G., and Wright, M. (2013). The web browser as synthesizer and interface. page 6.
- Spiegel, L. (1975). The expanding universe: 1970s computer music from bell labs by laurie spiegel. disponível em [http://www.retiary.org/ls/expanding\\_universe/](http://www.retiary.org/ls/expanding_universe/). *Retiary*.
- Srikumar, S. (2013). Taming the script processor node. disponível em <http://sriku.org/blog/2013/01/30/taming-the-scriptprocessornode/>.
- User:Markc (2013). Xruns: From the alsa wiki. disponível em <http://alsa.opensrc.org/Xruns>.
- W3C (2012). Web audio API.
- Wyse, L. and Subramanian, S. (2014). The viability of the web browser as a computer music platform. *Computer Music Journal*, 37(4):10–23.

# Musical Papers, Posters and Compositions

# Interação e convergência entre o instrumental e o eletrônico em *Poussières Cosmiques*

Danilo Rossetti<sup>1</sup>

<sup>1</sup>Instituto de Artes UNICAMP/Centre de recherche  
Informatique et Création Musicale Université Paris VIII

danilo\_rossetti@hotmail.com

**Resumo.** Neste artigo procuramos descrever e analisar o processo criativo da peça *Poussières cosmiques*, abordando sua concepção instrumental e os tratamentos eletrônicos. Os processos de interação e convergência entre estes dois meios são relacionados à ideia de morfologia sonora, fundamentada nos modelos ondulatório e granular de construção e/ou decomposição do som. Concluímos através da discussão a respeito da dualidade entre o contínuo e o descontínuo, a qual perpassa os diferentes níveis temporais da peça.

**Abstract.** In this article we describe and analyze the creative process of the work *Poussières cosmiques*, addressing its instrumental conception and the electronic treatments. The procedures of interaction and convergence between these two universes are related to the idea of sound morphology, based on the undulatory and granular models of sound construction and/or decomposition. We conclude addressing the duality between the continuous and the discontinuous, which passes by the different temporal levels of the work.

## 1. Introdução

*Poussières cosmiques*, obra para piano e eletrônica em tempo real, foi composta entre novembro de 2014 e fevereiro de 2015, durante o período sanduíche de nosso doutorado, junto ao CICM (*Centre de recherche Informatique et Création Musicale*) da *Université Paris VIII*. A peça possui duas versões, sendo a primeira uma versão pedagógica, interpretada pelo pianista Adrien Udo no concerto *Modes de Jeux* n° 8, realizado no auditório AmphiX da *Université Paris 8*, em 25 de março de 2015. Adrien Udo é aluno de piano do segundo ciclo do Conservatório de Saint Denis e construiu sua interpretação conjuntamente com seus professores que participaram do projeto.

A segunda versão, que apresenta uma escrita pianística mais elaborada, foi executada por Sophia Vaillant em seu recital *Musiques mixtes pour le piano*, em 12 de junho de 2014, também no AmphiX. Sophia Vaillant, além de executar a parte do piano, controlou a sequência dos efeitos eletroacústicos através de um pedal MIDI. A eletrônica das duas versões, cujos tratamentos são aplicados sobre o som captado do piano, apresenta os mesmos processos e a mesma programação, gerando um resultado sonoro semelhante. Neste artigo, analisaremos e discutiremos os processos de composição empregados na segunda versão da peça.

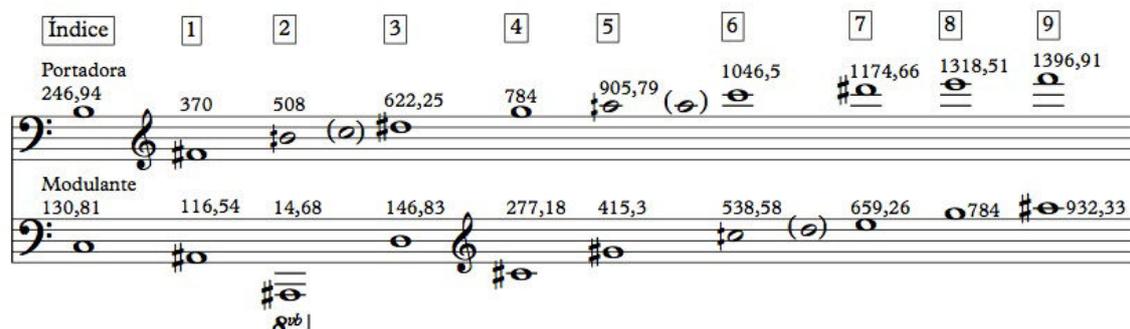
## 2. Concepção instrumental da obra

Considerando a ideia de síntese instrumental proposta por Gérard Grisey (1979, 1991), buscamos, como processo criativo, algumas possibilidades de utilização e transferência

de procedimentos utilizados na música eletrônica para o contexto da escrita instrumental. Desta feita, para a aplicação desta ideia, nos baseamos na concepção teórica da síntese por modulação de frequência [CHOWNING, 1973] para o estabelecimento das principais alturas musicais utilizadas na peça.

A síntese por modulação de frequência (FM), por definição, gera novos parciais a partir da interação entre duas frequências iniciais, uma onda portadora e outra modulante, apresentando como uma de suas principais características a possibilidade de implementação de modificações espectrais (no timbre) da onda resultante ao longo do tempo. Novas frequências são geradas a partir da adição e subtração das duas ondas envolvidas, sendo o valor desta última ainda dependente de sua multiplicação pelo índice de modulação ( $i$ )<sup>1</sup>. As novas frequências são geradas por este processo através da fórmula  $f = p + (i * m)$  ou  $f = p - (i * m)$ . Para a utilização deste procedimento no âmbito da música instrumental, as alturas foram determinadas através da transposição dos valores frequenciais em Hertz para alturas musicais<sup>2</sup> (considerando Lá 4 = 440Hz).

Como valores das ondas portadora e modulante, adotamos os valores frequenciais relativos ao intervalo de sétima maior entre o Dó 3 (130,81Hz) e o Si 3 (246,94Hz). A partir destes valores, utilizando a fórmula da síntese por modulação de frequência apresentada, geramos os seguintes alturas (aplicando a modulação de frequência dentro do conceito de síntese instrumental), com o índice de modulação variando entre 1 e 9.



**Figura 1. Geração das alturas a partir do processo de síntese instrumental por modulação de frequência**

Em relação às aproximações necessárias durante a conversão dos valores de frequência para alturas musicais, os índices 2, 5 e 6, por exemplo, apresentam valores em Hz que, quando convertidos para alturas, geraram intervalos de quartos de tom. Estes intervalos foram aproximados para os semitons cromáticos superiores.

Na manipulação das durações da peça, para a variação dos andamentos, utilizamos o procedimento de modulação métrica por *accelerandi* e *rallentandi*, tal como manejado pelo compositor José Manuel López López<sup>3</sup>. Para a modulação dos

<sup>1</sup> O índice de modulação ( $i$ ), cuja variação é proporcional à amplitude da onda modulante, representa o fator através do qual a onda modulante é multiplicada, antes de seu valor ser somado ou subtraído da onda portadora. O valor do índice é calculado pela razão entre o desvio de pico ( $d$ ) e o valor da frequência modulante ( $m$ ).

<sup>2</sup> No caso do piano, que é um instrumento de afinação temperada, algumas aproximações foram necessárias durante a transposição das frequências para alturas musicais.

<sup>3</sup> O método de modulação métrica utilizado por López López é similar ao conhecido método homônimo implementado por Elliott Carter.

andamentos, antecipa-se o novo valor da pulsação métrica através da utilização de quiálteras que correspondam ao andamento que se deseja alcançar. Neste contexto, transitamos entre os valores de 48, 60, 90, 72, 60, 75 e 94 pulsações por minuto, valores estes obtidos através das transições métricas apresentadas na tabela a seguir.

**Tabela 1: Modulações métricas em *Poussières cosmiques***

Andamento inicial	Andamento a ser atingido	Relação métrica
♩ = 48	♩ = 60	♩ <sub>5</sub> = ♪
♩ = 60	♩ = 90	♩ <sub>3</sub> = ♪
♩ = 90	♩ = 72	♩ = ♪ <sub>5</sub>
♩ = 72	♩ = 57,6 $\cong$ 60	♩ = ♪ <sub>5</sub>
♩ = 60	♩ = 75	♩ <sub>5</sub> = ♪
♩ = 75	♩ = 94	♩ <sub>5</sub> = ♪

### 3. Tratamentos eletrônicos

O *patch* utilizado na peça foi realizado em Max, a partir do objeto *process~*, que realiza diferentes tratamentos sonoros, além da espacialização do som em ambisonia. Este objeto está presente na biblioteca HOA (*High Order Ambisonics Library*), desenvolvida pelo CICM/*Université Paris VIII*. Na peça, tratamentos de granulação, decorrelação microtemporal, modulação em anel e *delay* são aplicados ao som captado do piano. Estes tratamentos foram utilizados separadamente ou através de uma combinação envolvendo dois deles, combinação esta definida em função da escrita pianística presente nos diferentes momentos da peça. Na nossa programação, os objetos *process~* realizam a espacialização em octofonia do som do piano captado, juntamente com os tratamentos determinados.

Descritivamente, o *patch* que implementamos funciona da seguinte forma: Há três microfones que captam o som do piano, cujos volumes de entrada podem ser controlados separadamente. Os três objetos que controlam o nível do sinal de entrada são enviados a dois objetos *gate~*, que definem a combinação dos tratamentos eletroacústicos aplicados ao som do piano. O primeiro objeto *gate~* faz a seleção entre os tratamentos de granulação (1), decorrelação microtemporal (2), modulação em anel (3), ou nenhum tratamento (0). O segundo objeto *gate~* seleciona os tratamentos de *delay* (1), decorrelação microtemporal (2), ou nenhum tratamento (0). Através da combinação entre os tratamentos selecionados pelos dois objetos *gate~* temos a produção do efeito eletroacústico desejado. Desta feita, podem haver momentos na peça em que não há nenhum processamento eletrônico (escolha do valor 0 nos dois objetos *gate~*), momentos em que há apenas um tratamento (valor 1, 2 ou 3 no primeiro *gate~* e valor 0 no segundo), ou momentos em que há dois tratamentos simultâneos (valor 1, 2, ou 3 no primeiro objeto e valor 1 ou 2 no segundo).

As saídas dos objetos *gate~* são conectadas aos seguintes objetos HOA *process~*: *Syn.Grain*, *Syn.Decorrelation*, *Syn.RingModulation* e *Syn.Delay*. Nestes objetos ainda ocorre a conversão do sinal mono para oito canais em modo ambisonia. Estes quatro objetos são colocados em paralelo e, como vimos, podem funcionar separadamente ou combinados em dupla. A saída destes objetos, em oito canais, é

enviada ao objeto *hoa.2d.decoder*, que decodifica os harmônicos espaciais da ambisonia e realiza sua conversão para oito sinais mono, os quais são enviados à saída do sistema em octofonia (*hoa.dac~1:8*). Uma amplificação direta do som captado do piano (cujo volume é definido separadamente) é enviada ao harmônico 0 do decodificador.

As valores aplicados às variáveis relativas aos tratamentos utilizados no objeto *process~* (granulação, descorrelação microtemporal, modulação em anel e *delay*) são registradas no objeto Max *patrstorage*, totalizando 27 *presets*. A transição entre os *presets* é realizada através de interpolações temporais cuja duração em segundos é estritamente definida. Desta maneira, a mudança do som resultante dos tratamentos é feita gradualmente, fazendo com que a alteração dos parâmetros dos efeitos não seja sentida de forma brusca pelos ouvintes. A seguir, apresentamos o diagrama em blocos do *patch* de *Poussières cosmiques*.

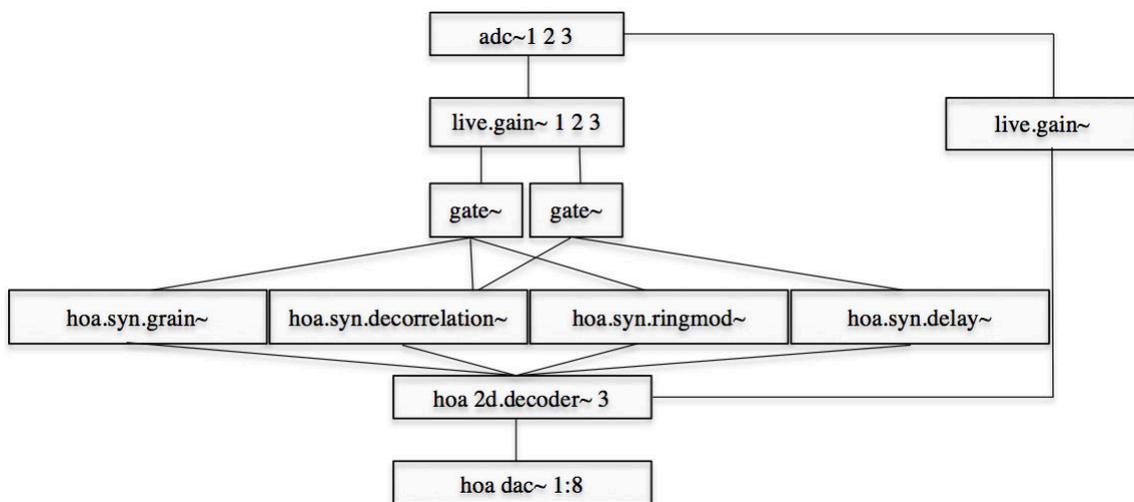


Figura 2: *Patch* de *Poussières cosmiques* representado em diagrama de blocos

#### 4. Interação e convergência

A escrita instrumental da peça é dividida em sete partes, para as quais foram atribuídas um tipo ou uma combinação de tratamentos eletrônicos. A escolha do tratamento eletrônico a ser aplicado à determinado tipo de escrita instrumental levou em consideração a morfologia resultante do som executado pelo pianista, principalmente em relação aos modelos granular (descontínuo) e ondulatório (contínuo) do som, questão esta que será desenvolvida mais adiante. Desta feita, os tratamentos atribuídos a cada parte da peça foram distribuídos da seguinte forma:

Parte 1: granulação

Parte 2: descorrelação microtemporal + *delay*

Parte 3: modulação em anel + *delay*

Parte 4: granulação

Parte 5: granulação

Parte 6: granulação + descorrelação microtemporal

Parte 7: granulação

Todas as sete partes da peça não têm uma relação direta entre si considerando os materiais utilizados, ainda que os tratamentos eletrônicos muitas vezes se repitam. A concepção formal da peça foi baseada na ideia de forma momento (*Momentform*), desenvolvida por Karlheinz Stockhausen (1960). A forma momento, de maneira geral, seria um tipo de justaposição de unidades de forma, ideia aplicada, entre outras, na sua obra *Kontakte* (1958 – 60). Por definição, o momento é toda a unidade de forma que possui, numa determinada composição, uma característica pessoal estritamente atribuível, possuindo um pensamento autônomo.

Apresentaremos, a seguir, três exemplos de combinações realizadas entre a escrita instrumental e os tratamentos eletroacústicos. No primeiro exemplo (Figura 3), referente aos compassos 26 a 28, temos uma escrita pianística concentrada na região extremamente grave do instrumento. Notas, em sua maioria semicolcheias, são escritas para as duas mãos, apresentando ritmos levemente diferentes, com o intuito de produzir pequenas decalagens temporais entre as duas vozes. A mão direita mantém um ritmo constante em semicolcheias, enquanto que a mão esquerda realiza variações métricas, passando por quáteras de 9:8, 11:8, 3:2, 6:4, 7:4, 5:4, 3:2, além de alguns trinados.

As pequenas variações temporais que produzem uma dessincronia entre as duas mãos geram auditivamente um tipo de granulação instrumental. Em convergência a este tipo de escrita pianística são aplicados os tratamentos da descorrelação microtemporal e do *delay*. A descorrelação microtemporal<sup>4</sup> é um tratamento similar ao *delay*, a qual gera a sensação de pequenas decalagens microtemporais entre as pistas (alto-falantes) de um sistema multicanal. Através destas pequenas decalagens e dependendo da fase do som (num plano de 360°) são produzidas mudanças na percepção espacial deste som através da criação de um campo sonoro difuso.

Como resultado desta combinação do instrumento com a eletrônica, temos a criação de um campo sonoro difuso (que é espacializado em oito canais), oriundo do som captado e transformado do piano pelos efeitos mencionados. Temos, desta forma, um tipo de granulação produzida a partir da escrita instrumental que é modulado por efeitos que enfatizam e amplificam as propriedades desta escrita granular, fazendo emergir uma nova sonoridade decorrente da interação entre estes dois meios.

Abaixo dos dois pentagramas do piano, temos uma pauta reservada à eletrônica na qual observa-se uma linha pontilhada que culmina na marcação 2.4, seguida de uma linha contínua. Este número remete à quarta variação dos parâmetros referentes aos efeitos utilizados na segunda parte da peça. A linha pontilhada demonstra o momento de interpolação entre os parâmetros determinados entre as variações 2.3 e 2.4, enquanto que a linha contínua significa que os parâmetros dos tratamentos eletroacústicos são mantidos constantes.

---

<sup>4</sup> Sobre a descorrelação microtemporal e sua aplicação na música eletroacústica, Cf. Kendall (1995), Vaggione (2002) e Sèdes (2015).

Figura 3: Partitura de *Poussières cosmiques*, compassos 26 a 28

O segundo exemplo se refere a um trecho da terceira parte da peça, no qual a escrita instrumental, que neste trecho valoriza a sobreposição de alturas num contexto ondulatório/harmônico/contrapontístico, é combinado ao tratamento eletrônico da modulação em anel. A modulação em anel [BODE, 1967], um tipo de modulação de amplitude, também se dá através da soma e da subtração das frequências de duas ondas, uma portadora e outra modulante. Neste caso, a onda portadora é o som captado do piano, sendo que o valor frequencial da onda modulante (uma onda senoidal) é controlado e definido por uma das variáveis do objeto *process~*.

A utilização da modulação em anel como tratamento eletrônico reforça o contexto ondulatório desta parte pois, além da grande quantidade de parciais sobrepostos decorrentes da sobreposição de várias alturas do piano, este tratamento ainda gera novos parciais oriundos da soma e da subtração das frequências portadora e modulante. Em relação à frequência modulante, foram selecionados apenas valores entre 1 e 20Hz, os quais produzem um efeito de vibrato (modulação de amplitude) no som resultante. Este fato ocorre porque, dentro desta escala de valores, não estamos no âmbito da percepção sonora, mas da percepção do ritmo.

Há também a aplicação de um *delay* cuja duração é curta, não excedendo 240ms. Este *delay* nos proporciona principalmente uma mudança no timbre do som resultante, já que o tempo de repetição determinado é bastante curto. No caso da escrita instrumental, a disposição dos acordes não nos remete a nenhuma função harmônica, ou seja, não há uma hierarquia funcional entre eles, os quais são auditivamente percebidos como “cores” diferentes.

O trecho apresentado na figura a seguir (entre os compassos 67 e 72) se refere ao final da terceira parte, no qual há uma direcionalidade dos acordes rumo à tessitura aguda do piano, além de um aumento gradual de sua intensidade, de *p* a *ff*. Quanto aos tratamentos eletrônicos utilizados, os valores da frequência modulante crescem gradualmente (de 5 a 19Hz) assim como o tempo de duração e a taxa de realimentação (*feedback*) do *delay*, que vão de 120ms e 20% a 240ms e 35%. No compasso 67, o *preset* que armazena as variáveis é o 3.5, o qual se mantém inalterado até o fim do compasso 68. No compasso 69, há a transição dos valores de 3.5 e 3.6, os quais se mantêm constantes até o compasso 72, havendo apenas mudanças harmônicas na escrita do piano.

Figura 4: Compassos 67 a 72 da obra

O terceiro exemplo abordado (Figura 5) se refere à sétima parte da peça, novamente um trecho no qual a escrita do piano se dá numa região bastante grave, a qual é combinada ao tratamento eletrônico da granulação. Há, na escrita instrumental, uma ênfase na utilização de técnicas que geram uma sonoridade descontínua, tais como trêmulos e trilos. Os trêmulos, que se iniciam no compasso 78, vão de uma dinâmica *pp* até *ff* (compasso 80), e novamente retornam a *pp*, no final do compasso 81.

Os trêmulos têm, como intervalos utilizados, uma sétima maior Dó – Si (mão esquerda), uma oitava abaixo do intervalo gerador do processo de síntese instrumental por modulação de frequência (Figura 1), além do intervalo Ré# - Fá# (mão direita), obtido através dos índices 3 e 1, respectivamente, deste processo de modulação. O trinado existente entre os compassos 82 e 84 é executado a partir da nota Lá#, altura também gerada a partir do índice 1 da modulação de frequência.

Quanto aos parâmetros utilizados no tratamento eletrônico da granulação, partimos, na parte 7, de um grão de duração de 200ms, um *delay* de 800ms e uma taxa de realimentação de 70%. No compasso 80, há uma transição para o *preset 7.1*, cujas variáveis têm os respectivos valores de 150ms, 600ms e 80%. O *preset 7.2*, que é atingido no último compasso da obra (o qual contém apenas a ressonância do piano e da eletrônica), apresenta os valores de 100ms, 1500ms e 100%. Há, portanto, uma direcionalidade de granulação no sentido de um tamanho menor do grão – uma sonoridade que se dirige ao ruído – além de um aumento no tempo do *delay* e de sua taxa de realimentação.

Figura 5: Compassos 78 a 85 da obra

## 5. Modelos ondulatório e granular

Propomos relacionar as questões de interação e convergência entre a escrita instrumental e os tratamentos eletroacústicos utilizados na peça (considerando a

morfologia sonora do som resultante) aos paradigmas ondulatório e granular de construção/manipulação do som. Como fundamentação teórica para o paradigma ondulatório, utilizamos o modelo acústico de Helmholtz (1885) – baseado Lei de Ohm – no qual o som é concebido como um fenômeno que pode ser decomposto em uma superposição de parciais. Em relação ao paradigma granular, partimos do modelo de Xenakis (1962, Capítulo II), no qual qualquer som pode ser pensado como uma integração de grãos, partículas elementares sonoras, quanta sonoros.

O objetivo em nosso trabalho composicional foi de relacionar os universos instrumental e eletroacústico, buscando uma convergência da morfologia sonora destes dois universos, de forma similar e complementar. Para uma visualização e discussão deste aspecto, apresentamos, na Figura 6, os espectrogramas das partes 2 (a partir de 2.3), 3 (a partir de 3.1) e 7 da obra. As partes 2 e 7 têm em comum a utilização do tratamento eletrônico da granulação, enquanto que na parte 3 foi utilizada a modulação em anel (relacionada ao modelo ondulatório). Em relação à escrita instrumental, optou-se, nas partes 2 e 7, a valorização de sonoridades descontínuas (através da execução de trilos, trêmolos ou figurações com pequenas decalagens temporais entre as diferentes vozes. Na parte 3, a escrita apresentava predominantemente acordes sem função harmônica definida, valorizando sua sonoridade intrínseca enquanto “cor”.

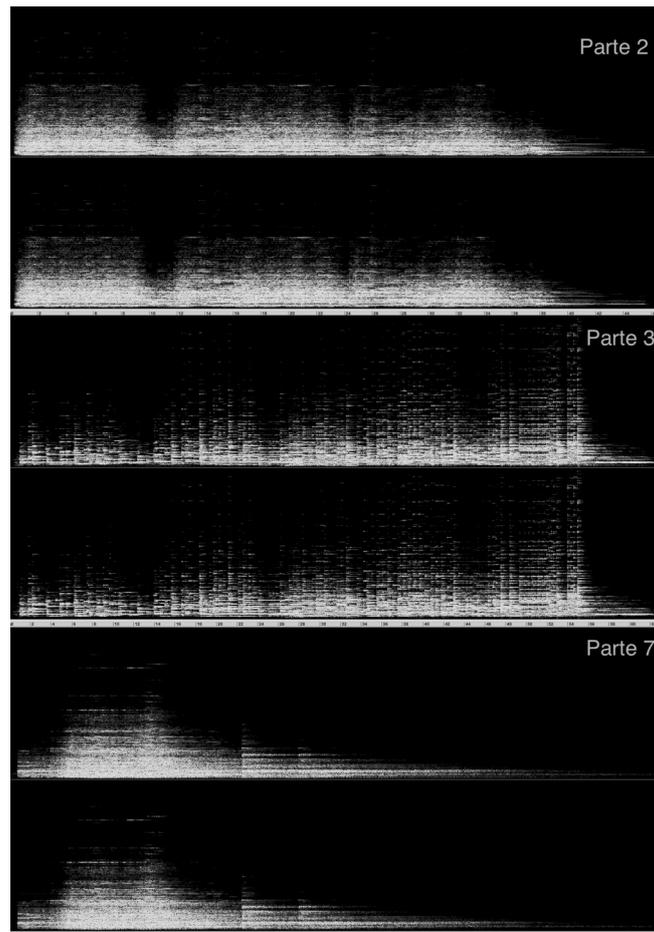


Figura 6: Espectrogramas das partes 2, 3 e 7<sup>5</sup>

<sup>5</sup> A duração dos três espectrogramas apresentados é diferente, respectivamente 46, 63 e 42 segundos.

O que se observa, a partir destes espectrogramas, é uma semelhança morfológica entre as partes 2 e 7, que podem ser interpretadas majoritariamente como uma massa sonora de diferentes densidades, cuja área abrangida é proporcional às alturas executadas pelo pianista. Por outro lado, no espectrograma da parte 3, observamos nitidamente a presença do som estruturado através da sobreposição de parciais, correspondentes ao processo de modulação em anel, realizado a partir das frequências executadas pelo piano (além da presença do próprio som do piano).

Esta dualidade contínuo/descontínuo permeia todo o processo composicional de *Poussières cosmiques*, desde o universo microtemporal da composição interna do som, representado pela dualidade dos paradigmas ondulatório e granular, mas também pelas questões de interação e convergência entre os universos instrumental e eletroacústico. No nível temporal médio (mesoescalar), no caso da definição das alturas principais utilizadas na peça, optou-se pela simulação instrumental do processo de síntese por modulação de frequência, processo claramente calcado no modelo ondulatório.

Esta discussão, de forma não menos importante, também é aplicada ao nível do macrotempo, ou seja, ao nível formal da peça. A escolha pela utilização da ideia de forma momento implica em uma escolha pela não continuidade entre as diferentes partes da peça. Estas partes, desta forma, se projetam como a afirmação da ideia do instante presente, já que este tipo de configuração formal nos leva a pensar o tempo a partir de cortes verticais, os quais retiram estes diferentes instantes da continuidade causal entre passado, presente e futuro. Os diferentes momentos nos transmitem a sensação de estarmos sempre vivenciando as tensões do presente, sem a necessidade de recorrermos à memória para relacionarmos o presente aos instantes que passaram, nem tampouco criarmos expectativas sobre os materiais que estão por vir.

## 6. Referências

- BODE, H. (1967). “The multiplier-type ring modulator”, In: *Electronic Music Review* vol. 1, Reynold Weidenaar e Robert Moog (Ed.), Trumansburg, Independent Electronic Music Center, p. 9 – 15.
- CHOWNING, J. (1973) “The synthesis of complex audio spectra by means of frequency modulation”, In: *Journal of the Audio of Engineering Society* v. 21 n° 7, [https://ccrma.stanford.edu/sites/default/files/user/jc/fm\\_synthesispaper-2.pdf](https://ccrma.stanford.edu/sites/default/files/user/jc/fm_synthesispaper-2.pdf).
- GRISEY, G. (1979) “A propos de la synthèse instrumentale”, *Écrits ou l’invention de la musique spectrale*, G. Grisey, Paris, Éditions MF, 2008, p. 35 – 37.
- \_\_\_\_\_. (1991) “Structuration des timbres dans la musique instrumentale”, *Écrits ou l’invention de la musique spectrale*, G. Grisey, Paris, Éditions MF, 2008, p. 89 – 120.
- HELMHOLTZ, H. (1885) *On the sensations of the tone*, Nova York, Dover Publications, 1954.
- KENDALL, G. (1995) “The decorrelation of audio signals and its impact in spatial imagery”, In: *Computer Music Journal* v. 19 n° 4, Cambridge, The MIT Press, p. 71 – 87.
- ROADS, C., *Microsound*, Cambridge, The MIT Press, 2001.

- ROSSETTI, D. (2015) Poussières cosmiques version 2, pour piano et électronique, Partitura.
- SÈDES, Anne. (2015) “Approche musicale de la décorrélation microtemporelle dans la bibliothèque HOA” In: Actes de Conférence des Journées d’Informatique Musicale, Montréal, [http://jim2015.oicrm.org/actes/JIM15\\_Sedes\\_A.pdf](http://jim2015.oicrm.org/actes/JIM15_Sedes_A.pdf).
- STOCKHAUSEN, K. (1960) “Momentform: Nouvelles corrélations entre durée d’exécution, durée de l’œuvre et moment”, In: Contrechamps n° 9, Paris, Éditions L’Age d’homme, 1989, p. 101 – 120.
- VAGGIONE, H. (2002) “Décorrélation microtemporelle: morphologies et figurations spatiales”, In: Actes de Conférence des Journées d’Informatique Musicale, Marseille, [http://jim.afim-asso.org/jim2002/articles/L27\\_Vaggione.pdf](http://jim.afim-asso.org/jim2002/articles/L27_Vaggione.pdf).
- XENAKIS, I. Musiques Formelles, Paris, La Revue Musicale Richard Masse, 1962.

# Comprovação; Abordagens Desde a Heurística Estética em Eco Composição

Luzilei Aliel<sup>1</sup>, Rogério Luiz Moraes Costa<sup>1</sup>, Damián Keller<sup>2</sup>

<sup>1</sup>NuSom – Núcleo de Pesquisas em Sonologia – Universidade de São Paulo (USP)  
Caixa Postal 05508-900 – São Paulo – SP – Brasil

<sup>2</sup>NAP – Núcleo Amazônico de Pesquisa Musical – Universidade Federal do Acre (UFAC) Caixa Postal 69.920-900 – Rio Branco – Acre – Brasil

{luzaliel, rogercos}@usp.br, dkeller@ccrma.stanford.edu

**Abstract.** *The purpose of this article is to investigate a methodological alternative to ecomposicionais practices through the use of systematic comprovação. With a focus on philosophical perspectives, manifest and artistic products linked to comprovação, we analyze this approach as a mechanism to contribute to the production and elucidation of composition and improvisation concepts from ecomposicional perspective.*

**Resumo.** *O objetivo deste artigo é investigar uma alternativa metodológica para as práticas ecomposicionais através da utilização da sistemática da comprovação. Procurando embasamento nas perspectivas filosóficas, em manifestos e nos produtos artísticos vinculados à comprovação, analisamos esse enfoque como mecanismo capaz de contribuir para a produção e elucidação dos conceitos de composição e improvisação desde a perspectiva ecomposicional.*

## 1. Introdução

O objetivo deste artigo é investigar uma alternativa metodológica para as práticas ecomposicionais através da utilização da sistemática da comprovação. Procurando embasamento nas perspectivas filosóficas, em manifestos e nos produtos artísticos vinculados à comprovação, analisamos esse enfoque como mecanismo capaz de contribuir para a produção e elucidação dos conceitos de composição e improvisação desde a perspectiva ecomposicional.

A primeira seção do artigo fornece uma revisão das tentativas de conceituação das práticas comprovisatórias. Focamos nas limitações das metodologias propostas e fornecemos dois exemplos musicais para ilustrar alguns aspectos dos enfoques discutidos. Na segunda seção apresentamos uma perspectiva que pode suprir parte das carências observadas nos trabalhos analisados. Sugerimos que estética heurística<sup>1</sup>, como

---

<sup>1</sup> Keller et al. (2015a) propõem que a heurística estética abrange o estudo da tomada de decisões no contexto das atividades criativas. “Aesthetic heuristics concern creative-activity decision making. In other words, the term aesthetic heuristics focuses on the decision processes involved in creative activities that target aesthetic choices. [...] As objects of study creative decisions are localized events that take place while creative processes unfold. [...] Recent research on computational creative methods has incorporated unstructured techniques to target activities that rely on resources that become available during the creative act. While computational intelligence relies on knowledge representation and formalized reasoning, idea processors usually set emphasis on broader (unstructured) thinking. [...] This expanded notion of creative heuristics ties the cognitive shortcuts used in decision-making with the intent to produce a creative output.” [Keller et al. 2015a].

vem sendo aplicada no campo da pesquisa em música ubíqua, pode fornecer uma alternativa viável para o trabalho teórico em improvisação.

## 2. Improvisação: definição, conceitos e relevância.

Nesta seção abordamos as tentativas recentes de estabelecer um quadro conceitual para as práticas que ficam a meio caminho entre as propostas de improvisação livre [Aliel et al. 2015; Costa 2014] e as práticas composicionais que visam a mais completa determinação dos eventos sonoros. Em um manifesto elaborado por um grupo de artistas eslovacos, Fujak (2011) propõe uma definição ampla com o propósito de emancipar e legitimar as práticas improvisatórias. São levantados alguns pontos que, para tal grupo são relevantes para uma estética improvisatória:

- O princípio que articula em seus métodos a arte contemporânea transversal em analogia às situações de vida de forma transparente (como resultante do que é rigorosamente planejado e do que está sujeito a mudanças imprevisíveis, para as quais devemos responder operacionalmente e criativamente);
- A escolha consciente das mutações em processos composicionais como os antecedentes, os princípios, as diretrizes e a improvisação situada no contexto contemporâneo de intertextualidade musical com recursos tecnológicos;
- O conceito que expressa o processo de tornar o ato criativo, o gesto artístico e sua percepção numa ação unificada [W. Welsch 1992];
- Avanço no processo de re-territorialização artística lidando com o que/como e por que somos o que somos [G. Deleuze 1980];
- Algo que também pode ser encontrado no espaço entre diferentes meios artísticos em tempos da desapareição do pós-modernismo (que refuta os preconceitos de “pureza formal” e da separação entre forma e conteúdo), em música, artes visuais, cinema, dança, teatro, arte-performance, *happenings*, literatura, fotografia, vídeo-arte, etc.;
- Algo que define a abertura no espírito das metáforas de bricolage, imagem-sônicas e texto-gesto como espaço aberto e tolerante, mas definindo as coordenadas da música e da arte multimídia de forma radicalmente não convencional;
- Uma das oportunidades para a transgressão da esfera AMA (som, música e artes audiovisuais) nas poéticas da *Transmusic Comp*, *Snow-u*, e os Cantos da Floresta que - no início da segunda década deste século e sem pretensões de serem estéticas da improvisação - reatualizam a cena artística eslovaca.

A fim de estreitar as definições metodológicas, nos apoiamos nos relatos dos processos criativos de Hannan (2006) que afirma ter adotado o termo improvisação (*comprovisation*) como denominação para composições que utilizam gravações de práticas improvisadas. Ele sugere que embora existam diversos elementos aleatórios ou intuitivos, a improvisação seria uma nova forma de se pensar a composição. Hannan (2006) define alguns parâmetros para caracterizar a prática da improvisação:

- *A intenção explícita de propósito*: mesmo que a ideia geral seja tentar criar novos materiais para a composição por meio de processos improvisatórios ou através da experimentação de formas livres de montagem do material, Hannan sugere que deve existir uma formulação explícita do propósito, podendo ser modificada se for julgada como ineficaz;
- *A análise aprofundada da literatura*: é necessário estar ciente da literatura escrita e gravada, incluindo as técnicas instrumentais não tradicionais e as técnicas da improvisação livre;
- *A adoção da metodologia de pesquisa*: o projeto deve ter a capacidade de explicar as escolhas e todas as relações estruturais entre os movimentos, seções, e frases. No caso das obras comprovadas, a perspectiva experimentalista pode ser descrita como um tanto arbitrária e não-sistemática. No entanto, há aspectos do enfoque experimental que mostram proximidade com a prática comprovatória.
- *A relevância sistemática e abrangente de dados musicais*: por exemplo, através da montagem de uma biblioteca de eventos de som para viabilizar o trabalho criativo;
- *A apresentação pública dos resultados visando uma avaliação mais aprofundada (reprodutibilidade)*: O processo crítico dos produtos musicais pode estabelecer uma contribuição única para o conhecimento compartilhado. Reprodutibilidade é outra questão.

Desde outra abordagem, podemos citar o trabalho de Bhagwati (2014). O autor despende questionamentos sobre a notação musical como recurso para obras fechadas e abertas em fluxo contínuo, visando resultados estéticos parcialmente estruturados. Porém, Bhagwati não se limita a desenvolver ou estabelecer propostas puramente formais para as obras comprovadas. Como requisito para desenvolver trabalhos que possam abranger as práticas da composição e da improvisação no contexto orquestral, Bhagwati (2014: 171) propõe a “perspectiva da notação”, que serviria para estruturar os elementos reproduzíveis (composição) e contingentes (improvisação) em oposição ao que ele denomina de “perspectiva da audição centralizada”. Esta última abrange a estruturação hierárquica da prática musical através da sincronização polifônica - geralmente envolvendo a organização métrico-tonal - e centralizando as decisões interpretativas na figura do regente. Na perspectiva centralizada, o objetivo não seria o equilíbrio entre os fatores contingentes e determinísticos, mas o enfeito dramático ou narrativo viabilizado pela organização cronológica da estrutura formal.

Bhagwati busca no recurso da notação musical um possível mecanismo de controle entre composição e improvisação capaz de unir as duas atividades. As “perspectivas da notação” propiciam uma ponte entre a “composição fixa” e a “improvisação livre”, tornando evidente o que todo músico experiente sabe: que nenhuma composição se destina a ser livre de contingências - assim como nenhuma improvisação abandona completamente as estruturas reproduzíveis. Levando esse raciocínio ao extremo, segundo Bhagwati toda criação musical pode ser vista como comprovatória. Para o autor, três modelos parecem dominar as práticas comprovatórias:

1. *Variações com dramaturgia plana*: utilizam fórmulas melódicas / harmônicas ou ciclos rítmicos estáveis; 2. formas "livres" de crescimento orgânico e/ou de desintegração gradual; 3. composições com improvisações incorporadas, por exemplo, improvisações com seções compostas em blocos;

É interessante observar que essas três categorias podem ser exemplificadas por estratégias composicionais utilizadas em perspectivas criativas surgidas no século XX. As metáforas orgânicas foram introduzidas por Varèse na sua proposta de abertura formal baseada em processos sonoros, envolvendo agregados sonoros que eram estruturados por mecanismos emergentes (*bottom-up*), como exemplificado na comparação entre o conceito de timbre e o processo de formação de cristais (Wen-Chung e Varèse 1966). A *stasis*, ou organização plana de elementos formais, foi claramente exemplificada nas obras de Feldman (1989). Feldman utiliza a metáfora da tela (*canvas*), comparando o processo composicional ao procedimento utilizado pelo pintor. Ao colocar múltiplas camadas de tinta, as cores vão sendo modificadas lentamente. Musicalmente, o efeito é a suspensão da percepção do tempo, por conta da expansão dos processos de organização sonora a escalas que vão além da capacidade de comparação entre os eventos locais. A forma emerge sem necessidade de impor estruturas narrativas. A terceira estratégia, a organização por blocos, foi amplamente explorada pela escola Polonesa dos anos 1960. Neste caso, podemos descrever o procedimento como de-cima-para-baixo (*top-down*): as relações temporais entre seções tomam precedência sobre as relações temporais entre eventos. Em particular, Lutoslawski (1959-1960) aplica técnicas improvisatórias dentro de obras estritamente estruturadas desde o ponto de vista da proporção temporal entre as seções. A escrita em bloco foi posteriormente adotada por múltiplos compositores dentro do repertório orquestral.

