



UNIVERSIDADE DE BRASÍLIA
PROGRAMA DE PÓS-GRADUAÇÃO MÚSICA EM CONTEXTO

DANIEL LIMA PANTOJA LEITE

**MODELAGEM DE PRINCÍPIOS CRIATIVOS: INVESTIGANDO A INTERAÇÃO
MÚSICO-COMPUTADOR POR MEIO DA IMPROVISACÃO MELÓDICA**

Brasília

2020

DANIEL LIMA PANTOJA LEITE

Modelagem de Princípios Criativos: Investigando a interação músico-computador por meio da Improvisação melódica

Dissertação apresentada ao Programa de Pós-graduação PPGMUS/ Música em Contexto da Universidade de Brasília-UnB para obtenção do título de Mestre em Música

Área: Processos e Produtos – Linha A

Orientador: Prof. Dr. Carlos Eduardo Vianna de Mello

Brasília

2020

Dissertação de autoria de Daniel Lima Pantoja Leite, sob o título “**Modelagem de Princípios Criativos: Investigando a Interação músico-computador por meio da Improvisação Melódica**”, apresentada ao Programa de Pós-graduação PPGMUS/ Música em Contexto da Universidade de Brasília-UnB, para obtenção do título de Mestre em Música, na área de concentração Processos e Produtos – Linha A, aprovada em 4 de dezembro de 2020.

Prof. Dr. Carlos Eduardo Vianna de Mello - PRESIDENTE

Instituição: Universidade de Brasília - UnB

Prof. Dr. Flávio Santos Pereira - MEMBRO do PPG-MUS/UnB

Instituição: Universidade de Brasília - UnB

Prof. Dr. Mauro Rodrigues - CONVIDADO

Instituição: Universidade Federal de Minas Gerais - UFMG

Àqueles que, pelo som, buscam re-significar o sentido de si próprios, recriando caminhos pela transgressão do caos e pela procura incessante das infinitas realidades que o silêncio engendra.

Agradecimentos

Aos professores do programa de Pós-graduação em Música da UnB: Prof. Dr. Carlos Eduardo Vianna de Mello por uma introdução aos conceitos básicos de Música e Tecnologia, pela motivação no projeto de desenvolvimento da Flauta MIDI, pela orientação interessada desta dissertação, bem como na elaboração e desenvolvimento do programa proposto; Profa. Dra. Delmary Vasconcelos de Abreu, por ser a “luz no fim do túnel”, dando novo suspiro a este programa de pós-graduação, a partir de sua coordenação comprometida; Prof. Dr. Flávio Santos Pereira pelas aulas de análise, pelo interesse no projeto e pela leitura atenta do texto de qualificação, no semestre em que co-orientou o projeto; Profa. Dra. Tatiana Olivieri Catanzaro, pelas críticas, sugestões ao texto e direcionamento da pesquisa na banca de qualificação; Prof. Dr. Daniel Junqueira Tarquínio por me receber na chegada a Brasília, ouvir meu projeto e me encaminhar para o meu orientador como sugestão, também pelas aulas de análise; Prof. Dr. Paulo Roberto Affonso Marins, pelas aulas do FOP - Fórum Orientado de Pesquisa, nos trazendo para a realidade e justeza que a pesquisa exige. Ao Prof. Dr. Alexandre Ricardo Soares Romariz do Dep. de Engenharia Elétrica da UnB, pelo esforço e disponibilidade efetivos em gerar uma parceria colaborativa entre os departamentos, oferecendo inclusive um bolsista para contribuir diretamente no desenvolvimento de nosso projeto.

Ao aluno da graduação e integrante do Lab. de Tecnologia Musical José Abreu, pelo apoio constante e imprescindível na elaboração do código e desenvolvimento do sistema interativo proposto, um dos objetos desta pesquisa.

Aos colegas do programa de Pós-graduação em Música da UnB, em especial: Ely Janoville e Thiago Rodrigues, por suas distintas colaborações, seja pelas discussões, seja pelas sugestões ao projeto; Givas Demore, pela contribuição direta e comentários na leitura atenta do artigo publicado pela revista TeMa/ 2019, e que corresponde ao cap. 1 e parte do cap. 2 desta dissertação; Carlinhos Veiga, Maria Clara, Wanderson Bonfim e Jhoninhas, por compartilharem aspectos de seus projetos, e fazerem sugestões enriquecendo em muito as trocas e experiências do grupo de pesquisa. Aos alunos da disciplina “Oficina de Construção & Improvisação com Instrumentos Eletrônicos”, por suas contribuições e “feedbacks” quanto ao sistema interativo proposto.

Felipe José pela dissertação de mestrado *O jogo musical Cobra (1984) de John Zorn: Reflexões sobre sua história e prática*, à qual, servindo de ponto de partida e fundamentação, foi muitíssimo importante para a realização da pesquisa que hora realizamos.

À CAPES e à DDS-UnB pela bolsa de estudos e pelo apoio, sem os quais esta pesquisa não teria sido possível.

Aos *Amigos, Musicus e Artistas sp*: Leandro Domith, Amigos do Estação Blues, O BancO, Cantares, amigos do Coral Mater Verbi, Laura Delgado e Contadores de História do Granbery, Marcos Marinho e Mezcla de JF, Fernando Tubarão (Guanabara), Gil e Thiagy Das em Londres/ UK, do grupo RAMO, Misturada Orquestra, Felipe José, Marcelo Chiaretti, Mateus Bahiense, Tarcísio Braga, Pedro Trigo Santana, Rafael Macedo, João Antunes, Edson Fernando, Maurício Ribeiro, Misturada Orquestra, Thiago Nunes, Rafael Martini, Théo Nicácio, LG Lopes, Luís Felipe Bisaggio e Afrânio Guimarães em BH e muitos outros; Juninho Ibituruna, Sara Cabral, Nana Mendonça, Antonino Coutinho, Romário e Nonoca, Nandy Xavier, Família Borges, Roberto, Saulo Carvalho, Cabeça, Carlão, Tadeu Felipe, Mengueles, Bobô da Cuíca e Rafael Formiga em Itabira/ MG por compartilharem de diversas etapas deste processo de auto-descoberta por meio da criação musical e da improvisação.

Especialmente à Daniel Christófaro e Gabriela Guerra, pelo incentivo e apoio, respectivamente, para que eu fizesse o mestrado;

Evandro Archanjo, pelo convite para realização da “Oficina de Construção de Flautas de PVC no IV Flautas Gerais” em Diamantina/ MG, oportunidade na qual pude concretizar mais uma etapa do desenvolvimento desta pesquisa.

Daniel Magalhães, pela generosidade com que se disponibilizou a partilhar de sua experiência de vida e conhecimentos pianísticos; Leonora Weissmann, Elisa Paraiso, Ana Flávia Figueiredo, Dora Vargas, Geovana Jardim, Bárbara Viggiano e Mariana Barros pela amizade e partilha poética; Aos *Amigos, Musicus e Artistas Brasiliensis sp* Mauro Souza, Edson Lima, Danielle Lemos, André Togni, Ricardo Nakamura, Filipe Togawa e Mavi, Lígia Moreno, Káthia Pinheiro, Nisvara, Dib Franciss, Antenor Bogéa, Páris Bogéa, Farley Derze, Anna Beatriz Gaertner, Diana Daher, Luciano e Júlio pela generosidade com que me acolheram, contribuindo, cada qual à sua maneira, para meu desenvolvimento e adaptação em Brasília; e Taís Vieira, também pelo interesse no projeto e empréstimo de livros importantes para esta pesquisa; Ao Prof. e Coordenador da Pastoral do La Salle, Edson Lima, pela amizade e incentivo; e ao Diretor Geral do La Salle, onde minha filha estuda desde então, Irmão Euclédes Casagrande, pelo apoio e pela acolhida na chegada em Brasília. Aos integrantes do Coral de pais Lassallistas pela amizade, em especial: Edson, Carol, Roberto, Josi, Shirley e Hudilei.

Aos mestres musicais: Flávio Pantoja, Annelise Kersten e Gilberto Bittencourt (Canarinhos de Petrópolis), Lena Horta (CBM/ RJ), Odette Ernest Dias, Kim Ribeiro, Estêvão Teixeira (CHFA/ JF), Toninho Carrasqueira; Artur Andrés, Mauro Rodrigues, Fábio Adour e Cliff Kormam (UFMG) e Matt Marvuglio (Berklee College of Music), *in memoria*, pela generosidade e inestimáveis contribuições ao meu desenvolvimento pessoal e musical.

Aos familiares: minha filha Stella, meus pais Ana e Flávio; meus irmãos: Davi, Petra Raissa e João Álvaro por tudo sempre; tia Tânia, pelo apoio e incentivo; e tio Sérgio pela contribuição na aquisição de livros da bibliografia do programa.

Meus sinceros agradecimentos à Ariadne Paixão por inúmeras contribuições, dentre elas a sensibilidade, incentivo e o apoio para com diversos estágios do desenvolvimento deste projeto entre outros, em especial neste último ano.

Criando Desejo...
Exerço a Liberdade Natural de Criatura Evolutiva que Sou.
Atento,
Deixo-me Guiar por Instâncias Superiores.
Intuo sobre o Verdadeiro Propósito da Experiência,
Faço-me pela Comunhão com o Outro.
Reconheço a Força da Oportunidade como Missão.
Aceito um Convite Original,
Desfaço os Labirintos do Ego;
Metamorfoseando-me e Transmutando-me, Supero Limites e Incongruências...
Pela Inspiração-Legado Supremo,
Fora do Tempo e do Espaço,
Vislumbro a Expressão Amplificada de Mim Mesmo.
Criar é estar nú, e Bonito!
Desfruto Integralmente da Criação em Todo o seu Potencial de Gozo.
Silencio por fim...

(Anônimo)

RESUMO

Modelagem¹ de Princípios Criativos: Investigando a Interação Músico-Computador por Meio da Improvisação Melódica

Este trabalho é fruto de uma pesquisa em andamento consistindo em uma Revisão de Literatura sobre sistemas interativos entre músico e computador. Nos ateremos aos vieses de interfaces computacionais interativas, portanto, e à improvisação melódica. Objetivamos assim, a partir da proposição de um sistema interativo músico-computador, definir orientações elaboradas sobre ‘Princípios Criativos’ de *Similaridade*² & *Contraste*³ e *Cronos* segundo Igor Stravinsky em sua *Poética Musical em 6 Lições* (1996) e o conceito de “*Building Blocks*” de John Zorn em seus sistemas composicionais coletivos, *Game of pieces*. Para tanto, estabelecemos tipos de Segmentação e Transformação, bem como critérios para avaliação, análise e comparação entre segmentos melódicos e sua posterior classificação em 6 níveis de similaridade e contraste numa régua contínua. Ao trabalharmos as técnicas de manipulação do material motivico, nos apoiamos, também, na obra *Fundamentos de Composição Musical* (2012) de Arnold Schoenberg. Partindo da biblioteca MuMRT (Musical Material Real Time), realizamos a modelagem a partir desses ‘Princípios Criativos’, averiguando a aplicabilidade do sistema proposto enquanto ferramenta composicional na interação com o computador, bem como em contextos coletivos de criação por meio da improvisação melódica. Partindo da proposição de práticas de criação musical coletiva, elaboramos abordagens de experimentação sistemática para o reconhecimento de gradientes dos parâmetros sonoros e aspectos formais do discurso musical. Como objetivos desta experiência, buscamos também, portanto, estimular a investigação musical criativa dos participantes, incentivando-os a compor instantaneamente, em grupo e por meio de processos improvisatórios. Assim, por meio de oficinas, realizadas entre 2011 e 2019 em instituições e festivais, bem como na disciplina “Oficina de criação e improvisação com instrumentos eletrônicos”, ofertada pelo Laboratório de tecnologia musical do Dep. de Música da UnB em 2019, constatamos resultados consistentes desta experimentação, uma vez que os participantes passam a agir e reagir musicalmente de forma mais consciente aos estímulos sonoros propostos no grupo e em relação à máquina. Também em relação ao desenvolvimento do programa, uma vez que pudemos averiguar a identificação entre Segmentadores, Transformadores, Blocos do Transformadores e níveis de Similaridade e Contraste dos possíveis OUTputs em relação aos INputs. Isto nos dá subsídio para o desenvolvimento do Gerenciador/ Inteligência do Sistema Interativo, uma vez que este será orientado a “escolher” OUTputs por meio da comparação entre segmentos melódicos transformados e o INput, orientados para um nível específico de Similaridade e Contraste em determinada Interação; sendo esta a etapa seguinte de implementação do programa.