Um exemplo de estruturação a partir de processos orgânicos de crescimento e desintegração é fornecido por [Costa 2014]. Em improvisações com grupos, encontramos interações sociais que envolvem negociação e decisões distribuídas até atingir o resultado sonoro. Os improvisadores utilizam-se de recursos compartilhados para estabelecer a troca musical com outros improvisadores. Desta feita, os resultados criativos tendem a funcionar melhor quando o repositório de conhecimento comum entre os indivíduos improvisadores é maior [Ferraz e Keller 2014]. Além da troca explícita, há uma empatia entre os participantes, fornecendo mecanismos heurísticos comuns que aumentam as chances de conhecer como o improvisador agiria ou agirá. Um exemplo deste tipo de relação em grupo é *Meditação* [Costa 2014]:

*A peça foi desenvolvida para três, quatro ou cinco instrumentos capazes de produzir sons longos em trocas sonoras com modelos computacionais. A ideia geral é de continuidade. A sensação geral deve ser de tempo estático, liso e de transformação gradual lentíssima. Em outras palavras: de longe, tudo é sempre igual, de perto, tudo é sempre diferente. Duração: aproximadamente 15' (até o silêncio acústico, eletrônico e visual). Dinâmica sempre entre ppp e mp (adaptações proporcionais podem ser feitas dependendo do espaço da performance). Apesar da dinâmica restrita, deve ser mantido um alto grau de tensão e energia. A performance deve estar sempre próxima do silêncio, mas as pausas devem ser evitadas, a não ser em pequeníssimas respirações utilizadas para a retomada de energia. Os movimentos devem ser sutis, porém revestidos de intencionalidade. É importante que os movimentos se diferenciem do gestual espontâneo relacionado à técnica instrumental. A não ser que estes sejam redimensionados*

*intencionalmente. O processamento eletrônico de áudio capta e interage com o fluxo sonoro acústico criando uma segunda camada que distorce de maneira sutil o resultado acústico original.*

*Exemplo 1: Meditação (fragmento - arquivo sonoro em anexo).*

Bhagwati (2014) nos proporciona importantes *insights* sobre abordagens alternativas para lidar com contingência e controle:

1. *Abandono da centralização sônica* ("a mistura ideal"), *espacial* ("o palco italiano") e *metodológica* (a notação determinística) em favor de múltiplas perspectivas de escuta não homogêneas;

2. *Elaboração de uma polifonia de práticas improvisacionais* que permitem camadas (*layerings*) horizontais de eventos sonoros, portanto, uma multiplicidade de relacionamentos heterofônicos, não só entre as diferentes "vozes" ou fontes sonoras, mas entre os diferentes parâmetros musicais dentro de cada camada sonora;

3. *Elaboração dos múltiplos níveis da estrutura como um fluxograma de opções* em que as decisões podem resultar em diversas formas de participação dos músicos, abrangendo: a interação sonora ou visual com seus pares e/ou com vários regentes, o "significado" musical ou estilístico a partir da referência a uma forma externa (musical, poética, matemática ou de outra natureza) ou a partir do resultado de um processo autônomo ou interativo;

4. *Perspectivas consolidadas*: manter ou mesmo expandir os benefícios do uso da narrativa dramaturgica e da estrutura formal (composição determinística), bem como a incorporação de elementos interativos dentro do uso de fluxogramas procedimentais - o regente como co-criador.

O último item faz referência às propostas vinculadas ao suporte tecnológico para viabilizar mecanismos de mediação entre músicos e ferramentas. Numa primeira aproximação ao problema, Dudas (2010) introduz os fatores tecnológicos dentro da prática improvisatória e propõe dois modelos que complementam os fatores citados por Bhagwati:

1) Compor para um "instrumento", podendo ser utilizado na improvisação em uma performance; 2) Improvisar com ferramentas tecnológicas, a fim de criar materiais pré-composicionais;

Os aspectos da improvisação em ambos os cenários são demarcados pelas limitações das ferramentas utilizadas, tanto em hardware quanto em software. Segundo Dudas, a definição do "instrumento" também incluiria as estruturas composicionais, mesmo quando o instrumento é utilizado em práticas improvisacionais [Dudas 2010: 29]. Desta forma, teríamos as denominações de "improvisação estruturada" ou "composição improvisada". Como já foi detalhadamente discutido na pesquisa em música ubíqua [Keller et al. 2014], o uso de conceitos extraídos da composição para instrumentos acústicos não contempla a complexidade e flexibilidade dos comportamentos criativos observados nas práticas criativas cotidianas [Pinheiro da Silva et al. 2013]. A redução do suporte para a atividade criativa ao uso de um instrumento, seja este acústico ou digital, deixa de fora aspectos como o contexto social, o uso de fontes sonoras ambientais [Truax 2002], e a participação da audiência no processo criativo [Keller e Capasso 2006]. Cobussen (2003) sugere que improvisadores

poderiam pressupor todo tipo de objetos, estruturas entre outros apetrechos com uma finalidade utilizável (sonoramente) no ambiente como fontes de suas experiências sonoras. Como vem sendo colocado nos múltiplos projetos embasados em cognição ecológica [Burtner 2005; Keller 2000; Keller et al. 2010], a proposta é que a ecologia dos materiais seja utilizada em prol da ação criativa. Por exemplo, quando um improvisador observa/escolhe uma cadeira, essa escolha pressupõe a limitação das ações em função das suas experiências prévias com cadeiras como recursos sonoros. Nesse contexto tem impacto não só as possibilidades sonoras do objeto, também tem um papel às crenças - ou vieses criativos - do participante. Ou seja, ele poderia tratar a cadeira como objeto percussivo, poderia criar sons contínuos com a mesma, e pode até mesmo utilizá-la para ter acesso a outros objetos que ficariam fora do alcance sem o auxílio da cadeira. Seu conhecimento prévio sobre o objeto “cadeira” - em termos ecocognitivos, as *affordances* ou *os canais de interação materiais* - fornecem informações de como agir para obter resultados sonoros.

Para exemplificar a aplicação de recursos tecnológicos em comprovação, podemos citar o *projeto DP* [Aliel e Fornari 2014]. O *projeto DP* é uma comprovação interativa, desenvolvida em PD (*Puredata*), que apresenta a possibilidade de entrelaçar dinamicamente materiais de paisagens sonoras oriundas de locais distintos e distantes. Ao invés de usar sons gravados, DP usa o *streaming* de áudio digital coletado diretamente da geração dinâmica do material sonoro. Desta forma, não há uma dependência de sincronização entre os seus agentes formantes, uma vez que o resultado final é produto da interação com os mecanismos de síntese e dos elementos presentes nas paisagens sonoras. Portanto, a estruturação ocorre através da interação entre o performer e o modelo computacional e é suprida pelos eventos sonoros gerados espontaneamente. Esta realimentação ocorre em seções pré-determinadas para improvisação, sendo que em cada nova fase mantém os áudios captados do *soundscape* e acrescenta novos conteúdos improvisados. Esse mecanismo de acumulação sonora acontece de forma gradual - adotando um enfoque similar ao utilizado nas obras de Varèse - tornando o crescimento e a desintegração sonora em processos semi-orgânicos. O resultado criativo apresenta uma sonoridade única que transcende a limitação do suporte fixo.

*Exemplo 2: Projeto DP (arquivo sonoro em anexo).*

A discussão precedente deixa claramente estabelecida que a comprovação não é um fenômeno novo, mesmo que ainda não haja um trabalho teórico sistemático que possa dar conta das múltiplas nuances dessa prática. Tendo em vista as propostas pioneiras de Hannan (2006), Dudas (2010) e Fujak (2011), a postura de Mailman (2013) de se auto-definir como um dos pioneiros dessa vertente é no mínimo questionável. Mailman considerar “a comprovação como um tipo de criatividade musical que se relaciona a composição e a improvisação de uma forma sem precedentes, algo que era impossível de se alcançar com tecnologias mais antigas” [Mailman 2013: 357]. Mailman defende que a comprovação se torne uma ferramenta significativa somente quando apoiada em processos interativos. A diferença entre a comprovação e a composição interativa (Chadabe 1983) seria o controle da expressividade do sistema (Mailman 2013: 358). A comprovação separaria os dados gerados algoritmicamente em características de nível macro (propriedades emergentes) e trajetórias expressivas que seriam manipuladas pelo performer. O executante pode expressar-se com seus movimentos, sabendo *exatamente* de que forma suas trajetórias podem afetar os

resultados. Segundo Mailman, as técnicas de geração algorítmica devem ficar limitadas a processos quase-estocásticos, amplamente explorados desde os anos 1960 [Xenakis 1971]. Dada essa definição restritiva, o autor postula os seguintes pontos como requisitos para a comprovação:

(1) A comprovação envolve compor música gerada a partir de algoritmos; (2) Envolve intencionalidade e manipulação de parâmetros através de movimentos, visando à expressividade.

### 3. Considerações Sobre as Propostas Comprovisatórias

Enxergamos uma posição aberta no manifesto proposto por Fujak (2011), visando buscar dar validade a uma linha da arte sonora contemporânea que já se expõe em diversas obras que ainda não têm uma denominação comum. Dentro das diversas formas de compreender a comprovação, podemos concluir que esta prática vai além de uma combinação de composição com improvisação, tendendo a se torna um mecanismo com regulação própria. Os procedimentos comprovisacionais têm adquirindo motivação própria, oriunda da fusão de processos transversais da arte contemporânea, abrangendo arte sonora, visual e performance [Fujak 2011]. Entre os exemplos apresentados, em destaque: a manipulação de objetos gravados a partir de conteúdos improvisados [Hannan 2006], a utilização de técnicas variadas de notação musical com intuito de obter materiais fechados, regulados ou organizados hierarquicamente (composição determinística) e materiais abertos, estocásticos ou heterofônicos (improvisação) [Bhagwati 2014]. A prática comprovisatória também inclui procedimentos auxiliados pelo suporte computacional, incluindo estéticas vinculadas à arte interativa que adotam enfoques embasados na tradição instrumental [Dudas 2010] e nos ideais românticos de expressividade e virtuosismo cultivados durante o século XIX [Mailman 2013] ou a adoção da intencionalidade como base do processo de organização sonora [Hannan 2006; Mailman 2013].

Desde uma perspectiva ecomposicional, a aplicação do paradigma acústico-instrumental<sup>2</sup> parece mais um retrocesso do que uma contribuição para o avanço das práticas criativas. A pesquisa em música ubíqua (*Ubimus*) vem apontando alternativas para o suporte à criatividade, tanto no âmbito profissional [Aliel et al. 2015; Keller et al. 2014] quanto nos contextos cotidianos [Pinheiro da Silva et al. 2013]. Mais especificamente, observa-se uma convergência entre as práticas improvisatórias [Costa 2014] e os enfoques exploratórios geralmente descritos como atividades epistêmicas [Keller et al. 2010; Keller et al. 2014]. Intimamente relacionado com os métodos determinísticos e indeterminados, é o processo de tomada de decisões nos ciclos criativos. Na próxima seção discutimos os últimos avanços dentro do contexto da pesquisa em *Ubimus*<sup>3</sup>.

---

<sup>2</sup> O conceito de paradigma acústico-instrumental, ou viés instrumentalista, vem sendo adotado dentro da pesquisa em música ubíqua para sintetizar as perspectivas que defendem que o instrumento acústico e o criador isolado são os agentes mais importantes na atividade criativa musical [Keller et al. 2014].

<sup>3</sup> *Ubimus* – referente à música ubíqua, ou seja, sistemas de agentes humanos e recursos materiais que possibilitam atividades musicais através do desenvolvimento de ferramentas sustentáveis de suporte à criatividade.

#### 4. Sobre o processo de tomada de decisões

Tratando do processo de tomada de decisões em *Ubimus*, Keller e coautores (2015a) propõem a heurística estética (*aesthetic heuristics*) dentro do âmbito da investigação musical ubíqua. Keller et al. (2015a) fornecem uma definição provisória de heurística estética, indicando o objeto de estudo e as motivações da proposta. A heurística estética foca o estudo da tomada de decisões durante os processos criativos. As decisões podem envolver tanto mecanismos cognitivos explícitos quanto processos que não podem ser verbalizados. Esta pesquisa é colocada no âmbito dos esforços atuais de suporte à criatividade, destacando os estudos recentes que visam à criatividade musical cotidiana. Os resultados experimentais apontam para uma lacuna entre as propostas teóricas em criatividade musical e os fatores que impactam as atividades musicais cotidianas. A *auto-referencialidade do construto teórico-experimental* e a *restrição prematura de domínio* são duas das limitações apontadas em trabalhos teóricos anteriores [Keller et al. 2011; Lima et al. 2012; Keller et al. 2014]. Keller et al. (2015) fornecem uma definição provisória de heurísticas estéticas, indicando o objeto de estudo e as motivações da proposta.

Em *Relational Properties in Interaction Aesthetics: The Turn Music Ubiquitous* Keller et al. (2014) adotam como ponto de partida os avanços teóricos recentes em design de interação e em interação humano-computador [Löwgren 2009] para analisar o processo de tomada de decisões nas atividades criativas. Sendo que esses processos apoiam-se em recursos materiais, sociais e cognitivos, conceitos e métodos específicos propostos para descrever aspectos da concepção e da materialização das experiências tecnológicas dentro da pesquisa em *ubimus*. A *maleabilidade* [*pliability* – Löwgren 2007] e a *ancoragem* [*anchoring* - Keller et al 2010] são duas qualidades de design que surgiram do impacto das interações entre as práticas criativas ecológicas e o enfoque estético em design de interação. Um exemplo da aplicação de conceitos cognitivo-ecológicos no suporte criativo para a música ubíqua é a metáfora de marcação temporal [Keller et al. 2010; Radanovitsck et al. 2011]. Múltiplos estudos de aplicação da marcação temporal serviram para exemplificar como as propriedades relacionais podem ser integradas dentro de sistemas para o suporte criativo [Ferreira da Silva et al. 2015; Farias et al. 2014; Keller et al. 2013; Pinheiro da Silva et al. 2013].

Duas abordagens para o design de interação - a estética da interação e a música ubíqua - são analisadas no capítulo *Interaction Aesthetics and Ubimus* [Keller et al 2015]. O engajamento, os padrões de comportamento, as formas alternativas de design com combinações inovadoras de materiais e as identidades dos participantes inseridos em contextos culturais específicos são identificados como temas comuns. Dispor-se à criatividade implica em lidar com fenômenos que não foram considerados dentro da perspectiva estética do design de interação. Os potenciais criativos, a criatividade cotidiana e a criatividade distribuída estão ganhando destaque na pesquisa recente em música ubíqua. Para estabelecer métodos comuns entre a estética da interação e a pesquisa em *ubimus*, Keller et al. (2015) discutem as implicações da adoção de propriedades relacionais materiais e de propriedades relacionais sociais como alvos para o trabalho experimental. As propriedades materiais são o foco de duas metáforas de interação: a marcação temporal - mencionada previamente - e a marcação espacial. A

marcação espacial consiste na utilização de pistas visuais para determinar as ações dos participantes em atividades criativas musicais [Keller et al. 2011].

Entendidas como um todo, essas três publicações fornecem uma base teórica inicial para abordar o estudo da tomada de decisões nas atividades criativas. Três metodologias foram destacadas pela pesquisa: a avaliação dos subprodutos criativos, a observação dos comportamentos criativos e da análise dos perfis dos participantes. Para avançar na frente metodológica, esses trabalhos propõem o design de novas metáforas de apoio à criatividade. Os exemplos discutidos em [Keller 2015] servem para ilustrar a aplicação musical das heurísticas criativas sugeridas por Amabile (1996). Amabile (1996) compilou uma lista das heurísticas mais citadas para a tomada de decisões no contexto de atividades criativas. Ao invés de focar em procedimentos predeterminados, cinco dos sete itens discutidos por Amabile descrevem estratégias para lidar com a atividade criativa, sem metas predefinidas:

1. Jogue com ideias [Wickelgren 1979].
2. Tente gerar hipóteses, analisando estudos de caso, utilizando analogias, respondendo por exceções e investigações incidentais paradoxais [McGuire 1997].
3. Faça o familiar estranho e faça o estranho familiarizar [Gordon 1961].
4. Tente algo contra-intuitivo [Newell et al. 1958].
5. (a) Use sessões de trabalho concentrados, em vez sessões de trabalho distribuídos; (B) Comece jogando com a mesma ideia de um número de formas diferentes, antes de entrar nas sessões de ideias geradas; (C) não conte com a sobre-aprendizagem de mecanismos de resposta [Mednick 1962].
6. (a) Tente reorganizar os elementos de um problema; (B) faça uma pausa; (C) comece a considerar uma solução que envolva os elementos mais importantes antes de elaborar a solução em detalhe; (D) tente considerar classes de interesses em nome de elementos específicos; (E) Quando a via para alcançar uma meta de sua posição parecer muito longe, o sistema auxilia a procurar formas mais curtas [Anderson 1980].
7. (a) Aplique *impossíveis intermediários*; (b) determine o potencial de uma ideia para desencadear novas ideias ao invés de validá-lo [Bono 1971].

Vemos que desde uma perspectiva mais ampla, não necessariamente focada nos aspectos específicos do fazer musical, mas no suporte para as decisões criativas, ganha importância o processo de preparação da atividade - o que no âmbito da pesquisa em música ubíqua é denominado *design*. As heurísticas 1, 2, 5, 6 e 7 não propõem como método a definição de um objetivo previamente ao início da atividade criativa (que se constitui em requisito das propostas formuladas por Mailman - intencionalidade - e Hannan - adoção da metodologia vinculada à solução de problemas). Muito pelo contrário, o suporte para as decisões criativas envolveria o engajamento ativo com os materiais e implicaria necessariamente um processo de auto-reflexão.

## 5. Considerações finais e perspectivas futuras na comprovação

Concluimos que o estado atual da prática improvisatória precisa de um afinamento conceitual drástico. Conceitos anacrônicos como a expressividade ou a formulação de decisões estéticas prévias a exploração do espaço epistêmico constituem uma visão hierárquica e direcionada do fazer musical. Essa visão encaixa bem com as práticas vinculadas ao que Bhagwati chama de “perspectiva centralizada”. A procura da intencionalidade, da expressão individual e da estruturação desvinculada da experimentação com os materiais é consistente com o paradigma acústico-instrumental, mas dificilmente é aplicável à visão ampla da comprovação defendida por Fujak (2011).

Apesar das repetidas expressões de “pioneirismo” encontradas na literatura sobre improvisação, observamos que esta prática remonta aos inícios do século XX. Dentre as práticas documentadas, a única vertente que pode ser considerada significativa pelo seu caráter plural é a eslovaca. Os outros exemplos mostram contribuições isoladas. O movimento improvisacional eslovaco pode ser comparado ao trabalho comunitário realizado no campo da música ubíqua no Brasil. Ele mantém um enfoque amplo, aberto para as contribuições de grupos de outras nacionalidades.

## 6. Referências

- Aliel, L., and Fornari, J., (2014). “Projeto Destino Pirlampo: Um Estudo sobre a Composição de Meta- Soundscapes em Música Ubíqua”. *Música Hodie*, v.14, p. 105 - 121.
- Aliel, L., Keller, D., Costa, R., Melo, M. T. S., Pinheiro da Silva, F., Andrade, L. S., (2015) "Práticas Ecocognitivas Improvisatórias: Dois Estudos de Caso", In: VI Ubimus, <http://lammax.lnu.se/ubimus/>
- Amabile T. M. (1996) “The Motivation for Creativity in Organizations”. In: *Harvard Business Review*, pp. 1-14
- Anderson, J. R. (1980) "Spatial and Hippocampal Synaptic Plasticity in Senescent and Middle-Aged Rats", In: O. Stein (Ed.), *Psychobiology og Aging: Problems and Perspectives* (pp. 253 - 272)
- Barbosa, R. B.; Lima, M. H.; Queiroz, M.; Pimenta, M. S.; Keller, D. (2010) “Validação de Procedimentos Composicionais: Estudo Exploratório”. In: 6 Encontro de Música e Mídia, 2010, São Paulo, SP. *Anais do 6 Encontro de Música e Mídia*. São Paulo: Musimid.
- Bhagwati, S. (2014) “Towards Interactive Onscreen Notations for Improvisation in Large Ensembles”. In: *Sound & Score: Essays on Sound, Score and Notation*.
- Bono, E. *Pensamento Criativo*. Rio de Janeiro: Vozes, 1971.
- Burtner, M. (2005) “Ecoacoustic and shamanic technologies for multimedia composition and performance”. *Organised Sound* 10 (1), 3-19. (Doi: 10.1017/S1355771805000622.)
- Costa, R. L. M. (2014) “Livre improvisação e ecologia sonora: uma aproximação a partir da estética da sonoridade”. *Opus*, Porto Alegre, v. 20, n. 1, p. 189-206.

- Cobussen, M. (2003) "The Omnipresence of Improvisation", [https://social.stoa.usp.br/articles/0035/9284/USP\\_Impro\\_Texts\\_Marcel\\_Cobussen.pdf](https://social.stoa.usp.br/articles/0035/9284/USP_Impro_Texts_Marcel_Cobussen.pdf)
- Chadabe, J. "Interactive Composing: An Overview", 1983.
- Deleuze, G., and Guattari, F. *Mille Plateaux. Capitalisme et Schizophrénie*, Paris, Les Editions de Minuit, 1980.
- Dudas, R. (2010) "Comprovisation: The Various Facets of Composed Improvisation within Interactive Performance Systems," *Leonardo Music Journal* 20 p. 29-31.
- Farias, F. M.; Keller, D.; Pinheiro da Silva, F.; Pimenta, M. S.; Lazzarini, V.; Lima, M. H.; Costalonga, L. L.; Johann, M. (2014) "Suporte para a Criatividade Musical Cotidiana: mixDroid Segunda Geração". In: *V Workshop on Ubiquitous Music (V UbiMus)*, 2014, Vitória, ES. *Proceedings of the V Workshop on Ubiquitous Music (V UbiMus)*. v. 1. p. 1-12.
- Feldman, M. (2000). "Give my regards to eighth street: Collected writing of Morton Feldman". Cambridge, Ma: Exact Change. (ISBN: 9781878972316)
- Ferraz, S. & Keller, D. (2014) "MDF: Proposta Preliminar do Modelo Dentro-Fora de Criação Coletiva". *Cadernos de Informática (UFRGS)*, v. 8, p. 57-67.
- Fujak, J. (2011) "Comprovisación – Notas para la discusión sobre la validez del concepto", *Oro Molido* No. 33, Madrid, Spain.
- Gordon, W. W. (1961) "Synectics: The Development of Creative Capacity", New York: Harper and Row.
- Hannan, M (2006) "Interrogating Comprovisation as Practice-led Research", in *Speculation and Innovation: applying practice led research in the creative industries*, Brisbane: Queensland University of Technology, <http://www.speculation2005.qut.edu.au>, now accessible at [http://epubs.scu.edu.au/sass\\_pubs/448/](http://epubs.scu.edu.au/sass_pubs/448/).
- Keller, D.; Capasso, A. (2006) "New concepts and techniques in eco-composition". *Organised Sound* 11 (1), 55-62. (Doi: 10.1017/S1355771806000082.)
- Keller, D., Lazzarini, V. & Pimenta, M. S., (ed.). (2014) *Ubiquitous Music*, Vol. XXVIII. Berlin and Heidelberg: Springer International Publishing. (ISBN: 978-3-319-11152-0.)
- Keller, D.; Barreiro, D. L.; Queiroz, M.; Pimenta, M. S. (2010) "Anchoring in Ubiquitous Musical Activities". In: *International Computer Music Conference, 2010*, New York, NY. *Proceedings of the International Computer Music Conference*. New York, NY: ICMA.
- Keller, D., Otero, N. & Costalonga, L. L. (2015a). *Aesthetic heuristics in ubimus*. In *Proceedings of the Electronic Visualisation and the Arts Conference (EVA 2015)*. London: BCS, Computer Arts Society Specialist Group.
- Keller, D.; Otero, N.; Lazzarini, V.; Pimenta, M. S.; de Lima, M. H.; Johann, M.; Costalonga, L. (2015) "Interaction Aesthetics and Ubiquitous Music". In: Nelson Zagalo, Pedro Branco. (Org.). *Springer Series on Cultural Computing*. 1ed. London, UK: Springer London, v. 1, p. 91-105.

- Löwgren, J. (2007) "Pliability as an experiential quality: Exploring the aesthetics of interaction design". *Artifact* 1, 2, 95, 85.
- Löwgren, J. (2009) "Toward an articulation of interaction esthetics", *The New Review of Hypermedia and Multimedia*, v.15 n.2, p.129-146.
- Lutoslawski, W. (1959/1960) "Musique funebre." In: *Nutida Musik* 24, no.1 (1980/1981).
- Mailman, J. B. (2013) "Improvising Synesthesia – Comprovisation of Generative Graphics and Music", *Leonardo Electronic Almanac (LEA)*, Volume 9, Issue 3, Leonardo/ISAST, San Francisco, California, USA,
- McGuire, W. J. (1997) "Creative Hypothesis Generating in Psychology: Some Useful Heuristics", *Annual Reviews of Psychology*, 48, 1-30. McHardy, P., and Allan, T. (2000). *Closing the Gap Between What Industry Needs and What He Provides*.
- Mednick, S. A. (1962) "The associative basis of the creative process". In: *Psychological Review*, 69, 220-32. Meichenbaum, D. H. and Bowers, K. S. (1984)
- Newell, A., Shaw, J. C. and Simon, H. A. (1958) "Elements of a Theory of Human Problem Solving", In: *Psychol.*, 12, 214-220.
- Pinheiro da Silva, F.; Keller, D.; Silva, E. F.; Pimenta, M. S.; Lazzarini, V. (2013) "Criatividade Musical Cotidiana: Estudo Exploratório de Atividades Musicais Ubíquas". *Música Hodie*, v. 13, p. 64-79.
- Radanovitsck, E. A. A., Keller, D., Flores, L. V., Pimenta, M. S. & Queiroz, M. (2011). *mixDroid: Marcação temporal para atividades criativas*. In *Proceedings of the XIII Brazilian Symposium on Computer Music (SBCM)*. Vitória, ES: SBC.
- Truax, B. (2002) "Genres and techniques of soundscape composition as developed at Simon Fraser University". In: *Organ Sound.*, p. 7:5–14
- Welsch, W. (1992) "Transkulturalitat. Lebensformen nach der Auflosung der Kulturen". In *Information Philosophie*.
- Wen-Chung, C. and Varese, E. (1966) "Perspectives of New Music". Vol. 5, N°1. p. 11-19.
- Wickelgren. W. A (1979) "Chunking and consolidation: A Theorical Synthesis", In: *Neuropsychologia*, 6, 244 - 235.
- Xenakis, I (1971) "Theory and Applications of Cellular Automata", Singapore: World Scientific. *Formalized Music*, Bloomington, In: Indiana University Press.

# ***Metaflauta: design e performance de instrumento aumentado via suporte computacional***

**Gabriel Rimoldi<sup>1</sup>, Jônatas Manzolli<sup>1</sup>**

<sup>1</sup>Núcleo Interdisciplinar de Comunicação Sonora –  
Universidade Estadual de Campinas (UNICAMP)  
Rua da Reitoria, 165 – 13.083-872 – Campinas – SP – Brasil

{gabriel.rimoldi,jonatas}@nics.unicamp.br

## ***Abstract.***

*This article describes the design, development and performance of Metaflauta, an augmented instrument based on a hybrid system of gestural capture. We explore both direct and indirect gestural acquisition of flutist as potential controls in multimodal environments. First we present the state of art on the use of sensors and gestural acquisition of flute and then discuss the criteria applied in the design and implementation of the computational system. Finally, we demonstrate the use of the system by means of sound synthesis and sound and visual processing modules developed.*

## ***Resumo.***

*Descrevemos neste artigo a concepção, desenvolvimento e performance através da Metaflauta, um instrumento aumentado baseado num sistema híbrido de captura gestual. Neste sistema, exploramos a captura direta e indireta dos gestos do flautista como potenciais controles gestuais aplicáveis a ambientes multimodais. Apresentamos primeiramente o estado da arte sobre o uso de sensores e aquisição gestual da flauta transversal e, em seguida, discutimos os critérios aplicados no design e na implementação do sistema computacional. Por fim, demonstramos a aplicação desse sistema no controle de síntese sonora digital, processamento sonoro e visual em tempo real.*

## **1. Introdução**

O desenvolvimento de novas Interfaces Musicais Digitais (Musical Digital Interfaces - MDI) tem propulsionado uma significativa multiplicação de meios de expressão musical nas últimas décadas. Nessa enorme e heterogênea quantidade de propostas de MDIs, os chamados “instrumentos aumentados” compreendem aqueles que utilizam aparatos externos (sensores) direta ou indiretamente acoplados a instrumentos acústicos, permitindo, assim, a desejada expansão de suas possibilidades sonoro-expressivas [1]. Há extensa literatura sobre a utilização da flauta transversal como instrumento aumentado [2] [3] [4] [5] [6]. Os propósitos do uso de sensores acoplados a esse instrumento variam desde a aplicação em sistemas musicais interativos [4], processamento e síntese sonora [3] até como ferramenta de pesquisa básica [5] [6]. Enquanto a maioria dos trabalhos têm explorado a aquisição gestual direta [2] [3] [4] [5] [6], isto é, através de um conjunto de sensores acoplados ao instrumento, outros

trabalhos tem utilizado o método de aquisição indireta [7] [8], através da análise e processamento do sinal de áudio capturado do instrumento [1] [9]. O trabalho que apresentamos neste artigo faz uso desses dois tipos de aquisição. Desta forma, denominamos a Metaflauta, como um “instrumento aumentado híbrido”. O sistema implementado associa sinais capturados por sensores, acoplados à flauta, ao processamento e extração de características do som utilizando descritores de áudio [10].

Apresentamos inicialmente um panorama sobre as diferentes aplicações da flauta transversal no contexto dos instrumentos aumentados e, em seguida, discutimos a concepção e implementação da Metaflauta. Por fim, descrevemos os resultados obtidos com a aplicação da Metaflauta no controle de sinais multimodais associados à improvisação do intérprete.

## 2. Flauta transversal como instrumento aumentado

Um dos trabalhos pioneiros no uso de sensores acrescidos a instrumentos acústicos foi a MIDI-flute desenvolvida no IRCAM em meados da década de 80 [2]. Seu protótipo inicial era composto de sensores óticos acoplados às chaves da flauta e um microfone que detectava a mudança de oitava a partir da análise do áudio. Sua versão final, composta de sensores de efeito Hall e imã acoplados às chaves, foi implementada por Michel Ducuor e Michel Starkier em parceria com o flautista Lawrence Beauregard. O objetivo de desenvolvimento da MIDI-flute foi a aplicação da mesma como guia para um sistema de score following [2]. Esse instrumento foi amplamente empregado em obras do repertório eletroacústico misto, tais como ...explosante fixe... (1991-93), de P. Boulez, Jupiter ou La Partition du ciel et de l'enfer (1989) de P. Manoury. Um dos aspectos mais interessantes dessa flauta é que ela não possuía cabos conectando-a à um processador externo, pois havia uma placa de circuito eletrônico acoplada no próprio instrumento.

A Digital Flute apresentada por Ystad e Voinier [3] é bastante similar à interface desenvolvida pelo IRCAM. As chaves dessa flauta foram acrescidas de 14 sensores de efeito hall e imã e um microfone foi posicionado junto à rolha no interior do tubo do instrumento. Diferentemente da MIDI-flute, os autores desenvolveram uma interface capaz de controlar síntese sonora digital até mesmo em situações em que o instrumentista não emitisse jato de ar através do tubo. O método de síntese utilizava tanto o sinal acústico como parâmetros de modelagem física. A extração de sinais produzidos pelo comportamento da coluna de ar no interior do tubo, permitia a manipulação de características do timbre do instrumento, simulando, por exemplo, o ressonador de um outro instrumento ou mesmo aumentar ou diminuir o tamanho do tubo virtualmente. Os autores utilizavam também quatro pedais para controlar parâmetros macroestruturais ou configurações dos sensores acoplados à flauta.

A Hyper-Flute, desenvolvida por Palacio-Quintin [4], baseou-se no sistema SensoLab, criado por J. Scherpenisse e A. J. Van den Broeck no Instituto de Sonologia de Haia. Essa interface foi utilizada em sistemas interativos musicais e dispunha de uma grande variedade de controles adicionados ao instrumento acústico. Essa diversidade implicava numa considerável complexidade de controle associado à performance instrumental, sendo necessário um aprendizado e treino para desenvolver os potenciais expressivos do instrumento. A autora apresenta ainda o desenvolvimento de uma Hyper-

Bass-Flute, que possui funcionamento similar à anterior, com a vantagem de possuir um maior espaço para o acoplamento de sensores.

Da Silva, Wanderley e Scavone [5] utilizaram sensores para mensurar aspectos relacionados à embocadura, mais especificamente à velocidade e direção do jato de ar. Os autores utilizaram dois sensores de pressão prolongados com duas pequenas sondas, cada qual posicionada numa extremidade lateral do porta-lábios da flauta. Esse sistema compreendia ainda um microfone, um sensor resistivo à pressão (FSR) e um atuador atado junto ao pescoço do instrumentista, o qual fornecia um feedback sobre a vazão do ar detectada pelos sensores de pressão.

Almeida e colaboradores [6], interessados em relacionar as transições de dedilhado da flauta transversal com aspectos acústicos decorrentes da dificuldade técnica de determinada passagem, analisaram a presença de transientes e a diminuição da amplitude entre duas notas ligadas. Para isto, utilizaram 16 sensores fotorresistores colados na borda inferior de cada chave. O sinal acústico do instrumento foi também gravado utilizando-se um microfone. Na Tabela 1, resumimos os principais tipos de sensores aplicados ao desenvolvimento de flautas transversais aumentadas encontradas na literatura.

**Tabela 1: Utilização de diferentes sensores nos trabalhos que exploram a aquisição gestual direta da flauta transversal**

Tipo de Sensores	Acelerômetro	FSR	LSR	Sensor de efeito hall	Ultrassônico	Chaves de mercúrio	Chave de contato	Microfone	Pedal	Sensor de Pressão	Atuador
MIDI-Flute [2]				●							
Ystad & Voinier [11]				●				●	●		
Palacio-Quintin [4]	●	●	●	●	●	●	●	●			
Silva et al. [5]		●								●	●
Almeida et al. [6]			●								

Outras pesquisas têm explorado ainda a extração de informações gestuais de maneira indireta, recuperando informação musical através da análise e processamento do sinal de áudio. Kereliuk e colaboradores [7] pesquisaram técnicas de aquisição indireta em tempo real para aferir posições de dedilhado da flauta na emissão de harmônicos. O objetivo da pesquisa foi detectar os diferentes padrões de dedilhado e pressão que resultam numa mesma frequência fundamental. Os autores extraíram informações temporais e espectrais dos sinais a partir da aplicação de análise de componentes principais (PCA) da energia dos sub-harmônicos de cada sinal. Verfaillie e colaboradores [8] realizaram experimento similar e utilizaram descritores de áudio para avaliar o efeito do aumento do fluxo de ar (overblow) na produção de notas harmônicas.

Encontramos ainda trabalhos que exploram a aquisição gestual direta através de sensores não acoplados ao corpo do instrumento ou instrumentista. É o caso, por exemplo, do Multimodal Music Stand (MMMS), um sistema de captura de gestos que utiliza sensores eletromagnéticos acoplados à estante, além de um conjunto de câmeras

e microfone que complementam a informação detectada pelos sensores [12]. A primeira experiência realizada com o sistema foi a obra interativa *timeandagain*, composição para flauta e computador de JoAn Kuchera-Morin. Para a referida obra, dois tipos de movimento distintos foram mapeados: a movimentação para cima e para baixo do corpo do instrumento e a rotação da cabeça no eixo transversal. O movimento do corpo da flauta no eixo longitudinal foi detectado pela câmera frontal e para o movimento da cabeça utilizaram a imagem capturada por uma câmera posicionada lateralmente. Visto que as informações extraídas com as câmeras eram pouco acuradas, devido ao ruído gerado pela variação de luz e outros fatores ambientais, os autores agregam informação originados através da interface de áudio e de sensores de campo elétrico acoplados ao MMMS.

### 3. Design e implementação da *Metaflauta*

A *Metaflauta* foi concebida como instrumento aumentado baseado num sistema híbrido de captura gestual, o qual associa informações diretas e indiretas produzidas a partir dos movimentos do flautista. A construção dessa interface teve como objetivo primordial a expansão das possibilidades de controle de sistemas interativos multimodais sem, entretanto, comprometer as possibilidades performáticas já propiciadas pelo instrumento acústico. A captura direta é realizada por meio de sensores acoplados ao instrumento enquanto que para a indireta utiliza-se processamento do sinal de áudio capturado via microfone.

Os sensores utilizados na *Metaflauta* focam a exploração de movimentos que não estão diretamente associados à produção sonora, o quais não podem ser recuperados por meio do processamento do áudio. Nosso principal propósito foi explorar possibilidades de controle através dos chamados *gestos auxiliares* [13] [9].

Em contrapartida, os gestos ou sonoridades advindas da produção sonora do instrumento são analisados em tempo real por um conjunto de descritores de áudio, os quais extraem as seguintes características: a) Temporais (energia média, zero crossing [14]), b) Espectrais (Centroide [15] e Fluxo [16]), c) Harmônicas (Chroma [17]) e d) Perceptivas (Loudness [18] e Escala Bark [19]). Tanto as informações gestuais capturadas diretamente via sensores como as capturadas indiretamente via análise e processamento do sinal de áudio são enviadas ao módulo de mapeamento, o qual combina e transforma as mesmas em múltiplos parâmetros de controle audiovisual.