Palavras-Chave: Sistemas Interativos Músico-Computador, Improvisação, Criação Musical Coletiva.

¹ Modelagem: segundo aqui expressamos, consiste na aplicação de modelos matemáticos e técnicas de programação computacional em C++ (código); com o objetivo de análise, compreensão e estudo da fenomenologia de problemas da área de computação musical e sistemas musicais interativos.

² Similaridade: qualidade, caráter ou condição das coisas similares. Característica, estado ou natureza do que é similar; semelhança. Particularidade dos objetos e/ou das coisas similares. Semelhança, similitude. (HUAISS, 2009) e (FERREIRA, 1998)

³ Contraste: grau marcante de diferença ou oposição entre coisas da mesma natureza, suscetíveis de comparação. O que sinaliza a oposição ou a distinção entre coisas ou pessoas, quando comparadas. (HUAISS, 2009) e (FERREIRA, 1998)

ABSTRACT

Modeling⁴ of Creative Principles: Investigating Musician-Computer Interaction through Melodic Improvisation

This work is the result of an ongoing research consisting of a Literature Review on interactive systems between musician and computer. We will stick to the biases of interactive computational interfaces, therefore, and to melodic improvisation. Thus, from the proposition of an interactive musician-computer system, we aim to define guidelines elaborated on 'Creative Principles' of *Similarity*⁵ & *Contrast*⁶ and *Cronos* according to Igor Stravinsky in his *Musical Poetics in 6 Lessons* (1996) and the concept of "*Building Blocks*" of John Zorn in his collective compositional systems, *Game of pieces*. For that, we have established types of Segmentation and Transformation, as well as criteria for evaluation, analysis and comparison between melodic segments and their subsequent classification in 6 levels of similarity and contrast in a continuous ruler. When working on the techniques for handling motivic material, we also rely on Arnold Schoenberg's work *Fundamentals of Musical Composition* (2012). Starting from the MuMRT library (Musical Material Real Time), we have performed the modeling based on these 'Creative Principles', investigating the applicability of the proposed system as a compositional tool in the interaction with the computer, as well as in collective contexts of creation through melodic improvisation. Starting from the proposition of collective music creation practices, we have elaborated systematic experimentation approaches for the recognition of gradients of sound parameters and formal aspects of musical discourse. As objectives of this experience, we also seek, therefore, to stimulate the creative musical investigation of the participants, encouraging them to compose instantly, in groups and through improvisational processes. Thus, through workshops, held between 2011 and 2019 in institutions and festivals, as well as in the discipline "Creation and improvisation workshop with electronic instruments", offered by the Music Technology Laboratory of the Music Department of UnB in 2019, we found consistent results of this experimentation, since the participants start to act and react musically in a more conscious way to the sound stimuli proposed in the group and in relation to the machine. Also in relation to the development of the program, since we were able to verify the identification between Segmenters, Transformers, Blocks of Transformers and levels of Similarity and Contrast of the possible OUTputs in relation to the INputs. This gives us support for the development of the Interactive System Manager / Intelligence, since it will be guided to "choose" OUTputs through the comparison between transformed melodic segments and INput, for a specific level of Similarity and Contrast for a given Interaction; this being the next stage of implementing the program.

Keywords: Musician-Computer Interactive Systems, Improvisation, Collective Music Creation.

⁴ Modeling: as expressed here, consists of the application of mathematical models and computational programming techniques in C ++ (code); in order to analyze, understand and study the phenomenology of problems in the area of music computing and interactive musical systems.

⁵ Similarity: quality, character or condition of similar things. Characteristic, state or nature of what is similar; similarity. Particularity of objects and / or similar things. (HUAISS, 2009) e (FERREIRA, 1998) - tradução

⁶ Contrast: a marked degree of difference or opposition between things of the same nature, susceptible of comparison. Which signals the opposition or the distinction between things or people, when compared. (HUAISS, 2009) e (FERREIRA, 1998) - tradução

Lista de Abreviações

1. *S&C* – Similaridade e Contraste
2. MuMRT – Biblioteca de códigos Musical Material Real Time
3. MuM – *MuMaterial*, Classe principal da biblioteca MuMRT

Lista de Figuras

Figura 1 – Demonstração das etapas de desenvolvimento do sistema	5
Figura 2 – Diagrama: Música Gerativa	11
Figura 3 – Sistema Interativo proposto: Fluxo de Informações (Dados).....	18
Figura 4 – OUPputs: segmentos do INput e segmentos transformados.....	20
Figura 5 – Classes da biblioteca MuMRT.....	25
Figura 6 – OUPputs válidos para um dado INput alterado por diversos Transformadores.....	35
Figura 7 – Correspondência entre Segmentadores, Transformadores e a régua de <i>S&C</i>	36
Figura 8 – Exemplo de Interação com Nível de Contraste Low (CLow).....	40

Lista de Gráficos

Gráfico 1 – Critérios para Avaliação de Segmentos Melódicos e níveis de <i>S&C</i>	37
Gráfico 2 – Segmentadores e níveis de <i>S&C</i>	38
Gráfico 3 – Transformadores e níveis de <i>S&C</i>	39

Lista de Tabelas

Tabela 1 – Critérios para Avaliação de Segmentos Melódicos.....	27
Tabela 2 – Segmentadores, Critérios para Avaliação de Segmentos Melódicos, Transformadores e níveis de Similaridade & Contraste.....	31
Tabela 3 – Blocos de Transformadores para cada nível de Similaridade & Contraste.....	33
Tabela 4 – Status do Desenvolvimento das Prerrogativas e Funções APLICADAS do Sistema.....	42
Tabela 5 – Status do Desenvolvimento das Prerrogativas e Funções NÃO APLICADAS do Sistema....	44

Lista de Apêndices

Apêndice 1 – Exemplos de possíveis Interações do Sistema.....	49
Apêndice 2 – Níveis de Similaridade & Contraste/ Critérios para Avaliação de Frases.....	50
Apêndice 3 – Funções de Transformação do Sistema.....	52
Apêndice 4 – Funções de Segmentadores do Sistema.....	55
Apêndice 5 – Alguns Testes do Segmentador (CSound/ ‘Partitura’).....	58
Apêndice 6 – Transformadores e Segmentadores: alguns testes válidos.....	60
Apêndice 7 – Código Completo do Programa Desenvolvido.....	64

Lista de Anexos

Anexo 1 – Lista de métodos da função principal da biblioteca MuMRT.....	65
---	----

Sumário

INTRODUÇÃO	1
i. Justificativa.....	2
ii. Objetivos.....	2
iii. Metodologia.....	3
1. Princípios Criativos: premissas para a proposição de um sistema interativo entre músico e computador	6
1.1 Similaridade & Contraste e <i>Cronos</i>	6
1.2 A interação músico-computador	9
1.3 A Improvisação enquanto abordagem composicional e “ <i>Building Blocks</i> ”.....	14
2. O Sistema Interativo proposto	17
2.1 Composição Algorítmica e a biblioteca MuMRT (Musical Material Real Time).....	20
2.1.2 A biblioteca MuMRT (Musical Material Real Time).....	22
2.1.3 MuMRT e o Improvisador.....	24
2.2 Segmentadores e critérios	25
2.3 Critérios para Avaliação de Segmentos Melódicos.....	26
2.4 Transformadores e critérios	28
2.4.1 Outros Transformadores	30
2.5 Blocos de Transformadores (Gerenciador/ ‘Inteligência’)......	31
3 Resultados & Conclusão	34
3.1 Discussão e Análise de Dados.....	35
3.2 Considerações Finais.....	42
Referências Bibliográficas	48
Apêndices	51

INTRODUÇÃO

Ao longo dos últimos anos venho desenvolvendo diversa atividade instrumental, tendo na flauta transversal, na composição e na improvisação, objetos e objetivos centrais de minha experiência, interesse e pesquisa. Nesse sentido, partindo da proposição de práticas de criação musical coletiva, elaborei abordagens de experimentação sistemática para o reconhecimento de gradientes dos parâmetros sonoros e estruturas formais do discurso musical. Por meio de oficinas de criação musical coletiva, realizadas entre 2011 e 2016, em instituições e festivais de Minas Gerais, pudemos averiguar resultados consistentes desta experimentação, uma vez que os participantes passaram a agir e reagir musicalmente de maneira mais consciente aos estímulos sonoros propostos no grupo. Como objetivos principais desta experiência, buscávamos até então, estimular a investigação criativa dos participantes, incentivando-os a compor instantaneamente em grupo e por meio de processos improvisatórios.

Na presente pesquisa procuramos realizar uma Revisão de Literatura ao investigarmos sistemas interativos e criativos entre músico e computador, como forma de ‘simular’ aquela experiência original. Nesse sentido nos ativemos também e portanto, aos vieses de interfaces computacionais interativas, improvisação e técnicas de composição coletiva. Pelo desejo de nos aprofundarmos ainda mais neste campo, partimos para a proposição de um sistema interativo músico-computador que pudesse nos dar suporte para o levantamento de dados desta pesquisa. A partir de disso, definimos orientações elaboradas sobre os ‘Princípios Criativos’ de *Similaridade & Contraste*, assim como o *Cronos* (tempo/ *timing*) na música, segundo Igor Stravinsky em sua *Poética Musical em 6 Lições* (1996); estabelecemos tipos de Segmentação e Transformação, bem como Critérios para Análise e Comparação de frases melódicas. Ao trabalharmos as técnicas de manipulação do material motivico, nos apoiamos, também, na obra *Fundamentos de Composição Musical* (2012) de Arnold Schoenberg. Partindo da biblioteca MuMRT (Musical Material Real Time)⁷, realizamos a modelagem (em C++) desses ‘Princípios’, sugerindo 6 níveis de *S&C* numa ‘régua’ contínua que pudesse orientar as manipulações do material, assim como as escolhas de saída.

⁷ Biblioteca de códigos disponível em: <<https://github.com/carlosemello/MuM>>

i. Justificativa

Com o propósito de ampliarmos a discussão na área, inserindo nossa produção em contextos acadêmicos e artísticos contemporâneos, propomos um sistema interacional músico-computador através da modelagem de ‘Princípios de criação musical’, também sendo nosso propósito responder à questão: *Tomando por base a interação músico-computador, como a modelagem desses ‘Princípios de criação musical’ pode contribuir para a compreensão de contextos composicionais coletivos, Instantâneos e improvisatórios?*

Na exploração sistemática de conceitos e parâmetros definidos para a investigação de processos interativos, tendo por abordagem a improvisação e por objetivo final a criação coletiva instantânea, nos deparamos inicialmente com questões relevantes, às quais orientaram em grande medida o sentido que buscamos e que nos foram úteis como suporte teórico e prático durante todo o percurso da pesquisa. Nesse sentido, mesmo que nossa questão central de pesquisa seja focada na validação de um sistema interativo proposto, pontos periféricos, não menos importantes, surgem como fatores a serem averiguados para um melhor direcionamento da mesma. Assim, buscamos restringir nosso campo de investigação para que possamos extrair o máximo de informações úteis e em tempo hábil para a pesquisa teórica, a construção e os testes do sistema interativo e sua validação posterior enquanto metodologia de composição e musicalização coletivas.