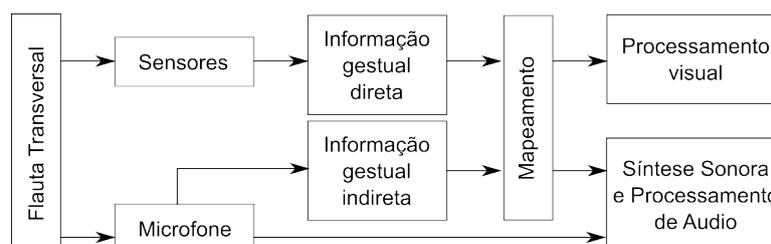


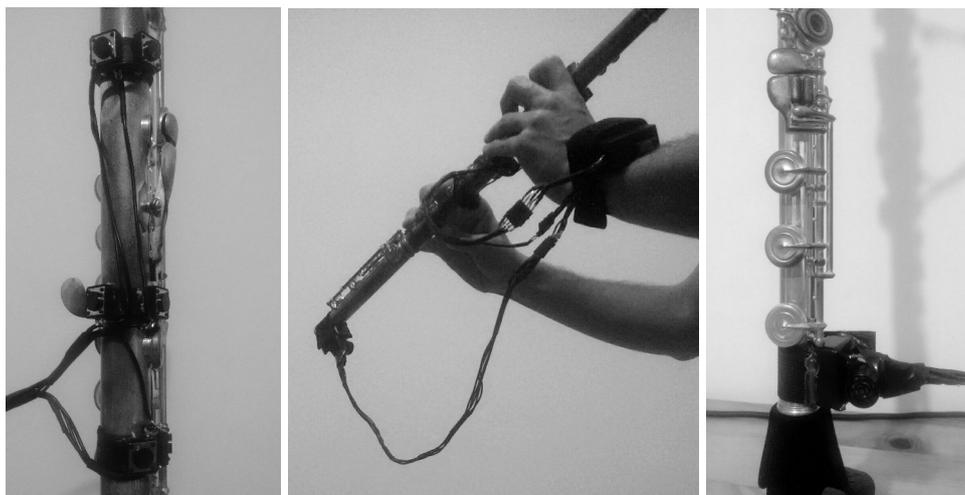
Figura 1: Diagrama de design da *Metaflauta*

### 3.1 Captura gestual direta

As informações capturadas pelos sensores são enviadas ao computador a partir de um protótipo Arduino<sup>1</sup>, com processador Atmel ATmega2560. Na Tabela 2, descrevemos os tipos de sensores utilizados no protótipo desenvolvido. O sinal de cada sensor pode tanto ser utilizado de modo independente como também combinado ao sinal de outros sensores, afim de se extrair informações mais acuradas e de mais alto nível em relação aos movimentos do performer. É o caso, por exemplo, do giroscópio, acelerômetro e magnetômetro, que formam uma composição MARG (*Magnetic, Angular Rate and Gravity*), por meio da qual obtemos os valores de posicionamento angular do instrumento em três eixos, a dizer *pitch*, *roll* e *yaw*. Informações de ordem superior também podem ser utilizadas pelo sistema, como por exemplo, os momentos estatísticos de aceleração durante um certo período de tempo.

**Tabela 2: Categorias e medida extraída de cada sensor na captura gestual direta da Metaflauta**

Sensores	Função/Descrição
1 Acelerômetro triaxial	Mede a aceleração em cada eixo do instrumento.
1 Giroscópio triaxial	Mede a variação angular de cada eixo do instrumento.
1 Magnetômetro triaxial	Mede a intensidade e direção do campo magnético exercido sobre cada eixo do instrumento.
1 sensor ultrassônico	Detecta a distância da mão direita do instrumentista em relação ao corpo do instrumento.
1 FSR ( <i>force sensitive resistor</i> )	Detecta a pressão do polegar direito.
6 chaves de contato	Controles discretos posicionados ao longo do corpo do instrumento sendo 3 controladas pelo polegar direito, 2 pelo polegar esquerdo e 1 pelo dedo mínimo esquerdo.



**Figura 2: Detalhes dos sensores acoplados à Metaflauta**

<sup>1</sup> <https://www.arduino.cc/>

### 3.2 Captura gestual indireta

A recuperação de informações gestuais relacionadas à produção sonora do instrumento foi realizada a partir da análise e processamento do sinal de áudio. As características extraídas do sinal de áudio foram agrupadas em quatro categorias, a dizer: temporais, espectrais, harmônicas e perceptivas. Como demonstrado pelo diagrama da Figura 3, o vetor de características temporais é obtido através do janelamento do sinal capturado pelo microfone. Cada janela temporal é processada via Transformada Rápida de Fourier (*Fast Fourier Transformation - FFT*), através da qual são extraídas as características espectrais do sinal. As características harmônicas e perceptivas são extraídas através do sinal obtido pela FFT por meio de modelos sinusoidais e relacionados à percepção auditiva humana, respectivamente.

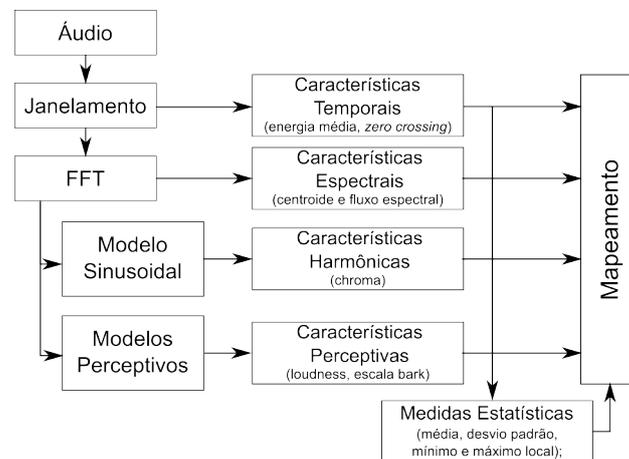


Figura 3: Diagrama de aquisição gestual indireta através da análise e processamento do sinal de áudio capturado.

Os sinais obtidos pela análise do áudio são então enviados ao módulo de mapeamento, responsável pela associação dos mesmos aos parâmetros de controle de processamento audiovisual. Medidas estatísticas que descrevem o comportamento de cada vetor de características ao longo do tempo são enviadas ao módulo de mapeamento, configurando-se as mesmas também como variáveis de controle do sistema. Para a análise e extração das características supracitadas, utilizamos a biblioteca *PDescriptors* implementada em Pure Data<sup>2</sup> por Monteiro e Manzolli [10].

## 4. Módulos de síntese sonora e processamento de áudio

### 4.1 Phase Vocoder

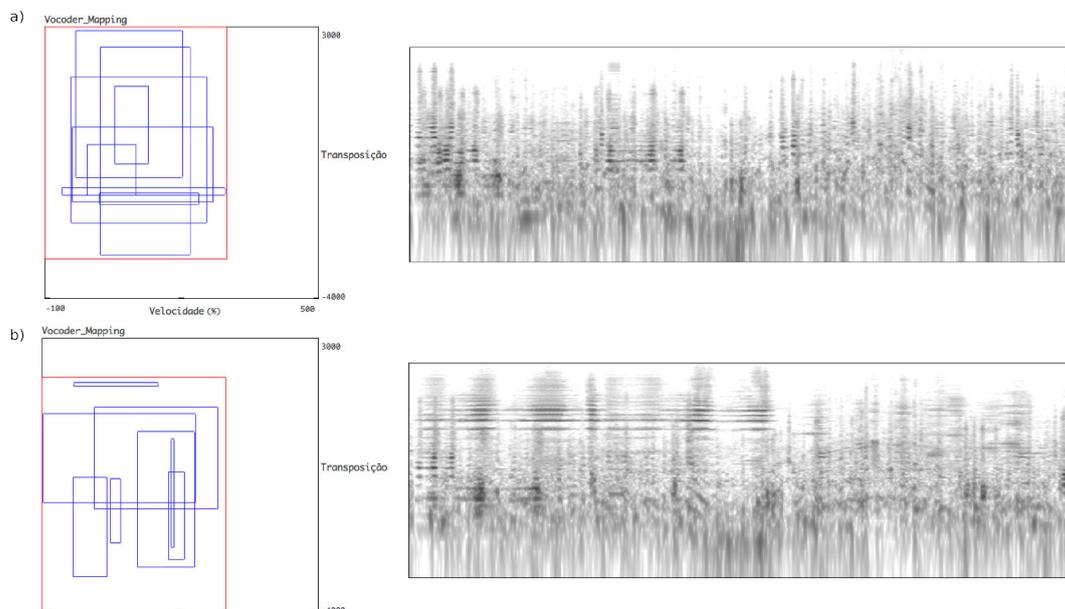
Neste módulo, o som é gravado e armazenado pelo performer em um *buffer*. O sistema compreende oito módulos independentes de síntese via *phase vocoder* que processam o sinal armazenado no *buffer*. O performer pode armazenar o som continuamente através

<sup>2</sup> [www.puredata.info](http://www.puredata.info).

de um *buffer* circular ou selecionar trechos específicos ao longo da performance. Cada módulo possui dois parâmetros de que são controlados interface gestual, a dizer: a velocidade de leitura do *buffer* e a taxa de transposição de altura, associados respectivamente ao *roll* e *yaw* da composição MARG. Os dados originados da interface gestual são mapeados de modo independente para cada módulo de síntese através de uma transformação afim. Para cada módulo  $k$ , o mapeamento entre os parâmetros controle transposição e velocidade e as variáveis *roll* e *yaw* são respectivamente mapeados como:

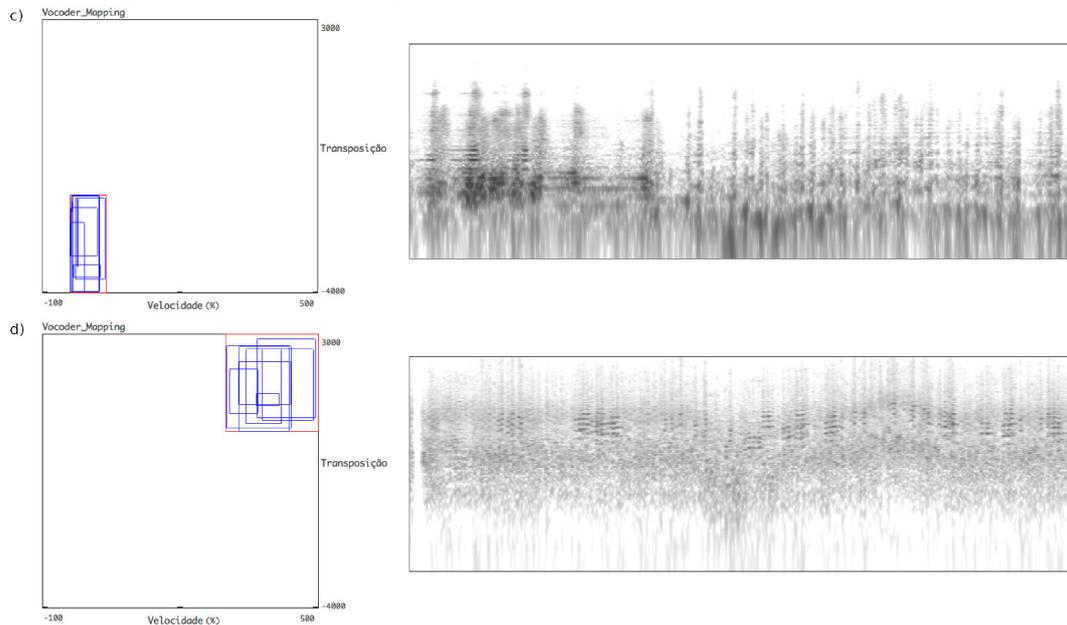
$$f(\text{roll})_k = A_k \text{roll} + B_k \quad g(\text{yaw})_k = C_k \text{yaw} + D_k$$

A manipulação dos valores de A, B, C e D associados a cada módulo  $k$  na transformação afim configura-se um controle macroestrutural que pode ser manipulado pelo instrumentista ao longo da performance. A partir de um espaço bidimensional de controle paramétrico, o performer controla o domínio sob o qual serão gerados randomicamente valores que modularão as variáveis de controle para cada módulo  $k$ . A alternância entre domínios previamente configurados pode ser realizada através de chaves de contato acopladas à flauta. Na Figura 4 demonstramos diferentes domínios estabelecidos entre as variáveis de controle e os parâmetros para cada módulo de síntese. Em cada um dos mapas (denotados pelas letras de *a* até *d*), o quadrado vermelho representa o domínio estabelecido no espaço paramétrico e os quadrados azuis o mapeamento atribuído a cada um dos oito módulos de síntese. Os espectrogramas<sup>3</sup> apresentados junto a cada mapa paramétrico apresentado pela figura demonstram as diferentes respostas do módulo através simulação de um mesmo vetor gestual armazenado da interface e de uma mesma amostra sonora utilizada no processamento<sup>4</sup>.



<sup>3</sup> Os espectrogramas foram gerados com o auxílio do software *Sonic Visualizer* (<http://www.sonicvisualiser.org/>), com taxa de amostragem de 44.1KHz, janelamento 4096 amostras, taxa de sobreposição de 512 amostras e escala logarítmica na representação das frequências.

<sup>4</sup> Os arquivos de áudio gerados na simulação apresentada pela Figura 4 encontram-se em anexo.



**Figura 4:** Mapa de controle macroestrutural das transformações afins associadas a cada módulo de síntese e as respectivas respostas de espectro obtidas pela simulação de um mesmo vetor gestual e mesma amostra sonora processada.

## 4.2 Filtragem espectral

Neste outro módulo, o som do instrumento é processado em tempo real por um conjunto de 24 filtros que recombina informações espectrais de amostras captadas e armazenadas do próprio instrumento. Ao longo da improvisação, o performer amostra o áudio executado utilizando uma das chaves acopladas ao instrumento. Destas amostras, são armazenadas tanto as informações de magnitude do espectro sonoro, bem como as informações gestuais diretas e indiretas associadas ao momento de execução das mesmas. O módulo possui a capacidade de armazenamento de até 500 quadros de análise de amostras. As informações gestuais diretas utilizadas neste módulo foram o deslocamento azimutal (*yaw*) e de elevação (*roll*) do corpo do instrumento, que constituem um espaço bidimensional de representação do movimento. As informações indiretas utilizadas foram o *loudness* e *chroma*, ambos obtidos via Transformada de Fourier (FFT) do sinal amostrado.

Cada filtro processa o sinal em tempo real a partir de uma combinação da magnitude espectral de diferentes amostras armazenadas. Tendo em vista o estabelecimento de um controle macroestrutural do performer em relação à seleção destas amostras, utilizamos o algoritmo *boids* como modelo de controle da interface. O algoritmo, desenvolvido por Craig Reynolds [20] e implementado em Pure Data por Eric Singer, Jasch e Andr Sier, simula o comportamento emergente do deslocamento espacial de pássaros e baseia-se num conjunto de regras simples e locais, tais como: separação, alinhamento e coesão entre vizinhos.

No presente módulo, cada filtro é associado a um *boïd*, que navega sobre o espaço bidimensional de representação do movimento, constituído por sua vez das informações gestuais diretas (*yaw* e *roll*) amostradas em *buffer*. A posição do *boïd* associado a cada filtro determinará quais amostras este utilizará no processamento do sinal do instrumento, constituindo deste modo o filtro uma soma ponderada da magnitude espectral das três amostras mais próximas ao referido *boïd* no espaço de representação do movimento. Matematicamente, denotamos por  $B_t = \{x_{bt}, y_{bt}\}_{b=1,2,\dots,24}$  o conjunto de coordenadas de cada *boïd*  $b$  no tempo  $t$  e  $A = \{x_a, y_a\}_{a=1,2,\dots,500}$  o conjunto de coordenadas espaciais associadas a cada amostra armazenada. A magnitude espectral associada ao filtro  $b$  resultará da soma ponderada das três menores distâncias entre  $A$  e  $B_t$ , de forma que:

$$f(b)_t = \frac{1}{\sum_{j=1}^3 \bar{d}(\min_j)} \sum_{j=1}^3 \left[ \sum_{k=0}^N \text{mag}(\min_j D(A, B_t)) \bar{d}(\min_j D(A, B_t)) \right]$$

A utilização deste algoritmo no controle de parâmetros musicais propicia o surgimento de comportamentos emergentes, que possuem um certo grau de complexidade e diversidade, sem, no entanto, sobrecarregar o performer com um demasiado número de parâmetros de controle. As variáveis de controle em tempo real do sistema estão associadas a parâmetros globais, tais como: a posição e força de um ponto atrator aos *boïds*, a velocidade global, mínima e máxima desempenhada pelos mesmos e a inércia ou prontidão para mudança de velocidade e aceleração dos *boïds*. Na Tabela 3 apresentamos as variáveis de controle associado ao controle de parâmetros globais do algoritmo.

**Tabela 3: Parâmetros globais do algoritmo *boïds* associado às variáveis de controle da *Metaflauta***

Parâmetros de controle do algoritmo	Mapeamento
Posição (x, y) do ponto atrator dos <i>boïds</i>	<i>Roll</i> e <i>Yaw</i> da composição MARG
Força de atração do ponto atrator	Magnitude da aceleração obtida do acelerômetro + <i>Loudness</i>
Velocidades global, máxima e mínima desempenhada pelos <i>boïds</i>	Valores de média, mínimo e máximo de velocidade integrada da composição MARG
Inércia (prontidão para mudança de velocidade e aceleração dos <i>boïds</i> )	<i>Fluxo Espectral</i> (desvio padrão local)

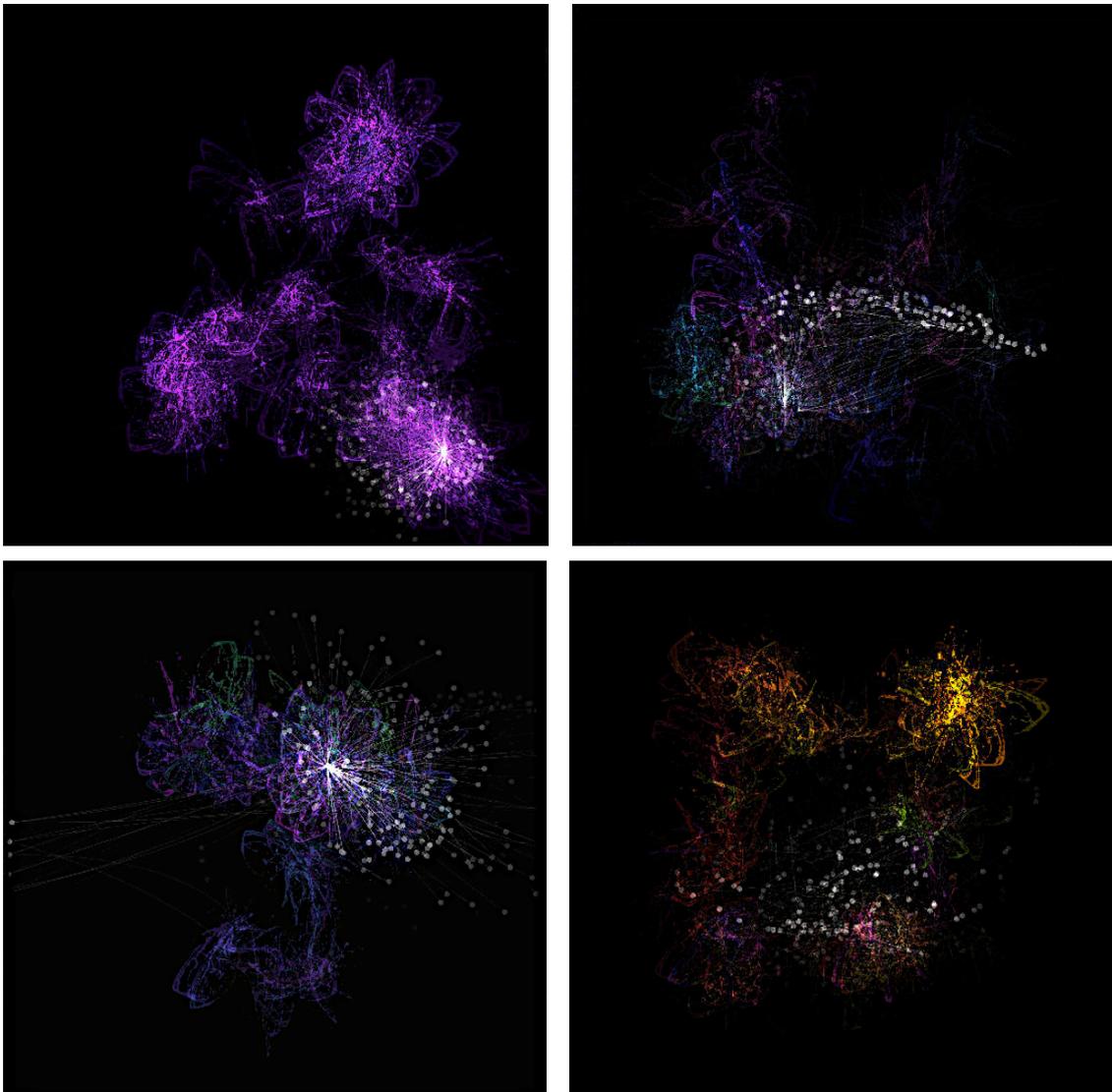
O algoritmo *boïds* é também utilizado como controle da espacialização sonora. A posição no eixo azimutal do sinal sonoro resultante de cada filtro é associado ao movimento do respectivo *boïd* no espaço de representação do movimento.

## 5. Processamento visual em tempo real

Nossa proposta de processamento visual controlado pela *Metaflauta* foi o estabelecimento de uma associação direta com os parâmetros de controle aplicados aos módulos de síntese e processamento do som. A composição gráfica foi realizada com o auxílio da biblioteca GEM implementa em Pure Data. As amostras sonoras armazenadas em *buffer* e utilizada no processamento de ambos os módulos de síntese sonora

referidos anteriormente são representadas pela renderização de múltiplas texturas. A posição de cada textura na imagem é definida pelas coordenadas da composição MARG registradas no momento de captura da amostra. A cor de cada da textura é definida pelo centro de massa do *chroma* e o tamanho e taxa de transparência da textura definida pelo *loudness*, ambas medidas extraídas da análise da referida amostra.

Associado ao controle de parâmetros do módulo de síntese via filtragem espectral, o deslocamento dos *boids* foi representado por meio de círculos e os filtros selecionados ao longo do tempo por cada *boid* representado por linhas curvas que conectam os referidos pontos no espaço de representação do movimento. Na Figura 5 apresentamos alguns resultados visuais obtidos pelo processamento em tempo real dos dados originados da *Metaflauta*.



**Figura 5: Amostras do processamento visual em tempo real gerado a partir dos dados extraídos da interface gestual da *Metaflauta***

## 6. Considerações Finais

Apresentamos neste trabalho a concepção e desenvolvimento da *Metaflauta*, um instrumento aumentado baseado num sistema híbrido de captura gestual. Em nosso modelo, a aquisição direta foi aplicada como estratégia para a captura de gestos auxiliares, ao passo que alguns gestos diretamente correlacionados à produção sonora foram capturados indiretamente, isto é, por meio da análise e processamento do sinal de áudio do instrumento. Discutimos ainda as estratégias de mapeamento das variáveis obtidas da interface no controle de parâmetros de processamento de som e imagem. Consideramos que a associação de informações gestuais capturadas direta e indiretamente demonstrou-se eficaz como estratégia de mapeamento de múltiplos parâmetros de controle dos módulos de processamento audiovisual.

Em trabalhos futuros, pretendemos ampliar a utilização das informações gestuais capturadas como variáveis de controle para os módulos de processamento audiovisuais. Tendo em vista o bom desempenho no controle de parâmetros através de informações de mais alto nível obtidas pela fusão de diferentes sensores, consideramos a utilização de procedimento similar aos outros dados capturados. A interpretação de informações de mais alto nível, sobretudo em relação às informações gestuais capturadas indiretamente, poderá assegurar uma maior robustez e precisão no controle de múltiplos parâmetros sem, no entanto, dispender um considerável aumento na complexidade de performance do instrumento. Numa perspectiva próxima, pretendemos ainda explorar a utilização de métodos de aprendizado não supervisionado de máquina no mapeamento da interface que possibilite um ajuste dinâmico das variáveis de controle mapeadas aos diferentes parâmetros de controle sonoro e visual.

## Referências

- [1] MIRANDA, E. R.; WANDERLEY, M. M. *New digital musical instruments: control and interaction beyond the keyboard*. Middleton, USA: A-R Editons, 2006.
- [2] POUSSET, D. "La flute-midi, l'historique & quelques applications" *Master's thesis, Université de Paris - Sorbonne*, 1992.
- [3] YSTAD S.; VOINIER, T. "A virtually real flute". *Computer Music Journal*, vol. 25, no. 2, pp. 13-24, 2001.
- [4] PALACIO-QUINTIN, C. "The hyper-flute" in *Proceedings of the International Conference New Interfaces for Musical Expression, NIME-03*, Montreal, 2003, pp. 206–207.
- [5] SILVA, T.; WANDERLEY, M. M.; SCAVONE, G. "On the Use of Flute Air Jet as A Musical Control Variable ," in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME05)*, Vancouver (Canada), 2005, pp. 105-108.
- [6] ALMEIDA, A.; CHOW, R.; SMITH, J.; WOLFE, J. "The kinetics and acoustics of fingering and note transitions on the flute" *Journal of Acoutical Society of America*, vol. 126, no. 3, pp. 1521-1529, 2009.
- [7] KERELIUK, C.; SCHERRER, B.; VERFAILLER, V.; DEPALLE, P.;

- WANDERLEY, M. M. "Indirect Acquisition of fingerings of harmonic notes on the flute" in *Proceedings of the International Computer Music Conference (ICMC'07)*, Copenhagen, Denmark, 2007, pp. 263-266.
- [8] VERFAILLE, V.; DEPALLE, P.; WANDERLEY, M. M. "Detecting overblown flute fingerings from the residual noise spectrum" *Journal of Acoustical Society of America*, vol. 127, no. 1, 2010.
- [9] TRAUBE, C.; DEPALLE, P.; WANDERLEY, M. M. "Indirect Acquisition of Instrumental Gesture Based on Signal, Physical and Perceptual Information" in *Proceedings of the 2003 Conference on New Interfaces for Musical Expression (NIME-03)*, Montreal (Canada), 2003.
- [10] MONTEIRO, A. C.; MANZOLLI, J. "A Framework for Real-time Instrumental Sound Segmentation and Labeling," in *Proceedings of IV International Conference of Pure data*, Weinmar, 2011.
- [11] YSTAD, S.; VOINIER, T. "Design of a Flute Interface to Control Synthesis Models," in *Proceedings of the International Computer Music conference ICMC99*, Beijing (China), 1999.
- [12] BELL, B. et al., "The Multimodal Music Stand," in *Proceeding of New Interfaces for Music Expression Conference, NIME07*, New York, 2007.
- [13] WANDERLEY, M. M.; VINES, B. "Origins and functions of clarinetist's Ancillary Gestures," in *Music and Gesture*, Ashgate, Hampshire, England, 2006.
- [14] GOUYON, F.; PACHET, F.; DELERUE, O. "On the use of zero-crossing rate for an application of classification of percussive sounds.," in *Proceedings of the 6th conference on digital audio effects*, Verone, Italy, 2000, pp. 147-152.
- [15] WESSEL, D. L. "Timbre space as a musical control structure," *Computer Music Journal*, vol. 3, no. 2, pp. 45-52, 1979.
- [16] ONG, B. S. "Towards automatic music structural analysis identifying characteristic within-song excerpts in popular music" *Master's thesis, Music Technology Group, UPF*, 2005.
- [17] FUJISHIMA, T. "Realtime Chord Recognition of Musical Sound: a System Using Common Lisp Music," in *CCRMA*, 1999.
- [18] ZWICKER, E.; FASTL, H. *Psychoacoustics: Facts and models*. Secaucus, NJ, USA: Springer-Verlag New York, 1990.
- [19] ZWICKER, E. "Subdivision of the audible frequency range into critical bands" *Journal of the Acoustical Society of America*, vol. 33, pp. 248-249, 1961.
- [20] REYNOLDS, C. "Flocks, herds and schools: A distributed behavioral model," in *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques (Association for Computing Machinery)*, New York, 1987, pp. 25-34.

# GuitarAMI: um Instrumento Musical Aumentado que Transpõe Restrições Intrínsecas do Violão

Eduardo Aparecido Lopes Meneses<sup>1\*</sup>, José Eduardo Fornari Novo Junior<sup>1</sup>

<sup>1</sup>NICS - Núcleo Interdisciplinar de Comunicação Sonora  
Universidade Estadual de Campinas

Rua da Reitoria, 165 - Cidade Universitária “Zeferino Vaz” – 13083-872 Campinas, SP

eduardo@edumeneses.com, tutifornari@gmail.com

**Abstract.** *Despite its huge spread and popularity, the acoustic guitar has intrinsic and characteristic restraints of its physical structure. This article discusses the construction and use of an augmented musical instrument (AMI) entitled GuitarAMI. While most of the digital musical instruments are designed to allow control of sound synthesis and transformations during the instrumental performance, this AMI was created in order to modify some of the intrinsic characteristics of the acoustic guitar, commonly considered restrictive in guitaristic practice.*

**Resumo.** *Apesar da sua enorme disseminação e popularidade, o violão apresenta restrições intrínsecas e características da sua estrutura física. Este artigo discute a construção e utilização de um instrumento musical aumentado (AMI) intitulado GuitarAMI. Enquanto grande parte dos instrumentos musicais digitais foram criados para possibilitar o controle de síntese e transformação sonora durante a performance instrumental, este AMI foi criado com o objetivo de modificar algumas das características intrínsecas do violão, comumente consideradas restritivas na prática violonística.*

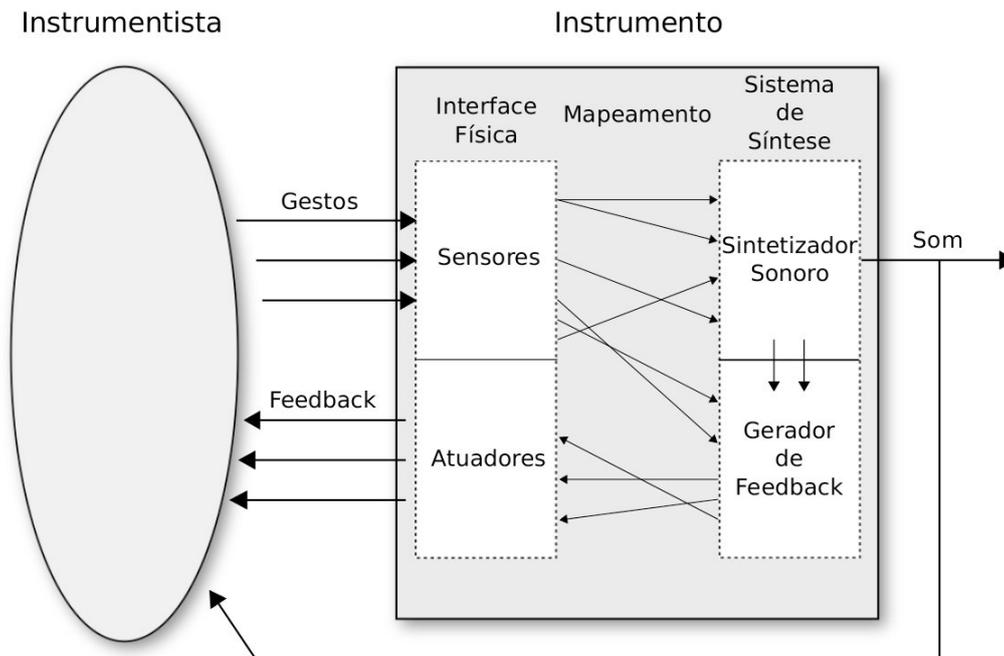
## 1. Introdução

A partir da segunda metade do século XX houve uma verdadeira revolução tecnológica, ocorrida com a chegada de diversos dispositivos e plataformas eletrônicas que, além de possuírem um baixo custo, promoveram formas facilitadas de construção (*hardware*) e programação (*software*), possibilitando a utilização destes recursos em atividades artísticas antes pouco beneficiadas pelos avanços tecnológicos, como a performance musical ou a criação de instalações artísticas. Entre outros, podemos citar como exemplos de implementações utilizando software-livre e interfaces gestuais de baixo custo, as descritas em [Aliel and Fornari, 2013] e [Fornari, 2012]. Esta facilidade de programação é de fundamental importância para diminuir a curva de aprendizagem sobre determinado dispositivo e o tempo investido em sua criação, como afirma Perry Cook:

A performance musical com novas interfaces [instrumentos] de computador é atualmente corriqueira como resultado da disponibilidade de dispositivos eletrônicos de baixo custo, novos sensores para capturar diversos parâmetros como intensidade [do movimento] e localização e novos programas de computador para síntese e manipulação do som em tempo real. [Cook, 2001, p. 1; Tradução e comentários nossos.]<sup>1</sup>.

\*Bolsista CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior).

<sup>1</sup>Musical performance with entirely new types of computer instruments is now commonplace, as a result of the availability of inexpensive computing hardware, of new sensors for measuring physical parameters such as force and position, and of new software for real-time sound synthesis and manipulation [Cook, 2001, p. 1].



**Figura 1: O modelo de DMI de Mark Marshall [Marshall, 2008, p. 26], onde pode-se visualizar a relação entre os diversos componentes. Um mesmo *hardware* pode originar diferentes DMIs (ou AMIs) alterando-se algoritmos de síntese e manipulação sonora executados pelo processador.**

Este cenário proporcionou a construção de diversos instrumentos musicais digitais (DMIs, sigla de *Digital Musical Instruments*) e instrumentos musicais aumentados (AMIs, sigla de *Augmented Musical Instruments*), além de uma considerável quantidade de controladores digitais inspirados ou similares à instrumentos acústicos tradicionais, alguns deles descritos por Miranda e Wanderley em [Miranda and Wanderley, 2006, p. 20]. Utilizamos neste trabalho a sigla DMI para designar de forma genérica controladores gestuais utilizados no fazer musical, sejam estes constituídos apenas de componentes eletrônicos, possuïrem partes eletrônicas e acústicas ou serem exclusivamente virtuais.

Os AMIs são DMIs específicos onde instrumentos musicais tradicionais foram aumentados com a inclusão de sensores e atuadores que permitem a este novo instrumento híbrido o controle de parâmetros sonoros ou mesmo a criação de novos sons [Miranda and Wanderley, 2006, p. 21-22].

## 2. GuitarAMI

Para a construção de um AMI deve-se considerar não somente a montagem de sensores e atuadores em um instrumento acústico, mas também o próprio instrumento acústico, o mapeamento entre os dados gestuais capturados pelo dispositivo e sua conversão em comandos para que a programação seja executada pelo computador responsável pelo processamento sonoro. Todos estes componentes são parte integrante de um AMI funcional.

No âmbito geral, os modelos de DMIs propostos por Bert Bongers, David Birnbaum, Marcelo Wanderley e Perry Cook foram analisados em [Marshall, 2008] e Mark Marshall criou um modelo de construção de DMIs que serviu de ponto de partida para a criação de um AMI utilizando o violão.

A construção e experimentação do primeiro protótipo de um AMI utilizando o violão foi construído durante o segundo semestre de 2014 [Meneses et al., 2015a]. Este dispositivo foi intitulado **GuitarAMI** e em sua criação foram utilizados *hardwares* e *soft-*



**Figura 2: Performance do trio B.E.A.T.<sup>7</sup> no evento *Palestras e Performances*, ocorrido no Auditório do Instituto de Artes da Unicamp em 24 de junho de 2015.**

ware livres. A escolha dos componentes do AMI também considerou a facilidade de programação e construção, já que as plataformas Arduino<sup>2</sup> e Pure Data<sup>3</sup> (PD), utilizadas no dispositivo, são amplamente documentadas e possuem como premissa a facilidade de uso.

O GuitarAMI foi utilizado tanto em performance quanto como ferramenta pedagógica onde, para cada uma das diferentes situações, foram utilizados *patches* (algoritmos desenvolvidos em PD) diferentes.

Para uma performance do trio B.E.A.T.<sup>4</sup>, mostrado na figura 2 e formado pelos músicos Alê Damasceno (bateria), Walmir Gil (trompete) e Edu Meneses (violão), o GuitarAMI foi utilizado executando um *patch* para enviar comandos no protocolo MIDI para o *software* Integra Live<sup>5</sup> onde foram utilizados algoritmos de síntese sonora controlados dinamicamente pela posição do violão.

Durante um curso ministrado no Programa Guri<sup>6</sup>, pelo autor principal deste trabalho, os alunos puderam utilizar o GuitarAMI para disparar e controlar eventos sonoros pré-programados, utilizando o sensor ultrassônico disponível no GuitarAMI, que coleta em tempo real dados da distância entre o instrumento e o instrumentista.

O comportamento do GuitarAMI é portanto diretamente controlado pelos algorit-

---

<sup>2</sup>Arduino é uma plataforma de *hardware* livre para computação baseado em uma placa de microcontrolador simples, trabalhando em conjunto com um aplicativo utilizado como ambiente de desenvolvimento para escrever algoritmos e enviá-los ao *hardware* [Meneses et al., 2015b]. Maiores informações: <http://arduino.cc/en/Guide/Introduction>.

<sup>3</sup>Pure Data, ou PD, é uma linguagem visual de programação de código aberto desenvolvido originalmente no IRCAM por Miller Puckette e projetado para criar algoritmos (*patches*) de geração, controle e manipulação de dados de controle, áudio e vídeo, em tempo real [PD, 2015].

<sup>4</sup>Acrônimo de *Brazilian Electronic Aleatorium Trio*.

<sup>5</sup>O Integra Live é um programa de computador para processamento de áudio interativo originalmente financiado pela Creative Europe Desk UK entre 2007 e 2013. Atualmente o Integra Live é mantido pelo Integra Lab no Birmingham Conservatoire, no Reino Unido [IntegraLive, 2015].

<sup>6</sup>O Programa Guri é um programa de educação musical e inclusão social iniciado em 2008 através de uma parceria entre a Secretaria de Cultura do Estado de São Paulo e a Santa Marcelina - Organização Social de Cultura (<http://www.gurisantamarcelina.org.br/>) [Meneses et al., 2015a].

<sup>7</sup>Vide nota de rodapé 4.

mos contidos neste *patch* do PD. Pode-se também concluir que, de acordo com o modelo de DMI de Marshall, foram utilizados dispositivos diferentes em cada uma das atividades relatadas.

### 3. Expandindo as possibilidades violonísticas

Diferentemente de DMIs, que podem ser modificados não somente pela adição ou subtração de componentes físicos mas também pela alteração dos algoritmos responsáveis pela síntese e manipulação sonora, instrumentos musicais acústicos possuem características estruturais (físicas) correspondentes a características acústicas que são praticamente imutáveis, decorrentes da forma com que estes instrumentos foram construídos. Tais características, sejam elas consideradas restritivas ou enriquecedoras, podem ser exploradas em performances e composições musicais, ou mesmo evitadas ou minimizadas, de acordo com os critérios estéticos ou escolhas criativas do interprete ou compositor.