A seguir apresentamos uma revisão dos objetivos e métodos, bem como da proposição do sistema interativo e suas atualizações a partir das perspectivas e aprofundamentos suscitados por meio das leituras, orientações, programações e testes realizados até aqui, nesta etapa da pesquisa.

ii. Objetivos

a) Gerais

- Realizar pesquisa de Literatura relacionada aos temas da pesquisa
- Propor e testar um sistema composicional instantâneo na interação entre performer-computador, por meio de processos improvisatórios e melódicos, tomando por base os ‘Princípios musicais criativos’ de *S&C*, *Cronos* e *BuildingBlocks*, sobre os parâmetros de Altura e Duração.

b) Específicos

- Adquirir conhecimentos na área de programação computacional com o objetivo de realizar a modelagem dos ‘Princípios para a criação musical’;
- Propor um Esboço do Sistema Interativo;
- Definir os Critérios para Avaliação de frases melódicas (Sistema de Pesos);
- Definir os níveis de *S&C* entre frases melódicas;
- Definir Funções de Segmentação e Transformação dos segmentos melódicas;
- Definir co-relação entre os níveis de *S&C*, Critérios de Avaliação de segmentos Melódicos, Segmentadores e Transformadores;
- Propor Blocos de Transformadores, orientados para os níveis de *S&C*;
- Desenvolver um software, a partir da biblioteca MuMRT, testando, transformando e criando algorítmicos com base no sistema composicional proposto;
- Realizar uma Oficina de Criação Musical Coletiva com objetivo de testar, revisar e calibrar as funções de segmentação, transformação e sintaxes do sistema proposto;
- Descrever o processo, avaliando os resultados e projetando possibilidades e desenvolvimentos futuros;
- Realizar apresentação experiencial e demonstrativa das aplicações preliminares do software desenvolvido;
- Defender a dissertação e escrever artigos relacionados ao tema com base nos capítulos desta.

iii. Metodologia

Trata-se de uma Revisão de Literatura correspondendo ao cap.1 e uma Pesquisa Aplicada, gerando produtos e processos a partir da ‘*pesquisa básica*’, além da construção de conhecimentos dirigidos para a solução de problemas específicos, como a elaboração de sistemas de manipulação, seleção e resposta, averiguando as vantagens e limitações (aplicabilidade) de um sistema interativo entre músico-computador. A pesquisa caracteriza-se como sendo tanto Exploratória quanto Explicativa: tendo como finalidade acumular informações sobre o assunto investigado, orientando a fixação dos objetivos e a formulação de hipóteses. A pesquisa

Exploratória permitirá o estudo do tema sob diversos aspectos, tais como: Performance e Improvisação, Criação coletiva, Sistemas Interativos e Composição Algorítmica.

É também e, por fim, pesquisa Experimental, uma vez que definido o objeto do estudo, investigaremos as variáveis que serão capazes de influenciá-lo, definindo as formas de controle e de observação dos efeitos que as variáveis produzirão no objeto a partir do sistema proposto.

Com o intuito de nos atualizarmos em termos dos recursos computacionais disponíveis, buscamos auxílio do PPG em Música da UnB, através de orientação especializada para o uso de ferramentas que nos serviram de suporte. Para citar alguns: CSound⁸, PD (Pure Data)⁹ e C++¹⁰. Após a programação e os testes, fizemos o levantamento de dados preliminares para as devidas correções do sistema. Nesta etapa, os registros fonográficos/gravações para documentação, discussão e ampliação do conhecimento acadêmico e artístico resultantes do processo foram essenciais. Feitas as revisões do sistema, procedemos à avaliação por meio da análise e projeção de dados.

Na etapa 1 realizamos a Modelagem decorrente da *delimitação de características dos segmentos melódicos em cada um dos critérios de análise*, buscando orientar os transformadores para a comparação entre os segmentos a partir de cada um dos 6 níveis de S&C previamente definidos. Na etapa 2, implementamos os *segmentadores e transformadores* sobre o INput¹¹. Na etapa 3, a interação entre músico e interface são testados, os possíveis OUTputs¹² são selecionados e avaliados enquanto válidos para a interação. Abaixo um esboço das etapas de implementação e desenvolvimento do programa:

⁸ *CSound*: is a Domain-specific computer programming language for audio programming. It is called Csound because it is written in C, as opposed to some of its predecessors.

⁹ *Pure Data (Pd)*: is a visual programming language developed for creating interactive computer music and multimedia works.

¹⁰ *C or C++*: is a general-purpose, procedural computer programming language. *C* is an imperative procedural language. It was designed to be compiled to provide low-level access to memory and language constructs that map efficiently to machine instructions, all with minimal runtime support. Despite its low-level capabilities, the language was designed to encourage cross-platform programming. A standards-compliant C program written with portability in mind can be compiled for a wide variety of computer platforms and operating systems with few changes to its source code.

¹¹ Para o termo INput, leia-se toda a frase ou segmento melódico executados pelo músico, via MIDI, na entrada do programa.

¹² Para o termo OUTput, leia-se toda a frase ou segmento melódico executado, via MIDI, na saída do programa.

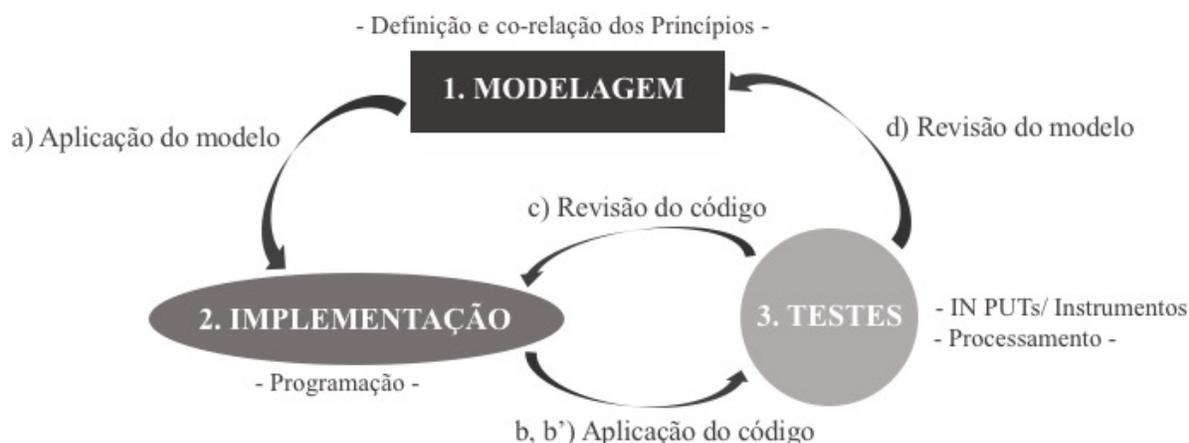


Figura 1- *Demonstração das etapas de desenvolvimento do sistema*

Fonte: *O Autor*

Procurando elaborar protocolos experimentais que nos auxiliassem no esboço do sistema, tomamos por base nossa experiência prévia na realização de oficinas, o embasamento na literatura de referência, bem como estudos realizados sobre as temáticas de Obra Aberta, Composição Coletiva Instantânea e Improvisação Coletiva. Uma “Oficina de Criação Musical Coletiva” foi realizada para testes e aprimoramento dos conceitos do sistema. Esta, que consistiu na realização de “Laboratórios Musicais”, foi realizada dentro da disciplina “*Oficina de criação e improvisação com instrumentos eletrônicos*”, ofertada pelo Laboratório de tecnologia musical do Dep. de Música da UnB no segundo semestre de 2019, sob orientação do Prof. Dr. Carlos Eduardo Mello. Na ocasião os estudantes tiveram a oportunidade de vivenciar diferentes práticas interativas, por meio da improvisação como ferramenta para a exploração sistêmica do elemento sonoro. Através de interações entre os participantes e o sistema proposto procuramos validar os prepostos à eficácia do mesmo. Elencamos a seguir alguns aspectos trabalhados nos encontros e que dinamizam as práticas junto ao sistema interativo proposto:

- a) Improvisação coletiva
- b) Elementos de Criação, Discurso e Linguagem musical (Estruturas, Forma e Variação Temática)
- c) Composição Coletiva
- d) Interação com o Sistema

A oficina constituiu assim, um laboratório/ campo de experimentação e testes das diversas fases de implementação do sistema interativo. Por meio desta pudemos gerar interações com INputs mais variados ou mesmo inesperados para a compreensão dos efeitos dos *Segmentadores e Transformadores sobre os segmentos do INput*, bem como a reatividade dos participantes aos OUTputs da máquina. Como objetivos para a proposição da oficina elencamos ainda:

- Testar e aprimorar um sistema interativo músico-computador
- Fomentar a exploração e interesse por processos de criação musical coletiva
- Proporcionar um espaço para a experimentação e improvisação
- Criar composições com o grupo

1. Princípios Criativos: premissas para a proposição de um sistema interativo entre músico e computador

Nesta seção procuramos estabelecer as bases teóricas que nos deram subsídio para a proposição do sistema interativo músico-computador. Consistindo em uma Revisão de Literatura, no cap. 1 buscamos explorar ‘princípios de criação musical’ que nos guiarão por todo o desenvolvimento do programa. Dispostos no item 1.1, são eles os conceitos de *Similaridade e Contraste* e *Cronos* (Relações temporais na música). Ainda, nesse sentido, investigamos na literatura disponível, processos interativos entre músico-computador no item 1.2, bem como processo improvisatórios ligados às experiências de Criação Coletiva Instantânea, assim como, no item 1.3, o conceito de “*Building Blocks*”, que nos será útil para a elaboração do programa.

1.1 Similaridade & Contraste (S&C) e Cronos

Na busca por um melhor direcionamento das oficinas, procuramos, inicialmente, embasamento em alguns autores que trouxeram grande contribuição para o aprofundamento e amadurecimento das questões suscitadas nos encontros. Igor Stravinsky, ao apresentando-nos seus ‘*Princípios de criação musical*’, no livro *Poética Musical em 6 lições*, coletânea de palestras realizadas em Harvard em 1939-40, proporcionou clareza filosófica a respeito da música enquanto linguagem, bem como seus aspectos técnicos composicionais mais herméticos de maneira reflexiva, estimulante e criativa. Entre estes podemos citar: os formais, os de

coerência do discurso, assim como algumas particularidades da cognição humana regidos pela percepção de características de *Similaridade & Contraste* no contato com o material sonoro. Estes últimos nos servindo para a proposição das ‘Linhas de interação’/ *sintaxe* ou “Regras de Engajamento” e “Operações”, terminologias e conceitos retirados da revista *Strategy & Tactics* de 1977 e que, utilizados por John Zorn para a elaboração da peça COBRA de 1984, do conjunto de composições *Game of pieces*, favoreceram a experimentação dos participantes, permitindo que criassem sem deixar de construir sentidos estruturais, fundamentais à condução do discurso musical e à comunicabilidade das interações.

Arnold Schoenberg, ao trazer importantes conceituações e exemplos de estruturas formais, bem como técnicas de variação motivico-temáticas em seu livro *Fundamentos da Composição Musical*, alicerçou os pontos de partida por meio de aspectos elementares, para a exploração sistêmica dos gestos sonoros propostos pelo grupo na realização das oficinas. Ao definir, por exemplo, que “[...] o termo *forma* significa que a peça é organizada, dotada de certa completude e adaptável à combinação com outras unidades similares” Schoenberg (2012, p. 27), o autor nos oferece um possível conceito de obra musical, o qual exploramos no curso desta pesquisa. Em se tratando de características melódicas, propriamente ditas, a conceituação dos termos *frase* e *motivo* ganham importância pra nós e são assim definidos: “A menor unidade estrutural é a *frase*... (é) constituída por ocorrências musicais unificadas, dotada de uma certa completude e adaptável à combinação com outras unidades estruturais”. (*ibid*, p. 29) Ressaltando também que “O *ritmo* é elemento particularmente importante [...] o autor contribui, ainda, para o interesse e variedade que a frase engendra, estabelecendo seu caráter e sendo frequentemente o fator determinante para a existência de sua unidade”. E ainda: “a preservação de elementos rítmicos produz, efetivamente, coerência [...]” (*ibid*, p. 30 e 36) Ao *motivo*, por sua vez, dá-se especial atenção pois ele “... é frequentemente considerado o germe da ideia...visto que quase todas as figuras de uma peça revelam algum tipo de afinidade para com ele” (*ibid*, p. 35), assim:

[...] o uso do motivo deve produzir unidade, afinidade, coerência, lógica, compreensibilidade e fluência do discurso. Os fatores constitutivos de um *motivo* são intervalares e rítmicos [...] (*ibid*, p. 29-35)

Na mesma obra o autor de Pierrot Lunaire reforça que os aspectos da “Formação, Utilização e Tratamento dados ao *motivo*, uma vez que englobam as noções de repetição e

variação, são justificadas em seu lugar de pertinência no contexto do trabalho.” (*ibid*, p. 35-37) Tais conceitos, associados aos princípios de *S&C* de Stravinsky passaram então a conduzir as definições de ‘*Linhas de interação*’ nos processos de criação coletiva, contribuindo, pois, para a formulação das sintaxes interativas, tomando os ‘princípios de coerência e coesão do discurso’ a partir daqueles eixos. Ao abordarmos esses princípios “que dominam o processo criador”, aos quais se fazem corresponder as noções essenciais de unidade e variedade, verificamos o posicionamento do compositor de Petrouchka:

[...] tenho considerado sempre que, em geral, é mais satisfatório proceder por similaridade do que por contraste. [...] O contraste produz um efeito imediato. A similaridade nos satisfaz a longo prazo. O contraste é elemento de variedade, mas divide nossa atenção. A similaridade nasce de um esforço pela unidade. A necessidade de se buscar a variedade é perfeitamente legítima, mas não devemos esquecer que o Uno precede o múltiplo. (STRAVINSKY 1996, p. 38)

Indo um pouco mais além, entendemos que a questão do ‘tempo musical’, será de importância capital para nós, uma vez que pretendemos uma interação músico-computador em tempo-real. Sob esta ótica, (*ibid*, p. 35), tendo já afirmado anteriormente que “...a música é uma arte cronológica...”, Stravinsky remete-nos ao filósofo russo Pierre Souvtchinski, e nos fala do *Cronos* musical pelo viés fenomenológico:

Qualquer música, quer se submeta ao fluxo normal de tempo, quer se dissocie dele, estabelece uma relação particular, uma espécie de contraponto entre a passagem de tempo, a duração da própria música e os meios técnicos e materiais pelos quais a música se manifesta. (*ibid*, p. 37)

Ressaltando, ainda, dois tipos de música no que tange à evolução temporal, Stravinsky sugere que o compositor agiria, então, sobre um certo ‘campo’ de expectativas do ouvinte, sendo o controle dessas duas experiências dado nas dimensões do ‘tempo ontológico’ e do ‘tempo psicológico’. O primeiro envolveria o ouvinte, introduzindo “...um sentimento de euforia em sua mente”, e que Stravinsky chamou de ‘calma-dinâmica’. Para nós este conceito poderia ser traduzido como a ‘expectativa de interação’, gerada pelo INptut. Já o segundo, diz ele, “...vai em direção contrária...”, “...não está encerrado em cada unidade momentânea”. Aqui a dinâmica de *contrastes* tomaria seu devido lugar a partir das variações (segmentações e transformações) aplicadas ao INput e devolvidas nos OUTputs com níveis alterados de semelhança e diferença em relação aos INputs. O autor então conclui:

A música que se apoia no tempo ontológico é geralmente dominada pelo princípio da Similaridade. A música que adere ao tempo psicológico tende a proceder por Contraste. Esses dois princípios [...] dominam o processo criativo[...] (*ibid*, p.37)

1.2 A interação músico-computador

Da área de Computação Musical, na qual nos inserimos, pelo objeto e metodologia de nossa investigação, Griffiths (1986)¹³, nos diz que até pouco tempo, podíamos distinguir claramente três grupos de música apoiadas na interação com o computador: a) a música na qual o computador gerava a partitura, que poderia ser executada por humanos, a exemplo da *Illiad Suite* para quarteto de cordas de Hiller e Isaacson; b) a música composta por humanos e executada por computadores, a exemplo dos experimentos de Síntese e sequenciamento de sons de Max Mathews; que também teve aprofundados vários desenvolvimentos de sua produção na pesquisa de Duarte (2014)¹⁴ sobre as peças *MUSIC N*, e ainda: c) a música composta e executada por computadores. Griffiths (1987, p. 156)¹⁵ nos lembra que Iannis Xenakis¹⁶, também usou computadores para auxiliá-lo em suas composições por meio de métodos estocásticos¹⁷; sendo um exemplo disso a peça *Eonta*, para piano e quinteto de metais (1963-64). Entretanto, o aumento gradativo no poder de processamento das máquinas, bem como sua capacidade de memória, afetou drasticamente a maneira como interagimos e geramos música com o auxílio do computador. Características mais elaboradas da Síntese Sonora, do Processamento de Áudio, da Composição Algorítmica e da Programação, entre outros, projetaram para além das peças criadas naqueles três grupos, o surgimento de um novo modelo composicional: o de peças criadas e executadas conjuntamente pelo músico e pelo computador, ambos compositores/ improvisadores e performers. Um exemplo disso foi OMAX¹⁸, desenvolvido por pesquisadores do IRCAM¹⁹. Nos aprofundaremos ainda sobre este programa mais adiante.

Griffiths (1979)²⁰, num contexto histórico e mais generalista, contribui também para uma maior compreensão dos desafios enfrentados pelo compositor no cenário musical a partir do séc.

¹³ *Encyclopaedia of 20th Century Music*

¹⁴ *A Contribuição de Max Mathews para a Música Computadorizada*

¹⁵ *Modern Music: a concise history* (título original, 1947)

¹⁶ *Formalized Music: Thought and Mathematics in Composition*

¹⁷ *Estocástico: método estatístico e probabilístico*

¹⁸ OMAX é um software de interação musical em tempo-real desenvolvido pelo IRCAM

¹⁹ IRCAM - Institut de Recherche et Coordination Acoustique/ Musique

²⁰ *A guide to electronic music*

XX e que confluíram, em alguma medida, por fim, para a música feita com o auxílio do computador. Em relação à busca por novas técnicas e estéticas, o autor nos alerta que “...compositores tais como Debussy e Mahler, Bartok e Schoenberg, Webern e Strauss [e por quê não Stravinsky?] foram rapidamente estendendo as gamas de sons admitidos em música”. É nesse sentido que a Música Eletrônica, surge também como opção, pela inserção de outros conceitos e meios de produção sonora na prática dos compositores. Referindo-se ao *Sketch for a New Aesthetic of Music* (1907) de Ferruccio Busoni, Griffiths exemplifica aquele cenário com as exclamações e indagações do autor:

A ‘Exaustão’, ele escreveu, ‘certamente nos espera ao fim desta longa trajetória que já foi percorrida. Para onde então deve se virar nossa atenção? Em que direção deve nos guiar o próximo passo? Ao som abstrato, à técnica..., ao material ilimitado.’ (apud GRIFFITHS 1979, p.9)

Hiller e Isaacson (1959)²¹, ao tentarem simular a tarefa composicional no computador empregam um gerador/ modificador/ seletor para estimular o processo.

Esta abordagem conduz à geração de ‘raw materials’ enquanto bases para a composição. Então aplicamos técnicas, tais como permutação e transformação geométrica, para manipular o material gerado. Por fim, aplica-se uma seleção de regras para escolher o material mais adequado a se tornar parte da composição. (apud HILLER; ISAACSON, 1959)

Abaixo, um quadro dispendo os tipos de sistemas que a pesquisa em música feita com auxílio do computador abrange:

²¹ *Experimental Music: Composition with an Electronic Computer*

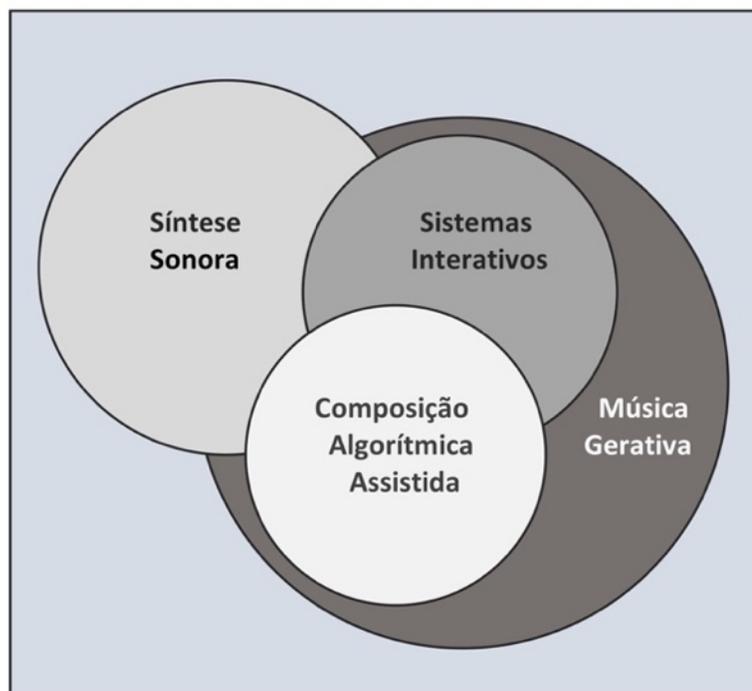


Figura 2 - Diagrama: Música Gerativa

Fonte: Wikipedia (Christopher Ariza)

Do ponto de vista dos processos interativos tomamos por exemplo as pesquisas de modelagem de estilo que, no intento de reproduzir a estética específica de um compositor ou improvisador, buscam identificar padrões rítmicos e melódicos que representem o que Cope (1996, p.27)²² denomina por “Musical Signatures”, presentes em seu *Signature Dictionary*, e que passam a guiar o algoritmo de escolhas feitas pela máquina para que esta componha no estilo de Mozart, com o sotaque dos solos de Coltrane ou à moda Barroca. Já o software OMAX, por exemplo, é um ambiente de software que também aprende em tempo real as características típicas do estilo de um músico e toca junto com ele de forma interativa, dando a sensação de uma máquina/ferramenta de co-improvisação. OMAX usa o OpenMusic²³ e Max²⁴, é baseado em pesquisas

²² *Experiments in Music Intelligence*

²³ *OpenMusic*: is an object-oriented visual programming environment for musical composition based on Common Lisp. It may also be used as an all-purpose visual interface to Lisp programming. At a more specialized level, a set of provided classes and libraries make it a very convenient environment for music composition.

²⁴ *Max*: also known as Max/MSP/Jitter, is a visual programming language for music and multimedia. Over its more than thirty-year history, it has been used by composers, performers, software designers, researchers, and artists to create recordings, performances, and installations. The Max program is modular, with most routines existing as shared libraries. Because of this extensible design, which simultaneously represents both the program's structure and its graphical user interface (GUI), Max has been described as the lingua franca for developing interactive music performance software.

sobre modelagem estilística realizadas por Cont, Dubnov e Assayag, (2007)²⁵ e em pesquisas sobre improvisação com o computador realizadas por Assayag, Bloch e Chemillier (1994)²⁶ no grupo Ircam Music Representations. Assim como OMAX re-injeta de várias maneiras diferentes o material do músico que passou por um estágio de aprendizado da máquina, permitindo uma representação em nível semântico da sessão, bem como uma recombinação e transformação deste material em tempo real, nós também, na proposição de nosso programa buscamos recuperar o material segmentado e transformado, embora a questão estilística não seja nosso foco, mas uma interação com elementos de ligação e coesão.

OMAX usa ainda variantes originais do algoritmo Factor Oracle especialmente adaptado para o processamento de músicas. Alguns destes algoritmos “aprendem” no corpo do processo ou do passado de uma sessão para produzir continuações estilisticamente adequadas e/ ou baseadas no gesto mais recente da improvisação. (CROCHEMORE *et al.*, 2007, p.56)²⁷

OMAX lida com dados MIDI, áudio e vídeo, nós utilizamos apenas entradas e saídas MIDI, com foco estrito nos parâmetros de altura e duração.

As sessões do OMAX Garson ocorreram em novembro de 2006 no estúdio de gravação de Mike Garson no norte de Los Angeles. Especialmente o modo “follow” que pode ser ouvido nos exemplos a seguir, onde a zona de memória na qual o OMAX improvisa realmente segue o músico com uma profundidade de memória variável. Especialmente para um músico que é capaz de reagir a milissegundos a qualquer variação que o software produz, esta opção é muito desafiadora e muito ‘re-assegurante’, já que o músico tem um melhor e mais rápido controle sobre as instruções tomadas pelo software. Mike também sugeriu transformação “algébrica”, como retrógrada, inversões, etc. (Site: IRCAM)²⁸

Ainda neste contexto, pesquisas levadas a cabo pelo NICS (Núcleo Interdisciplinar de Comunicação Sonora da Unicamp), especialmente nos trabalhos desenvolvidos por Schaub *et al* (sd)²⁹, nos servem de esteio pela natureza de sua abordagem sistêmica, sua proposta composicional e simultaneamente improvisatória, características desta pesquisa:

Um improviso interativo (será) considerado aqui como um sistema musical capaz de acompanhar o jogo entre um ou mais músicos (músico e computador) improvisando ao vivo. Supõe-se que a saída (Out put) fornecida pelo computador

²⁵ *Anticipatory Model of Musical Style Imitation*

²⁶ *Omax Brothers: a Dynamic Topology of Agents for Improvisation Learning*

²⁷ *Algorithms on strings*

²⁸ <http://repmus.ircam.fr/omax/home> (tradução nossa)

²⁹ *Interactive Free Improvisation Using Time-domain Extrapolation of Textural Feature.*

não se reduza a conjuntos pré-estabelecidos de fórmulas, mas que seja elaborado “on the fly” (em tempo real) a partir de informações extraídas da sessão de que se torna um participante ativo. Assim como em outros sistemas..., o formulário de saídas deve primar pela coerência estilística geral e contextual, objetivo geral do desenvolvimento desse sistema. (SCHAUB *et al.*, sd)

Por fim, a passagem para a ‘improvisação interativa’ foi conduzida a partir de uma variedade de recursos algorítmicos. Neste ínterim o *Continuator* de Pachet (2002)³⁰, por exemplo, usa árvores de decisão probabilísticas. Este algoritmo permite a interação musical em tempo real por meio da improvisação. Outros tipos de abordagem também podem ser mencionados como o GenJam, que usam algoritmo genético.

Tais algoritmos funcionam por meio de ‘métodos evolucionários’, ou métodos evolutivos de compor. Através da ‘mutação e seleção natural’, diferentes soluções evoluem para uma peça musical adequada. A ação iterativa do algoritmo elimina as soluções ‘ruins’ (indesejadas) e cria novas soluções daqueles que sobrevivem ao processo. Os resultados do processo são supervisionados pelo ‘crítico’ (Gerenciador), uma parte vital do algoritmo que controla a ‘qualidade’ das composições criadas. (BILES, 1994)³¹

Desde os primeiros experimentos de Lewis (2000)³² com a Voyager nos anos 80, vários sistemas digitais interativos foram propostos e são considerados pertinentes para a pesquisa que hora realizamos. Schaub *et al* (sd), no entanto, consideram que...

[...] a natureza interativa desses sistemas pode ser ‘obscura, misteriosa e opaca’ e mudaram seus esforços para tentar ‘produzir imitações’ de comportamento interativo em direção a critérios mais gerais. (SCHAUB *et al*, sd)

Ainda segundo estes autores, nesse sentido a noção de ‘antecipação’ (ou expectativa), por exemplo, pode ser encontrada na teoria e na análise da música através do trabalho de Meyer (1956)³³, e também conhecendo desenvolvimentos importantes, nesse sentido, desde então, sendo Huron (2006)³⁴ um exemplo disso. Nos trazendo *insights* ainda, a pesquisa em psicologia do comportamento improvisatório, pelo menos desde Pressing (1998)³⁵, integra pertinente reflexão sobre os aspectos atitudinais do performer durante a prática da improvisação. Mais

³⁰ *The Continuator: Musical Interaction with Style*

³¹ *GenJam: A Genetic Algorithm for Generating Jazz Solos*

³² *Too Many Notes: Computers, Complexity and Culture in Voyager*

³³ *Emotion and Meaning in Music*

³⁴ *Sweet Anticipation: Music and the Psychology of Expectation*

³⁵ *Improvisation: Methods and Models*

recentemente, chegamos ao desenvolvimento de improvisadores interativos automáticos, com contribuições notáveis de Cont *et al* (2007)³⁶ que concluem: “Pensando a ‘improvisação’ por parte do computador, poderíamos inferir que ela, de alguma forma, estimularia a criatividade do músico ao fornecer estruturas (musicais) reconhecíveis.” Assim, na performance ao vivo, o nosso sistema passaria a re-injetar o material do INput de várias maneiras, buscando uma representação em nível semântico da sessão em curso, bem como uma re-combinação ‘inteligente’ de transformações deste material em tempo real.

Em se tratando da questão temporal da música realizada na interação com a máquina, Cont (2008)³⁷ nos propõe um sistema de acompanhamento modular e antecipatório, sendo capaz de executar partituras eletrônicas em sincronização com uma performance ao vivo e usando vários controles ao longo do tempo. Em seu uso básico, é um sistema clássico de partitura, mas em uso avançado permite a representação simultânea e o reconhecimento de diferentes descritores de áudio (não apenas alturas). O programa atua sobre várias escalas de tempo usadas na música escrita e possibilita a interação temporal entre a execução e a partitura eletrônica. *Antescofo* vem com uma linguagem de partitura simples para escrita flexível de tempo e interação em música de computador.

1.3 A Improvisação enquanto abordagem composicional e “Building Blocks”³⁸

No presente trabalho, desenvolvemos um sistema interativo entre músico e computador. A abordagem que propomos é a da improvisação, porém sem nos atermos às condicionantes de estilo. Nesse sentido, ao nosso ver, embora não haja uma terminologia completamente adequada, nos baseamos, entre outros, no conceito de improvisação não-idiomática, com as devidas ressalvas, uma vez que nossa proposta limita a interação aos parâmetros de altura e duração, bem como pode estar inserida em contextos de tonalidade, modalidade e/ou ainda contextos mais amplos como atonalidade, politonalidade, entre outros; dependendo do tipo de interação proposto no grupo. Assim, Bailey (1992)³⁹, propondo o termo,

³⁶ *Anticipatory Model of Musical Style Imitation*

³⁷ *Antescofo: Anticipatory Synchronization and Control of Interactive Parameters in Computer Music*

³⁸ *Building Blocks*: Memória/ Biblioteca de componentes temáticos, estilísticos, motivicos e rítmicos – unidades de conteúdo musical que correspondem aos *elementos constitutivos* e cuja manipulação é um importante componente improvisatório em diversas culturas.

³⁹ *Improvisation: its nature and practise in music*

nos elucidada sobre algumas de suas características, “[...] por não estar condicionada às questões de estilo, evitando por exemplo o que de antemão nos remeteria a condicionamentos pré-concebidos”. No entanto, encontramos duas justificativas do termo que nos servem como ponto de partida para os contextos de interação com a máquina e enfoque que desejamos dar para nosso programa:

Frequentemente os sistemas de improvisação interativa automática são aplicados a ambientes improvisatórios não-idiomáticos. Isso pode ser atribuído ao fato de que seu domínio dos estilos verdadeiramente idiomáticos ainda é um tanto superficial e que a exploração de novos horizontes musicais foi, desde o início, uma importante motivação por trás de seu desenvolvimento. (SCHAUB *et al.*, sd)

Partimos da premissa de que, para se desenvolver no discurso musical, a expressão através da improvisação deve buscar suas inflexões, “gestos” e modos alternativos de tocar, em elementos e princípios de coerência. Estes, por sua vez, passam a desempenhar um papel central na construção de sentido por meio de suas dimensões estruturais e formais. Segundo Pressing (1998) “[...] quando nenhuma gramática ou referência é pré-estabelecida, os princípios dos quais a continuidade e a coerência geral do fluxo musical devem seguir são (mais) difíceis de identificar.” Ao afirmar que “[...] a prática da improvisação livre só será possível a partir da configuração de uma escuta contemporânea do pós-guerra, múltipla e intensiva”, Rogério Costa nos ajuda a entender tanto o cenário no qual esta música interativa surgiu - o da Democracia e dos conceitos de Liberdades e de Direitos Individuais – quanto os vieses condicionados pelo ‘hibridismo’ e à experimentação, conceitos recorrentes do *metier* artístico desde então.

A improvisação livre ou não idiomática é uma possibilidade que se configura a partir de uma série de fatores, dentre os quais a crescente dissolução ou permeabilidade de fronteiras entre os idiomas e sistemas musicais e o conseqüente cruzamento que se dá entre as diversas linguagens em determinados contextos da prática musical contemporânea. (COSTA 2003, p.6)⁴⁰

Em sua dissertação *O jogo musical COBRA* de John Zorn, Abreu (2013)⁴¹ nos traz ampla pesquisa sobre a história da Improvisação e suas diversas abordagens, bem como sobre métodos de práticas criativas coletivas por meio da Improvisação e das “*Regras de Engajamento*”, aproximando assim a improvisação e a composição:

⁴⁰ *O Músico enquanto meio e os territórios da livre Improvisação*

⁴¹ *O jogo musical Cobra (1984) de John Zorn: Reflexões sobre sua história e prática*

[...] partes do mesmo substrato criativo, improvisação e composição são muitas vezes apontados como pontos opostos numa mesma linha, mas seus territórios e limites não são facilmente discerníveis. Até que ponto a improvisação - criação no curso da performance - não é uma espécie de composição em tempo real, também sujeita a regras, padrões e estruturas? [...]

Mais recentemente, Sundberg (2005)⁴² reforça de maneira singular o viés da nossa pesquisa ao afirmar “[...] a existência de uma ‘mão dupla’ na relação performer e computador”. De um lado, a performance e seus contextos expressivos orientam o processo de modelagem de parâmetros na programação computacional, de outro lado, a busca pelo desenvolvimento de programas interativos que sejam ‘mais musicais’/ (mais orgânicos talvez), fomentam o aprofundamento de estudos no que tange aos pontos essenciais na pesquisa sobre performance, especialmente a criatividade, nosso foco principal. Sob esta ótica, Domenici (2012)⁴³ afirma que “[...] ao discutirmos o aspecto participativo do intérprete no contexto do ‘fazer musical’, acreditamos estar contribuindo para uma visão de seu papel enquanto construtor de sentido a partir da interação com a ‘obra’”. Investigando os desdobramentos desta afirmação: performers agindo a partir de ‘Princípios de criação musical’ definidos encontrariam um viés paramétrico de comunicação sonora ao expressar-se em um dado cenário musical. Um dos desafios para toda pesquisa nessa área constatamos ser a geração desse contexto na interação com a máquina. Esta, por sua vez, igualmente, respondendo e ‘reagindo aos estímulos’ propostos na interação.

No contexto da composição através da performance, Polaschegg (2007)⁴⁴ nos elucida sobre o papel que a improvisação vem assumindo desde a década de 1950 até as mais recentes experiências de interação com o computador, nos advertindo sobre sua intrínseca relação com a composição no contexto contemporâneo. Afirma Polaschegg (*ibid*) que “uma emerge da outra”, propondo, a partir disso, o conceito de “com-provisation”.

A proposição da geração de repositórios composicionais/ memórias, nos levou, em nossa pesquisa, ao conceito de “*Building Blocks*”, o qual aproveitamos dos *Game of pieces* de John Zorn, para a elaboração das funções de memória e recuperação de segmentos (transformados ou

⁴² *Computer synthesis of music performance*

⁴³ *A voz do Performer na Música e na Pesquisa*

⁴⁴ *Interweavings - towards a new view of the relation between composition and improvisation*

não) a serem dispostos nos OUTputs do programa. É importante para nós, neste estágio da pesquisa, antecipar alguns processos a partir do armazenamento de segmentos transformados ou não. Estes, por sua vez deverão ser ‘marcados’ com os níveis de S&C alcançados até então. A partir daí seriam resgatados para os OUTputs, caso alcancem os níveis de S&C desejados para a interação, ou ainda conduzidos para novos processos de Transformação até que estejam finalmente disponíveis para a escolha das saídas do programa. No contexto das técnicas de Improvisação Coletiva, sugeridas por John Zorn para as peças do *Game of pieces*, por exemplo, *prompters*⁴⁵ são levantados por um ‘regente’. Em formato de placas, estes buscando dar orientações mais gerais para o todo ou partes do grupo, no sentido da abordagem comportamental dos integrantes durante a execução da peça. *Prompters* mais específicos, em termos musicais, orientam os integrantes para conjuntos de notas, regiões de oitavas, densidade melódica, memorização e resgate de eventos, entre outros.

A partir da construção de cenários para a elaboração de obras instantâneas, conjuntamente criadas e executadas com o computador, transportamos a ideia do performer/ intérprete ao ponto de abordagem desta pesquisa: o do improvisador/ compositor. É nesse sentido que, conceitos tais como a Livre Improvisação, Composição Instantânea e Obra Aberta se ligam direta e intimamente ao nosso projeto.