Como grande parte dos instrumentos de cordas, o violão possui uma grande variedade timbrística, porém outras características do instrumento, como envelope dinâmico de intensidade sonora com pouca sustentação ou impossibilidade de controle dinâmico após o ataque, são usualmente consideradas severas restrições impostas pela estrutura física do violão.

Uma das possibilidades de utilização do GuitarAMI é na execução de algoritmos programados na forma de *patches* em PD para reverter as restrições descritas anteriormente. Para esta tarefa foi criado um *patch* que utiliza o som emitido pelo violão e realiza manipulações sonoras com a finalidade de sustentar um elemento sonoro registrado pelo computador ou controlar o volume deste mesmo elemento.

Para que seja possível a sustentação da intensidade de um som emitido pelo violão por tempo indeterminado, a solução adotada foi a utilização da Transformada Rápida de Fourier, ou FFT<sup>8</sup>. Com a FFT é possível analisarmos um sinal de áudio em determinado momento no tempo para obtermos uma representação contendo a amplitude de todas as frequências do espectro deste mesmo áudio. O resultado desta análise é utilizado para gerar um som de duração indefinida que pode ser sintetizado com a utilização da operação inversa da FFT [Martin, 1986]. Este algoritmo é comumente denominado *Spectral Freeze*, ou simplesmente *Freeze* conforme mostra a figura 4.

Para a programação deste algoritmo utilizamos referencialmente o *patch freeze3.pd* disponibilizado por um membro ativo<sup>9</sup> do fórum do Pure Data [PD-Forum, 2015]. A comunidade de desenvolvedores do PD é extremamente ativa e muitos dos algoritmos criados por usuários são disponibilizados, alterados e estudados pelos membros da comunidade. Para se adequar à proposta de utilização pelo GuitarAMI, a programação do algoritmo *freeze3.pd* foi modificada com a inclusão de opções de sensibilidade para a captura do áudio a ser sintetizado, parâmetros mapeados de forma direta com os sensores do AMI e um comando para liberação (*release*) do áudio sintetizado. Houve também acréscimo da programação referente ao controle de intensidade sonora do áudio sintetizado.

Os parâmetros disponíveis para controle desta manipulação sonora são o comando instantâneo de captura do áudio e início da sustentação, o comando instantâneo de liberação (*release*) deste mesmo áudio e o controle de volume do evento sonoro gerado pelo computador (o som sustentado indefinidamente). Para que o instrumentista controle estes parâmetros, foram utilizados dois interruptores momentâneos modelo SPST (botões ou *foot switches* de uma via de acionamento simples) disponíveis na base do GuitarAMI

<sup>8</sup>Sigla de *Fast Fourier Transform*.

<sup>9</sup><http://forum.pdpatchrepo.info/user/ralf>.

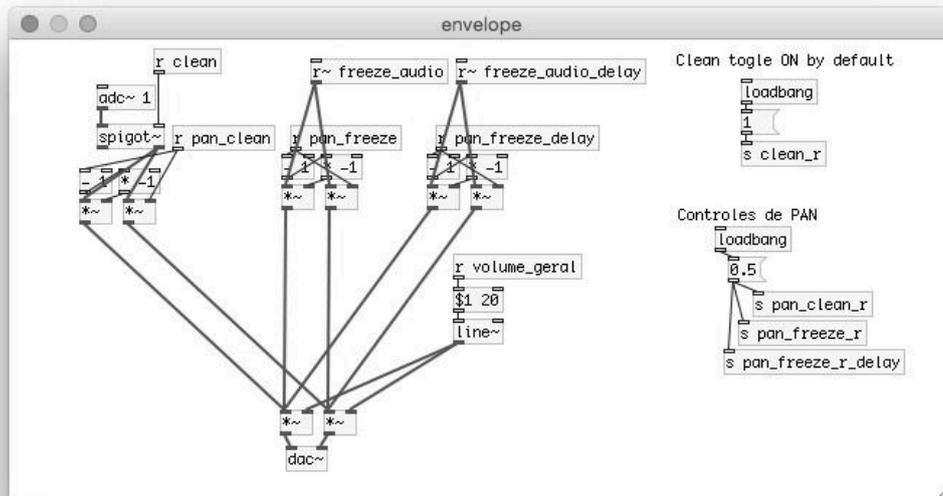


Figura 3: *Subpatch* contendo a programação do algoritmo *Env-shaper*.

(conectada ao computador) e que foram configurados para captura e liberação (*release*) do som do violão pelo algoritmo, além de um dos eixos do acelerômetro disponível no módulo do GuitarAMI (afixado na faixa do violão) para controle de volume geral.

Assim, o instrumentista pode capturar e liberar a síntese sonora ao acionar os interruptores com os pés, além de controlar a intensidade sonora da síntese, afastando ou aproximando o violão em relação ao seu corpo. Para esta movimentação do instrumento efetuamos o mapeamento gestual de forma que o parâmetro de controle de intensidade do *Freeze* (mostrado na figura 4) seja controlado simultaneamente pelos 3 eixos de sensibilidade do acelerômetro. Este tipo de mapeamento é chamado de *many-to-one* [Hunt et al., 2000] e, ao acrescentar maior complexidade no mapeamento, proporciona também uma experiência de controle mais próxima àquela oferecida pelos instrumentos acústicos tradicionais [Wanderley and Depalle, 2004].

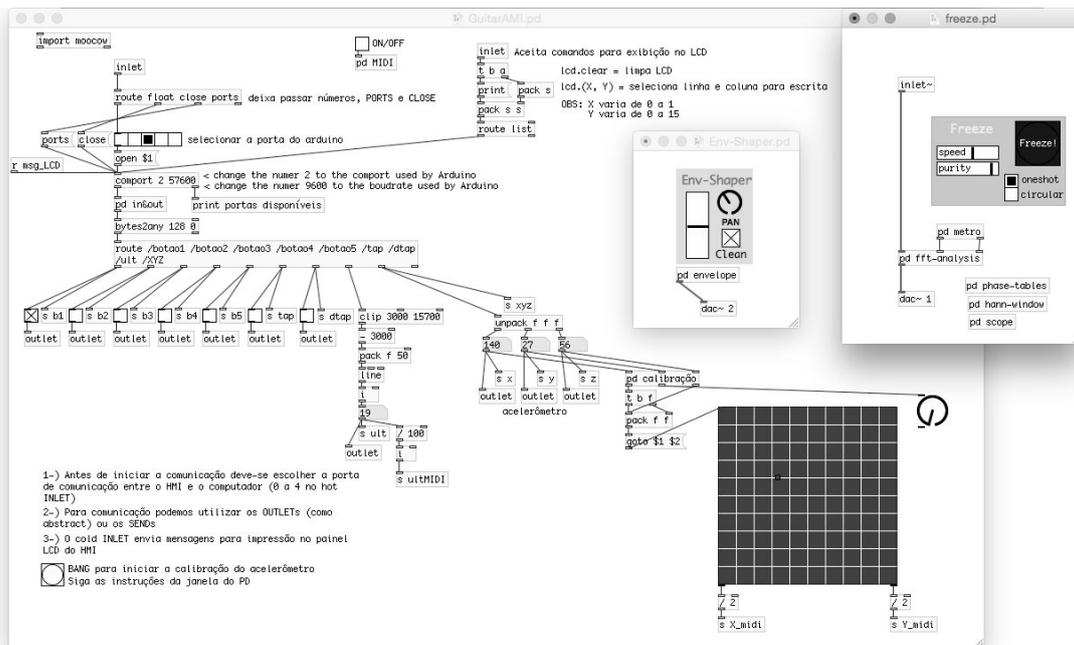
Para que o instrumentista possa controlar a intensidade de um determinado som após sua emissão, de forma independente e sem a necessidade de utilizar o algoritmo *Freeze*, foi programado um *subpatch* dentro do *patch Freeze*. Este é responsável exclusivamente pela modelagem de envelope dinâmico em tempo real (*real-time envelope shaper*) da intensidade sonora; ou *Env-shaper*.

As informações geradas pela interação entre instrumentista e o sensor ultrassônico foram coletadas pelo módulo do GuitarAMI e convertidas em comandos que podem ser mapeados de forma direta (*one-to-one*) a um algoritmo de controle de intensidade sonora atuando diretamente sob o som captado do violão.

Este sensor ultrassônico é responsável pela aquisição de gestos instrumentais efetivos [Cadoz and Wanderley, 2000] realizados com uma das mãos do instrumentista que modula a intensidade sonora captada no violão. É, desta forma, possível ao instrumentista gerar desta forma envelopes dinâmicos antes impossíveis no violão tradicional.

#### 4. Discussão e Conclusão

Este artigo apresentou o projeto e desenvolvimento de um violão acústico expandido, intitulado GuitarAMI. Observou-se que a utilização dos *patches Freeze* e *Env-shaper* de



**Figura 4:** Patch para Pure Data contendo os algoritmos *Freeze* e *Env-shaper*, além da versão preliminar do algoritmo de controle e mapeamento gestual do GuitarAMI.

fato expandiram a sonoridade do violão, conforme apresentado na seção 3.

O GuitarAMI se comportou como esperado e com o patch apresentado na figura 4 foi possível alterar o envelope dinâmico ou sustentar indefinidamente sons emitidos pelo instrumento, desde que estes estejam dentro do espectro de sons que possam ser captados pelo equipamento utilizado, microfone ou captador específico do instrumento.

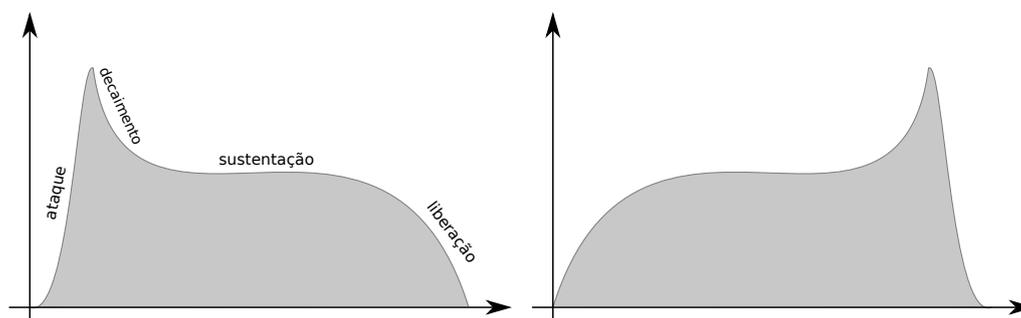
É interessante relatar algumas particularidades percebidas durante os testes de performance com o GuitarAMI. Como esperado, na utilização do algoritmo *Freeze* observou-se que o momento de acionamento do interruptor é crucial para definir a qualidade do som sustentado. Ao efetuar a captura durante o momento de ataque do envelope sonoro original do violão foi enviado ao algoritmo uma janela de captura contendo diversos ruídos transientes característicos desta fase, como o som das unhas do instrumentista friccionando as cordas. O momento ideal para registrar a janela de captura parece ser imediatamente após o ataque, quando grande parte dos ruídos transientes (estocásticos) já perderam energia e temos a presença mais definida da parte tonal (determinística) do sinal sonoro do violão [Serra, 1989].

A síntese sonora realizada pelo *Freeze* se torna mais distante do som original quanto maior for a sua complexidade. Acordes e sons muito ricos em harmônicos podem originar sons sintetizados com características sonoras bem diferentes das esperadas pelo instrumentista. Esta característica pode ser utilizada expressivamente durante uma performance, porém se distancia do objetivo do algoritmo, que foi programado para produzir um som que seja auditivamente equivalente ao som original, porém com duração indefinida.

O controle da intensidade sonora do som sintetizado no algoritmo *Freeze* requer posicionamento e movimentação precisas do violão. Parâmetros de sensibilidade da captura gestual realizada pelo acelerômetro devem ser calibrados de acordo com o instrumentista para fornecer uma resposta expressiva condizente com a intenção estética deste.

O algoritmo *Env-shaper*, apesar de sua simplicidade, requer alguma prática do

instrumentista para poder apresentar resultados satisfatórios durante a performance. É possível recriar em tempo real envelopes sonoros recorrentes na música acusmática e que antes eram realizados com a retrogradação do envelope sonoro, conforme exemplificados na figura 5.



**Figura 5:** À esquerda exemplo de um envelope sonoro com seus componentes; À direita a retrogradação do mesmo envelope (uma das possibilidades de utilização do algoritmo *Env-shaper*). Utilizando o *Env-shaper* o instrumentista é capaz de criar diferentes formatos de envelope sonoro em tempo real.

## 5. Trabalhos Futuros

O GuitarAMI apresenta uma série de possibilidade de futuros desenvolvimentos e expansões. Entre outras, com relação à diferentes parametrizações dos algoritmos utilizados. O algoritmo *Freeze* possui diversos parâmetros de programação passíveis de alteração e que podem alterar o resultado sonoro apresentado. Ainda são necessários ajustes para obter uma síntese sonora psicoacusticamente mais próxima do som emitido pelo violão.

Pode também vir a ser desejável incorporar permanentemente estes algoritmos à programação do GuitarAMI e fornecer ferramentas para que o instrumentista possa mapear, de forma intuitiva, os movimentos recebidos e processados pelo dispositivo. Um *patch* especificamente programado para interconectar o GuitarAMI a quaisquer outros algoritmos desenvolvidos pelo instrumentista ou por um compositor poderá vir a contribuir no sentido a expandir as possibilidades de uso desta plataforma em futuros estudos e desenvolvimentos.

## Referências

- Aliel, L. and Fornari, J. (2013). Creating an Ecologically Modeled Performance Through the Remote Manipulation of Multiple Soundscapes. In *Arquivos do XIV Simpósio Brasileiro de Computação Musical (SBCM)*, Brasília - DF - Brasil. Grupo de Computação Musical - IME/USP.
- Cadoz, C. and Wanderley, M. M. (2000). Gesture-music. In Wanderley, M. M. and Battier, M., editors, *Trends in Gestural Control of Music*, pages 71–94. Editions IRCAM – Centre Pompidou, Paris.
- Cook, P. (2001). Principles for designing computer music controllers. *Conference on New Interfaces for Musical Expression (NIME-01)*, pages 1–4.
- Fornari, J. (2012). Designing Bodiless Musical Instruments. In *Anais do 10<sup>o</sup> Congresso de Engenharia de áudio*, number 10, pages 41–46, São Paulo. 16<sup>a</sup> Convenção Nacional da AES Brasil.

- Hunt, A., Wanderley, M. M., Kirk, R., Pompidou, I. C., and France, P. (2000). Towards a Model for Instrumental Mapping in Expert Musical Interaction. Technical report.
- IntegraLive (2015). Integra live - web site. In <http://www.integralive.org/>. Último acesso em 13 de março de 2015.
- Marshall, M. T. (2008). *Physical Interface Design for Digital Musical Instruments*. PhD thesis, McGill University, Montreal.
- Martin, G. V. (1986). Final project report: Real-time Sound Freezing and Spectral Shifting. Technical report.
- Meneses, E. A. L., Fornari, J., and Wanderley, M. M. (2015a). Projeto e construção de uma interface gestual para improvisação livre. In *Anais do 13º Congresso de Engenharia de Áudio da AES Brasil*, pages 75–81. AES – Audio Engineering Society – Brazil Section.
- Meneses, E. A. L., Fornari, J., and Wanderley, M. M. (2015b). A study with hyperinstruments in free musical improvisation. In *Anais do XI Simpósio Internacional de Cognição e Artes Musicais*, Pirenópolis/GO. Associação Brasileira de Cognição e Artes Musicais (ABCM).
- Miranda, E. R. and Wanderley, M. M. (2006). *New Digital Musical Instruments: Control and Interaction Beyond the Keyboard*. Computer Music and Digital Audio Series. A-R Editions, Inc.
- PD (2015). Pure data - web site. In <http://puredata.info/>. Último acesso em 20 de abril de 2015.
- PD-Forum (2015). Pure data oficial forum. In <http://forum.pdpatchrepo.info/>. Último acesso em 14 de setembro de 2015.
- Serra, X. (1989). *A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition*. PhD thesis, Stanford University.
- Wanderley, M. M. and Depalle, P. (2004). Gestural control of sound synthesis. *Proceedings Of The Ieee*, 92(4):632–644.

## ***Caminho das águas, instalação audiovisual interativa: implementação computacional e interação dos visitantes***

**Clayton R. Mamedes<sup>1,2</sup>, Artemis Moroni<sup>2,4</sup>, Jônatas Manzolli<sup>1,2</sup>, Denise Garcia<sup>1,3</sup>**

<sup>1</sup>Instituto de Artes - UNICAMP

<sup>2</sup>Núcleo Interdisciplinar de Comunicação Sonora – NICS-UNICAMP

<sup>3</sup>Centro de Integração, Documentação e Difusão Cultural – CIDDIC-UNICAMP

<sup>4</sup>Centro de Tecnologia da Informação Renato Archer - CTI

claytonmamedes@nics.unicamp.br, artemis.moroni@cti.gov.br,  
jonatas@nics.unicamp.br, d\_garcia@iar.unicamp.br

**Abstract.** *This paper presents the computational implementation and analyses the behavior of visitors of the audiovisual installation Caminho das águas. Based on an analysis method developed during the research that includes survey, statistics about motion data and esthetic analysis about the audiovisual results reached by visitors, we observed the occurrence of gestural patterns as well as correlations about the interactive process and the artistic background of visitors.*

**Resumo.** *Este artigo apresenta a implementação computacional e analisa o comportamento dos visitantes da instalação audiovisual Caminho das águas. Baseados em um método de análise desenvolvido durante a pesquisa que inclui análises de questionários, análises estatísticas sobre os dados de movimento e análises estéticas sobre o resultado audiovisual gerado pelos visitantes, observamos a ocorrência de padrões gestuais e de correlações entre o processo interativo e a formação artística dos visitantes.*

### **1. Introdução**

O termo instalações vêm sendo empregado largamente pelo meio artístico, especialmente após a década de 1990 [Reiss 1999, p. 136]. Com o aumento na produção de obras e a interação de diferentes configurações entre objetos, espaço e som, as margens que permitiriam compartimentar obras do gênero em classes começam a apresentar limitações. Observamos na literatura a pertinência dos seguintes tópicos: 1) percepção integral do espaço pelo visitante como uma única obra; 2) experiência multis sensorial pelo visitante; 3) engajamento e imersão do visitante.

A percepção integral do espaço como uma única obra implica tratar todo o espaço da instalação como uma única situação, ao invés de tratá-lo como uma galeria para exibição de obras separadas [idem, pág. 14]. Desta forma, o uso simultâneo dos sentidos contribui para uma experiência que se manifesta através das dimensões, forma, disposição e materiais que constituem o ambiente, e sua relação com luminosidade, sons, temperatura e fragrâncias [Stallabrass 2004, p.26]. Esta experiência artística multimodal conduz à busca pela imersão do público. Claire Bishop e Graham Coulter-Smith definem que os processos de imersão se caracterizam pelo isolamento da realidade externa para o público

dentro de um espaço controlado por uma proposta artística. A percepção multimodal do espaço físico [Bishop 2005] e a consciência da proposta narrativa da obra [Coulter-Smith 2006] são tipos de imersão que consideramos complementares e que ocorrem em níveis variados.

Interação em obras de arte é objeto de pesquisa de vários autores. Nestas obras “a atividade do receptor [...] é a fonte primária de sua experiência estética” [Kwastek, 2013, p. xvii]. Podemos exemplificar o emprego de diferentes processos interativos com as instalações *Op\_era*, *ADA*, *AURAL* e *The Treachery of Sanctuary*. Estas instalações exemplificam a convergência dos principais tópicos explorados em nossa pesquisa: o controle de resultados audiovisuais pelo visitante, o emprego de estratégias para estabelecer relações entre seus gestos e estes resultados, analogias a experiências intuitivas e o emprego de captura de movimento como interface entre o visitante e a obra. Em *Op\_era* [Garcia e Campos 2003] o visitante utiliza um sistema móvel do tipo *joystick* para controlar imagens e eventos sonoros que são ajustados de acordo sua posição espacial. Esta solução simula a posição da fonte sonora dentro do espaço tridimensional da instalação. Em *ADA* [Wasserman et al. 2005], sensores no piso e microfones rastreiam a posição do visitante e, de acordo com regras pré-estabelecidas, procuram expressar emoções sintéticas baseadas no comportamento dos visitantes. *AURAL* [Moroni et al. 2009] considera trajetórias de robôs no espaço da obra como parâmetro de entrada de dados, enquanto o visitante pode interagir desenhando curvas em uma interface gráfica que modulam o resultado sonoro associado às trajetórias dos robôs. Por fim, *The Treachery of the Sanctuary*, de Chris Milk, tríptico no qual a imagem dos visitantes é projetada em três telas e controla revoadas de pássaros. Nesta obra há o controle do número de visitantes – apenas um por tela, máximo de três visitantes no total. Entretanto, situações orientadas como esta não compreendem a maior parcela de instalações que observamos.

Várias aplicações usam dados gestuais para controlar instalações e performances musicais. A maioria destas ferramentas foram desenvolvidas para aplicações específicas. As pesquisas de Christopher Dobrian e Antonio Camurri, que buscam interpretar o movimento através dos aspectos qualitativos dos gestos segmentados, exemplificam abordagens aos modelos de captura de movimento. Dobrian [Dobrian e Bevilacqua 2003] desenvolveu um conjunto de descritores baseados em propriedades físicas e dados derivados de coordenadas espaciais. Camurri [Camurri et al. 2005, Camurri e Trocca 2000] implementou soluções baseadas em redes neurais, tabelas de decisão e conjuntos de treinamento. O autor utiliza estes descritores para identificar movimentos em performances de música e dança.

A análise sobre o comportamento do público em instalações também vêm sendo objeto de estudo recente. As pesquisas de Eva Horneck [Horneck e Nicol 2011, Horneck e Stifter 2006] sobre instalações e ambientes interativos em museus utilizam uma sistemática baseada em entrevistas e observações sobre os visitantes. O método que desenvolvemos em nossa pesquisa partiu da análise baseada em questionários abertos e observações do artista-pesquisador sobre a interação dos visitantes. Este método foi utilizado em nossos estudos sobre as instalações *Abstrações* e *Cerejeira* [Mamedes et al, 2012a e Mamedes et al., 2012b]. Para o presente estudo, aos questionários associamos análises estatísticas sobre os dados de movimentos realizados pelos visitantes e estruturamos a metodologia de nossa análise sobre o comportamento dos visitantes tomando como referência as análises estéticas de J. J. Nattiez [2002, p. 15], aplicando-a sobre o resultado audiovisual

gerado pelos participantes. Procuramos analisar o processo interativo dos visitantes da instalação *Caminho das águas* com o objetivo de construir uma base que nos permita observar pertinências no processo interativo e verificar a eficiência na implementação da proposta artística, criando um ambiente propício à experiência engajada dos visitantes. Nas próximas seções apresentamos a descrição da instalação, seguida por comentários sobre as propostas de design sonoro, sobre o sistema computacional e o sistema de captura de movimento. Finalmente, analisamos o processo interativo dos visitantes utilizando o método apresentado.

## 2. Descrição da Instalação

*Caminho das águas* é uma instalação audiovisual interativa para dois projetores, oito canais de áudio e duas câmeras infravermelhas controlada por um sistema computacional implementado em MaxMSP. O tema da instalação aborda a estiagem ocorrida no outono e verão de 2014-2015. As referências visuais e sonoras da obra compreendem o repertório imagético de seca e cursos d'água minguantes, contrapondo a paisagem audiovisual natural às interferências humanas no ambiente. Essas referências aparecem revisitadas por recursos de edição, redução formal e processamento computacional em tempo real, os quais realizam a desconstrução do conteúdo original e remetem às características essenciais do material audiovisual que guia o desenvolvimento e o caráter imersivo da obra.

O material visual consiste em gravações associadas a cada seção temporal da instalação. O material sonoro compreende sons de paisagem sonora e texturas compostas previamente. Os alto-falantes são dispostos de forma a revelar aspectos da obra conforme os visitantes se deslocam pelo espaço da instalação. Essa disposição cria planos perspectivos de apreciação sonora, com foco centralizado nas regiões de melhor apreciação das projeções de vídeo, onde ocorre a detecção dos movimentos dos visitantes. As seções da instalação compreendem os seguintes estados formais:

- 1) início - seção inspirada em figuras de ondulação, linhas curvas e baixa luminosidade, distorcidos por *blurring* (borrado) e efeitos de superposição da mesma imagem por convolução de vídeo; materiais sonoros de gotas e passos na água.
- 2) a água na natureza - seção inspirada em paisagens naturais, apresentando mananciais, quedas d'água e rios processados por *zooming* (aproximação); predominância de paisagem sonora com rumores de água e vento.
- 3) a água dominada e a água fugida - seção inspirada em paisagens de seca ou que sofreram interferência humana; predominância de processamento e edição articulada (instrumental) de materiais sonoros, com pouca referência ao material sonoro original.

Parâmetros de processamento de vídeo e áudio são controlados em tempo real a partir de movimentos dos visitantes. Segmentação gestual e propriedades físicas do movimento são interpretados para controle expressivo da instalação. A interação no aspecto visual da obra controla o processamento de efeitos, sequenciamento, luminosidade e superposição de camadas de vídeo. No aspecto sonoro, a interação controla a seleção de gravações, superposição de camadas texturais, processamento de efeitos, intensidade sonora e difusão sonora. O controle da instalação se inspira no trabalho de orquestração, onde o visitante determina a densidade expressiva e o contraponto entre os materiais audiovisuais. Cada seção possui duração entre 2 e 3 minutos, totalizando cerca de 8 minutos para apreciação da obra na íntegra. Não há interrupção entre as seções, nem

indicação de sua ciclicidade, tornando o ambiente apto à interpretação temporal individual dos visitantes.

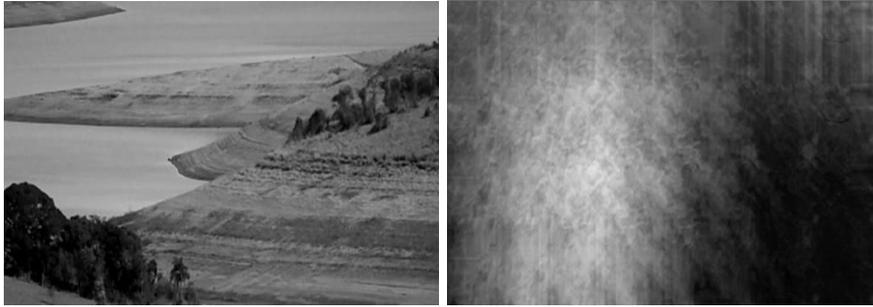


Figura 1. Excertos visuais de *Caminho das águas*.

### 3. Design sonoro em *Caminho das águas*

Nesta seção apresentamos o design sonoro utilizado na instalação e indicamos as características pelas quais classificamos o conteúdo sonoro da obra de acordo com a espectro-morfologia de Denis Smalley [Smalley, 1986]. Esta classificação foi utilizada para organizar a apresentação dos eventos sonoros de acordo com regras de unidade propostas para cada seção da obra e para cada tipo de controle gestual. Para o controle gestual selecionamos os descritores de velocidade, segmentação por velocidade, esforço e a posição angular dos membros em relação ao centro da cinesfera. Classificamos a relação entre controle gestual e sons em três categorias:

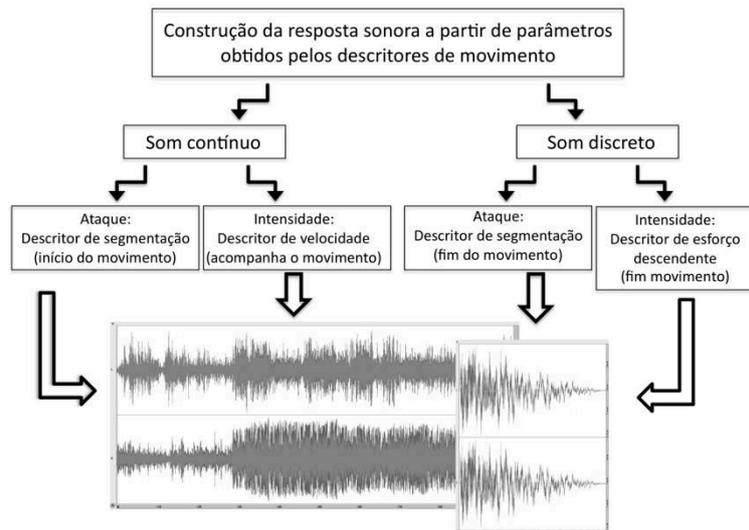
- 1) Intenção de um controle discreto do evento sonoro: sons de perfil morfológico do tipo ataque-ressonância, associados à interrupção do movimento. Controlado pelo descritor de segmentação gestual por velocidade, que determina seu início, e pelo descritor de esforço, que determina sua intensidade sonora.
- 2) Intenção de um controle contínuo do evento sonoro: sons de perfil morfológico do tipo contínuo, associados à sonificação de movimentos. Controle realizado de maneira contínua pelo descritor de velocidade, que controla o início e a intensidade dos eventos sonoros.
- 3) Evento sonoro tocado quando não há movimentação dos visitantes: controle realizado pela inversão dos resultados obtidos pelo descritor de segmentação gestual por velocidade. A intensidade foi pré-definida com a intenção de manter esta camada sonora em um nível de intensidade inferior ao dos eventos sonoros resultantes de interação.

Podem se superpor cinco planos sonoros: quatro de interação e um de acompanhamento. A classificação do conteúdo sonoro permitiu construirmos um modelo de articulação formado por duas etapas: quando o visitante inicia o movimento, este é sonificado por sons contínuos; ao interrompê-lo, sons de controle discreto marcam a segmentação gestual por um evento de perfil ataque. Como os eventos se encadeiam, cria-se um vínculo entre os eventos contínuos e sua conclusão em eventos de perfil ataque (Fig. 2).

A espacialização é baseada em uma abordagem de fontes sonoras<sup>1</sup>. A posição destas fontes dentro do espaço da instalação é simulada através do software Spat e controlada pelo descritor de relações angulares para as mãos direita e esquerda. Este descritor associa

<sup>1</sup> Fontes sonoras correspondem à posição do som no espaço. Fontes do sinal acústico correspondem à posição das caixas acústicas.

os dados de azimute para controle rotacional da fonte sonora no eixo horizontal do espaço acústico, os dados de elevação para controle rotacional no eixo vertical, e os dados de distância para controle de profundidade da fonte sonora e reverberação. Definimos que os dados serão sempre calculados em relação à posição do visitante no espaço da instalação, ou seja, o centro de sua cinesfera.



**Figura 2. Fluxograma apresentando os parâmetros de controle sonoro que compõem cada gesto do visitante.**

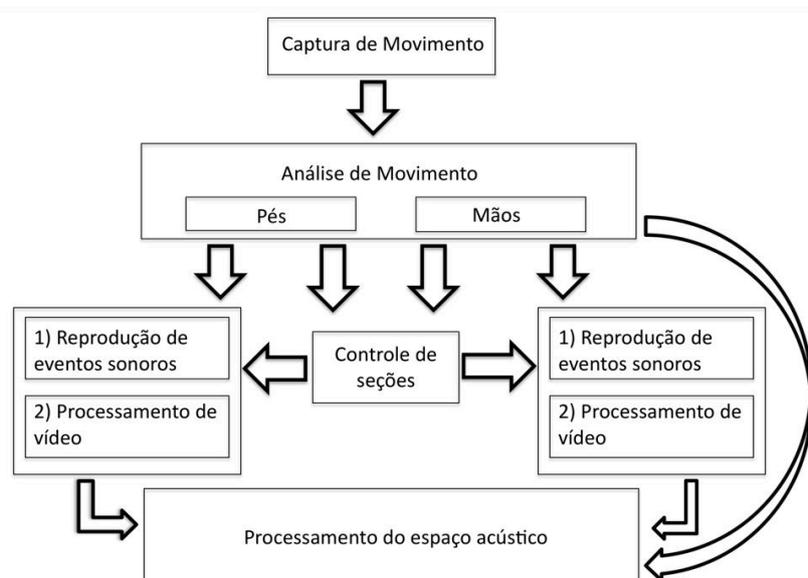
#### 4. Implementação computacional

O sistema de captura de movimento utiliza câmeras de vídeo infravermelhas que reportam coordenadas cartesianas baseadas em modelos de esqueleto humano. O processo de desenvolvimento e teste dos descritores de movimento foi implementado utilizando o sistema Vicon V460<sup>2</sup>. Por questões de aplicabilidade em situações de apresentação pública, empregamos o Kinect como dispositivo de captura de movimento. Os dados foram recolhidos durante a exibição da obra utilizando este dispositivo. Do modelo de esqueleto adotado no Kinect utilizamos as coordenadas referentes às mãos direita e esquerda, aos pés direito e esquerdo, e aos ombros direito e esquerdo. Os ombros foram considerados o centro da cinesfera dos visitantes, a partir do qual obtemos sua posição no espaço incluindo a rotação de seu corpo.

A implementação da instalação segue uma estrutura modular, organizada a partir de mapeamentos por blocos que conduzem o processamento de dados. O fluxograma apresentado na Figura 3 ilustra a estrutura geral da instalação. Exemplificando esta estrutura modular, o módulo para análise de movimento de *Caminho das águas* emprega os descritores citados na seção anterior, selecionados dentre aqueles apresentados em [Mamedes et al, 2013]. O módulo para controle de seções da instalação é derivado de um modelo inspirado em máquina de estados e apresentado em [Mamedes et al, 2011]. Os módulos restantes foram implementados especificamente para esta obra. A instalação

<sup>2</sup> Etapa da pesquisa realizada junto ao IDMIL-McGill University, sob orientação do Prof. Dr. Marcelo Wanderley, a quem agradecemos.

emprega dois computadores comunicando-se via OSC (*Open Sound Control Protocol*) em uma rede UDP (*User Datagram Protocol*). O computador principal hospeda a placa de áudio, um dispositivo Kinect para captura de movimento e uma projeção. O segundo computador, modular, hospeda um Kinect e uma projeção; esta máquina envia os dados para controle de áudio ao computador principal por rede. Devido a esta construção modular, a instalação pode ser expandida para configurações maiores ou reduzida a uma versão quadrifônica e com uma projeção.



**Figura 3. Fluxograma descrevendo a rede de processamento de dados**

A estrutura do mapeamento entre os módulos de captura de movimento e de análise de movimento é do tipo *um-para-vários* [Malloch et al. 2007], ou seja, as coordenadas cartesianas controlam vários descritores simultaneamente: velocidade, segmentação gestual por velocidade, esforço e relações angulares para os quatro membros. Já o mapeamento entre a análise de movimento e os módulos de controle de seções e processamento do espaço acústico (difusão) são do tipo *vários-para-um*; a distância e as relações angulares entre os membros superiores e o centro da cinesfera do visitante controlam a posição das fontes sonoras no espaço acústico. Por sua vez, o mapeamento entre a análise de movimento e os módulos de reprodução e processamento operam com os três tipos de mapeamento: *um-para-um*, *um-para-vários* e *vários-para-um*. Como exemplo desta variabilidade de mapeamentos, a velocidade controla apenas a intensidade dos sons contínuos; a segmentação gestual controla o início dos sons discretos assim como o início e o fim dos sons contínuos e dos sons tocados quando não há movimentação; os sons discretos têm seu início definido pela segmentação gestual, sua intensidade definida pelo descritor de esforço, sua posição no espaço pelo descritor de relações angulares e a reverberação pela distância em relação ao centro da cinesfera.

## 5. Estudo com participantes: método

O estudo com participantes avaliou a percepção de controle da obra pelos visitantes, relatada através de questionários; estes dados foram comparados às análises estatísticas realizadas sobre dados de captura de movimento e às análises estéticas realizadas sobre o resultado audiovisual gravado. Este estudo, transversal, foi aplicado uma única vez.

Contou com 31 participantes, visitantes voluntários da exposição. O diagrama apresentado na Figura 4 ilustra a utilização do espaço durante o experimento.

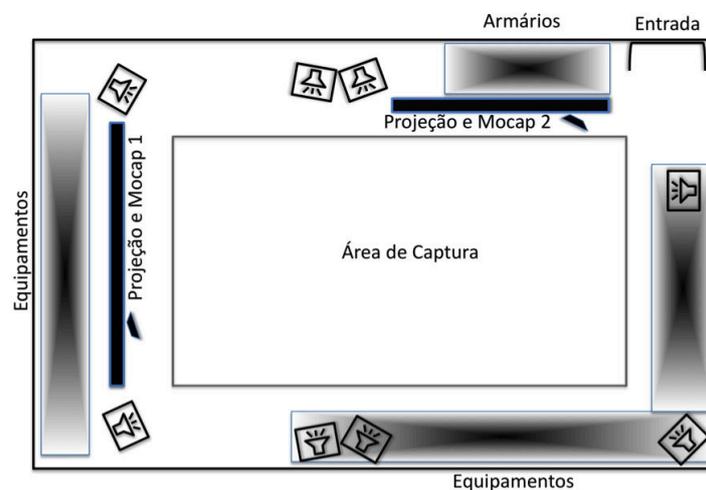


Figura 4. Croqui do espaço de experimento.

### 5.1. Análise dos Questionários

O estudo compreendeu a análise do conjunto de visitantes da instalação, com o objetivo de observar pertinências no processo interativo. As questões abaixo foram montadas de forma a induzir uma abordagem crítica sobre a experiência artística:

- 1) Comente sua apreciação sobre a instalação multimídia interativa que você acabou de experienciar. Sinta-se à vontade para dizer se gostou ou não gostou, críticas ou elogios que gostaria de compartilhar. A sua avaliação final foi negativa, neutra ou positiva?
- 2) Comente sobre sua apreciação do processo de interação com a obra. Sinta-se à vontade para dizer se percebeu que seus gestos modificavam a obra, se conseguiu relacionar algum gesto seu ao controle de alguma propriedade tanto no áudio quanto no vídeo ou se você não percebeu que estava controlando a obra. Se sua percepção de controle demorou a ocorrer, indique aproximadamente quanto tempo você levou para percebê-la.

Os participantes do estudo foram agrupados de acordo com seu nível de formação artística autodeclarado, constituindo três grupos: artistas profissionais (35% dos participantes), estudantes de artes (39%) e não-artistas ou amadores (26%). Verificamos que 86% de nossos participantes perceberam o mecanismo de controle e 14% não o perceberam ou ficaram frustrados por não estabelecer uma relação de controle. Dentre os participantes que perceberam seu controle sobre a instalação, 58% sentiram domínio sobre o processo interativo enquanto 42% não perceberam dominar a interação com o conteúdo audiovisual.