2. O Sistema Interativo proposto

Ao desenvolvermos este sistema interativo músico-computador, apresentamos uma proposta de ‘Linhas de interação’ musical, de sintaxe específica, baseada na exploração dos ‘Princípios de criação musical’, sendo estes estruturados a partir de dois eixos principais interligados:

- I. Parâmetros do Som: Altura e Duração - Investigando *Similaridades & Contrastes* melódicos, a partir de Igor Stravinsky em sua *Poética Musical em 6 Lições*. Este primeiro eixo nos orienta para a elaboração de todo tipo de critério avaliativo do sistema.
- II. Forma e Coerência no Discurso musical - A partir da exploração do material motivico, propondo estruturas, técnicas de variação melódica e elementos de articulação, com base nos

⁴⁵ Placas com números, símbolos, formas geométricas e orientações propriamente ditas.

Fundamentos da Composição Musical de Arnold Schoenberg. Este segundo eixo nos orienta para a elaboração de todo tipo de função de alteração do *material musical* melódico.

Entendemos, *a priori*, que nossa pesquisa teve início pelo interesse nos processos de criação musical coletiva que partissem da prática de improvisação. Ao propormos oficinas de criação coletiva, buscamos respaldo em experiências anteriores para que pudéssemos elaborar práticas consistentes no que tange ao aprimoramento perceptivo e conceitual dos participantes enquanto experimentam conscientemente o material sonoro. A proposta de aprofundarmos essas práticas, a partir da investigação de processos criativos na interação com o computador, nos conduziu à uma pesquisa de literatura; esta por sua vez, nos dando subsídio para a proposição e desenvolvimento de um programa/ sistema Interativo que pudesse gerar OUTputs orientados para determinados níveis de similaridade e contraste, a partir da segmentação, transformação e análise de INputs improvisados. A seguir, um gráfico demonstrativo das etapas de funcionamento do sistema, bem como do fluxo de dados sugeridos:

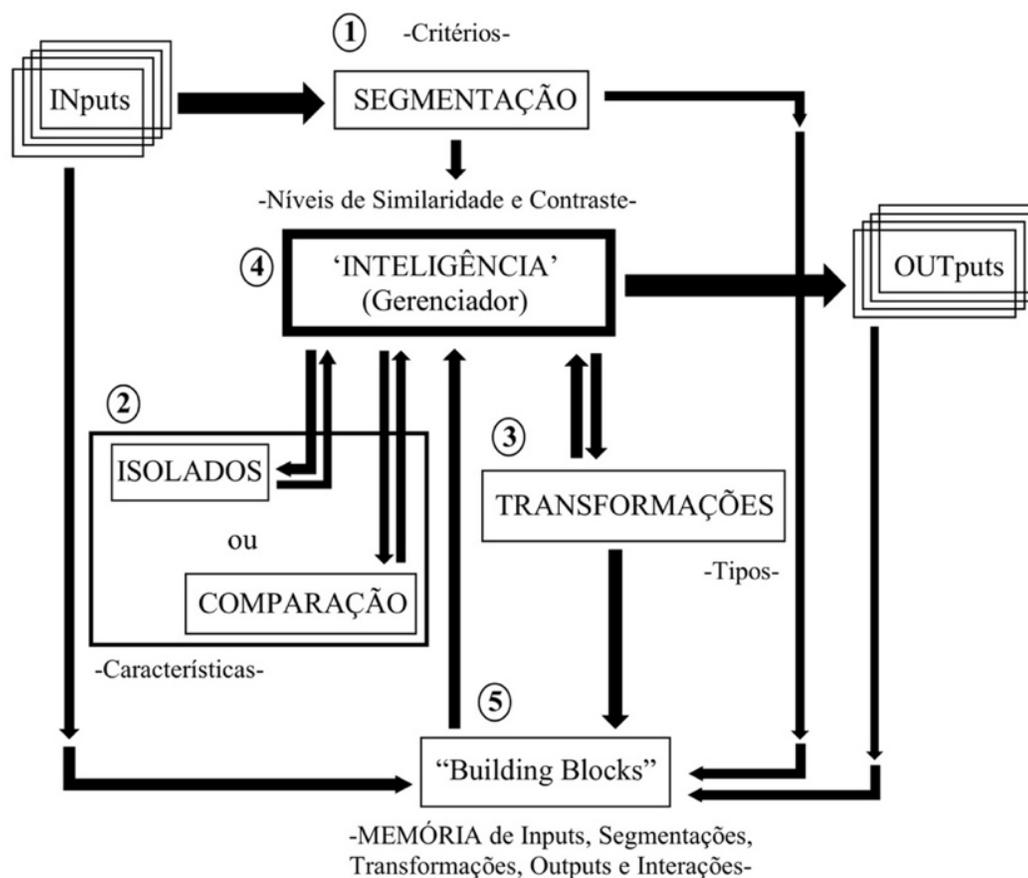


Figura 3 - Sistema Interativo proposto: Fluxo de Informações (Dados)

Fonte: O Autor

A seguir apresentamos, em detalhe, a sugestão de funções que o programa dever realizar:

- a) Captação de INputs MIDI improvisados pelo músico (Alturas e Durações)
- b) Avaliação das características melódicas do INput, segundo critérios definidos
- c) Segmentação do INput e Avaliação, segundo critérios definidos
- d) Alteração dos segmentos melódicos por meio de Transformadores e Blocos de Transformadores
- e) Comparação dos níveis de S&C entre o INput e os possíveis OUTputs
- f) Memorização/ Armazenamento de INputs, Segmentos Transformados e OUTputs, bem como Interações completas/ Input + OUTput no “Building Blocks”
- g) Re-produção de OUTputs orientados para determinado nível de S&C

Partindo deste projeto, o sistema resultante vem sendo testado quanto à sua coerência e reatividade, em sessões de improvisação gravadas e em oficinas de criação musical coletiva nas quais o computador é um dos integrantes.

A partir do momento em que objetivamos orientar a interação músico-computador para um dado nível de S&C, torna-se crucial, na construção do nosso sistema, elaborarmos definições precisas no que tange às variantes dos segmentos do INput, bem como das variantes dos segmentos transformados para as escolhas de OUTputs. Isto se justifica uma vez que, potencialmente, todos os segmentos passam a configurar no âmbito das saídas do computador durante a interação; servindo também e portanto, ao refinamento da análise dos segmentos a ser realizada na etapa seguinte de desenvolvimento da “inteligência” do programa. No sentido de esclarecermos algumas nuances desta análise, diferenciamos a seguir o que entendemos por *Possíveis OUTputs*, *OUTputs desejáveis* e *OUTputs válidos e não válidos*.

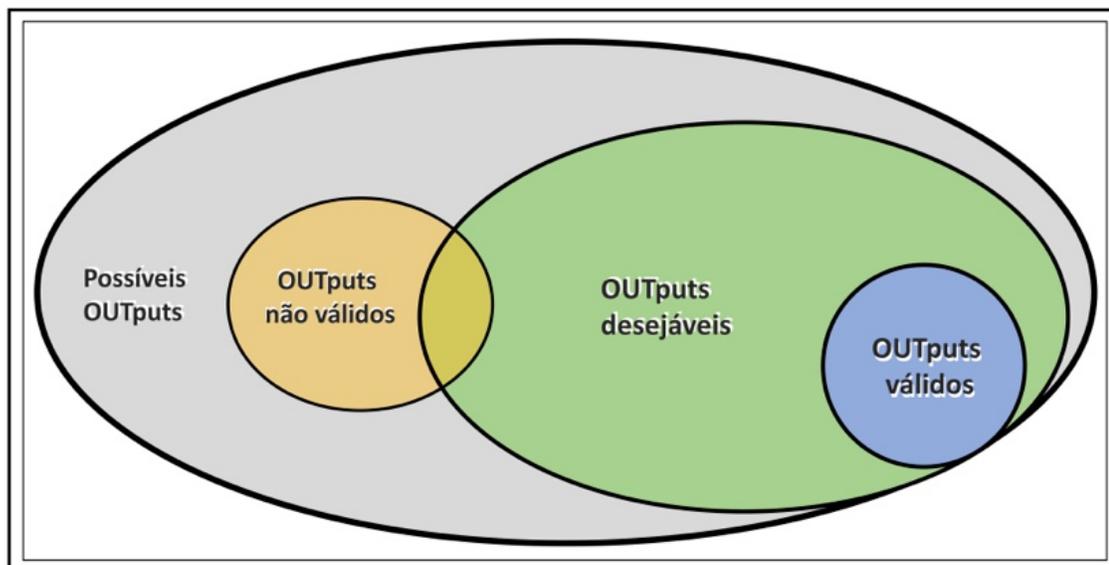


Figura 4 – *OUTPUTs: segmentos do INPUT e segmentos transformados*

Fonte: o Autor

Quanto aos *Possíveis OUTPUTs*, entendemos que sejam todos os segmentos do *INPUT*, transformados ou não, que aguardam por uma análise comparativa do gerenciador na memória “Building Blocks” do sistema. Assim, uma vez sendo definido que determinada interação será orientada para certo nível de *S&C*, entendemos que *OUTPUTs desejáveis* sejam aqueles que, já tendo sido segmentados e transformados, apresentem potencial, pela análise do gerenciador, para, em algum momento, serem parte da interação na saída do programa. Entendemos, portanto, que *OUTPUTs válidos* sejam aqueles, entre os *OUTPUTs desejáveis* e que, efetivamente, foram escolhidos pelo gerenciador do nosso sistema para a saída do programa em determinada interação. Já os *OUTPUTs não válidos*, por sua vez, são aqueles descartados pela análise do gerenciador, após sua segmentação e transformação, por não terem atingido os níveis aceitáveis ou estipulados de *S&C* para determinada interação. No gráfico acima ilustramos as variantes dos *OUTPUTs* e suas inter-relações.

2.1 Composição Algorítmica e a biblioteca MuMRT (Musical Material Real Time)

Em se tratando de composição algorítmica, partiremos inicialmente da biblioteca MuMRT (Musical Material Real Time), desenvolvida pelo prof. Dr. Carlos Eduardo Mello,

coordenador do Núcleo de Computação Musical – NCM, do Laboratório de Computação Musical do Dep. de Música da UnB e orientador deste trabalho. Esta biblioteca consiste em um *framework* de Classes e Objetos construídos na linguagem C++ para servir de suporte à implementação de algoritmos de composição musical automatizada. Justificamos a escolha da programação em C++, não apenas por ser a linguagem da biblioteca, que já nos traz um número considerável de funções para a manipulação de material musical, mas também por possibilitar uma maior flexibilidade no desenvolvimento de funções específicas, próprias para a construção do nosso programa. Contamos também, nesse sentido, com a importante colaboração de José Abreu, integrante do Núcleo de Tecnologia Musical do Dep. de Música da UnB, que contribuiu sobremaneira com autor na elaboração do sistema, bem como na programação das funções de Segmentação e Transformação do sistema Interativo.

Em seu artigo *Musical Material and Algorithmic Composition*, o Prof. Dr. Carlos Eduardo Mello nos traz breve panorama da composição algorítmica, inserindo o desenvolvimento da biblioteca MuMRT nesse contexto. Mello (2009) *apud* ARIZA (2005)⁴⁶, nos esclarece que este, embora traga outra proposta para modelos de composição (algorítmica) auxiliada por computador, contribui efetivamente para a conceituação da biblioteca quando discute uma definição para *materiais musicais* e sua utilização em sistemas de composição algorítmica. Partindo de conceitos mais gerais como notas isoladas, intervalos, células rítmicas, motivo melódico, frase, progressão de acordes ou mesmo seções inteiras de uma peça, o autor do artigo assim se expressa:

[...] a característica mais importante deste conceito (*musical material*) é a sua flexibilidade. Se qualquer combinação de notas pode ser tratada como um *musical material*, então qualquer transformação (*musical procedures*) aplicada a estes materiais produzirá mais materiais. De fato, a história mostra que muitas composições musicais apresentam estruturas que revelam a utilização de diversas transformações sobre uma relativamente pequena coleção de materiais básicos. [...] Tais composições musicais reforçam a evidência de que uma discreta, ainda que escalonável, unidade de representação musical pode contribuir para o desenvolvimento da composição algorítmica; essa unidade poderosa presta-se a ser implementada por meio de um modelo de orientação ao objeto. [...] (MELLO, 2009)

⁴⁶ *An Open Design for Computer-Aided Algorithmic Music Composition*

No contexto do aproveitamento de algoritmos de outras áreas para elaboração e funções específicas, Robert Rowe, no cap. 3 - Live Computer Music, de seu livro *Interactive Music Systems* (1993), observa que Dannenberg (1991)⁴⁷, conduzindo importantes trabalhos no campo do design de sistemas interativos multimídia, nos adverte: “...uma vez que as aplicações (de algorítmicos genéricos) são estendidas às interações em tempo-real”, “[...] algumas das técnicas relativamente estáveis (em outros contextos), quando usadas em sistemas puramente musicais, podem requerer uma revisão. Exemplo disso são “[...] os agendamentos não previstos (de OUTputs), os *nonpreemptive scheduling processes*.” Estes, por sua vez, em algum sentido, ainda orientariam as escolhas de saída do nosso sistema. A seguir aprofundaremos um pouco mais nas características e utilizações da biblioteca no desenvolvimento do nosso programa.

2.1.2 A biblioteca MuMRT (Musical Material Real Time)⁴⁸

MuMRT é uma biblioteca criada para composição algorítmica. O termo MuM se refere a *Musical Material* (Material Musical). Esse *framework* foi desenvolvido para facilitar a manipulação de informações, ao se implementar algoritmos de composição musical, e é construído em torno da ideia de "Material Musical", um conceito aberto e aplicável a qualquer quantidade de informação musical que possa ser manipulada para criar informação musical mais complexa. Nesse sentido, um material musical pode ser um motivo, uma melodia, um acorde, um coral com várias vozes, uma obra orquestral completa ou simplesmente um único intervalo musical. Em outras palavras, qualquer material do qual um compositor possa extrair materiais para novas composições. A biblioteca MuMRT modela essa informação em uma classe de objetos que pode armazenar qualquer combinação de notas e vozes. MuMRT é baseado num concepção discreta de eventos sonoros, na qual uma composição é, em última instância, representada como um conjunto organizado de notas. Embora exista suporte para a manipulação e armazenamento de informação sobre timbre, o foco principal é na organização de ritmos e alturas. Assim, o *framework* não é particularmente útil para a construção de música centrada na transformação contínua do som. A biblioteca MuMRT foi construída com a linguagem C++ standard (ANSI). Consequentemente é facilmente portátil para várias plataformas na forma de código fonte.

⁴⁷ *Practical aspects of Midi conducting programs*

⁴⁸ Disponível em: <<https://github.com/carlosemello/MuM>>

*MuMaterial*⁴⁹ é a classe mais importante da biblioteca MuMRT. Ela incorpora a ideia principal no design do *framework*, a de um objeto flexível que pode conter qualquer combinação de notas distribuídas em vozes. É através desse objeto que o código do usuário acessa e manipula as notas de modo a criar algoritmos específicos para uma composição. *MuMaterial* incorpora diversos métodos para inserir notas, copiar vozes inteiras ou carregar arquivos de partituras no formato Csound⁵⁰. Oferece também diversas transformações básicas de seus dados internos tais como transposição, re-escalamento temporal, inversão, retrogradação, concatenação, mixagem e muitos outros.

A classe inclui métodos que permitem ao código do usuário organizar sua estrutura interna em vozes separadas, para uso com diferentes instrumentos ou trilhas. Entretanto, embora esses métodos comportem a maioria das operações de rotina necessárias para lidar com essa estrutura, tais como criar remover copiar e limpar cada voz independentemente, o uso da classe é frequentemente mais intuitivo quando essas tarefas são realizadas de forma transparente por um de seus objetos. A classe *MuMaterial* foi construída de modo a exigir o mínimo de especificação pelo usuário, ao acessar cada um de seus recursos. Para conseguir isso, a maioria dos métodos é sobrecarregada, de modo que existe frequentemente uma chamada que aplica transformações sobre todo o conteúdo do objeto e outra, ou outras, que direcionam a ação para vozes específicas dentro do objeto. Assim, se o código do usuário usar apenas as chamadas "gerais", estará operando num escopo mais intuitivo, ou seja, sobre todo o conteúdo, mas quando fizer sentido, poderá operar também sobre uma voz específica.

A saída de dados em MuMRT pode ser realizada em arquivos de partitura do programa Csound. Porém, em sua versão mais recente, a biblioteca incorporou uma nova classe de objetos (*MuPlayer*)⁵¹ para lidar com a performance musical em tempo real. Isso significa que ela possibilita também a geração e transmissão assíncrona de eventos MIDI, de modo que se possa tocar o conteúdo de qualquer material durante a execução do programa. De forma semelhante, a biblioteca implementa um mecanismo de entrada de dados via MIDI (*MuRecorder*)⁵², para que um programa construído com suas classes possa receber notas em tempo real para usar nos algoritmos de composição.

⁴⁹ Código: Definição da Classe no Anexo 1, pág. 63

⁵⁰ Conforme exemplo do apêndice 5, pág. 58

⁵¹ Descrição da Classe *MuPlayer* disponível em: <<https://github.com/carlosemello/MuM>>, entre outras

⁵² Descrição da Classe *MuRecorder* disponível em: <<https://github.com/carlosemello/MuM>>, entre outras

2.1.3 MuMRT e o Improvisador

Ambos os mecanismos de comunicação de informações MIDI mencionados acima foram utilizados na implementação do sistema de improvisação descrito no presente trabalho. A classe de entrada MIDI possibilitou a interação com conteúdo musical produzido pelo usuário durante a execução do programa, de modo que esse conteúdo pudesse ser armazenado, segmentado e modificado para produzir possíveis reações ao interlocutor musical. A classe de *Playback*, por sua vez foi central para que o programa pudesse ser tocado em tempo real, apresentando assim as novas frases musicais para a interação com o músico. Além disso, diversas partes do sistema incorporaram objetos da biblioteca (*MuMaterial*, *MuNotes*⁵³, etc.) para armazenar, copiar, editar as informações musicais geradas pelo programa, de modo que essas pudessem ser facilmente multiplicadas, testadas e modificadas. Diversos componentes centrais do programa, tais como os *Segmentadores* e os *Transformadores*, fazem uso extensivo dos objetos de MuMRT. Particularmente, a flexibilidade da classe *MuMaterial*, permitindo a expansão contínua e a facilidade de se multiplicar conteúdos musicais relativamente complexos de forma simples e rápida, ajudaram armazenar e organizar a enorme quantidade de dados gerados pelas rotinas do programa. Abaixo exemplificamos os componentes da biblioteca MuMRT usados na elaboração de funções específicas do nosso sistema:

⁵³ Descrição da Classe *MuNote* disponível em: <<https://github.com/carlosemello/MuM>>, entre outras

C cs_pitch	Pitch information in Csound format
C EventQueue	Event Queue - MIDI events to be played
C MuError	Error Class
C MuMaterial	Musical Material Class
C MuMIDIBuffer	MIDI Buffer structure
C MuMIDIMessage	MIDI event structure
C MuNote	Note Class
C MuParamBlock	Parameter Block Class
C MuPlayer	MuPlayer Class
C MuRecorder	MuRecorder Class
C MuVoice	Voice Class

Figura 5 – *Classes da biblioteca MuMRT*

Fonte: *site GitHub*

Observamos que o anexo deste trabalho incluem uma lista dos métodos disponíveis nas principais classes da biblioteca MuMRT.

2.2 Segmentadores⁵⁴ e critérios:

Uma vez debruçados sobre o desenvolvimento das funções do sistema interativo, nos deparamos com algumas questões iniciais, e estas por sua vez, nos conduziram, entre outras, à proposta de criação do que denominamos *Segmentadores*. As primeiras questões que nos ocuparam nesse sentido, surgiram a partir da prerrogativa de que, se a entrada (INput) for contínua, uma vez que o computador estaria sempre aberto aos gestos introduzidos pelo músico, qual deveria ser o momento em que a máquina faria sua avaliação do INput? Em que parte ou momento do INput ela deveria ser iniciada? Em que parte ou momento do INput ela deveria terminar? Estas questões, entre outras, nos trouxeram as bases para a elaboração de critérios para as primeiras análises que o programa deveria realizar. Sendo assim, guiados também pela experiência musical, tais critérios nos orientaram para a elaboração de *Segmentadores* do INput. Os segmentos resultantes serão armazenados, transformados, avaliados e selecionados nas fases

⁵⁴ O código desta Classe, entre outros ainda em desenvolvimento, encontram-se no apêndice 7 desta dissertação

seguintes da interação. A seguir descrevemos os critérios de funcionamento e implementação de alguns *Segmentadores* de INputs, já em uso no programa, a começar pelo critério Temporal (para outros segmentadores ver apêndice 4). Propomos assim os seguintes *critérios de segmentação*, partindo de valores pré-definidos para os sorteios das durações dos segmentos (em segundos) com os seguintes valores: 1, 2, 3, 4, 6, 8, e 10.

- a) Gesto a Gesto - Começando do ponto “0”, tempo (zero) da interação, é feito um ‘sorteio’ da duração do segmento, entre alguns valores pré-definidos. O próximo segmento tem início após a última nota do segmento anterior e um novo ‘sorteio’ é realizado para determinar sua duração e assim por diante.
- b) Aleatório - São ‘sorteados’, tanto o ponto de início quanto a duração de cada segmento, sendo esta última com valores de tempo pré-definidos.
- c) Tempo a Tempo - Começando do ponto “0”, tempo (zero) da interação, entre valores pré-definidos, é feito um ‘sorteio’ da duração do segmento, porém neste caso uma segmentação é realizada a partir de cada tempo (em segundos) da interação.
- d) Nota a Nota - Segmentam a cada nota/ altura
- e) Por Direcionalidade - Segmentam toda vez que o INput mudar de direção (Asc. ou Desc.)
- f) Por Grau conjunto - Segmentam os trechos de grau conjunto separados por salto
- g) Por Saltos - Segmentam os trechos de saltos separados por grau conjunto

2.3 Critérios para Avaliação de Segmentos Melódicos

Após as segmentações, a etapa de avaliação inicial realizada pelo programa, corresponde à análise de características melódicas dos segmentos do INput a partir de 6 critérios. O objetivo aqui é entrar efetivamente no material musical dos segmentos, buscando compreender sua estrutura melódica e rítmica, bem como características específicas das variantes, combinações e gradientes dessa estrutura. A esta análise corresponderá, em desenvolvimentos futuros do código, um índice de pesos das porcentagens de variação de cada característica do segmento em relação ao INput. Assim, reconhecidas as características de cada um dos segmentos do INput, o programa deverá proceder à etapa de transformações seguindo orientações sobre os níveis de S&C desejados para os OUTputs. Desejamos assim, averiguar também, a hipótese de que, indo

de 1 a 6, as alterações realizadas nos primeiros critérios corroboram para um maior contraste na modificação do material que os últimos, sendo a manipulação desses, ao contrário, direcionados para as interações de maior similaridade. Abaixo encontramos uma lista dos critérios, partindo do de maior peso, em termos do grau de alteração do material:

Critérios para Avaliação de Segmentos Melódicos	Pesos em % de alteração do material
1. Tom/Modo (Conjunto de Alturas)	
2. Rítmica (Durações e Padrões)	
3. Direcionalidade (Contorno Melódico)	
4. Predominância de graus Conjuntos e Saltos	
5. Densidade Melódica (Alturas/ tempo)	
6. Região de oitavas (Tessitura)	

Tabela 1 – *Critérios para Avaliação de Segmentos Melódicos*

Fonte: *O Autor*

Para vários desses critérios já podemos reconhecer suas correlações com os transformadores a seguir, podendo planejar algumas linhas de ações, indo desde as segmentações, passando pela análise e transformações, até a escolha de OUTputs, numa direção específica de *similaridade ou contraste*. As alterações de porcentagens dessas características, relacionadas aos níveis de *S&C*, estão descritas no apêndice 2.