Tabela 1. Comparação entre percepção de domínio declarada e nível de formação artística

	Artistas	Estudantes	Não-Artistas
<b>Sentiram domínio</b>	30%	66%	86%
<b>Não sentiram domínio</b>	70%	34%	14%

A Tabela 1 apresenta a distribuição dos participantes que relataram domínio sobre o processo interativo de acordo com nossos três grupos. Podemos observar que a percepção de domínio foi consideravelmente menor entre os artistas, com um salto considerável para os estudantes e abrangendo grande parcela dos participantes que não são artistas. Esta

comparação levanta dúvidas interessantes sobre o tipo de uso que o participante pretendeu ao interagir com a instalação, como um nível de exigência e precisão elevado para controle do processo interativo que poderia indicar uma busca pelo controle instrumental da instalação. As demais análises não inclusas neste artigo apresentaram baixas variações entre as amostras, o que não permitiu estabelecermos correlações com os dados observados, ou apresentaram porcentagens de recorrência abaixo de nosso limite definido em 19% do total de participantes.

## 5.2. Análise estatística

O objetivo desta análise foi o de verificar possíveis pertinências entre o tempo de atividade de cada participante e as respectivas medidas de soma, média e desvio-padrão para os intervalos temporais de atividade. A medida de assimetria temporal foi obtida pelo cálculo de sua distribuição probabilística através de curtose. Abaixo apresentamos as equações empregadas para o cálculo do tempo de atividade  $A_k$  e de silêncio  $S_k$  dos participantes. A identificação da atividade do participante é definida como uma função da velocidade  $v$ , a partir de um limite  $l$  que elimina ruídos originados pelo dispositivo de captura de movimento.

$$f(v_k) = \begin{cases} 1 & \text{se } v_k \geq l \\ 0 & \text{se } v_k < l \end{cases} \quad (\text{Eq.1})$$

O silêncio  $S_k$  do participante é definido pelos momentos em que os valores de tempo entre os intervalos amostrados atendam à seguinte condição:

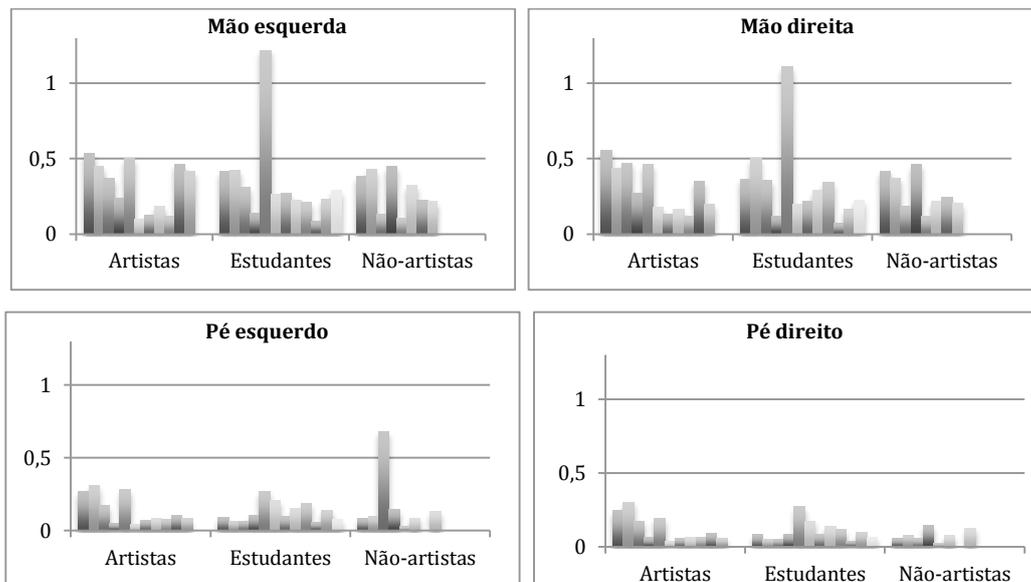
$$S_k\{f(v_k)\} = \begin{cases} T_k - T_{k-1} & \text{se } f(v_k) = 0 \\ 0 & \text{se } f(v_k) = 1 \end{cases} \quad (\text{Eq.2})$$

A atividade  $A_k$  do participante é definida pelos momentos em que os valores de tempo entre os intervalos amostrados atendam à seguinte condição:

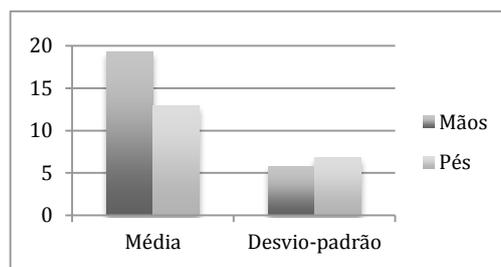
$$A_k\{f(v_k)\} = \begin{cases} T_k - T_{k-1} & \text{se } f(v_k) = 1 \\ 0 & \text{se } f(v_k) = 0 \end{cases} \quad (\text{Eq.3})$$

Os resultados são apresentados com base no dispositivo de captura de movimento associado à tela maior. Observando o gráfico 1, que apresenta os dados dos participantes agrupados pelo nível de formação artística, podemos verificar que os artistas se movimentaram muito ou pouco. Já os estudantes e os não-artistas têm uma distribuição de variação distribuída em todo o âmbito observado, à exceção do participante 14, que se movimentou acima do padrão observado em todos os grupos.

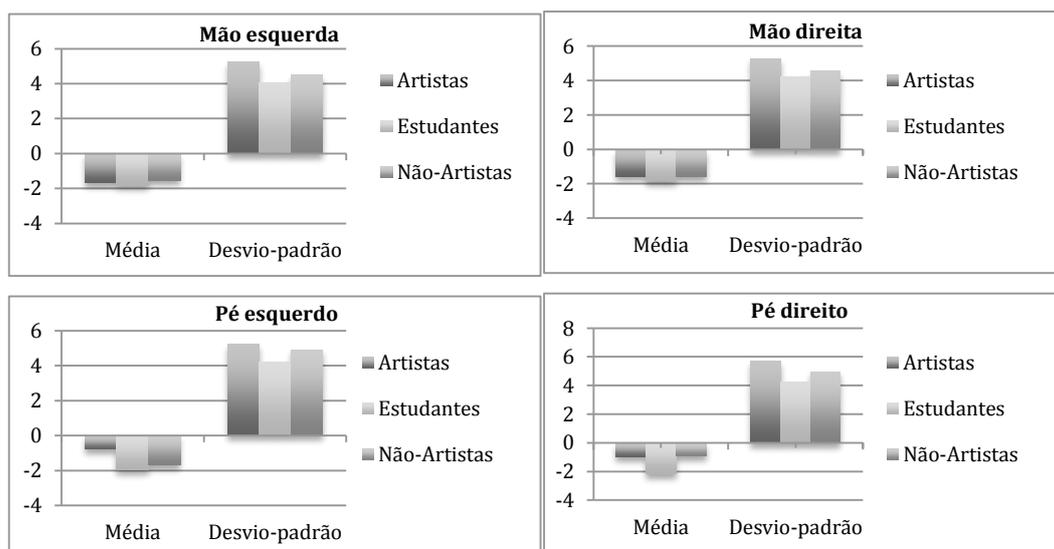
A medida de variação entre as mãos esquerda e direita foi calculada a partir da Norma L2 implementada pela equação  $N = \sqrt{\sum_k (x_k - y_k)^2}$ , onde as variáveis  $x$  e  $y$  correspondem à atividade das mãos esquerda e direita (respectivamente) de todos os participantes. A mesma equação foi utilizada para cálculo da variação entre os pés direito e esquerdo. O cálculo desta variação permite estimar a independência entre as mãos e os pés. Observamos uma maior independência para as mãos quando comparadas aos pés. Os valores de variação foram considerados baixos pelos pesquisadores. Como os dados estão normalizados em  $t = 100$ , podemos considerar os resultados apresentados no gráfico 2 como a porcentagem de variação entre os membros. Os resultados apresentam pouca variação em relação ao nível de formação artística dos participantes.



**Gráfico 1. Proporção entre atividade e silêncio dos participantes comparada pelo nível de formação artística**



**Gráfico 2. Variação na movimentação entre as duas mãos e os dois pés utilizando a Norma L2**



**Gráfico 3. Relação entre os níveis de formação artística e a média para a medida de curtose para os membros dos participantes.**

A distribuição dos dados no tempo foi analisada através de curtose, medida que descreve sua dispersão através de uma curva de distribuição de probabilidade. Esta medida nos

permite interpretar a assimetria entre os intervalos temporais observados. Observamos a pertinência da simetria entre mãos e pés, confirmando a alta similaridade observada durante a análise pela Norma L2. Os resultados médios indicam a predominância da distribuição platicúrtica, com predominância de intervalos temporais distantes da média. Esta relação é ligeiramente ressaltada para os membros superiores dos estudantes e muito próxima para os artistas e para os não-artistas. Já o desvio-padrão indica maior variação para os artistas, seguidos pelos não-artistas e pelos estudantes, respectivamente. Os resultados podem ser visualizados no gráfico 3, apresentado em escala logarítmica de base 10.

### 5.3. Análises estéticas

Nesta seção realizaremos análises estéticas sobre o processo interativo dos visitantes. Seguindo a metodologia de J. J. Nattiez [2002, p. 15], assumimos a postura de receptores e analisamos as performances dos visitantes atribuindo significações a suas experiências. Considerando que o tempo total dos experimentos foi de 3 horas e 40 minutos, selecionamos aqui três participantes considerando como critério ilustrar a variabilidade de estratégias empregadas para o controle da instalação, cada um integrando um dos grupos: artistas, estudantes e não-artistas.

O participante 3, artista profissional com experiência em música e dança, apresenta resultados caracterizados pela exploração do conteúdo sonoro e pela tentativa de construção de um discurso musical. Observamos a sequência de três padrões de exploração da instalação. Primeiro, o participante explora movimentos largos e sequenciados, com pouco intervalo entre cada gesto. Após tocar o som de perfil ataque-ressonância que marca o início da terceira seção da obra, o participante muda sua estratégia de interação e passa a construir estruturas que podem ser associadas a frases musicais, explorando a intensidade sonora através de variações na velocidade de seus gestos. Na parte final do experimento, o participante retoma o padrão exploratório inicial.

O participante 21, estudante com experiência em música, apresenta resultados caracterizados por uma interação negativa. O participante reportou incômodo com “uma batida grave [...] aos poucos me mexia tentando não acionar este som”. A estratégia adotada por este participante estabelece como premissa de controle negar materiais sonoros indesejados. Esta opção contraria nosso projeto para o modelo interativo, baseado no reconhecimento de ações gestuais. O processo interativo inicial do participante consiste na exploração da instalação até o início da terceira seção da instalação, quando é tocado o evento sonoro descrito, marca de seu início. Podemos observar uma súbita interrupção no disparo de novos eventos sonoros e da exploração do ambiente. Este novo padrão de movimentação perdurará até o final do experimento, onde o participante move-se pouco e realiza gestos com largos intervalos temporais.

O participante 25, não-artista, tentou controlar a interação a partir de gestos baseados no controle de interfaces de toque, movendo apenas as mãos. O participante se deslocou pelo espaço da instalação apenas para melhor visualizar a projeção. Desta forma, observamos uma forte dependência (norma L2) entre os membros inferiores devido aos pés parados simultaneamente. O participante também apresentou baixos valores de velocidade máxima, o que resultou em uma performance de baixa intensidade sonora. Observamos que suas mãos acompanhavam o processamento de vídeo por convolução, processo que transforma a imagem em uma textura caracterizada por faixas verticais e horizontais. O

participante conseguiu alcançar o ponto de equilíbrio onde a velocidade de seus movimentos e o deslocamento da textura visual estavam em sincronia.

## 6. Conclusões

O conjunto dos resultados obtidos com este experimento nos indica uma correlação geral, segundo a qual a percepção de controle e a variabilidade de possibilidades foi influenciada pelo nível de formação artística. Os resultados de análises sobre os questionários apontaram que os participantes com maior nível de formação artística perceberam com mais facilidade o controle sobre a instalação, enquanto aqueles com menor nível de formação artística demonstraram maior satisfação, relatando a sensação de domínio sobre o processo interativo. As análises estatísticas indicaram uma maior regularidade temporal para os movimentos dos participantes artistas, seguidos pelos não-artistas e estudantes, respectivamente. Já a distribuição da relação entre atividade e silêncio foi caracterizada por dois grupos bem claros para os artistas: pouca ou muita movimentação; esta relação apresentou uma distribuição mais equilibrada para estudantes e não-artistas. As questões que os resultados obtidos levantam são: a formação artística do visitante aumenta a exigência de controle sobre a obra? Seria este controle inspirado em uma abordagem instrumental, criando uma expectativa por mapeamentos mais complexos? Estudantes estariam dispostos a explorar mais padrões diferentes que artistas e não-artistas? Estas questões apontam a necessidade de pesquisas sobre a convergência do processo interativo em instalações multimídia com estudos cognitivos, observando com maior profundidade a relação entre apreciação artística e controle interativo. Por fim, consideramos positivos os resultados alcançados pelo modelo interativo empregado na instalação, que aplicou parte dos conceitos desenvolvidos durante a pesquisa ao processo criativo de *Caminho das águas*.

## Agradecimentos

Os autores agradecem ao apoio financeiro da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), processos nº 2011/01553-8 e 2012/21039-0, ao Prof. Dr. Tiago Tavares (FEEC-Unicamp) pela implementação do programa que grava os dados obtidos pelos descritores de movimento, e ao Prof. Dr. Marcelo Wanderley que nos auxiliou a desenvolver estes descritores junto ao IDMIL-McGill University.

## Referências

- Bishop, C. (2005) *Installation Art: A Critical History*. Routledge.
- Camurri, A., Trocca, R. (2000) "Movement and gesture in intelligent interactive music systems". Em: *Trends in Gestural Control of Music*, Editado por M. M. Wanderley e M.M. e M. Battier. IRCAM - Centre Pompidou. Paris.
- Camurri, A., Volpe, G., Poli, G.D., Leman, M. (2005) "Communicating expressiveness and affect in multimodal interactive systems". Em: *IEEE Multimedia*. Vol.12-1, pp. 43-53.
- Coulter-Smith, G. (2006) *Deconstructing Installation Art*. CASIAD, Southampton.
- Dobrian, C., Bevilacqua, F. (2003) "Gestural control of music: using the vicon 8 motion capture system". Em: *Proceedings of the 2003 Conference on New Interfaces for Musical Expression*. pp. 161-163.

- Garcia, D. H. L.; Campos Jr., J. I. (2003) "Projeto e implementação sonora em *Op\_era*, um ambiente imersivo-interativo desenhado em sistemas de realidade virtual. In: Anais do IX SBCM Brazilian Symposium on Computer Music, pp.187-193.
- Hornecker, E. Nicol, E. (2011) "Towards the Wild: Evaluating museum installations in semi-realistic situations. Re-Thinking Technology in Museums, Limerick, Ireland.
- Horneck, E. Stifter, M. (2006) "Learning from interactive museum installations about interaction design for public settings". Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments, pp. 135-142.
- Kwastek, K. (2013) *Aesthetics of Interaction in Digital Art*. MIT Press.
- Malloch, J., Sinclair, S., Wanderley, M. M. (2007) "From controller to sound: tools for collaborative development of digital musical instruments". In: Proceedings of the International Computer Music Conference. pp. 65-72.
- Mamedes, C. R., Fornari, J., Manzolli, J., Garcia, D. H. L. (2011) *Abstrações: An audiovisual installation based on motion recognizing*. In: 4th International Congress on Image and Signal Processing (CISP), 2011. vol. 2, pp. 586-589. Shanghai.
- \_\_\_\_\_ (2012a) *Abstrações: apontamentos sobre o comportamento dos visitantes*. Revista do Encontro Internacional de Música e Arte Sonora. Juiz de Fora.
- \_\_\_\_\_ (2012b) *Cerejeira: delimitadores ambientais para controle de diversidade utilizando algoritmos genéticos interativos*. In: Anais do XXII Congresso da Associação Nacional de Pesquisa e Pós-Graduação em Música. pp. 683-690. João Pessoa.
- Mamedes, C. R. Wanderley, M. M. Manzolli, J. Garcia, D. H. L. (2013) *Strategies for Mapping Control in Interactive Audiovisual Installations*. Proceedings of the 10th International Symposium on Computer Music Multidisciplinary Research, pp. 766-778. Marseille.
- Moroni, A. M. F. S., Guimaraes, E., Paula, G., Manzolli, J. (2009) "AURAL: An Evolutionary Interface for a Robotic Sonification Process" Em: *Intelligent Computer Graphics*, vol.2. Springer.
- Nattiez, J. J. (2002) "O modelo tripartite de semiologia musical: o exemplo de La Cathédrale Engloutie, de Debussy". Em: *Revista Debates nº6: Cadernos do Programa de Pós-graduação em Música*. Unirio. Rio de Janeiro, pp. 7-39.
- Reiss, J. H. (1999) *From Margin to Center: the spaces of installation art*. MIT Press. Cambridge.
- Smalley, D. (1986) "Spectro-morphology and structuring processes". Em: *The language of electroacoustic music*, Edited by Simon Emmerson, The Macmillan Press, Londres.
- Stallabrass, J. (2004) *Art Incorporated: The Story of Contemporary Art*. Oxford University Press. New York.
- Wassermann, K. C., Eng, K., Verschure, P. F. M. J., Manzolli, J. (2003) "Live soundscape composition based on synthetic emotions". *IEEE Multimedia* 10(4), pp. 82-90.

# SoMo: Um Instrumento Musical baseado em Movimento – Características Estéticas e Técnicas e potenciais Usos

Tiago Brizolara da Rosa<sup>1</sup>, Marcos Goularte Moritz

<sup>1</sup>Departamento de Música – Universidade Estadual de Maringá (UEM)

tiago.brizolara@gmail.com, marcosmoritz@gmail.com

**Abstract.** *This paper describes the conception, development and current characteristics of a movement-based musical instrument developed using the free open-source tools Pure Data and Processing. The instrument can be assembled using an usual personal computer and even with a low cost video camera (we use PSEye®) we attained a responsiveness of up to 100 frames analysed per second and real-time audio generated. The system was tested in Linux Ubuntu and Windows 7 operating systems. Along with its low cost, the project is open-source, turning it into more than a musical instrument, a potential tool for education and experimentation.*

**Resumo.** *Este artigo descreve a concepção, desenvolvimento e atuais características de um instrumento musical baseado em movimento desenvolvido utilizando as ferramentas livres de código aberto Pure Data e Processing. O instrumento pode ser montado usando um computador pessoal comum e mesmo com uma câmera de vídeo de baixo custo (nós usamos a PSEye®) obtivemos uma responsividade de até 100 quadros analisados por segundo e áudio gerado em tempo real. O arranjo foi testado nos sistemas operacionais Linux Ubuntu e Windows 7. Além de seu baixo custo, o projeto é de código-fonte aberto, tornando-o mais que um instrumento musical, uma potencial ferramenta educacional e experimental.*

## 1. Introdução

Este artigo descreve as características técnicas e estéticas do estado atual de um instrumento musical digital baseado em movimento, desenvolvido de forma independente, com código aberto hospedado em <https://bitbucket.org/mmoritz/somo>. Iniciamos apresentando como nossa concepção dialoga com o conceito de instrumento musical.

### 1.1 Do conceito de Instrumento Musical

O conceito de instrumento musical vem há mais de cem anos sendo revisitado. Em 1901 acontecia a primeira audição pública do Telharmonium, instrumento eletromecânico capaz de mimetizar diversos instrumentos tradicionais, produzir sons jamais ouvidos e de transmitir sua música ao vivo através de linha telefônica - e também de criar oscilações na rede elétrica e interferências nas conversas telefônicas (Brend, 2012). Logo no começo do século XX, o movimento futurista aborda recursos eletrônicos e de outras tecnologias como parte de sua estética, trazendo como instrumentos musicais materiais que até então jamais seriam assim denominados - como turbinas de avião e buzinas. Em meados do séc. XX temos na França a *Musique Concrète*, concebida por

Pierre Schaeffer, com a teorização de objeto sonoro passando pela experiência que o compositor batiza de *acusmática*, onde importa somente o som e não suas fontes. A música concreta é elaborada manipulando a própria fita magnética na qual há sons gravados. Na Alemanha, a *Elektronische Musik* se propõe a produzir música por meio exclusivamente eletrônico<sup>1</sup>, explorando sons até então não disponíveis aos músicos - ao menos não de forma controlada.

No seu *Traité des Objets Musicaux* (Schaeffer 1961, p. 40), Schaeffer, pai da Música Concreta, coloca que “todo dispositivo que nos permite obter uma variada coleção de objetos sonoros ... mantendo no seu cerne a permanência de uma causa é um instrumento musical, no sentido tradicional de uma experiência comum a todas as civilizações” (tradução livre). Ruviaro (2012) propõe uma extensão da definição de Schaeffer para trabalhar o conceito nas *Lorks (Laptop Orchestras)*, onde comumente há uma fricção entre a expectativa de uma experiência não-acusmática (os músicos estão no palco “executando” seus laptops) e a experiência de fato onde à plateia não se apresenta com clareza uma relação entre os sons e a presença dos instrumentistas com seus laptops e falantes (esse tipo de experiência com música computacional é precisamente a motivação do *survey* conduzido por Magnusson and Hurtado (2008) com praticantes de instrumentos digitais e acústicos). Para Ruviaro, os três conceitos a seguir devem ser levados em conta ao pensarmos em instrumento musical:

- Presença: a aparência física do instrumento e como o instrumentista se posiciona em relação a ele;
- Movimento: gestos e movimentos necessários para executar o instrumento; gestos e movimentos que são idiomáticos mesmo sem gerar som; uma máxima relativa aos instrumentos musicais tradicionais: “quando o movimento pára, o som pára ou começa a extinguir-se”;
- História: um instrumento faz parte de uma ou mais culturas. Há contextos onde ele costuma estar, repertório típico e forma típica de execução para cada contexto.

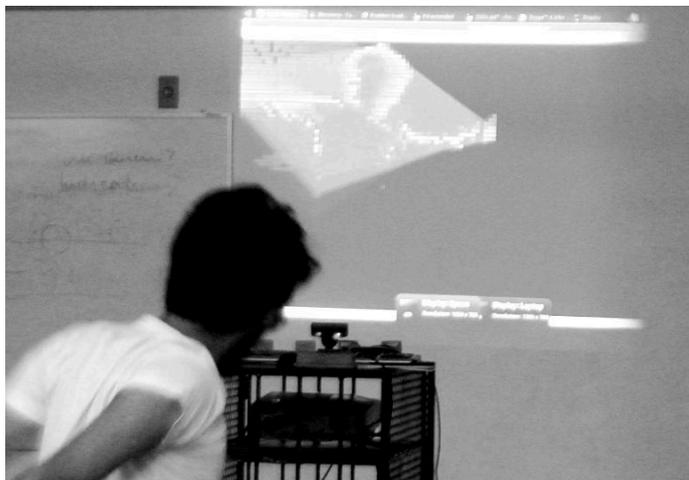
## 1.2 Concepção Inicial do SoMo

A ideia básica do SoMo (das palavras som e movimento, e recordando *soma* - corpo, em grego) é a de um instrumento cuja interface é o vídeo, capturando o executante. Foi tentadora a ideia de utilizar o Microsoft Kinect®: com ele poderíamos mapear as 20 *joints* fornecidas pelo sensor (cabeça, mãos, quadris, pés, etc) a parâmetros de um sintetizador. Não enveredamos por esse caminho por causa da baixa taxa de dados que esse sistema fornece: 30 pacotes de dados por segundo (ou quadros por segundo, abreviados doravante por FPS - *frames per second*), o que só permitiria fineza na dimensão temporal com movimentos lentos. Segundo Menezes (2004), seções 3.4.4, 3.4.5], podemos dizer que o limiar de reconhecimento dos sons por parte do ouvido é da ordem de 50 ms - para o que 30 FPS bastariam -, mas objetos sonoros de mesma proveniência instrumental possuindo variação de altura entre si podem ser distinguidos com apenas 15 ou 20 ms - muito menos que a duração de um quadro a 30 FPS, que é de

---

<sup>1</sup> O que não excluiu a criação de obras mistas por parte desses compositores. Por exemplo, a peça *Mixtur*, de Karlheinz Stockhausen, 1964, foi composta para orquestra, três osciladores senoidais e 2 moduladores em anel.

33,3 ms. Assim, consideramos 30 FPS uma taxa desconfortável por estar ao redor de limitações do ouvido. Além disso, na seção 3 falamos sobre como as altas taxas de FPS conseguidas na captura e processamento do vídeo foram vitais não só para o áudio, mas para a experiência de imersão do executante com o uso do *feedback* visual (a decisão pelo *feedback* foi feita durante o desenvolvimento e é abordada nas seções 2.1.5 e 3.3).



**Figura 1. SoMo em uso. A projeção representa as alterações no vídeo em relação ao *frame* anterior e serve como *feedback* para o executante. Ao centro, sobre a mesa, encontra-se a câmera PSEye®. Os alto-falantes não aparecem na imagem.**

Ao eleger o vídeo como interface há as desvantagens de não termos a abstração das partes do corpo do executante, mas pudemos nos valer de outras possibilidades abertas: podemos usar objetos; nos afastar ou aproximar da câmera à vontade (enquanto o Kinect tem uma cobertura pequena de distâncias possíveis da pessoa ao sensor); ou mostrar qualquer porção do corpo no vídeo ao invés do corpo todo.

Podemos dizer que as quatro seguintes decisões estéticas - não totalmente independentes - foram as chaves para guiar a construção do instrumento:

1. *O instrumento não deve ser um simples sonificador de dados*, como os instrumentos tradicionais também não são. Até mesmo quando tocados de modo alheio ao fazer artístico, como por robôs ou quando um instrumento é excitado acidentalmente (numa queda, ao ser atingido por algo...), percebemos que podem gerar uma gama muito grande e interessante de sons. Dizer que um piano é um “sonificador de dedos” pode até ter rigor de um ponto de vista estritamente mecânico, mas faz pouco sentido ao falarmos de música;
2. *O som deve ser produzido pelo movimento* (mudanças no vídeo) e não somente pela presença (estado instantâneo do vídeo). Essa decisão encontra eco na característica de movimento dos instrumentos musicais levantada na seção 1.1. Uma leve diferença é que no SoMo, quando não há movimentos no vídeo ainda resta um som em baixa intensidade;
3. *O som deve ser sintetizado* ao invés de serem trabalhadas sonoridades tradicionalmente consideradas musicais. Porque almejamos uma experiência onde o executante sente estar extraíndo material sonoro do seu movimento e não

realizando uma experiência de manipulação/transformação de certos sons ao seu movimento ou efetuando um *DJing*<sup>2</sup>;

4. Esta decisão está aparentada à primeira. *A experiência de tocar o instrumento não deve se esgotar facilmente*, deve existir uma profundidade. Não há problema que o instrumento não seja fácil de dominar inicialmente - ninguém sai fazendo musica num violoncelo na primeira tentativa.

Os detalhes da descrição do som do SoMo estão na seção 2.2.

## 2. Materiais e Métodos

O arranjo da plataforma atual de testes é: câmera Playstation PSEye® (o preço de uma nova varia entre R\$ 60,00 e R\$ 100,00) no modo de 100 FPS; um notebook com processador Intel Core i3 2.20 GHz, 3 GB de RAM e sistema operacional Linux - tivemos dificuldades em fazer a câmera PSEye funcionar com o Java no Windows - ; uma placa de áudio PreSonus AudioBox 22VSL. Os softwares utilizados são todos de código-fonte aberto. Para a captura, processamento e geração de vídeo fizemos um programa em Java utilizando as bibliotecas do Processing 2.0<sup>3</sup>, com o ambiente de desenvolvimento integrado Eclipse. A cada quadro do vídeo passamos os dados da sua análise para um *patch*<sup>4</sup> da plataforma de programação de fluxo de dados Pure Data Extended<sup>5</sup> (Pd, em abreviatura).

### 2.1 Processamento do vídeo

O processamento do vídeo tem dois objetivos: 1) detectar movimentos, analisá-los e traduzir essa análise na forma de quantidades geométricas (doravante denominadas *descritores*) a serem enviados como parâmetros para o sintetizador de áudio; 2) fornecer *feedback* para o instrumentista acerca de seu próprio movimento e trajetória.

Ambas as abordagens devem ser otimizadas, visto que busca-se a análise na mesma taxa de atualização da câmera, que pode alcançar 100 FPS. As principais etapas lógicas estão resumidas na Tabela 1.

**Tabela 1. Algoritmos de processamento de vídeo**

Algoritmo	Objetivo
Análise dos Setores	Detectar movimento; Atenuar menores variações, inclusive ruído
Clusterização (opcional)	Reconhecer agrupamentos isolados de setores ativos (chamados de <i>clusters</i> ).
Envoltória Convexa	Definir um polígono que englobe os setores de um <i>cluster</i> e criar um objeto que definiremos como <i>corpo</i> com função de complementar o feedback visual e disponibilizar propriedades geométricas associadas a um <i>cluster</i> .

<sup>2</sup> O trabalho do DJ (disc-jóquei).

<sup>3</sup> <http://processing.org>

<sup>4</sup> Os programas feitos com o Pure Data costumam ser denominados *patches*.

<sup>5</sup> <https://puredata.info/downloads/pd-extended>

### 2.1.1 O Setor de Análise

Ambos os requisitos - detecção de movimento e geração de *feedback* - são alcançados através da implementação de uma abstração que chamamos de **setor de análise** de vídeo e implementamos através da classe *AnalysisSector*, dividindo a tela em uma grade de setores retangulares (Figura 2). Cada setor é uma instância desta classe, contendo atributos que definem:

- A **posição** do setor;
- O **tamanho** em pixels na horizontal e na vertical;
- A **amostra** da análise: quantidade de pixels a serem analisados no setor (definimos o padrão como  $\frac{1}{4}$  dos pixels abarcados pelo setor), como esquematizado na Figura 3;
- A **cor extraída** do setor.

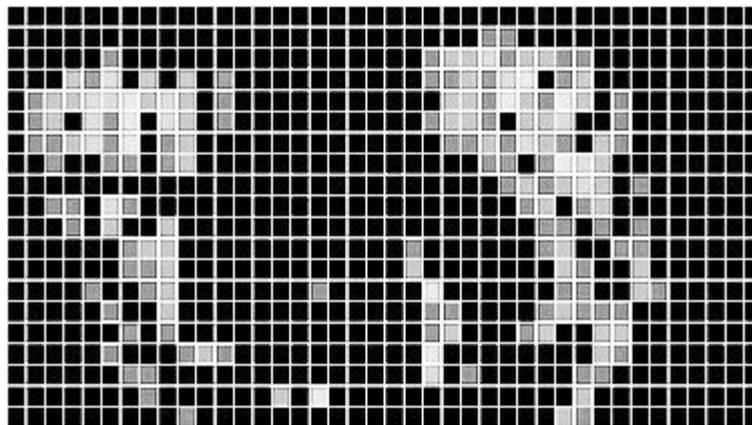


Figura 2. Os setores não pretos representam porções da imagem onde houve movimento (as linhas brancas são apenas para fins de ressaltar os setores nesta imagem e o polígono representando a envoltoria convexa está omitido). A estratégia de divisão em setores minimiza o efeito de ruídos e economiza processamento.

A **cor extraída** de um setor é a média das cores dos pixels da amostra do setor. Essa média ajuda a suavizar variações de um *frame* para outro, evitando falsos positivos advindos de ruído e movimentações sutis no vídeo. Se de um *frame* para o próximo essa média varia além de uma tolerância previamente fixada, determinamos que houve movimento naquele setor e ele passa a estar no estado **ativo**.

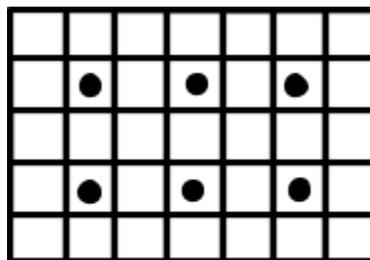


Figura 3. Representação esquemática de um setor de análise de 7 x 5 pixels. Apenas os pixels indicados com um círculo terão suas cores contabilizadas na média que confere a cor extraída do setor.

O fato da amostra ser menor que a quantidade de pixels abarcados pelo setor (ao redor de  $\frac{1}{4}$ ) confere economia de processamento.

Na configuração atual a grade de setores tem 80 colunas e 48 linhas.

### 2.1.2 Corpos

Utilizando os setores ativos mais externos construímos a envoltória convexa (o conjunto de todas as combinações convexas de envoltórias formadas pelos centros dos setores ativos), que chamaremos simplesmente de **corpo** (Figura 4). A representação do corpo como envoltória convexa favorece a extração de propriedades geométricas, faz parte da estratégia - em fase de avaliação - de identificar vários corpos e é utilizada no *feedback* visual como um polígono com transparência, como podemos ver nas Figuras 1, 4 e 5.

### 2.1.3 Separação em Clusters (opcional)

Para que pudéssemos identificar movimento de partes isoladas da imagem, i.e., vários corpos, elaboramos um algoritmo capaz de interpretar se a nuvem dos setores ativos pertence ou não à uma única fonte de movimento, na forma de uma versão do algoritmo de classificação *K-Means* que, dado um número de agrupamentos (*clusters*), associa cada setor a um *cluster*. Neste algoritmo, um número  $k$  de centros  $c_i$ ,  $i \in [1, k]$  são posicionados arbitrariamente e cada setor é classificado como pertencente ao *cluster* cujo centro está mais próximo de sua posição. Os centros são recalculados a cada iteração pela média da posição do setores associados à cada *cluster*. O algoritmo converge quando nenhum setor muda de *cluster* em duas iterações consecutivas.

O resultado deste algoritmo depende da posição inicial dos centros, que é aleatória, e o algoritmo precisa rodar diversas vezes por *frame* para garantir um resultado confiável.

Cada *cluster* será matéria prima para construção de um corpo. A Figura 4 mostra um exemplo com reconhecimento de 2 clusters. No entanto, na configuração padrão do SoMo, não há clusterização - consideramos sempre que há um único corpo.

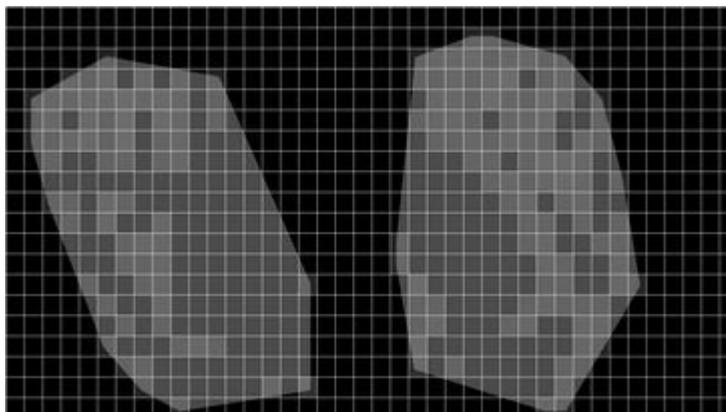


Figura 4. Grupos de setores são classificados como pertencentes ao *cluster* cujo centro se localiza mais próximo dele. Os pontos em cinza claro pertencem ao *cluster* 1, os em cinza escuro ao *cluster* 2. Os demais pontos são setores inativos. Nas configurações padrão do SoMo essa clusterização não é utilizada - definimos apenas um corpo.

### 2.1.4. Análise do movimento

A cada *frame* são extraídos os seguintes descritores do *corpo* (ver também Figura 7):

- n. Quantidade de setores ativos;
- c. Coordenada horizontal do centróide dos setores ativos;
- w. Cobertura horizontal compreendida pelos setores modificados no último *frame*;
- $\check{y}1$ . Distância vertical do limite superior do corpo ao centro da tela;
- $\check{y}2$ . Distância vertical do limite inferior do corpo ao centro da tela;

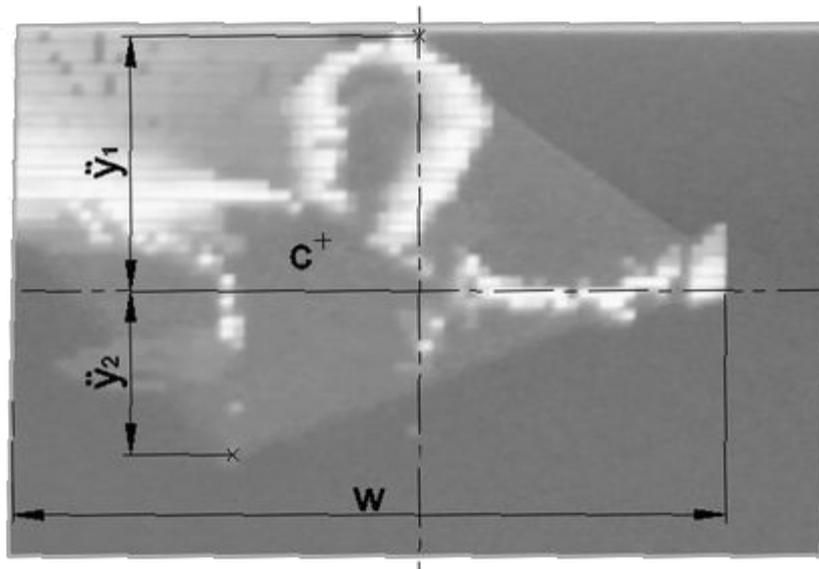


Figura 5. O vídeo de *feedback* consiste dos setores que acusaram modificação do *frame* anterior para o atual, mais um polígono representando a envoltória convexa do *corpo* definido por esses setores. Dessa envoltória são extraídos os seguinte descritores, para serem mapeados depois nos parâmetros do sintetizador: cobertura horizontal  $w$ , centróide  $c$ , extremos verticais  $\check{y}1$  e  $\check{y}2$  e quantidade de centróides modificados  $n$  (este não representado na imagem).

### 2.1.5. Feedback

O vídeo de *feedback* fica voltado para o executante e é exibido a 100 FPS. Como mostrado nas Figuras 1, 2, 4, 5, 6, apenas os setores ativos são exibidos no vídeo, sob fundo preto. Os setores que estavam ativos nos quadros anteriores também aparecem, com um grau de transparência proporcional a seu tempo de existência, gerando uma clara representação da trajetória dos movimentos (Figura 6).

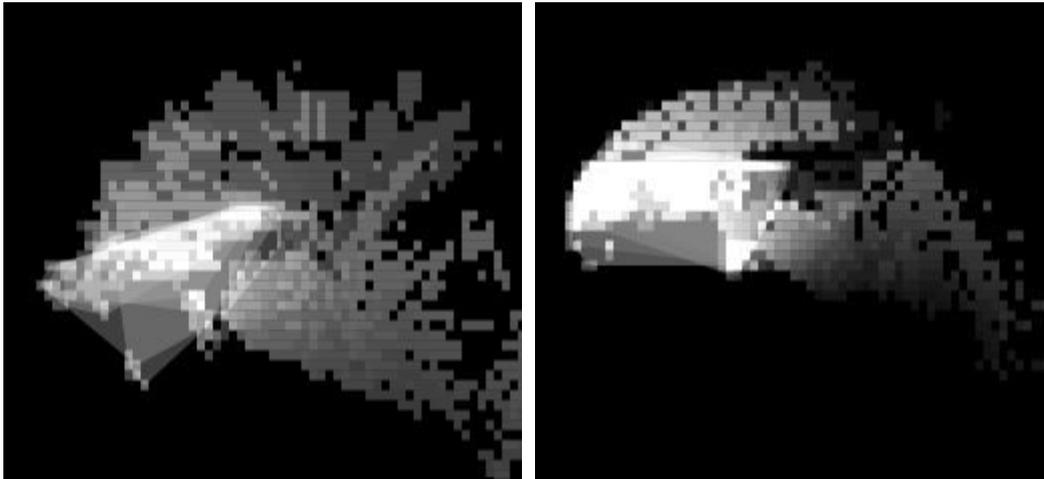


Figura 6. O *feedback* de vídeo inclui setores ativos de quadros anteriores, conferindo a noção de trajetória. Na figura temos um movimento rápido de braço. Esquerda: *feedback* de vídeo gerado a 60 FPS. Direita: *feedback* de vídeo a 100 FPS. Em ambos os casos a captura de vídeo está a 100 FPS.

A *envoltória convexa* do eventual *corpo* detectado é exibida como um polígono vermelho com transparência. Portanto, sua presença indica que o movimento está influenciando o áudio (omitimos sua presença na Figura 2).

## 2.2. Geração do Áudio

O áudio é gerado por um sintetizador desenvolvido na forma de um *patch* de *Pure Data*. Os valores dos descritores extraídos do vídeo são enviados continuamente do aplicativo Java para o *patch* via protocolo OSC (*Open Sound Control*) e cada um deles passa por uma função de calibração e é então roteado para um parâmetro diferente do sintetizador.

Indo ao encontro das decisões estéticas apresentadas na Introdução, a sonoridade buscada trabalha com os limites da sensação de altura definida, em um *patch* consistindo em:

- *OsSo* ("oscilador solitário"). Um oscilador senoidal que muda de frequência dezenas de vezes por segundo, sempre para uma frequência aleatoriamente escolhida dentro de certo intervalo. Seus parâmetros:
  - Amplitude  $A$ ;
  - Frequência central  $F$ ;
  - Intervalo ou janela de frequências  $W$ ;
  - Período de mudança de frequência  $T$
- *Distorter* (opcional). Um nível de valor  $D$  adicionado ao sinal final, no sentido de saturar o sinal, distorcendo o áudio;
- *Uivo* (opcional). Um oscilador senoidal com frequência igual a  $F/2$  (metade da frequência central do *OsSo*).

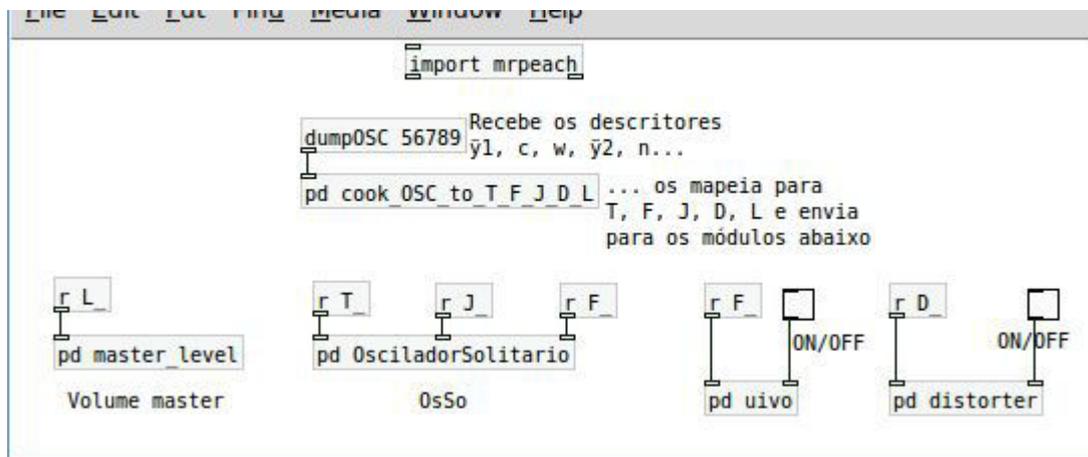
O som é espacializado em estéreo de forma coerente com a posição horizontal do centroide  $c$ . Quando não há movimentos no vídeo ainda resta um som em baixa intensidade.

Os parâmetros estão mapeados como indicado na Tabela 2:

**Tabela 2. Mapeamento dos descritores do vídeo para os parâmetros do sintetizador (width e height são as dimensões do vídeo)**

Parâmetro do sintetizador	Intervalo	Descritor do vídeo	Intervalo
Amplitude $A$	{0,1.2}	$n$ . Quantidade de setores ativos no último <i>frame</i>	{0, 80x40} <sup>6</sup>
Frequência central $F$	{100, 460} Hz	$c$ . Coordenada horizontal do centróide do corpo	{0, width}
Janela de frequências $J$	{2, 460} Hz	$w$ . Cobertura horizontal compreendida pelos setores ativos no último frame	{2, width}
Período de mudança de frequência $T$	{10, 20} ms	$y1$ . Distância vertical do último setor ativo mais superior ao centro da tela.	{height/2., 0}
Nível do Distorter $D$	{0, 1}	$y2$ . Distância vertical do setor ativo mais inferior ao centro da tela.	{height/2., height}

Os baixos valores de  $T$  fazem com que mesmo com apenas um oscilador (caso do *uivo* desligado), grandes quantidades de frequências sonoras estejam presentes em curtos intervalos de tempo. Podemos pensar no OsSo como uma espécie de síntese de frequência modulada (FM) com a modulação sendo dada por um sinal aleatório com características dadas pela frequência central  $F$  e janela de frequências  $J$  - não é exatamente o caso porque no OsSo a senóide não muda de frequência antes de completar um ou mais ciclos.



**Figura 7. Patch do sintetizador, mostrando o mapeamento dos descritores do vídeo para os parâmetros do sintetizador, constituído pelos sub-patches [master\_level], [OsciladorSolitario], [uivo], [distorter].**

Ao instrumentista diminuir a janela de frequências  $J$  através da ocupação de menos espaço horizontal na tela, obterá um som parecido com um vibrato errático em torno de uma nota, e pode ser trabalhoso manter uma pequena porção do vídeo ativa sem deslocar-se horizontalmente. Ao aumentar  $J$ , o som ganha características de um borbulhar, agregando um pouco de ruído quando a janela é demasiadamente aberta, o que pode ser amenizado aumentando  $T$  se o instrumentista atingir porções superiores do vídeo. O *uivo* ligado fornece a possibilidade de construir linhas melódicas sem o esmero

<sup>6</sup> 80x40 são as dimensões da grade de setores.

de manter uma pequena janela  $J$  - além, de obviamente, mudar a qualidade (timbre) do som. O *Distorter*, mais pronunciado quando o instrumentista alcança porções inferiores do vídeo, é um recurso timbrístico.

### 3. Resultados

#### 3.1. Implementação

Do lado técnico, conseguimos com o arranjo de baixo custo descrito na seção 2.1. utilizar a câmera PSEye a 100 FPS (não conseguimos acessar os modos de captura de 125 FPS); executar o aplicativo Java também a 100 FPS - isto é, depois de bastante otimização conseguimos fazer a análise completa de cada *frame* antes de receber o próximo - e atualizar o sintetizador no Pd de acordo com os parâmetros em tempo real. Testes sem uma placa de áudio dedicada também não mostraram prejuízo da experiência.

#### 3.2. Uso

O parâmetro de período de mudança de frequência  $T$ , mapeado pelo descritor de distância vertical superior  $y/l$ , pouco influenciou na experiência. E se aumentarmos o intervalo de  $T$ 's possíveis, a sonoridade muda demasiadamente - o ouvido passa a tentar acompanhar cada variação de frequência -, de um modo que nos parece perturbar a coerência da experiência. Portanto, consideramos o parâmetro  $T$  e seu mapeamento como não acabados.

Algumas características que ressaltamos dos testes preliminares:

*Imersão* - Nos testes preliminares, executantes advindos de experiências distintas - música, artes visuais, artes cênicas, ciências exatas, entre outras - avaliaram o uso do SoMo como de grande grau de imersão e estimulador da exploração.

*Funcionamento* - os princípios do funcionamento do instrumento não foram óbvios aos participantes, embora facilmente explicáveis aos músicos. Aos não músicos bastou uma explicação como: “imagine que você tem um monte de macaquinhos aos seus dois lados, espalhados sobre as teclas de um piano e tocando-as freneticamente. Quanto mais espaço você ocupar na horizontal do vídeo, mais espalhados no piano estarão os macaquinhos, gerando sons então por uma grande faixa desde graves até agudos. Quanto menos espaço horizontal você ocupa, mais os macaquinhos estarão junto a você, tocando, assim, um pequeno intervalo de teclas. Já seus movimentos na vertical mudam a sonoridade do instrumento”.

*Percussão* - O caráter imediato da resposta sonora e visual do sistema possibilitou o uso do instrumento como percussão. Mesmo questões de variação de sonoridade como as relatadas por Silva et al (2013) como difíceis de conseguir com interfaces digitais, podem ser alcançadas com variação de origem e sentido do gesto, forma do objeto móvel (mão aberta ou fechada, braço, objeto, proximidade com a câmera, etc), presença ou não de *Clipper* e *Uivo* e ainda a presença de mais partes móveis, para abertura da janela de frequência  $W$ . Tudo isso (exceto a presença/ausência do *Distorter* e *Uivo*) ainda pode ser variado durante o gesto. Novamente, acreditamos

que as estratégias que adotamos tiveram sucesso em conseguir a *experiência com profundidade* que foi norteadora do projeto.

*Formas de uso* - Até o momento, diversas formas de uso foram observadas. Em uma delas, um instrumentista se posiciona num extremo lateral e o outro, no extremo oposto e tocam o instrumento alternadamente quando querem uma janela pequena e juntos quando querem uma janela (bastante) grande<sup>7</sup>. Em outra, um instrumentista *toca* tríades alternando movimentos de uma mão, cabeça e a outra mão. Com *Uivo* e *Distorter* ligados, o abrir e fechar de uma mão próxima à câmera, deslocada à esquerda, soa como um coração pulsante.

*Gestuais típicos* - Gestos típicos foram o mover rápido de dedos para obter corpos bem localizados e o abaixar de um braço ou os dois ou o corpo todo com braços abertos, gerando sonoridades tipicamente encorpadas e delineadas no tempo.

### 3.3. Feedback visual, imersão

Se nos remetermos ao Theremin, um dos instrumentos eletrônicos mais antigos e de maior sucesso e longevidade, podemos vir a pensar que num instrumento todo *feedback* (exceto o sonoro, claro) pode ser dispensado.

No entanto, Magnusson e Hurtado (2008) mostram que a ausência de *feedback* tátil de vários instrumentos digitais é muito sentida. No nosso caso, verificamos que a implementação do *feedback* de vídeo, inicialmente apenas para análise, mudou radicalmente a experiência. Isso se deve em grande parte pelo instrumento não detectar movimentos mínimos, em razão de sua resolução e de um limiar programado. Tendo o *feedback*, ao começar a se movimentar, o instrumentista percebe o quanto e que partes de seu corpo estão se movendo, e quando o movimento atinge o limiar, a *envoltória convexa* do *corpo* gerado é exibida, deixando claro que o movimento passa a gerar e modificar o som.

O fato de nem todo movimento influenciar o instrumento propôs naturalmente aos instrumentistas uma estratégia de mover-se sorratamente até preparar-se para executar determinados movimentos. Consideramos que esse tipo de uso, bastante constante, foi um fator contribuinte no estabelecimento de períodos de grande imersão dos instrumentistas.

Como o *feedback* visual exhibe os rastros do movimento, avaliamos que para não prejudicar essa imersão, deveríamos ter grandes FPSs de vídeo exibido, o que nos levou a chegar aos 100 quadros por segundo - caso contrário, a trajetória aparece muito cortada, como uma imagem estroboscópica (ver Figura 6). Não fosse esse fator de imersão, poderíamos ter usado 60 FPS, que haviam se mostrado satisfatórios para controle do sintetizador e que já são uma alta taxa para vídeo.

## 4. Observações finais e Trabalhos futuros

A transformação em diferentes instrumentos pode ser feita facilmente roteando os descritores a qualquer outro sintetizador em *Pd* ou em outra plataforma que possa interpretar o protocolo OSC, ou seja, tratando o módulo de vídeo como uma interface.

<sup>7</sup> Exemplo, entre 0:05 e 00:10s do vídeo [https://www.youtube.com/watch?v=XWqqz\\_XgEHU#t=5](https://www.youtube.com/watch?v=XWqqz_XgEHU#t=5)

Acreditamos que em razão do baixo custo do arranjo, o SoMo tem grande potencial para o uso no contexto educacional (não necessariamente em nível superior).

Futuramente, pretendemos explorar possibilidades de outros descritores, como razão largura/altura do corpo, eixo principal, velocidade do centroide e mesmo análises mais complexas que possam dizer mais do que se passa com o corpo do instrumentista - por exemplo, utilizando visão computacional. Também aventamos a possibilidade de identificar e responder a gestuais.

O SoMo ainda é novo. Podemos, por fim, propor descompromissadamente formas de execução que ainda não experimentamos a contento, como:

- Lançar objetos - movimentos tipicamente rápidos de objetos lançados são possíveis graças à responsividade do SoMo. Com isso, consegue-se gestos que não seriam possíveis utilizando o corpo.
- Pêndulo - objetos pendurados podem ser excitados pelo executante. A característica rítmica desse tipo de dispositivo pode vir a ser esteticamente interessante.
- Ponto luminoso - apagar as luzes faz com que um objeto luminoso - como um telefone celular, um fósforo... - atue como um ponto voador, entidade mais difícil de obter com movimentos corporais
- Plataformas - ao atuar sobre uma plataforma, tem-se um renivelamento das transformações verticais de timbre

## 5. Agradecimentos

A Élder Sereni por testes iniciais e ideias, Dimitri Camorlinga e Fabio Yudi Yokomizu por testes, ideias e experimentações em montagens no curso de Artes Cênicas na Universidade Estadual de Santa Catarina-UDESC.

## 6. Referências

- Brend, M., *The Sound of Tomorrow: How Electronic Music Was Smuggled into the Mainstream*, Bloomsbury Publishing USA, 2012.
- Ruviaro, R. (2012) "From Schaeffer to \*Lorks: An extended definition of Musical Instrument in the context of laptop orchestras", In: *Proceedings of the 1st Symposium on Laptops Ensembles and Orchestras*, Louisiana State University, Baton Rouge, Louisiana.
- Magnusson, T. and Hurtado, M. (2008) "The Phenomenology of Musical Instruments: A Survey", [http://econtact.ca/10\\_4/magnusson\\_hurtado\\_survey.html](http://econtact.ca/10_4/magnusson_hurtado_survey.html) (acessado em 12/09/2015).
- Menezes, F., *A Acústica Musical em Palavras e Sons*, Atelie Editorial, 2004.
- Schaeffer, P., *Traité des Objets Musicaux*. Traducción de Araceli Cabezon de Diego, Alianza Editorial, 2003.
- Silva et. al. (2013) "Avaliando Interfaces Gestuais Para Prática de Instrumentos Virtuais de Percussão", In: *Simpósio Brasileiro de Computação Musical*, Escola de Música de Brasília, Distrito Federal.

# Colagem, recorte e erros em um processo composicional utilizando o Music21.

Guilherme Lunhani<sup>1</sup>

<sup>1</sup>Instituto de Artes e Design – Universidade Federal de Juiz de Fora  
Juiz de Fora, MG

gcravista@gmail.com

**Abstract.** *This article describes a case study in Computer Generated Assistance, for music analysis and didactic composition. The m21 command was programmed to automate routines based on [Music21 2015] library such: selecting, grouping and cutting scores in a J.S.Bach's corpus. In other words, creative exercises were made to test an m21's operation called glitch, applied in this corpus. At the end, comment on bugs, compositional problems and future plans of composition.*

**Resumo.** *Este artigo descreve um estudo de caso em Assistência Gerada por Computador para análise musical e composição didática. O comando m21 foi programado para automatizar rotinas da biblioteca [Music21 2015] como: selecionar, agrupar e recortar partituras de um corpus bachiano. Em outras palavras, exercícios criativos foram feitos para testar uma operação do m21 chamada glitch, aplicada ao corpus. Ao final comentarei bugs, problemas composicionais e planos futuros de composição.*

## 1. Introdução

Este artigo trata de um protótipo de ferramenta, *m21.py*, um binário *Python* para manipulação criativa e analítica do [Music21 2015]. Na seção 2, relacionamos alguns textos à elaboração do programa. Na seção 3 uma justificativa para o projeto.

Uma descrição dos métodos de desenvolvimento, e do método criativo (aplicado aos materiais pré-composicionais gerados), são explicados na seção 4. Na seção 5 são apresentados alguns detalhes da ferramenta, úteis no processo de geração do material. Na seção 6, é apresentado um exemplo prático.

Na seção 7 são apresentados problemas técnicos observados, e uma autocrítica. Na seção 8, os planos futuros de desenvolvimento.

## 2. Trabalhos relacionados

Este trabalho iniciou com a investigação do Music21 que, segundo [Soares 2015]:

É uma biblioteca projetada para trabalhar com manipulação e análise de corpus de arquivos partituráveis. Prepara a conversão entre diversos arquivos de dados musicais.(...) Music21 tem uma abordagem voltada para uma "musicologia assistida por computador" e já tem incorporada em suas classes algumas ferramentas comuns a esta prática como: numeração de grau funcional de acorde, numeração de classes de altura usando a

classificação de Allen Forte : a implementação dos algoritmos de detecção de tonalidade elaborado por Krumhansl (1990) e aperfeiçoada por Temperley (2001), busca de padrões como transposições e inversões e outros.[Soares 2015, p. 71-72]

Por um lado, o desenvolvimento do *m21* busca encoraja a utilização de métodos numéricos que podem ser usados na Análise Musical<sup>1</sup>. Por outro, estimula a transposição de técnicas, nas palavras de [Cascone 2000], pós-digitais. Isto é, uma tradução de técnicas do universo eletroacústico para o universo da notação em partitura. No caso do universo eletroacústico, Cascone esclarece:

A estética pós-digital foi desenvolvida em parte como resultado de uma experiência imersiva, de um trabalho em ambientes repletos de tecnologia digital: computadores, zumbido de *fans*<sup>2</sup>, impressoras a laser produzindo documentos, sonificação de interfaces de usuário, e o barulho abafado de discos rígidos. Mas mais especificamente, surgia das falhas da tecnologia digital que surgiram neste novo trabalho: falhas, *bugs*, aplicação de erros, quebras de sistema, clipagem, serrilhamento, distorção, quantização, e mesmo o ruído de fundo de placas de som são materiais crus que compositores procuram incorporar em sua música.[Cascone 2000, p. 393]<sup>3</sup>

Este artigo relata uma experiência de tecnomorfismo, mais especificamente, da aplicação controlada de erros em partituras, como seleção de compassos, agrupamento de notas e corte de acordes.

O resultado harmônico, dos pontos de vistas da percepção aural, e da notação tradicional, é característico. Como buscamos demonstrar na seção 6, é semelhante àquelas composições que, segundo Soares, estavam em:

Um limite que teve seu auge um pouco antes do início da história da música algorítmica, antes da preocupação imediata com os timbres ou da era das manipulações de amostras sonoras - e de certa maneira ainda proto-serialista. Uma música por vezes chamada politonal, polimodal ou usando o termo de [Straus 2004]: pós-tonal. [Soares 2015, p. 19]

Não sendo um tipo composição nova, é relevante do ponto de vista didático. Neste sentido foi programada uma ferramenta para ser usada em processos criativos musicais, considerando fatores pedagógicos.

<sup>1</sup>Por exemplo, a plotagem de um gráfico contendo histogramas de classes de altura de uma peça.

<sup>2</sup>“Ventiladores” do *cooler*.

<sup>3</sup>Tradução de *The "post-digital" aesthetic was developed in part as a result of the immersive experience of working in environments suffused with digital technology: computer fans whirring, laser printers churning out documents, the sonification of user-interfaces, and the muffled noise of hard drives. But more specifically, it is from the "failure" of digital technology that this new work has emerged: glitches, bugs, application errors, system crashes, clipping, aliasing, distortion, quantization noise, and even the noise floor of computer sound cards are the raw materials composers seek to incorporate into their music.*

## Questões diversas

O compositor Francisco Zmekhol Nascimento de Oliveira sugeria anotar em partituras, peças com um mesmo processo, mas que segundo a contingência do momento, são diferenciadas por seus eventos musicais. Esta abordagem encontrou respaldo nas estratégias composicionais propostas por [Koellreutter 1987]. Após a geração dos materiais no *m21*, uma improvisação de partituras-planimétricas auxiliou articular sons e silêncios, possibilitando fraseados semelhantes aos corais de J.S. Bach.

## 3. Justificativa

Este trabalho capacitou a produção de um número considerável de exercícios criativos, para piano solo. A intenção é oferecer uma ferramenta para geração de materiais pré-composicionais, de maneira quase imediatista. Pode ser útil em cursos de Composição Assistida por Computador, em universidades ou em oficinas de arte.

A programação de um algoritmo de erro (ver seção 5) foi um processo de aprendizagem do [Music21 2015], para programar peças pós-tonais de maneira didática. Isto concorda com o discurso de Cascone (é sugerido transpormos o termo “processamento de sinais digitais” para “intervalos musicais”):

Porque as ferramentas usadas neste estilo de música incorporam conceitos avançados de processamento de sinal digital, a utilização das ferramentas por artistas glitch tendem ser baseadas mais na experimentação do que em uma investigação empírica. Desta forma, usos não-intencionais se tornaram uma segunda permissão garantida. Dizem que alguém não necessita de treino para para usar programas de processamento de sinais - apenas “fuçar” até obtermos o resultado desejado. Algumas vezes, não conhecer a operação teórica da ferramenta pode resultar em casos mais interessantes, por “pensar fora da caixa”[Cascone 2000, p. 397]<sup>4</sup>

## 4. Metodologia

### Organização dos códigos

O programa foi separado em três arquivos: *i*) um binário em *Python* que realiza tarefas gerais da linha de comando (*m21*); *ii*) rotinas do Music21 (*m21utils.py*); e *iii*) um para rotinas externas (*tools.py*)<sup>5</sup>.

### Categorização do software

Nas palavras de [Cope 2008, p. x-xiii], o *m21* pode ser classificado como uma ferramenta para uma Assistência Gerada por Computador (*Computer Generated Assistance* ou CGA). Dentro das sub-categorias de CGA propostas por Cope, o *m21* pode ser incluído nos três modos abaixo: *1*) uso de uma Linguagem de Programação em texto

<sup>4</sup>Tradução de *Because the tools used in this style of music embody advanced concepts of digital signal processing, their usage by glitch artists tends to be based on experimentation rather than empirical investigation. In this fashion, unintended usage has become the second permission granted. It has been said that one does not need advanced training to use digital signal processing programs-just “mess around”until you obtain the desired result. Sometimes, not knowing the theoretical operation of a tool can result in more interesting results by “thinking outside of the box.”*

<sup>5</sup>Todos códigos, exemplos e documentação estão disponíveis <https://www.github.com/jahpd/m21>.

(PLs)(*Programming Languages*) ao invés de uma linguagem de programação visual (VPL); 2) o material partitural é gerado para performance humana ao invés de uma performance eletroacústica; 3) utilização de regras diversas (*Rules Based*) e composição dirigida por dados (*Data-Driven*).

### Método de composição

O exemplo apresentado na seção 6, foi realizado com alguns comandos apresentados na seção 5. Um material bachiano, BWV1, passou por um processo de: *i*) subtração de compassos, aleatoriamente; *ii*) compressão de notas contidas nos compassos restantes em um bloco harmônico; *iii*) embaralhamento de oitavas das notas do bloco, dentro dos limites da tessitura do piano, para derivar novos simultanóides [Koellreutter 1987]; *iv*) eventualmente, erros não previstos acontecem por *bugs* inerentes ao programa desenvolvido. Por exemplo, notas que deveriam ser comprimidas, segundo regra *ii*, não são comprimidas.

O embaralhamento pode resultar em 5 tipos de blocos harmônicos, dependendo do grau de erro escolhido pelo usuário (0 a 4): 0) o simultanóide será arranjado em uma posição fechada; 1) o simultanóide será arranjado em uma posição semi-aberta; 2) o simultanóide será arranjado em uma posição super-aberta; 3) uma nota, aleatória, será separada do simultanóide, como uma appoggiatura; 4) duas notas, aleatórias, serão separadas do simultanóide (se este tiver pelo menos três notas).

O material resultante do processo acima foi editado no [MuseScore 2015]. Algumas interferências pessoais, não previstas, foram incluídas. Seguimos com a observação de “organicidades” próprias do material gerado: fraseados, pontos culminantes e pontos de relaxamento, criando um pequeno discurso musical.

Por último, a peça foi diagramada no [Lilypond 2015].

## 5. M21

Listamos, na tabela 1, algumas opções úteis do comando `./m21`. Foram usadas para a elaboração de materiais pré-composicionais, apresentados na seção 6.

Um manual de instalação e operação mais detalhado está disponível junto com o código-fonte<sup>6</sup>.

<sup>6</sup>Disponível em <https://www.github.com/jahpd/m21/doc/manual.pdf>.

**Tabela 1: Tabela das opções utilizadas para produção e análise de um arquivo partiturável. Fonte: autor.**

Nome	Comando	Abreviação	Execução
<b>Compositor</b>	<code>--composer</code>	-c	Campo de procura no corpus pelo nome de um compositor. Usado em conjunto com a opção "Index".
<b>Index</b>	<code>--index</code>	-i	Indexação de uma peça (catálogo, p.e., bwv123). Usado em conjunto com a opção "Compositor"
<b>Arquivos xml</b>	<code>--xml</code>	-x	<i>Flag</i> indicativa que será fornecido um arquivo <i>musicXml</i> [MusicXML 2015] ao invés de uma obra do corpus.
<b>Composição Assistida por Computador</b>	<code>--CAC</code>	-C	<i>Flag</i> indicativa de uma operação de transformação em uma partitura. Usada em conjunto, as opções "Compositor" e "Index".
<b>Glitch</b>	<code>--glitch</code>	-g	<i>Flag</i> indicativa do tipo de operação de transformação. A peça é desorganizada e verticalizada em blocos harmônicos, de maneira randômica, limitada apenas por regras de tessitura do piano.
<b>Apresentar em um editor de partituras</b>	<code>--Show</code>	-S	<i>Flag</i> indicativa que, o resultado obtido será executado em um editor de partituras apropriado, no caso deste trabalho, o [MuseScore 2015].
<b>Plotar gráficos analíticos</b>	<code>--plot-*</code>	Não possui	<i>Flag</i> indicativa que um gráfico analítico será gerado. O Símbolo "*" representa a diversidade dos tipos de gráficos possíveis.

## 6. Resultados

Diversas opções diferentes do mesmo comando foram utilizadas para a estruturação das peças didáticas, realizadas por colagem de materiais da bachianos, recorte por erros, e readequações do discurso harmônico pós-tonal. Apresentamos dois exemplos de comandos no Código 1. O primeiro é sua forma em extenso, e a segunda, na sua forma comprimida. Ambos realizam o mesmo procedimento (mas irão gerar resultados diferentes).

Em um momento específico, no dia 08/07/2015, o comando foi executado para a elaboração da peça discutida na 6.1.

```
1 | ./main.py --show --CAC --composer bach --index bwv1 --glitch 2
2 | ./main.py -S -C -c bach -i bwv1 -g 2
```

**Listing 1: Código utilizado para composição do Coral #1. Na linha 1 apresentamos sua versão estendida, e na linha 2, sua versão comprimida. Fonte: Autor.**

### 6.1. BWV1

Experimentamos uma aplicação de erros no sexto movimento de *Wie schön leucht der Morgenstern* (Cantata para a festa da Anunciação, 1725, ver figura 1). Foi gerado um conjunto de simultanóides apresentados na figura 2.

**Music21 Fragment**

The image displays a musical score fragment for BWV1, titled "Music21 Fragment". It consists of five staves: Horn 2, Soprano, Alto, Tenor, and Bass. The music is in 4/4 time and features a complex harmonic structure with various accidentals and dynamics. The fragment is divided into four measures, with measure numbers 2, 3, and 4 indicated above the staves. The title "Music21" is written in the top right corner.

**Figura 1: Fragmento do sexto movimento do BWV1. Fonte: [Music21 2015].**

Cada bloco harmônico é constituído de notas de um determinado compasso, escolhido ao acaso pelo programa, e comprimidas em um único evento. A aplicação de erro no material bachiano deriva um fraseado harmônico (como por exemplo, a partir do primeiro

## Music21 Fragment



Figura 2: Sequência de simultanéides gerados. Fonte: Autor.

tempo do primeiro compasso ao terceiro tempo do terceiro compasso na figura 2), que vai da região média aos graves, percebida após repetidos usos do comando descrito.

Detalhamos o método composicional um pouco mais:

- A ordem dos simultanéides foi mantida;
- Foram escolhidos pontos que delimitam fraseados (silêncio como articulado de frases);
- Foi estabelecido que a quantidade de notas em um bloco como fator de tensão;
- Improvisamos dinâmicas como um dispositivo de ênfase do fraseado harmônico;
- Foram modificados um si bemol para si natural no terceiro simultanéide, e um si natural para si bemol no anti penúltimo simultanéide, para criar uma unidade harmônica.
- Aplicação de Deslocamento, retardamento ou subtração de elementos de notas.

A peça finalizada está na figura 3.

### Breve Análise

As ferramentas analíticas do *m21* auxiliaram na observação de semelhanças e diferenças entre a peça original e a peça variada. Por exemplo, com BWV1 de J.S.Bach, plotamos um histograma na figura 4, a partir do comando abaixo

```
1 | ./m21 --show --composer bach --index bwv1 --plot-histogram-pitch-space
```

Este histograma revela fatos analíticos óbvios, como uma ênfase do *pitch class* 5, Fá, 1º grau; Dó, 5º grau, seguido de outras classes, como Lá (3º grau) e Sol (2º grau); outros, como Ré (6º grau) e Si bemol, possuem a mesma quantidade. Por último Mi (7º grau) e si natural possuem os menores números, talvez como dispositivos cadenci-ais (quinto compasso, primeiro a terceiro tempo, da figura 1).

Um histograma também foi gerado para a peça resultante (figura 5), de acordo com o comando abaixo:

```
1 | ./m21 --show --xml bwv1_material.xml --plot-histogram-pitch-space
```

É possível notar que algumas proporções foram mantidas de maneira aproximada, mesmo com uma quantidade de eventos diminuta. Com a segmentação da peça, um centro tonal ainda é mantido (com os 1º e 5º graus em maior número), embora de maneira ambígua (2º, 3º e 6º graus possuem a mesma quantidade de aparições).

para Glerm Soares

**Coral #1**

Após uma extração de alturas do BWV1 em 08/7/2015

Bach / music21

Guilherme Lunhani

*Largo*

Piano

*f*  $\text{cresc.}$  *ff*  $\text{decresc.}$  *p* *f*

5 *ff* *sfz* *mp*

10 *pp* *sfz* *f*  $\text{cresc.}$  *mf*

Figura 3: Peça resultante das intervenções. Fonte: autor.

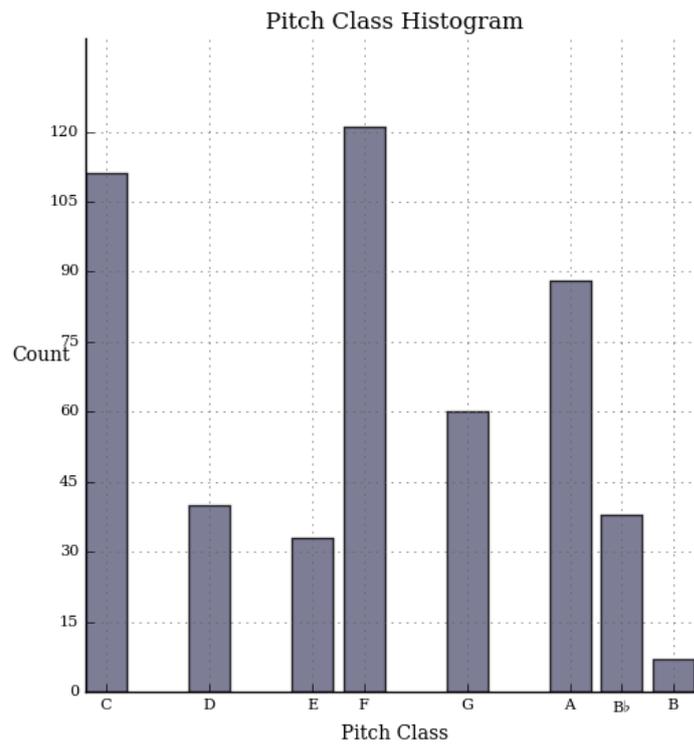


Figura 4: Histograma de *pitch-class* do BWV1. Fonte: Autor

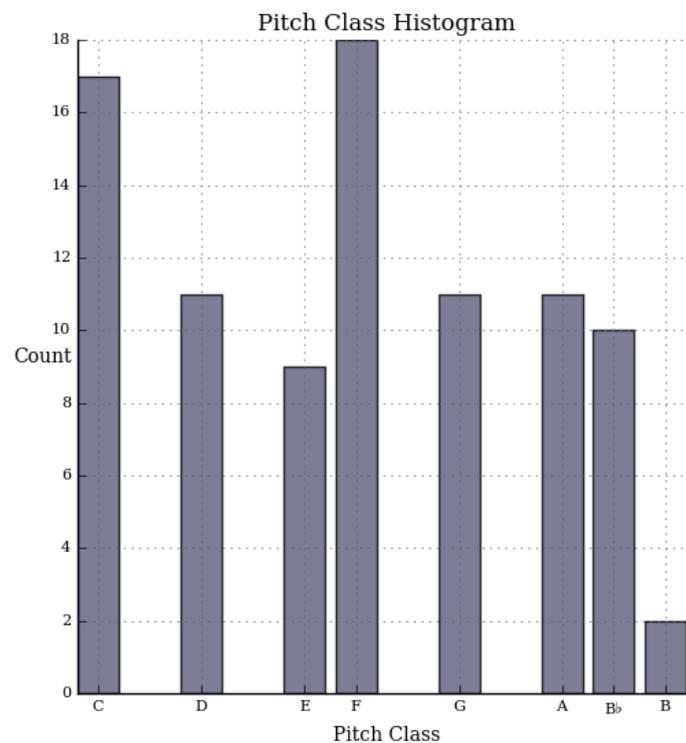


Figura 5: Histograma de *pitch-class* da peça Coral #1. Fonte: Autor

## 7. Conclusão

O *m21* possibilitou automatizar rotinas do [Music21 2015], que podem ser laboriosas em exercícios de composição ou análise.

Das rotinas analíticas possíveis, destacamos os gráficos de histogramas de classes de altura. Das rotinas composicionais, enumeramos procedimentos criativos como colagem, compressão, troca de oitavas e fragmentação do material resultante, nesta ordem.

Uma variedade de materiais pré-composicionais foram gerados. Apresentamos um exemplo prático: a partitura foi elaborada em um período reduzido de tempo, entre duas e três horas, considerando a geração do material, edição do discurso harmônico, escolha de pausas, fraseados, articulações, dinâmicas, pausas, e editoração.

No entanto, existe uma certa homogeneidade nos diferentes materiais gerados. O que pode ser um fator de desinteresse. Neste sentido, o *m21* ainda carece de funções que manipulem alguns processos do parágrafo acima.

Sumarizando, o *m21* é uma de ferramenta computacional e pedagógica para processos criativos e analíticos. A intenção é oferecer uma ferramenta, para docentes e discentes de composição, e análise musical, que se interessam pela assistência que um programa pode oferecer.

## 8. Planos Futuros

Investigação e correção de *bugs*. Implementação de um novo comando que fragmente várias obras do corpus em um único material pré-composicional (comando `--popp` ou `-p`). Continuação de novas composições para o ciclo.

## 9. Agradecimentos

Ao Guilherme Rafael Soares por apresentar a biblioteca *music 21*. Aos desenvolvedores do *musescore* e *lilypond*. FAPEMIG pelo financiamento da pesquisa.

## Referências

- Cascone, K. (2000). The aesthetics of failure: 'post-digital' tendencies in contemporary computer music. *Computer Music Journal*.
- Cope, E. D. (2008). Prefacio a OM composer's book vol. 2. In *OM Composer's Book*, volume 2, pages ix–xv. Editions Delatour.
- Koellreutter, H. J. (1987). *Introdução à estética e à composição musical contemporânea*. Movimento, 2 edition.
- Lilypond (2015). Lilypond.
- MuseScore (2015). Musescore.
- Music21 (2015). Music21.
- MusicXML (2015). Musicxml.
- Soares, G. R. (2015). Luteria composicional de algoritmos pós-tonais v1.1final.
- Straus, J. N. (2004). *Introduction to Post-Tonal Theory (3rd Edition)*.

# A Música Visual como alternativa aos modelos computacionais utilizados no controle da Síntese Granular em Tempo Real

Fernando Falci de Souza<sup>1,2</sup>

<sup>1</sup>Instituto de Artes – Universidade Estadual de Campinas (UNICAMP)  
Cidade Universitária Zeferino Vaz - Campinas – SP – Brasil

<sup>2</sup>Núcleo Interdisciplinar de Comunicação Sonora (NICS)  
Universidade Estadual de Campinas (UNICAMP)

fernando\_ffs@yahoo.com.br

**Abstract.** *Granular Synthesis is a technique for synthesis, analysis and processing of sound well known for its use in electroacoustic music that frequently presents challenges for real time control. Many computer-based models were proposed as solutions, each of which with its own features. In this paper, we describe a possible alternative to the use of such models presenting Visual Music a high-level aesthetical model that not only gives the performer a visual feedback in real time but also adds significant value to the work of art.*

**Resumo.** *A Síntese Granular é uma técnica para a síntese, análise e processamento de som, bastante utilizada na composição de música eletroacústica, que frequentemente apresenta desafios para o controle em tempo real. Diversos modelos computacionais foram propostos como solução, cada qual com características particulares. Neste artigo descrevemos uma possível alternativa para o uso destes modelos na qual a Música Visual é utilizada como um modelo estético de alto nível que, além de fornecer ao performer um feedback visual em tempo real, também adiciona valor expressivo à obra de arte.*

## 1. Introdução

Neste artigo descrevemos o desenvolvimento de um conjunto de sistemas e o processo criativo de uma obra artística audiovisual na qual texturas complexas são obtidas a partir da organização de elementos simples denominados grãos de som e imagem. Paralela à nossa pesquisa está a área conhecida como Síntese Granular, que se trata de uma técnica para síntese de som, e no outro extremo está a área da Música Visual, que delimita uma prática artística audiovisual específica. Embora se tenha obtido muito progresso ao longo de décadas de pesquisa nestas duas áreas, os avanços tecnológicos atuais permitem novas formas de integração entre som e imagem, levantando novas questões e paradigmas. Estas modernas tecnologias requerem naturalmente novos modelos para organizar o conteúdo audiovisual, com novos desafios à criatividade humana.

Na Seção 2, apresentamos a Síntese Granular e as dificuldades que surgem no controle em tempo real de uma quantidade descomunal de parâmetros de grãos sonoros. Descrevemos brevemente algumas soluções encontradas ao longo de décadas de pesquisa nas quais modelos computacionais foram utilizados no controle de alto-nível (global) dos parâmetros da síntese sonora, cada qual com suas vantagens e desvantagens.

Apresentamos também a motivação que nos leva a propor a Música Visual como alternativa ao uso dos modelos computacionais e algumas pesquisas recentes que representam esforços nesta mesma direção. Em seguida, na Seção 3, descrevemos os experimentos iniciais que fizemos no decorrer do desenvolvimento dos sistemas “*Granular Streamer*” e “*Granular Drawer*” que implementam a Síntese Granular Audiovisual. Na Seção 4 apresentamos o modelo que denominamos Música Visual Granular e possíveis formas de correspondência entre grãos de som e grãos de imagem. Um estudo de caso foi realizado com a criação da obra “*Contrapontos*” que é descrita na Seção 5. Finalmente, na Seção 6 apresentamos nossas conclusões deste trabalho e algumas propostas de pesquisa futura.

### 1.1. Agradecimentos

Este artigo é parte de uma pesquisa de doutorado realizada com apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), processo nº 2011/03713-2. As opiniões, hipóteses e conclusões ou recomendações expressas neste material são de responsabilidade do (s) autor (es) e não necessariamente refletem a visão da FAPESP.

## 2. O controle da Síntese Granular em tempo real

A Síntese Granular é uma técnica digital de composição fundamentada na Teoria do Quantum Acústico do físico Dennis Gabor. Gabor demonstrou que sons de qualquer complexidade e duração podem ser decompostos em células acústicas de conteúdo espectral simples (por exemplo, senóides com envoltórias gaussianas) e de duração da ordem de 10 a 200 ms [Gabor 1947]. Estas células, denominadas grãos de som, formam, na prática, uma base de representação para qualquer som no mesmo espírito da Série e da Transformada de Fourier ou ainda as chamadas *Wavelets*.

Logo após sua publicação, o trabalho de Gabor inspirou compositores a realizar o processo inverso, isto é, compor sons, timbres e texturas a partir da organização de grãos de som. Porém, devido ao seu enorme espaço de possibilidades sonoras e à grande quantidade de grãos de som que são necessários para a composição de um segmento apreciável de música, a pesquisa sobre o uso desta técnica é frequentemente voltada à busca por métodos e modelos que visam simplificar a tarefa de selecionar e organizar os tipos de grãos adequadamente, levando em consideração fatores como automatização, flexibilidade, a capacidade computacional e ainda o custo da tarefa (tempo de trabalho).

Historicamente, o compositor grego Iannis Xenakis foi o primeiro a utilizar o termo “*Microsom*” para fazer referência a objetos sonoros construídos a partir de grãos de som e, conseqüentemente, cujo controle é realizado no nível micro-temporal, como por exemplo, em sua peça “*Concrete P.H.*”, na qual pequenos recortes de uma fita magnética contendo uma gravação são reorganizados [Xenakis 1971], [Roads 2001]. Stockhausen também compôs peças de natureza granular como, por exemplo, em “*Kontakt*”, onde utiliza geradores de pulso analógicos para criar os grãos de som [Stockhausen 1957]. Em ambos os casos o processo de organizar foi extremamente trabalhoso, impondo fortes restrições na quantidade total de grãos e nas possibilidades sonoras, o que reforça o espírito pioneiro destes compositores.

A partir da década de 70, com o advento da tecnologia digital, a técnica da Síntese Granular tornou-se mais viável. Aqui podemos destacar as contribuições de Curtis Roads, que realizou inúmeros experimentos musicais com síntese granular criando uma vasta

taxonomia para classificar tanto os tipos de grãos como também sua forma de organização [Roads 2001], e de Barry Truax que, por meio do desenvolvimento de programas específicos para o “DMX-1000”, um processador de sinais digital, foi o primeiro a implementar a Síntese Granular em tempo real [Truax 1988].

Uma vez estabelecida a possibilidade de realizar a Síntese Granular em tempo real, durante os anos de 1990 a 2010, aproximadamente, a pesquisa na área foi direcionada à experimentação com modelos computacionais que permitissem que o usuário controlasse a síntese de uma grande quantidade de grãos de som através de um conjunto reduzido de parâmetros de alto nível. Neste sentido, DiScipio controlava os parâmetros de síntese através de Modelos Fractais, Sistemas Dinâmicos e Teoria do Caos [DiScipio 1990], Maia e Miranda utilizaram Conjuntos Fuzzy e Cadeias de Markov [Maia e Miranda 2005], a Teoria dos Grafos foi proposta como modelo de controle por Valle e Lombardo [Valle 2003]. Também experimentamos a aplicação de uma solução deste tipo em nossa pesquisa anterior com o desenvolvimento do sistema “EVOGrain” no qual um Algoritmo Genético controla em tempo real a evolução de uma população de grãos de som [Souza e Maia 2009].

## 2.1. Motivação

Embora a aplicação de modelos computacionais no controle da Síntese Granular tenha obtido resultados interessantes durante ao menos duas décadas, ao nosso ver, o uso de tais modelos, ao mesmo tempo em que simplifica e automatiza a seleção e organização dos grãos de som, implica também em duas desvantagens. Primeiramente, em sua maioria, os modelos restringem demasiadamente o espaço de possibilidades sonoras da Síntese Granular, chegando a um ponto em que a sonoridade resultante é tão particular do modelo que nos leva a questionar se este é que está servindo à criação musical ou se o que ocorre de fato é a sonificação do modelo.

Em segundo lugar, a indagação que fizemos no decorrer de nossa pesquisa é que, do ponto de vista de um usuário com formação musical tradicional, a Síntese Granular já apresenta alguns parâmetros de controle não usuais tais como, por exemplo, “densidade de grãos por segundo”, “largura de banda da distribuição dos grãos”, entre outros. O uso de um modelo computacional pode dificultar ainda mais a compreensão do sistema para este usuário caso sejam apresentados parâmetros de controle inerentes ao modelo, como ocorre no sistema “EVOGrain”, no qual são utilizados parâmetros como, por exemplo, “tamanho da população”, “velocidade de reprodução”, “taxa de evolução”, etc.

Sendo assim, partimos da hipótese que oferecer ao usuário um feedback visual em tempo real melhora a sua percepção sobre os processos sonoros em andamento e pode auxiliar no controle da Síntese Granular reduzindo a necessidade de automação na seleção dos grãos de som ao mesmo tempo em que impõe poucas restrições ao espaço sonoro tornando o processo de controle de parâmetros mais intuitivo.

## 2.2. Estado da Arte

Encontramos pesquisas recentes que contribuem para a integração da Síntese Granular ao domínio visual que, embora se diferenciem fundamentalmente de nossa pesquisa, demonstram os esforços que vem sendo realizado neste sentido. Uma destas pesquisas integra os domínios do som e da imagem por meio de um sistema de partículas [Callear 2011]. O usuário controla o sistema provendo dados para simulações feitas no sistema de

partículas. As informações resultantes da simulação são mapeadas paralelamente para o domínio sonoro e convertidas em grãos de som e para o domínio visual e convertidas em imagem. É possível notar uma forte presença do modelo nas obras resultantes e também um paralelismo constante entre os domínios do som e da imagem.

Em outro trabalho, o sistema “Kortex” é utilizado para performances audiovisuais [Batty et al 2013], [Batty 2014]. Trata-se de uma extensão visual da técnica conhecida como granulação, que consiste da fragmentação de um arquivo de áudio e da reorganização de seu conteúdo mantendo em grande parte a relação temporal. A granulação é apenas uma forma de organizar grãos de som dentre tantas outras utilizadas na Síntese Granular. Nestes trabalhos o conteúdo visual não explora a geração de texturas e os domínios do som e da imagem também mantêm-se constantemente paralelos.

### 3. Síntese Granular Audiovisual

Diante da motivação descrita anteriormente, demos continuidade pesquisa de um sistema para Síntese Granular em tempo real que ao mesmo permitisse explorar todo o espaço de possibilidades sonoras desta técnica sem utilizar modelos computacionais com parâmetros específicos e estranhos ao uso comum. Considerando o poder de processamento de um computador pessoal atual, verificamos a viabilidade de apresentar ao usuário um feedback visual que reproduzisse fielmente os processos em andamento na síntese granular de som tornando assim o sistema mais fácil e intuitivo de usar. Com o intuito também de permitir portabilidade para os diversos sistemas operacionais, desenvolvemos em JAVA dois sistemas modulares, o “*Granular Streamer*” e o “*Granular Drawer*”, que se comunicam por meio de mensagens *Open Sound Control* (OSC).

#### 3.1. O sistema “*Granular Streamer*”

O “*GranularStreammer*” realiza a Síntese Granular de som em tempo real e, por não estar vinculado a um modelo de alto nível, não apresenta restrições em seu espaço de possibilidades sonoras. Este sistema é organizado em um número configurável de fluxos granulares independentes. Para cada fluxo o usuário deve configurar os grãos de som escolhendo uma forma de onda e uma envoltória de amplitude dentre uma ampla biblioteca que inclui formas de onda como, por exemplo, senoidal, dentes de serra, quadrada, *wavetable*, entre outras, e envoltórias como, por exemplo, gaussiana, trapezoidal, exponencial, etc. O usuário também controla dinamicamente cada fluxo por parâmetros típicos da Síntese Granular tais como frequência, amplitude, duração dos grãos, posicionamento e densidade, através de uma interface com controles deslizantes (*sliders*) que podem ser facilmente associados à outros sistemas e dispositivos (controladores gestuais) habilitados a enviar mensagens OSC. Neste sistema, sempre que um grão é sintetizado e emitido pela placa de som, uma mensagem OSC contendo todos os parâmetros do grão é enviada para um endereço de rede configurado pelo usuário. Esta mensagem é identificada pelo nome “*/grain created*” e formada pelos parâmetros “identificador do fluxo”, “frequência do grão”, “duração do grão”, “amplitude do grão” e “posicionamento no campo estereofônico” (também conhecido como *pan*).

#### 2.2. O sistema “*Granular Drawer*”

O “*GranularDrawer*” é um sistema, ou melhor, é um conjunto de classes desenvolvidas em JAVA, utilizando as bibliotecas do *Processing* (<https://processing.org/>), que realiza

as tarefas de receber as mensagens “/grain\_created” enviadas pelo “*GranularStreamer*” e de desenhar em tempo real as representações visuais para os grãos de som. Na primeira aplicação que desenvolvemos com estas classes, os grãos de som foram representados por círculos desenhados na parte inferior da tela, e então deslocados lentamente para cima em um movimento conhecido como *scrolling*, desta forma representando o tempo no eixo vertical da visualização (Figura 1). Nomeamos estas representações visuais de “grãos de imagem”, estabelecendo um paralelo com os grãos de som e estabelecendo um mapeamento do domínio sonoro para o domínio visual, convertendo frequência em cor, duração em tamanho, amplitude em transparência e *pan* na posição horizontal do grão na tela.

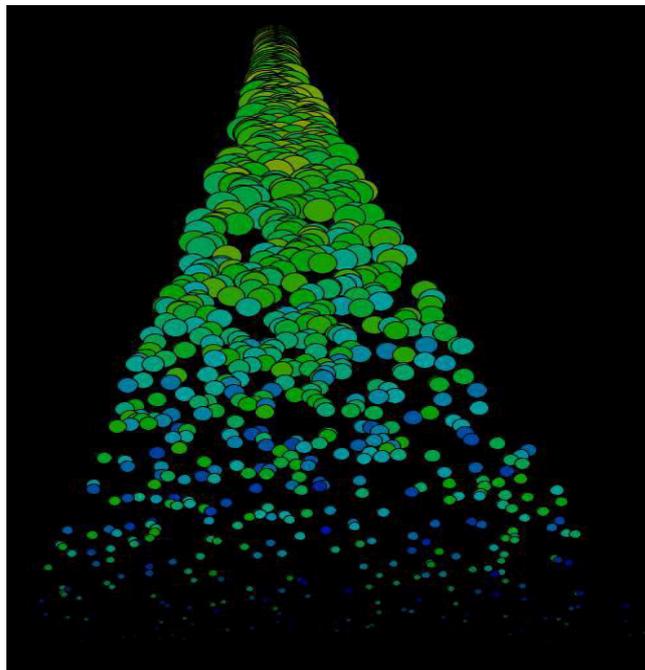


Figura 1. Primeira implementação da Síntese Granular Audiovisual.

### 2.3. Primeiros experimentos

Neste primeiro experimento, criamos um mapeamento de um para um entre os parâmetros do som e da imagem, o que permite uma análise visual bastante clara e direta do processo de transformação do material sonoro. Independente da escala de representação, podemos perceber que os grãos inicialmente ocupam uma posição central dentro do campo estereofônico, têm tamanho grande e frequências médias representadas por cores predominantemente verdes (parte superior da figura). Gradualmente, os grãos passam a ser espalhados por todo o campo sonoro, desde o extremo esquerdo até o direito, têm seu tamanho reduzido e frequências em registros mais altos, representadas por cores azuis (parte inferior da imagem).

Após realizarmos diversas demonstrações deste sistema pudemos fazer algumas observações. Em primeiro lugar, o *feedback* visual é de grande importância para o usuário, ou intérprete, que passa a controlar os parâmetros da Síntese Granular auxiliado não apenas pelo som emitido pelo sistema, mas também pela distribuição dos grãos de

imagem. Quanto à representação do tempo, o momento atual é o mais importante para ser exibido ao usuário, não havendo necessidade de acumular os eventos anteriores utilizando o recurso do *scroll*. Ainda, dentro de um processo granular específico, não haveria necessidade de representar visualmente os parâmetros que não são alterados. Por exemplo, se em um processo granular, a duração dos grãos de som é mantida constante, não há necessidade de representar visualmente esta informação. Isto poupa o usuário de excesso de informação focando a sua atenção em seu trabalho artístico. Finalmente, e mais importante, o feedback visual demonstrou um bom potencial para tornar-se parte da performance artística adicionando valor expressivo à obra.

#### 4. Música Visual Granular

Diante dos bons resultados obtidos com os primeiros experimentos em Síntese Granular Audiovisual, nossa pesquisa mudou ligeiramente de direção. Ao invés de procurar desenvolver um sistema que abarcasse todas as possibilidades sonoras da Síntese Granular provendo ao usuário um feedback visual único e abrangente, passamos a focalizar em casos específicos de criação artística nos quais som e imagem são gerados em tempo real.

Neste sentido, a área de pesquisa conhecida como Música Visual estuda uma forma de arte particular dentro de um grupo maior conhecido como Arte Audiovisual o qual inclui também o Filme Abstrato, Filme Experimental, Vídeo Arte, entre outras categorias. A arte audiovisual pode ser apresentada em diferentes configurações, desde vídeos em suporte fixo até performances ao vivo e instalações interativas, e é uma forma de arte na qual, com ajuda da tecnologia, a imagem em movimento acompanhada por som, é planejada, produzida, comentada e apreciada [Zielinski 1999]. A Música Visual, mais especificamente, localiza-se em uma zona fronteira entre os sentidos da visão e da audição, na qual imagem e som tendem a se fundir por meio de representações mutuamente abstratas. Historicamente, o termo “*Lumia*”, utilizado no Século XVIII, refere-se ao que hoje recebe o nome de Música Visual, isto é, uma forma de arte que permite ao artista visual “tocar imagens da forma como músicos tocam sons”. Seus elementos principais de expressividade no domínio visual são a cor, a forma e o movimento [Collopy 2000]. A Música Visual é um termo utilizado para descrever um amplo leque de práticas artísticas, espalhadas temporal e geograficamente, porém com a ideia comum de que as artes visuais podem aspirar as qualidades dinâmicas e não objetivas da música [Jones e Neville 2005].

Portando, se estamos utilizando grãos de som para a geração de sons, timbres e texturas, que por sua vez são organizados musicalmente, ao representar este material sonoro no domínio visual, utilizando grãos de imagem com certo conteúdo expressivo, estamos criando imagens e animações de natureza granular, isto é, Música Visual Granular.

##### 4.1. Correspondências entre som e imagem

Por se tratar de uma forma de arte na qual o som e a imagem estão intimamente relacionados, a criação de trabalhos de Música Visual frequentemente envolve discussões sobre possíveis correspondências entre os domínios sonoro e visual. Neste sentido, grandes esforços foram realizados para encontrar uma correspondência absoluta entre frequências sonoras e as cores. Porém, embora som e luz possam ser representados matematicamente por modelos ondulatórios ou de partículas, estes se tratam de

fenômenos físicos com propriedades bastante distintas, e não existe de fato uma correspondência física ou concreta para a conversão de som em imagem. [Edmonds 2004] ou, colocando de outra forma, a correspondência é arbitrária, dependente de modelos físicos, matemáticos ou de outras imaginações. No entanto, com o propósito de analisar os processos de criação de Música Visual que temos em mente, abaixo oferecemos uma classificação geral emprestada das teorias das possíveis correspondências.

Em contrapartida, seja qual for o modelo de criação de música visual, considerando o indivíduo e seus mecanismos de percepção ao invés de observar os fenômenos físicos da luz e do som, diversos experimentos realizados no Século XX demonstraram que a cor é um forte estímulo para a percepção do ambiente ao nosso redor podendo afetar o nosso ritmo cardíaco, a sensação de tempo, estimativas de tamanho, peso e temperatura de objetos e também a percepção da amplitude de som e de ruído, sendo que este último resultado indica fortemente uma relação cognitiva entre a percepção de som e imagem [Mahnke 1996]. Os modelos dos objetos do mundo real, construído em nossa cognição e armazenados em nossa memória, tem característica multissensorial, isto é, são modelos de objetos nos quais som, imagem, cheiro, sabor, temperatura, textura, entre outras qualidades, estão amalgamados e relacionados a outros objetos em diversas combinações interativas [Haverkamp 2009].

#### 4.2. Articulando grãos de som e de imagem

Foi demonstrado que quando um evento auditivo e um evento visual ocorrem dentro de um curto intervalo de tempo, a sensação será atribuída a um objeto multissensorial específico [Kohlrausch 2005]. Sendo assim, em intervalos de tempo pequenos, qualquer imagem ou efeito visual pode ser considerado um grão de imagem. A Música Visual faz uso deste fato para construção de segmentos complexos de uma obra. A Figura 2 apresenta a formalização do modelo que estamos utilizando para a criação de Música Visual Granular no qual, no domínio sonoro, temos como material básico grãos de som, e no domínio visual, grãos de imagem, e alguns possíveis tipos de correspondências entre estes grãos.

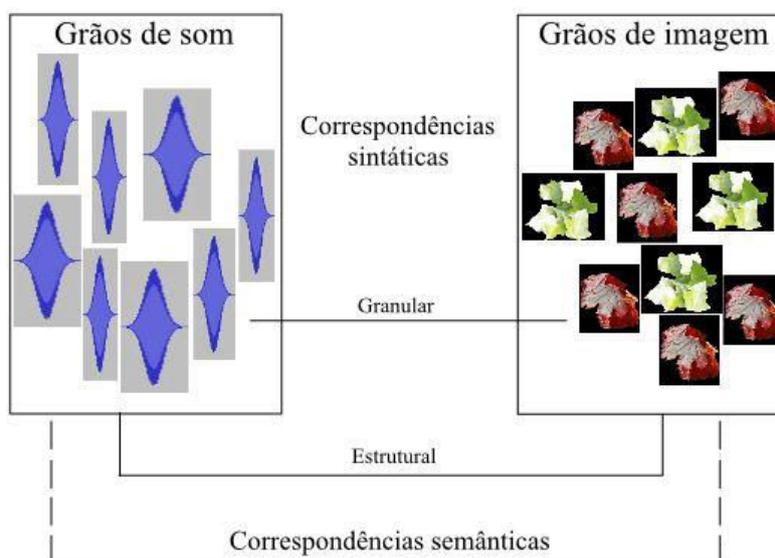


Figura 2. Material básico para criação de Música Visual Granular.

As Correspondências Sintáticas são criadas através de mapeamentos dos atributos próprios do material de um domínio no outro domínio. Podem ocorrer em dois níveis: no nível granular, quando são utilizados os parâmetros do grão, seja de som ou de imagem (envoltória, forma de onda, frequência, duração, amplitude, brilho, cor, forma, posição) ou no nível estrutural, quando o mapeamento é criado sobre parâmetros globais de um conjunto de grãos (quantidade de grãos, frequência média dos grãos, área de distribuição dos grãos, amplitude resultante dos grãos).

As Correspondências Semânticas associam o material audiovisual por meio de uma informação interpretada do som e da imagem por parte do compositor ou performer. Durante o processo criativo de uma obra de Música Visual Granular, em algum momento deverão ser definidos os tipos de grão de som utilizados, especificando suas envoltórias de amplitude, formas de onda e gravações de áudio. Também serão definidos os grãos de imagem diante de uma imensa possibilidade de formas abstratas, fragmentos de fotografias, vídeos, efeitos visuais, etc. Esta escolha pode definir uma Correspondência Semântica ao material, criando “relações gramaticais” de sinonímia, antonímia, metáfora e outras.

## 5. “Contrapontos”: um estudo de caso

“Contrapontos” é uma obra aberta em um processo constante de experimentos e melhorias. A partir de um material reduzido e fragmentado apresentamos contrastes que favorecem a reflexão sobre os sentidos da audição e da visão, o som e a imagem, o simultâneo e o defasado, o sintético e o amostrado, o puro e o ruidoso. Também se apoia poeticamente ( e semanticamente) em contrapontos sociais tais como a imensidão de uma metrópole e o confinamento dos espaços fechados, a liberdade e os padrões sociais e o caos e a ordem. Uma performance completa deste trabalho pode ser acessada pelo link [https://www.youtube.com/watch?v=HxgeCnO-y\\_E](https://www.youtube.com/watch?v=HxgeCnO-y_E).

Nesta obra, utilizamos os sistemas “*GranularStreamer*” e “*GranularDrawer*” controlados pela interface nanoKontrol2 fabricada pela Korg (<http://www.korg.com>). Em sua parte principal, este dispositivo possui oito conjuntos de controles, cada qual composto por um *slider*, um *knob* e três botões (Figura 3). Foram criados oito fluxos de grãos independentes e a cada um deles associamos um destes conjuntos para controlar certos parâmetros escolhidos da Síntese Granular Audiovisual.



Figura 3. Controlador nanoKontrol2 da Korg.

Esta obra está dividida em três grandes seções, A, B e C, cada qual com um material audiovisual específico explorando, respectivamente, os tipos de Correspondência Granular, Estrutural e Semântica, descritos em nosso modelo da Música Visual Granular. Ao longo de sua execução, o performer pode ir e voltar a cada uma das

seções, rerepresentando e retrabalhando o seu material característico, de forma que ao término da obra, a distribuição do tempo dispendido em cada seção seja aproximadamente o mesmo.

### 5.1. Contrapontos granulares

Na seção A, foram utilizados os três primeiros conjuntos de botões, 1, 2 e 3, associados a três fluxos de grãos audiovisuais. No domínio sonoro, utilizamos formas de onda senoidais e quadradas e envoltórias gaussianas e exponenciais, ou seja, utilizamos o som puro e sintetizado como material sonoro básico. Para cada grão de som emitido por estes fluxos, será criado também um grão de imagem equivalente, representado por um quadrado de cores cheias e com bordas pretas, em uma relação de um para um entre o som e a imagem (Figura 4). Embora visualmente esta peça faça referência à estética da obra de Piet Mondrian, a escolha deste material audiovisual se apoia em uma relação semântica que estabelecemos entre grãos de som e imagem quanto à simplicidade e a artificialidade do som e da forma. Nesta seção, os grãos de som e de imagem podem ser articulados simultaneamente ou independentemente, ou seja, podemos ter grãos de som sem que exista um correspondente visual e vice-versa, ou com certa defasagem entre um e outro. Desta forma, evitamos o paralelismo constante entre os domínios que passam a se complementar como se fossem duas vozes independentes de um contraponto musical.

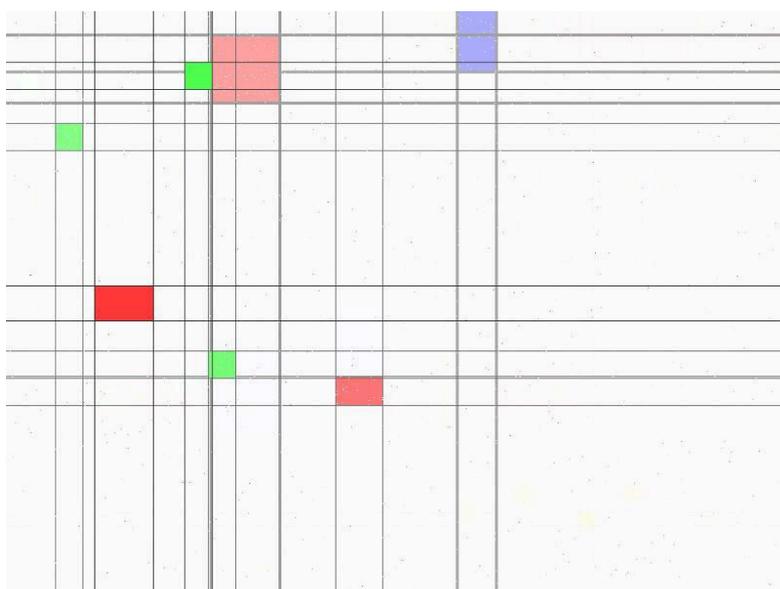
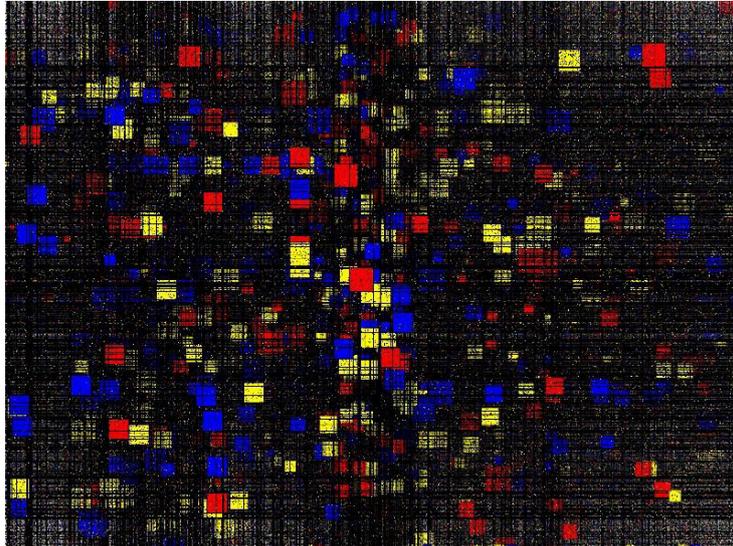


Figura 4. Correspondência Granular na Seção A de *Contrapontos*.

### 5.2. Contrapontos estruturais

Na seção B, utilizamos dois fluxos de grãos de som cuja forma de onda é o ruído branco, associados aos conjuntos 4 e 5 de controles do nanoKontrol2. Visualmente, estes grãos sonoros são representados por um efeito global que embaralha certa quantidade de pixels proporcional à quantidade de grãos de som, porém não em uma relação biunívoca de um pixel para cada grão de som. Desta forma, estabelecemos uma correspondência mais flexível entre os domínios a qual denominamos Correspondência Estrutural (Figura 5).

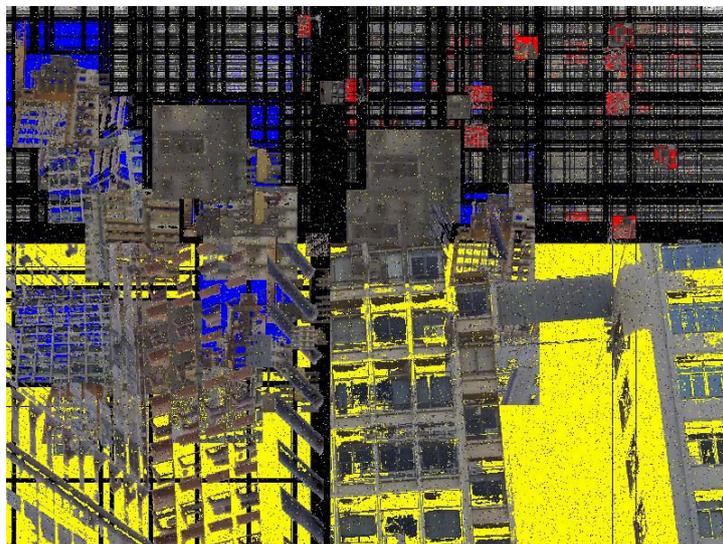
Nesta seção, são explorados objetos audiovisuais que consistem em uma grande quantidade de grãos emitidos por segundo (alta densidade) que são percebidos como texturas.



**Figura 5. Correspondência Estrutural na Música Visual Granular.**

### 5.3. Contrapontos semânticos

Por fim, na seção C, utilizamos grãos de som cuja forma de onda é obtida a partir de gravações do som das ruas e do tráfego da cidade de São Paulo no turbulento horário de pico. Utilizamos três fluxos sonoros associados aos conjuntos 6, 7 e 8 do controlador. Para estabelecer uma relação semântica entre este material sonoro e o material visual, escolhemos utilizar imagens de prédios da cidade, criando um contraponto entre a imensidão da metrópole e o espaço confinado entre construções ou dentro de veículos.



**Figura 6. Correspondência Semântica entre som amostrado e fotografias.**

## 6. Conclusão

Neste trabalho descrevemos, sucintamente, nossa pesquisa atual que envolve o desenvolvimento de um sistema audiovisual para criação de obras artísticas que utilizam como material básico grãos de som e de imagem.

Inicialmente, apresentamos uma breve história da Síntese Granular e de seu frequente desafio de automatizar os processos de organização dos grãos e ao mesmo tempo manter a flexibilidade de possibilidades sonoras desta técnica. Durante duas décadas de pesquisa, foram propostos diversos modelos computacionais de alto-nível, cada qual com suas características específicas.

Em seguida introduzimos a Música Visual, que não se trata de um modelo computacional, mas de uma área estética dentro das artes audiovisuais, e detalhamos como foi possível criar um sistema para Síntese Granular no qual o feedback visual se apresenta como auxílio no controle em tempo real da síntese ao mesmo tempo em que adiciona valor expressivo à obra.

Com a criação da obra “Contrapontos”, um estudo de caso em Síntese Granular Audiovisual, validamos a aplicabilidade tanto do modelo formal que denominamos Música Visual Granular quanto dos sistemas “*GranularStreamer*” e “*GranularDrawer*”, na produção de arte.

Nas últimas décadas, correlações audiovisuais em tempo real foram tratadas por diversos pesquisadores e artistas, ainda assim a questão permanece em aberto. Com este trabalho foi proposto um modelo integrado de correlação sintática e semântica entre objetos visuais e de áudio. Novas formas de interpretar estas correlações entre som e imagem trazem contribuições para a área de Computação Musical, seja na elaboração de sistemas para produção de som e imagem ou mesmo no projeto de interfaces gráficas alternativas.

## Referências

- Batty, J., Horn, K., Greuter, S. (2013) “Audiovisual granular synthesis: micro relationships between sound and image”, Proceedings of The 9th Australasian Conference on Interactive Entertainment: Matters of Life and Death, Melbourne.
- Batty, J. (2014) “Audiovisual granular synthesis: creating synergistic relationships between sound and image”, Doctor of Philosophy (PhD), Media and Communication, RMIT University.
- Callear, S. (2011) “Audiovisual Particles: Parameter Mapping as a Framework for Audiovisual Composition”, Doctor of Philosophy (Ph.D.), Music and Performing Arts, Bath Spa University.
- Collopy, F. (2000) "Color, Form and Motion: Dimensions of a Musical Art of Light", Leonardo, Vol.33, No.5, pp.355-360.
- DiScipio, A. (1990) "Composition by Exploration of Nonlinear Dynamical Systems", Proceedings of the 1990 International Computer Music Conference. San Francisco: International Computer Music Association, pp.324-327.
- Edmonds, E. A., Martin, A. e Pauletto, S. (2004) "Audiovisual Interfaces in Digital Art", ACE 2004, Singapura.

- Gabor, D. (1947) "Acoustical Quanta and The Theory of the Hearing", Nature 159. pp.591-594.
- Haverkamp, M. (2009) "Look at that Sound: Visual Aspects of Auditory Perception", III Congresso Internacional de Sinestesia, Ciencia y Arte, Granada.
- Jones, R. e Nevile, B. (2005) "Creating Visual Music in Jitter: Approaches and Techniques", Computer Music Journal, Vol.29, No.4, pp.55-70.
- Kohlrausch, A. e Van de Par, S. (2005) "Audiovisual Interaction in the Conext of Multi-Media Applications", In: Communication Acoustics edited by Blauert, J. Springer, Berlin, Heidelberg, pp.109-138.
- Maia, A. e Miranda, E.R. (2005) "Granular synthesis of sound through Markov chains with fuzzy control", Annals of the International Computer Music Conference (ICMC). Barcelona.
- Mahnke, F. H. (1996) "Color, Enviroment and Human Response", John Wiley and Sons Inc., pp.71-77.
- Roads, C. (2001) "Microsound", Cambridge: MIT Press.
- Souza, F. F. e Maia JR., A. (2009) "Evolutive Processes for Granular Synthesis in Extended Gabor Space", Anais do 7o Congresso de Engenharia de Audio - AES Brasil: São Paulo.
- Stockhausen, K. (1957) "...How time passes....", Die Reihe vol.3. Vienna: Universal Edition.
- Truax, B. (1988) "Real-time Granular Synthesis with Digital Signal Processing Computer", Computer Music Journal 12(2). pp.14-16.
- Valle, A. e Lombardo, V. (2003) "A two-level method to control granular synthesis", Proceedings of the XIV colloquium on musical informatics. Firenze.
- Xenakis, I. (1971) "Formalized Music: Thought and Mathematics in Composition", Hillsdale: Pendragon Press.
- Zielinsky, S. (1999) "Audiovisions: Cinema and Television as Entr'acts in History", Trad. Gloria Custance, Amsterdam University.

# Expert systems and genetic algorithms applied to the twelve-tone musical composition

Anderson Rodrigo Barretto Teodoro<sup>1</sup>, Flávia Toledo Martins Bede<sup>2</sup>, Denílson Paulo Souza dos Santos<sup>2</sup>, Paulo André Lima de Castro<sup>1</sup>

<sup>1</sup>Technological Institute of Aeronautics (ITA)  
São José dos Campos, SP – Brazil

<sup>2</sup>National Institute for Space Research (INPE)  
São José dos Campos, SP – Brazil

andersonrbteodoro@gmail.com, toledomartins@gmail.com,  
denilson.paulo@gmail.com, paulo.al.castro@gmail.com

***Abstract.** We develop a prototype system in this work using an artificial intelligence technique through genetic algorithms, for musical composition within the twelve-tone serial system. The prototype was developed seeking to create compositions with melody and harmony within the atonal system, trying to keep the melodic coherence in the works. We create a set of rules inspired by expert systems techniques used in objective function, seeking the best intervals combinations to prevent sudden dissonance conflicts, maintaining the harmonic coherence and leaving only the atonal characteristic in evidence.*

## 1. Introduction

The creativity is one of the main requirements for an artistic work creation in general. Among many ways to express art, music stands out as one of the most widespread. Musicians and researchers of the artificial intelligence fields and computer music still have a big challenge to create new compositions of melodies through computers, because this involves emotion and creativity (Macdonald et al., 2006).

The relevance of the study was also observed in one of the objections provided by Turing, addressed in Russell and Norving (2004). This objection was quoted by Professor Geoffrey Jefferson, which states that only when a machine has the ability to write a musical composition, thinking and feeling emotions and not by the provision of the sounds at random, it would be possible to agree that the machine could equate to the brain (Russell and Norving, 2004).

In this work we propose a technique that born on the brain, not in the heart or ear. This approach can deal with an infinite number of compositions through a twelve-tone structure and create a twelve-tone composition taking care of some harmonic structures, leaving only the twelve-tone characteristic in evidence. The technique was implemented using Java, with jMusic library (Brown, 2005).

## 2. Methodology

We create a database with many rhythm structures to deal with the rhythmic values. The next step was coding the twelve-tone series using to perform the population for the genetic algorithms (Holland, 1992), that is developed looking for the best combination for all possible series.

The serial system was built based on the twelve-tone system. This series was constructed with the goal of establishing a structure for melodic ideas to form the basis for the creation of new compositions, in accordance with the serialism rules. From these series all the base of ideas for the compositions is done (Whittall, 2008).

We considered using genetic algorithms in order to select the best serie's combination from the twelve-tone system using this for melody and harmonization. The codification of the serial system considered each series of the twelve-tone system as an individual. We considered a binary codification, where each individual is encoded by its horizontal and vertical position on the system, as illustrated in Figure 1.

We created the complete serial system using the first two bits representing the reading position of individuals in the matrix and the others bits representing each individual position following the series. The individual's quantity on the population was considered constant throughout the evolutionary process. This means that for each new individual generated, another should be eliminated from the population.

	00	0000		00	0001		00	0010		00	0011		00	0100		00	0101		00	0110		00	0111		00	1000		00	1001		00	1010		00	1011
11	0000	G4	C4	E4	D4	B4	A4	CS4	FS4	AF4	F4	EF4	BF4	10	0000																				
11	0001	D4	G4	B4	A4	FS4	E4	GS4	DF4	EF4	C4	BF4	F4	10	0001																				
11	0010	BF4	EF4	G4	F4	D4	C4	E4	A4	CF4	AF4	FS4	DF4	10	0010																				
11	0011	C4	F4	A4	G4	E4	D4	FS4	B4	CS4	BF4	GS4	EF4	10	0011																				
11	0100	EF4	AF4	C4	BF4	G4	F4	A4	D4	E4	DF4	B4	FS4	10	0100																				
11	0101	F4	BF4	D4	C4	A4	G4	B4	E4	FS4	EF4	CS4	GS4	10	0101																				
11	0110	DF4	FS4	BF4	GS4	F4	EF4	G4	C4	D4	B4	A4	E4	10	0110																				
11	0111	GS4	DF4	F4	EF4	C4	BF4	D4	G4	A4	FS4	E4	B4	10	0111																				
11	1000	FS4	B4	EF4	DF4	BF4	AF4	C4	F4	G4	E4	D4	A4	10	1000																				
11	1001	A4	D4	FS4	E4	CS4	B4	DS4	GS4	AS4	G4	F4	C4	10	1001																				
11	1010	B4	E4	GS4	FS4	DS4	CS4	ES4	AS4	C4	A4	G4	D4	10	1010																				
11	1011	E4	A4	CS4	B4	AF4	GF4	BF4	EF4	F4	D4	C4	G4	10	1011																				
	01	0000		01	0001		01	0010		01	0011		01	0100		01	0101		01	0110		01	0111		01	1000		01	1001		01	1010		01	1011

Figure 1. Codification scheme for the individuals within the twelve-tone series.

The individual's performance was evaluated using an objective function inspired by the expert systems techniques. In this evaluation, the best combinations of intervals were selected to avoid sudden conflicts of dissonance, maintaining the melodic coherence and leaving only the atonal nature of the twelve-tone serialism in evidence. Thus, it was created a set of rules, responsible for assigning the fitness at every crossing-over operation between the individuals.

### 3. Results

We developed a web system prototype in order to generate the compositions. This platform works from basic input parameters that describe the basic information about the composition. These parameters are set from the composition name, amount of bars and style of the work. The developed system provides the option to write songs in jazz, sacrum, canon and minuets style. The amount of bars is variable according to the user's inspiration in each composition.

The first compositions were generated by a random selection of the series obtained by the twelve-tone system. Figure 2 shows a score fragment with the selected series and the indications of critical intervals in this work, where the red rectangles indicate a critical situation and the yellow ones should be avoided. After, we applied the genetic algorithm to deal with this scenario, as illustrated in Figure 3 and Figure 4.

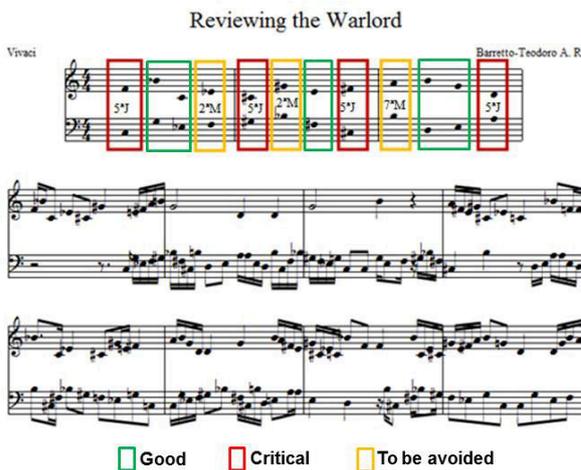


Figure 2. Composition fragment created from random series.

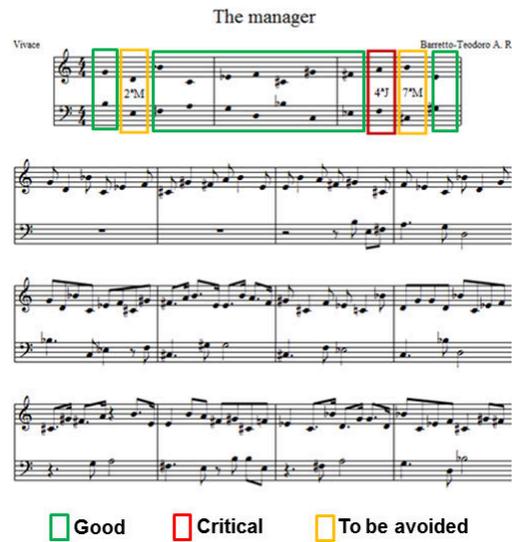


Figure 3. Composition fragment created by the genetic algorithm selection.

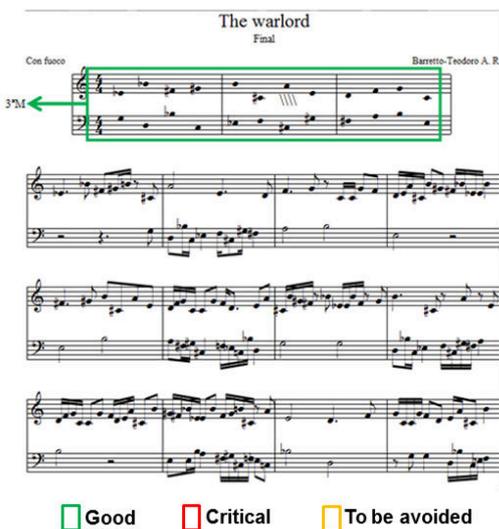


Figure 4. Best selection results obtained by the genetic algorithms.

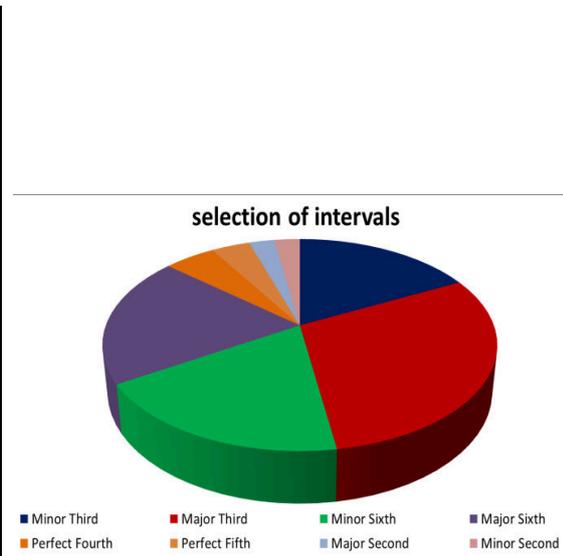


Figure 5. Number of occurrences for each interval.

The results confirm the good functioning of the genetic algorithms for the series' choice in the twelve-tone system and many of these results reached the good fitness desired as we illustrated in Figure 4. The total occurrences for each interval through the genetic algorithm selections are illustrated in Figure 5. We collected all the compositions generated using this algorithm and created the "Undergraduate Collection" album.

#### 4. Conclusions

We presented in this paper a technique that is able to create musical compositions with melody and harmony from the serialism techniques using an artificial intelligence methodology, by genetic algorithms and expert systems. This work demonstrates that it is possible to create new compositions through computers and mix known elements, from a knowledge base. The application of artificial intelligence methods through genetic algorithms and expert systems at the atonal scenario were essential for best results.

The results obtained from the rigid rules of tonality can be more efficient in some harmonic situations. However, serialism has two major advantages: flexibility and creative freedom of creation. The proposal to work with the atonal system, through serialism, applying artificial intelligence techniques, expands the horizon of possibilities to create new compositions. This characteristic gets stronger by the fact that this new musical proposal born in the brain, not in the heart or ears.

We intend to extend this technique to others styles of compositions, like: waltzes, blues, tango and others. Another important task will be dealing with a bigger set of rhythms structure, working with compasses in quaternary, binary, ternary and both in simple, as in compounds.

#### 5. Acknowledgments

We thank the CNPq for financial support and we are grateful to Samuel Soares Oliveira, because this project owes much of his musical guidance besides of his useful comments and suggestions.

#### References

- BROWN, A. R. **Making music with Java: An introduction to computer music**, java programming and the jMusic library. Andrew R. Brown, 2005.
- HOLLAND, J. H. **Genetic algorithms**. Scientific American, v. 267, n. 1, p. 66-72, 1992.
- MACDONALD, R.; BYRNE, C.; CARLTON, L. **Creativity and flow in musical composition: An empirical investigation**. Psychology of Music, v. 34, n. 3, p. 292-306, 2006.
- RUSSELL, S; NORVIG P. **Artificial Intelligence: A Modern Approach**. Prentice Hall, Englewood Cliffs, NJ (1995).
- WHITTALL, A. **The Cambridge Introduction to Serialism**. Cambridge, UK; New York: Cambridge University Press, 2008.

# Gest'Ação I: A hybrid computer music guitar performance mediated by instrumental gestures

Jorge L. de L. Santos<sup>1</sup>, José Fornari<sup>2</sup>

<sup>1</sup>Instituto de Artes – Universidade Estadual de Campinas (UNICAMP)  
Campinas – SP – Brazil

<sup>2</sup>Núcleo Interdisciplinar de Comunicação Sonora (NICS) – UNICAMP  
Campinas – SP – Brazil

{jorgelsantos02, tutifornari}@gmail.com

**Abstract.** *With all computer music improvements in the last decades, musical gestures started to be used as parameter for controlling dynamic processes that compound a musical performance. This article intends to describe the processes and methods employed in a composition of this type: "Gest'Ação I" for classical guitar and live electronics which uses a set of 'instrumental gestures' for the guitar as part of the musical performance. The intent here was to elaborate a musical piece that could join, in a free process, the idea of 'extended technique' to the "writing/instrument extended" using open-sourced musical software. This article briefly discusses the concept of "gesture" as defined by Wanderley (2000) and Delande (1998), especially concerning the guitar performance. It is also discussed the concepts of "extended technique" and "extended instrument" as defined by Padovani and Ferraz (2011). Here two open-source softwares were used: MuseScore2.0 (for the score writing) and Pure Data (for the audio processing in real time).*

## 1. Introduction

Since the rising of solid state electronics (with the invention of the transistor, in 1947), which made possible controlling the electric stream throughout another electric stream, a significant advance has been made in technology and communication. Amongst many areas of development, in 1990's, it has emerged a computational platform to create musical algorithms; the Pure Data (Pd), initially developed by Miller Puckette. Pd is distributed freely and developed by a community that has expanded it to a visual programming language of performative multimedia processes ([www.puredata.info](http://www.puredata.info)). It is currently possible to explore performative processes of computer music in real time using several types of sensors - which is also known by the term "live electronics". Among many examples of such type, there are the "hyperinstruments" (MACHOVER, 1989), which use traditional musical instruments coupled to "live electronics", to operate in a hybrid manner, in which the acoustic sound generated by the instrument also controls in real time the audio and/or synthesis process. This article reports a similar development; describing the process and methods employed in a composition of hybrid computing and instrumental music, named "Gest'Ação I", for guitar and live electronics. Starting from an inventory of "instrumental gestures", already established for guitar, we sought to elaborate a piece that unifies, in a free approach, the concept of "extended technique" to the concept of "extended écriture/instrument" making use of open software of music.

## 2. Instrumental gestures and the guitar gestures

The term "gesture" is directly related to "movement". When we think about our daily

life, the "natural gestures" are movements which depend on a series of external factors such as intention, attention and adequacy. In music, the use and function of gestures vary significantly and comprehend a wide scale of practices and meanings for the composer, for the performer and for the listener (WANDERLEY, 2000).

François Delande (1998) was the first one to create a categorization of gestures related to musical instruments after his observations about the instrumental practices of the pianist Glen Gould. This researcher elaborated the following categories: 1) Effective gestures: those related to sound production; 2) Accompaniment gestures: body movements that accompany the performance itself. (Ex.: movement of feet or head); and 3) Figurative gestures: gestures perceived by the listener and that has no straight correspondence to the performer's movements (Ex.: articulations, melodic variations). From that taxonomy, this study deals specifically with "effective gestures", among which it is found the "instrumental gestures". Those functions take place in the "gestural channel" where occurs the receiving and sending of information. On the guitar, most of the gestural channel is in the hands. Wanderley (2000) summarizes the concept of "instrumental gesture" as a specific modality of the gestural channel, complementary to the "empty hand gestures" and featured as follows: 1) It is applied to material objects and has a physical interaction with it; 2) On the physical interaction, specific phenomena are produced, where formal and dynamic development can be mastered by individuals; 3) These ones can become the basis for communicational messages and/or the basis for the material production of action.

Jonathan Norton (TORRES, 2011) explains that the fundamental principles of the guitar playing are basically the same, whatever the genre, and could be classified in: (1) instrument's supporting, (2) left-hand functioning, and (3) right-hand functioning. Among the three aspects, we are interested in those related more directly to the instrumental gesture. In relation to the left and right hands, the author defines three kinds of "gestural articulation": a) Attacks: articulations involved in originating the sound; b) Sustains: articulations involved in the intermediate phase of the sound; and c) Release: articulations involved in ceasing the sound.

### **3. Extended techniques and extended writing/instrument: the musical role of computers**

In a broader sense, in contemporary music, extended techniques could be defined as unusual techniques, i.e., ways of playing an instrument, or singing, that explores instrumental gestural and sounding possibilities that are uncommon or sparsely used, in a specific historic and aesthetic context (PADOVANI, FERRAZ, 2011). Especially in the second half of 20th Century, the practice of extended techniques started to have a more clear definition and systematic application. It was noticeable the construction of new paradigms around this subject, on one hand, aiming for the creation of complex and new sonorities, such as multiphonics or microtonal sounds; and on the other hand, as a compositional approach strongly focused on the instrumental mechanism and the gestural possibilities of the instrumentalists (IBID, p. 26). For the purpose of the piece here proposed, there are computational resources applied to the creation and performance as an interactive systems. In this context, a computer model can be created to track this performance which controls a sound synthesis processing in real time according to specific contexts of the piece. The most popular computer music software are, among others: Max/MSP, SuperCollider, Csound. These computing environments allow the possibility of creating a computer model that extend technically the traditional instruments in inventive and unpredictable ways (PADOVANI, FERRAZ, 2011).

### **4. Open software: real time audio processing and musical notation**

Most of softwares are developed commercially, and so, subordinated to the financial and market constrictions. Nonetheless, the last decades have seen a growing number of

software developed by volunteers and distributed for free. These are called free software and their code is normally opened. In this paper, we employ two open-source softwares, which are described as follows:

#### 4.1 Musescore

For the writing of the score "Gest'Ação I", we used the musical score editor Musescore. Released under Creative Commons Attribution-ShareAlike, this software is free and also open to the contribution of all. The version 2.0, released in 2015, reached great advances in capacity and quality. This software has proven to be a viable alternative to the properties notation software, more renowned, such as the case of Sibelius and Finale. Despite some clear advances in this new version, Musescore presented some challenges to the less conventional notation which was the kind we proposed in the piece here described. The program still lacks a wider set of graphic symbols for contemporary writing. Although it has a great number of symbols they are not flexible, not allowing certain fitting to some needs of writing contemporary pieces. One simple example is the lack of lines with arrows, or the possibility to manipulate the thickness of lines available. Fig. 1 shows an excerpt that could be better notated if there were the possibility to manipulate the direction, format and thickness of the lines and arrows.

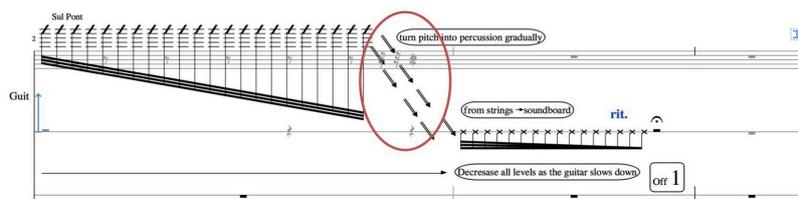


Figure 1. Excerpt of "Gest'Ação I" edited on Musescore 2.0

#### 4.2 Pure Data

Pd's programming exists by the manipulation and association of objects without the necessity of line codes writing. The algorithmic structures created with the objects are named "patches" and are placed on a screen named "canvas". The "patch" developed for the piece "Gest'Ação I" (Fig. 2) is composed of four modules that are executed and manipulated throughout the piece. The first one is a "reverb" which includes also the resource of "freeze reverb" in which in a certain chosen moment of time (a time window of milliseconds) the sound, and that effect acts as if being frozen, generating something like a continuous of that acoustic moment. The second module is composed essentially by two oscillators that receive the signal and allow the manipulation of pitch, velocity and depth. The third module simulates an usual effect in classical guitar, notwithstanding restrict to a very specific technique on the instrument; the tremolo. In this case, the effect could be applied to any kind of sound generated on the instrument, re-setting the notion itself of that specific instrumental gesture. Finally, the fourth module is a "spectral delay". The term spectral refers to the audio frequency spectrum, i.e., the distribution of frequency partials that compound each instant of sound.

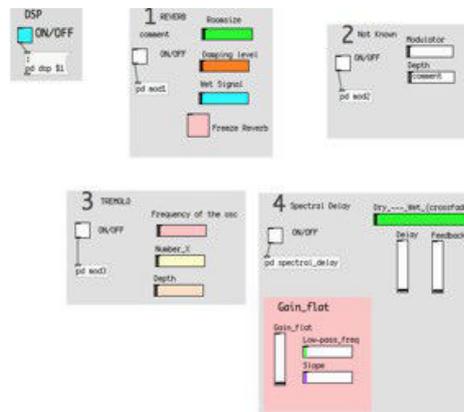


Figure 2. Patch of "Gest'Ação I" created on Pure Data Extended

## 8. Conclusion

This article presented the steps of development and conceptualization of the musical piece entitled “GestAção I” for classical guitar and live electronics. It is possible to understand this composition throughout two main perspectives, which we intended to relate at the end. On one hand, a very specific characteristic of the classical guitar – its gestures – is re-elaborate as part of its instrumental extended technique. On the other hand, there are a myriad of possibilities available by the computer music which make possible an actual timbre re-invention as well as the expansion of the musical context in which the instrument could be employed. We sought to use two open software tools to the pursuing of these goals, considering that the use of open software tools also points out to a new ideological paradigm and strategic path in the contemporary world where, each time more, we become more dependent of technology and, at the same time, a great inequality still remains in the access of this resource.

## References

- DELALANDE, F. La gestique de Gould, éléments pour une sémiologie du geste musical. In: G. Guertin (Ed.), *Glenn Gould pluriel* (pp. 85–111). Montreal, Québec, 1988
- FERRAZ, Silvo. PADOVANI, José Henrique. Proto-história, evolução e situação atual das técnicas estendidas na criação musical e na performance. *Música Hodie*. Vol. 11 -Nº2 – 2011
- MACHOVER, Tod; Chung, J. “Hyperinstruments: Musically intelligent and interactive performance and creativity systems” In *Proc. Intl. Computer Music Conference*. 1989.
- TORRES, Heber Manuel Pérez. An Analysis to the Guitar Lab's gesture acquisition prototype with the aim of improving it. *Universitat Pompeu Fabra*. Master thesis, 2011.
- WANDERLEY, Marcelo. *Non-obvious Performer Gestures in Instrumental Music*. Paris, 2000. Available at: <<http://recherche.ircam.fr/anasyn/wanderle/Gestes/Externe/GW99F.pdf>> Accessed in: June 12<sup>th</sup> 2015.

# Music Compositions

## Bandas Críticas(2015)

Ariane Stolfi

Programa de Pós-Graduação em Música

Universidade de São Paulo (USP)

arianestolfi@gmail.com

duração 9:30min

### 1. Resumo

A peça explora as frequências centrais e limítrofes das bandas críticas da audição, sobrepondo harmônicos e criando dissonâncias a partir de sons fora das escalas musicais tradicionais. Surge de um processo de composição metalinguístico, de investigar a própria percepção acústica, em um desejo de materialização de conceitos psicoacústicos em objetos sonoros.

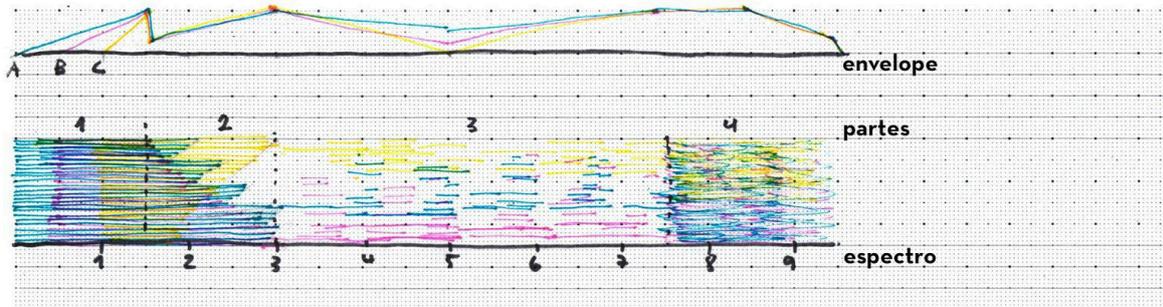


figura 1 - partitura gráfica

Surge de um processo de composição metalinguístico, de investigar a própria percepção acústica, em um desejo de materialização dos conceitos psicoacústicos em objetos sonoros. No caso, representa uma materialização do conceito de bandas críticas, buscando explorar sensações audíveis e artefatos emergentes da exoração desse conceito. A performance é programada para 3 músicos, cada um conectado a uma caixa de som diferente, para enfatizar as diferenças entre as frequências exploradas.

A peça é baseada em 44 osciladores, vibrando nas frequências centrais e limítrofes das bandas críticas da audição humana. A idéia é criar uma saturação baseada em sons puros e com o mínimo de mascaramento. A peça se divide em 4 partes: na primeira, um contínuo linear com todos osciladores ligados, vibrando uma onda senóide pura, onde os músicos vão entrando sucessivamente. Após esse contínuo, acontece uma queda abrupta e então os músicos começam a desligar os osciladores um a um. Cada músico desliga os osciladores de uma forma determinada, de modo a restar apenas um pequeno número de geradores ligados. Na parte seguinte, os músicos tem liberdade para improvisar a partir de suas frequências determinadas, acrescentando e retirando harmônicos à suas frequências de base, ou ligando e desligando geradores dentro de uma faixa previamente definida. Como as frequências não correspondem as notas da escala temperada, esse processo gera uma série de artefatos sonoros, bem como consonâncias e dissonâncias não convencionais. Ao final, os músicos passam a alterar manualmente o timbre através do desenho da onda, processo que acrescenta camadas de ruído ao som que era originalmente puro. O espaço espectral se satura novamente, mas desta vez com um som mais

ruidoso. Os músicos continuam a desenhar as ondas de amplitude cada vez mais baixa, até chegarem a uma linha reta próxima ao zero. (figura 1). A peça explora o uso de um *patch* (figura 2) desenvolvido em pure data especialmente para ela, organizado em três módulos. O primeiro consiste em uma série de osciladores baseados em tabelas (tipo tabosc), disparados por objetos que reconhecem a posição do mouse, que permite que os geradores podem ser ligados e desligados com um simples passar do cursor.

Os geradores são relacionados com uma tabela dinâmica, que está configurada para receber senóides e 16 harmônicos, mas que também pode receber um input direto através do desenho da onda. Um terceiro módulo faz o controle geral da dinâmica dos envelopes de amplitude sonora, que variam de formas lineares ao longo da peça, e dá indicações para os músicos das partes da música.

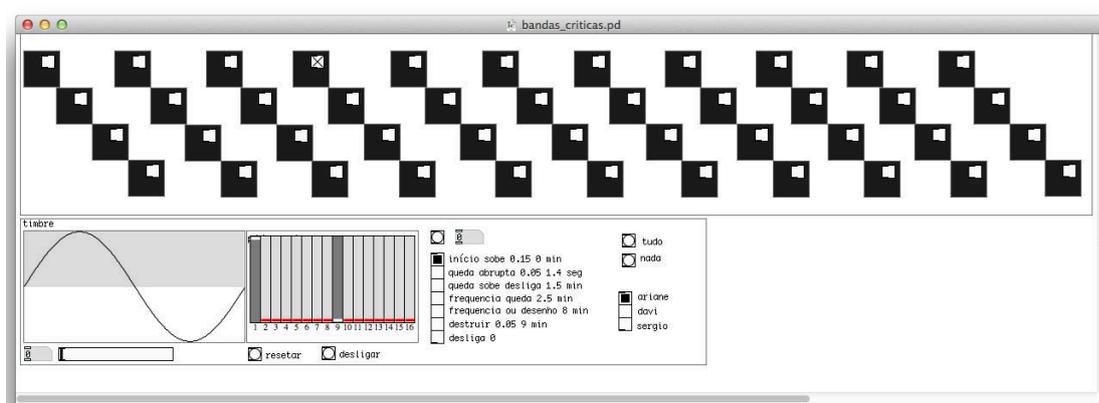


figura 2 – interface do patch bandas críticas

No simpósio, iremos utilizar um patch específico em pd/gem para um acompanhamento gráfico visual, que irá explorar relações diretas entre som e imagem a partir de análise das frequências sonoras em tempo real.

## 2. Biografia da compositora

Ariane Stolfi é arquiteta, compositora e programadora e transita por várias linguagens. É mestre em design e arquitetura pela FAU-USP e doutoranda em Sonologia pela ECA-USP, com orientação do professor Fernando Iazzetta. Foi editora da revista *Contravento* e organiza o portal de música experimental *finetanks.com*. Desenvolve interfaces interativas em HTML e Pure Data, como o "hexagrama essa é pra tocar" para a exposição *Gil70*, em parceria com Gabriel Kerhart e Daniel Scandurra e a performance "cromocinética", com o coletivo 24h. Participou dos festivais *Submidialogias* (2010), *#Dis Experimental* (2011) e da *Virada Cultural* (2012) e atualmente está desenvolvendo um sistema para a digitalização da revista de poesia concreta *Código*, projeto contemplado pelo Rumos do Itaú Cultural.

Produz música eletrônica desde 2005, baseada principalmente em princípios metalinguísticos, tendo organizado um álbum "Protomúsica" lançado de maneira independente pelo selo *finetanks*. Desde 2010, tem se dedicado à produção de música experimental baseada em pure data, desenvolvendo patches autorais voltados principalmente para a performance de improvisação livre.

# “CONTRAPONOTOS”

**Fernando Falci de Souza**

Instituto de Artes – Universidade Estadual de Campinas (UNICAMP)  
Cidade Universitária Zeferino Vaz - Campinas – SP – Brasil

Núcleo Interdisciplinar de Comunicação Sonora (NICS)  
Universidade Estadual de Campinas (UNICAMP)

fernando\_ffs@yahoo.com.br

## 1. Resumo

Contrapontos é um dos resultados artísticos de um projeto que une duas áreas de pesquisa, a Síntese Granular, um método de síntese sonora no qual sons, timbres e texturas são obtidos a partir da organização de uma grande quantidade de partículas sonoras de conteúdo espectral simples e de curta duração chamados de “grãos de som”, e a Música Visual, uma estética particular dentro das artes audiovisuais cujo conteúdo imagético é abstrato e os elementos de expressão são a forma, cor e movimento. Neste sentido, criamos uma extensão da Síntese Granular do domínio sonoro para o domínio visual e especificamos tipos de correspondências entre grãos de som e grãos de imagem, desta forma definindo um modelo para a Síntese Granular Audiovisual.

Nesta obra utilizamos tal modelo para construir um discurso de texturas sonoras e visuais a partir de uma biblioteca de grãos de som e de imagem que inclui tanto formas abstratas ou sintéticas quanto concretas e amostradas, apresentando contrapontos que favorecem a reflexão sobre os sentidos da audição e da visão, sobre o som e a imagem, o junto e o separado, o sintético e puro e o amostrado e ruidoso, e que também propõe questionamentos sobre, a vida na imensidão de uma grande cidade e a vida no confinamento de veículos e apartamentos, a liberdade e os padrões sociais. Contrapontos é uma obra audiovisual aberta, em constante evolução. Em cada performance novos elementos são adicionados e combinados de maneira única. A obra é executada em tempo real utilizando sistemas desenvolvidos em JAVA durante nossa pesquisa. A duração aproximada é de 15 minutos.

## 2. Biografia do Autor

Fernando Falci de Souza é doutorando em música pela Universidade Estadual de Campinas, atualmente desenvolve pesquisa em busca de uma extensão visual para o método de síntese sonora chamado Síntese Granular, localizando-se na interdisciplinaridade que envolve diversas áreas como a música, arte sonora, arte audiovisual e uso o criativo de novas tecnologia. É Mestre em Música pela Universidade de Campinas com foco em Música Computacional, e possui graduações em Ciências da Computação pela Universidade de São Paulo e Música Popular pela Faculdade Santa Marcelina.

### **3. Especificações Técnicas**

A obra será executada em tempo real pelo autor. Será necessário um projetor com resolução mínima de 800 x 600, conexão através de cabo VGA ou HDMI, superfície de projeção (tela) e sistema de som estereofônico com possibilidade para conexão no computador através de um conector “p2 estéreo”.

## *Unangebracht* (2015)

**Darwin Pillar Corrêa**

Programa de Pós-Graduação em Música  
Universidade Federal do Rio Grande do Sul (UFRGS)

darwincor7@gmail.com

### **1. Resumo**

*Unangebracht* é uma peça acusmática em 8.0 canais. Foi produzida no contexto do Centro de Música Eletrônica da UFRGS, como parte dos estudos realizados na disciplina Música e Tecnologia ofertada pelo PPGMUS-UFRGS. Contando com as instalações de som da Sala dos Sons, a principal sala de concertos de música eletroacústica do CME-UFRGS, foi possível explorar as possibilidades técnicas e de espacialização da música acusmática em 8.0 canais, através de um processo composicional que integra a investigação que está sendo realizada pelo autor durante seu mestrado em composição. Nessa investigação, busca-se - através da exploração de distintas temporalidades, interrupções e quebras abruptas no discurso musical - meios para criar trajetórias musicais marcadas pela impermanência dos eventos.

Para *Unangebracht* a intenção foi trabalhar com uma fonte sonora instrumental que é familiar ao compositor, o violão clássico, buscando evitar conotações miméticas relativas ao som tradicional do instrumento, com vistas a realizar um discurso musical aural integrado ao foco da investigação composicional. Como interface de trabalho foram utilizados os recursos do software *Nuendo* e plug-ins *GRM Tools*.

O processo composicional de *Unangebracht* iniciou com a gravação em estúdio de um catálogo de sons composto por improvisações livres, do próprio compositor, tocadas ao violão clássico com a utilização de técnicas tradicionais e estendidas. Essa definição teve como objetivo a obtenção de objetos sonoros variados e complexos, e também a obtenção de gestos musicais de maior duração que poderiam servir de base para seções da obra. Gestos rápidos em *rasgueado* com as cordas abafadas, percussão de diferentes partes do instrumento e dedilhado em diferentes regiões das cordas foram algumas das estratégias empregadas nas improvisações livres. A etapa seguinte correspondeu ao trabalho de organização e seleção do catálogo de sons, que consistiu na escolha de trechos das improvisações a serem trabalhados na composição da peça. Os materiais selecionados passaram por um processo de manipulação sonora realizada através de plug-ins *GRM Tools* buscando principalmente modificações de timbre e criação de massas sonoras.

### **2. Biografia do Autor**

Darwin Pillar Corrêa é acadêmico do curso de Mestrado em Música, Área de Concentração: Composição, do Programa de Pós-graduação em Música da UFRGS, sob orientação de Celso Loureiro Chaves. Estudou Composição Eletroacústica com Eloy Fritsch, Stephen David Beck e Amaro Borges. Suas obras têm sido apresentadas no Rio Grande do Sul.

### 3.Especificações técnicas para apresentação

Esta obra requer um sistema de áudio com 8.0 canais. Não é necessária mixagem em tempo real. A montagem utilizada durante o processo composicional de *Unangebracht* está representada na **Figura 1**:

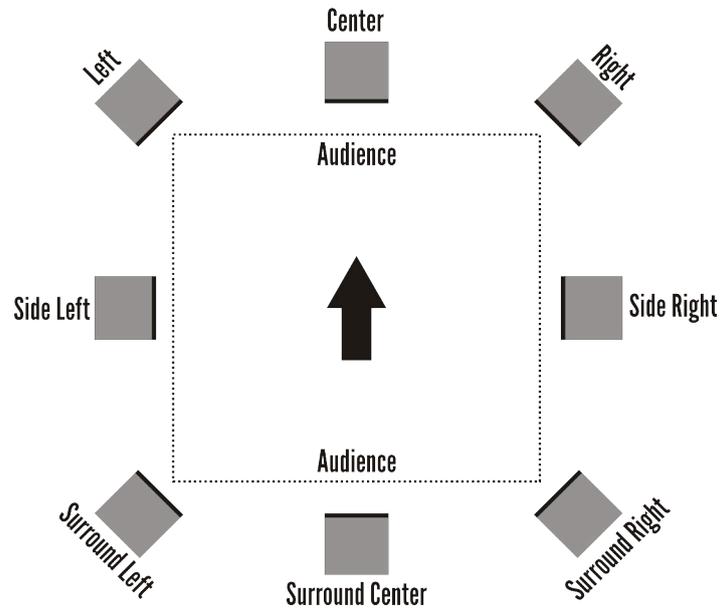


Figura 1. Exemplo de montagem para apresentação de *Unangebracht*

## Author Index

- Aliel, Luzilei, 169  
 Antunes, Micael, 68  
  
 Barbosa, Jeronimo, 3  
 Bedê, Flávia de Toledo Martins, 247  
 Borges, Raphael, 13  
  
 Cabral, Giordano, 25, 142, 146  
 Camporez, Higor, 127  
 Casthelo, Luan, 76  
 Corrêa, Darwin Pillar, 260  
 Costa, Rogerio, 169  
 Costalonga, Leandro, 76, 127  
 Coura, Daniel, 76  
  
 da Rosa, Tiago Brizolara, 213  
 de Castro, Paulo André Lima, 247  
 de Paula, Hugo, 13  
 Dias, Luca, 146  
  
 Faria, Regis, 123  
 Fornari, José, 72, 150, 193, 251  
 Fujinaga, Ichiro, 3  
  
 Garcia, Denise, 201  
 Gasco, Aline, 123  
 Gonçalves, Clenio Jr., 138  
 Gonçalves, Luan, 60  
 Goulart, Antonio, 68  
  
 Haeser, Estevao, 107  
 Homem, Murillo, 138  
  
 Johann, Marcelo, 84, 111  
  
 Keller, Damian, 169  
  
 Laia, Marcos, 48  
 Loureiro, Maurício, 13, 36  
 Lunhani, Guilherme, 154, 225  
  
 Madeira, Daniel, 48  
 Magalhaes, Tairone, 36  
 Mamedes, Clayton, 201  
 Manzolli, Jônatas, 96, 181, 201  
 Mateus, Ariane, 76  
 Mckay, Cory, 3  
 Meneses, Eduardo, 193  
 Miletto, Evandro, 107  
 Monticelli, Patrícia, 123  
 Moritz, Marcos, 213  
 Moroni, Artemis, 201  
 Mota, Davi, 36  
 Mota, Trajano, 127  
  
 Netto, Ayer, 76  
  
 Olios, Amanda, 76  
 Oliveira, Aluizio, 36  
  
 Padovani, José H., 96  
 Paulo, Avner, 48  
 Penteado, Antonio, 72  
 Pessoa, Leandro, 119  
 Pimenta, Marcelo, 84, 111  
 Pirotti, Rodolfo, 111  
  
 Ramalho, Geber, 25, 142, 146  
 Rimoldi, Gabriel, 181  
 Rocha, Helder, 127  
 Rodrigues, Renato, 25  
 Rolla, Vitor, 115  
 Rossetti, Danilo, 159  
  
 Santo, Guilherme, 68  
 Santos, Alexandre, 150  
  
 Santos, Denilson Paulo, 247  
 Santos, Jorge, 251  
 Schiavoni, Flávio, 48, 60, 119, 154  
 Silva, Eduardo, 142  
 Souza, Fernando Falci, 235, 258  
 Stolfi, Ariane, 256  
  
 Teixeira, Euler, 13  
 Teodoro, Anderson Rodrigo Barretto, 247  
  
 Velho, Luiz, 115  
  
 Zattera, Vilson, 72, 150

## Institution Index

CENPRA, 201

Faculdade de Artes Alcântara Machado – FAAM, 68

Independente, 213

Instituto Federal do Rio Grande do Sul – IFRS, 107

Instituto Federal do Rio Grande do Sul / Campus Porto Alegre – IFRS/POA, 107

Instituto Nacional de Matemática Pura e Aplicada – IMPA, 115

Instituto Nacional de Pesquisas Espaciais – INPE, 247

Instituto Tecnológico de Aeronáutica – ITA, 247

McGill University, 3

PUC Minas, 13

Universidade de São Paulo – USP, 68, 123, 169, 256

Universidade de São Paulo – USP / FFCLRP, 123

Universidade Estadual de Campinas – UNICAMP, 72, 96, 150, 159, 181, 193, 201, 235, 251, 258

Universidade Estadual de Maringá – UEM, 213

Universidade Federal da Paraíba – UFPB, 96

Universidade Federal de Juiz de Fora – UFJF, 154, 225

Universidade Federal de Minas Gerais – UFMG, 13, 36

Universidade Federal de Pernambuco – UFPE, 25, 142, 146

Universidade Federal de São Carlos – UFSCAR / Sorocaba, 138

Universidade Federal de São João del Rei – UFSJ, 48, 60, 119, 154

Universidade Federal de Viçosa – UFV, 119

Universidade Federal do Acre, NAP – UFAC, 169

Universidade Federal do Espírito Santo – UFES, 76, 127

Universidade Federal do Rio Grande do Sul – UFRGS, 84, 111, 260



## Organizing Committee

ANA LIA DELLA TORRE  
GERSON BRANCO ABDALA  
GISELE TEREZINHA GOMES DA SILVA  
FRANCIELI MOREIRA  
ANGELICA CONTRERAS PEÑA  
MIGUEL RUBIO CLEMENTI  
DJALMA DE CAMPOS GONÇALVES JUNIOR  
SANDRA REGINA CIELAVIN  
RAFAEL BRIENZA DE ALMEIDA  
TIAGO FERNANDES TAVARES  
CHARLES DE PAIVA SANTANA  
GABRIEL RIMOLDI DE LIMA  
ADRIANA DO NASCIMENTO ARAUJO MENDES  
ELIZABETH FERNANDES  
ALEXANDRE HENRIQUE DOS SANTOS  
MATTEO RICCIARD  
JOSE EDUARDO FORNARI NOVO JUNIOR

## Scientific Committee

FLAVIO LUIZ SCHIAVONI  
FERNANDO WILLIAM CRUZ  
MARCIO DA COSTA PEREIRA BRANDAO  
MARCELO SOARES PIMENTA  
GIORDANO RIBEIRO EULALIO CABRAL  
RODRIGO SCHRAMM  
EDILSON FERNEDA  
LEANDRO LESQUEVES COSTALONGA

## SPONSORSHIP



## PROMOTION



[compmus.ime.usp.br/sbcm2015](http://compmus.ime.usp.br/sbcm2015)