Como exemplo das técnicas de composição algorítmica, Alpern (1995)⁵⁵ nos trouxe importantes conceitos, tais como *Randomness*, *Melodic Contour* e *Meta-Contour*⁵⁶, entre outros. Estes nos serviram diretamente para a elaboração de funções de análise das características dos segmentos do INput, assim como dos possíveis OUTputs da interação, uma vez que embasaram

⁵⁵ *Technics for Algorithmic Composition*

⁵⁶ *Randomness*: Uso de números randômicos para juntar (ou formar) fragmentos melódicos/ *Melodic Contour*: Visão geral sobre o número de intervalos ascendentes ou descendentes, bem como dos picos e vales de uma melodia/ *Meta-Contour*: Análise do *Contorno Melódico* de sub-fragmentos de uma melodia, sendo o compasso um exemplo disso. (tradução nossa)

alguns dos critérios de avaliação de segmentos melódicos na implementação do código. A classe, em desenvolvimento, referente à análise de características do material é no código reconhecida por *AnaliseIndividual*⁵⁷. Os desdobramentos desta Classe orientarão, futuramente, o desenvolvimento do Gerenciador para as escolhas de OUTputs do programa.

2.4 Alguns Transformadores⁵⁸ e seus critérios:

Uma vez segmentados e analisados, os fragmentos do INput são direcionados para o processo que, efetivamente, irá modificar o material. Esta etapa consiste em aplicar tipos de *Transformadores*, em número de vezes e ordem variável, e com objetivo de atingir por variação do material, em maior ou menor grau, níveis de *S&C* previamente definidos em relação ao INput para determinada interação. Atingido o nível desejado, um OUTput será escolhido. Não alcançado o nível desejado, os segmentos são novamente direcionados para que novas *Transformações* tomem curso. Lembramos aqui, que todo segmento será armazenado na memória (“*Building Blocks*”) para possível re-introdução futura no curso da interação. Para a proposição e construção dos *Transformadores* tomamos por base, além da experiência, conceitos de variação motivica oriundos dos *Fundamentos de Composição Musical* de Arnold Schoenberg. As classes referentes a esta função são no código reconhecidas por *TransformationFactory* e *TransformationClassTree*.

Em seguida, descrevemos a implementação de dois *Transformadores* sobre os segmentos gerados na etapa anterior (para outros *Transformadores* ver apêndice 3). Estes, com caráter de transposição do material e cada qual contendo ‘modos’. Nos testes tomamos por referência as alturas da escala de Dó Maior. São eles portanto:

- a) ChangeNote - Alteração (transposição) de uma única nota. É uma transformação que aceita de 0 a 2 argumentos e, dependendo do número de argumentos definidos, altera seu comportamento. Os dois argumentos possíveis são *nota alvo* (nota a ser modificada no segmento) e quantos graus diatônicos esta nota deve ser modificada.

⁵⁷ O código da Classe *AnaliseIndividual*, entre outros ainda em desenvolvimento, encontram-se no apêndice 7 desta dissertação

⁵⁸ O código das Classes *TransformationFactory* e *TransformationClassTree* da função de Transformação do programa, entre outros ainda em desenvolvimento, encontram-se no apêndice 7 desta dissertação

- Modo 0: (sem argumentos) ‘Sorteio’ da nota alvo que é transposta no segmento, bem como da transposição em si; esta podendo variar entre -3 (descendente) e +3 (ascendente) intervalos/ graus diatônicos, a partir de uma dada escala, até uma 4ª Justa ou Aumentada de distância, portanto.
- Modo 1: (argumento único) A nota alvo é determinada pelo argumento (aqui determinamos partir o segmento ao meio). Em seguida a transposição segue os mesmos passos do item anterior, ou seja, por meio de um ‘sorteio’, esta deve variar entre -3 (descendente) e +3 (ascendente) intervalos/ graus diatônicos, a partir de uma dada escala, até uma 4ª Justa ou Aumentada de distância, portanto.
- Modo 2: (dois argumentos) A transposição agora é completamente determinística, obedecendo o alvo dado e a transposição solicitada. Neste caso escolhemos o alvo no meio do material e este será transposto uma 5ªJ ou diminuta acima, dependendo da nota escolhida.

*Observamos que: As alterações de alturas entre -3 e +3 graus diatônicos permitem transposições de 2ª m, 2ª M, 3ª m, 3ª M, 4ªJ e 4ªAumentada, considerando a escala escolhida (Dó Maior)

- b) ChangePhraseNote – Nesta função os argumentos seguem os mesmos procedimentos dos modos do ChangeNote. Porém aqui, a nota alvo representa o ponto a partir do qual o segmento é transposto. Para valores positivos a transposição ocorrerá a partir da nota alvo até o fim do segmento, para valores negativos a transposição ocorrerá da nota alvo para trás, ou seja, dela para o início do segmento.

Partindo de 1 único INput, aplicando-se 3 *Segmentadores* e 2 *Transformadores*, o computador tem à disposição 18 processos diferentes, gerando inúmeros OUTputs entre válidos e não válidos. Na sequência à aplicação de outros tipos de *Segmentadores* e *Transformadores*, o Gerenciador/ Inteligência deverá avaliar os segmentos - isolados ou em comparação aos INputs e OUTputs anteriores, com a finalidade de identificar os níveis de *Similaridade & Contraste* (ver

apêndice 2) entre eles, por meio dos Critérios de Análise de segmentos/ frases, gerando e/ ou selecionando então OUTputs para comporem o ciclo, completando assim a interação.

2.4.1 Outros Transformadores

Dando prosseguimento ao desenvolvimento do programa interativo músico-computador, buscamos ampliar e diversificar a utilização de *Transformadores* para que pudéssemos aprofundar as ‘nuances interativas’ das sintaxes do sistema. Visando aprimorar o sistema com objetivo de aproximá-lo de uma interação com níveis de complexidade com tendência às interações reais, ou seja, entre músicos, partimos para o refinamento do processo de variação do material. Buscamos, aqui, dar ao sistema um mínimo de variantes que o permita avaliar e categorizar, futuramente, os segmentos transformados para níveis específicos de *S&C* definidos em cada interação. Os *Transformadores* seguintes foram desenvolvidos mais recentemente, a partir de funções da biblioteca MuMRT, e complementam a estrutura da fase atual de testes do programa. São eles:

- c) ExtractNote - Retira notas dos segmentos do INput segundo critérios

- d) AddNote - Adiciona notas aos segmentos do INput segundo critérios

- e) RepeatNote - Escolha de notas dos segmentos do INput a serem repetidas um número de vezes variável, porém com outro ritmo em relação ao segmento transformado

- f) Re-ordenação - Re-organiza alturas e durações:
 - 1. CicleRithm - Mantém a ordem rítmica dos segmentos do INput, alterando apenas as alturas segundo critérios

 - 2. CiclePitch - Mantém a ordem de alturas dos segmentos do INput, alterando apenas os ritmos segundo critérios

 - 3. Retrógrado - Inverte a ordem das alturas e durações

 - 4. Permutação Aleatória - Sorteia a ordem de alturas e durações segundo critérios

Apresentamos a seguir uma tabela com um panorama das funções de Segmentação, Transformação e Análise atual do programa:

SEGMENTADORES	CRITÉRIOS para AVALIAÇÃO de SEGMENTOS (%)	TRANSFORMADORES	Níveis de SIMILARIDADE & CONTRASTE
a) Tempo a Tempo	1. Tom/ Modo (notas/ alturas)	a) ChangeNote	1. SHigh
b) Gesto a Gesto	2. Rítmica	b) ChangePhrase	2. SMedium
c) Nota a Nota	3. Direcionalidade/ Contorno melódico	c) AddNote	3. SLOW
d) Direcionalidade	4. Predominância de Graus conjuntos e Saltos	d) RepeatNote	4. CLOW
e) Grau Conjunto	5. Densidade melódica (notas/ tempo)	e) ExtractNote	5. CMedium
f) Salto	6. Região de oitavas (Tessitura)	f) Re-ordenação: 1. CicleRithm 2. CiclePitch 3. Retrógrado 4. Permutação Aleatória	6. CHigh

Tabela 2 – Segmentadores, Critérios de Avaliação de Segmentos Melódicos, Transformadores e níveis de Similaridade & Contraste

Fonte: O Autor

2.5 Blocos de Transformadores (Gerenciador/ ‘Inteligência’)

Ao determinarmos protocolos que gerem OUTputs desejáveis quanto aos níveis de *S&C* em comparação com o Input, bem como a partir da análise inicial dos segmentos do INput, propomos:

- a) a combinação de tipos de *transformadores e segmentadores*
- b) a ordem em que os transformadores se aplicam aos segmentos melódicos
- c) o número de vezes que determinados arranjos de transformadores são aplicados ao INput e seus segmentos

À essas combinações e arranjos chamamos *Blocos de Transformadores*, os quais, somados à análise das características dos segmentos melódicos, funcionarão como ferramentas base da ‘*Inteligência*’ do programa. Esta, mesmo que rudimentar, ao realizar a análise dos segmentos melódicos oriundos das aplicações desses blocos, selecionará os OUTputs desejados, segundo uma deliberação prévia da interação, em relação ao nível de *S&C*, bem como arranjos ou combinações desses níveis para sessões específicas de interações. Lembramos que esta análise é feita com base nas características das frases melódicas do INput e avaliação por comparação das porcentagens de características transformadas nos possíveis OUTputs já segmentados e transformados. Se o nível determinado não for atingido pelos *Blocos de Transformadores* aplicados numa primeira tentativa, uma cópia dos segmentos é direcionada para a memória ou “*Building Blocks*”, enquanto os segmentos já transformados passam novamente por outras transformações pela aplicação de um mesmo ou outro *Bloco de Transformadores*, seguindo-se uma nova análise. Os segmentos transformados e guardados na memória passam a constar como material repositório, passível de novas transformações, para escolha de possíveis OUTputs do programa, para esta ou futuras interações. Uma vez observados os efeitos dos transformadores, aplicados isoladamente sobre os INputs segmentados, pudemos averiguar aqueles que geram maior ou menor grau de alteração do material. Assim, os primeiros *Blocos de Transformadores* foram sugeridos para que se pudesse testar sua eficácia para cada um dos 6 níveis de *S&C* (ver apêndice 2). Abaixo exemplificamos os blocos inicialmente propostos para cada nível:

BLOCOS de TRANSFORMAÇÕES	Bloco 1 (1x) - 2a2 -	Bloco 2 (2x) - 3a3 -	Bloco 3 (combinações) - 4a4 -
Níveis de SIMILARIDADE & CONTRASTE			
1. SHigh	<ul style="list-style-type: none"> • ChangeNote • AddNote 	<ul style="list-style-type: none"> • ChangePhrase • RepeatNote • AddNote 	<ul style="list-style-type: none"> • ExtractNote • AddNote • ChangePhrase • RepeatNote
2. SMedium	<ul style="list-style-type: none"> • ChangePhrase • RepeatNote 	<ul style="list-style-type: none"> • ExtractNote • CycleRithm • ChangePhrase 	<ul style="list-style-type: none"> • ExtractNote • CycleRithm • ChangePhrase (2x)
3. SLow	<ul style="list-style-type: none"> • ChangeNote • CyclePitch 	<ul style="list-style-type: none"> • AddNote • ChangePhrase • RepeatNote 	<ul style="list-style-type: none"> • AddNote • ChangePhrase • RepeatNote • CyclePitch
4. CLow	<ul style="list-style-type: none"> • CycleRithm • ChangePhrase 	<ul style="list-style-type: none"> • CyclePitch • AddNote • ChangePhrase 	<ul style="list-style-type: none"> • CyclePitch • ChangePhrase • RepeatNote • ExtractNote
5. CMedium	<ul style="list-style-type: none"> • Retrógrado • RepeatNote 	<ul style="list-style-type: none"> • CycleRithm • Retrógrado • ChangePhrase 	<ul style="list-style-type: none"> • CycleRithm • ChangePhrase (2x) • AddNote
6. CHigh	<ul style="list-style-type: none"> • Permutação Aleatória • ExtractNote 	<ul style="list-style-type: none"> • Permutação Aleatória • RepeatNote • CyclePitch 	<ul style="list-style-type: none"> • Permutação Aleatória (2x) • Retrógrado • ExtractNote

Tabela 3 – *Blocos de Transformadores para cada nível de Similaridade & Contraste*

Fonte: *O Autor*

Na tabela acima, podemos observar a proposição de *Blocos de Transformadores*, atuando em conjuntos de 2, 3 e 4 Transformadores por Bloco. Na linha 3, por exemplo, para testes do sistema propomos, em uma interação com nível de Similaridade Low (SL), as combinações de transformadores organizados pelos Blocos 1, 2 e 3. Nesta ordem buscamos averiguar gradientes internos daquele nível de interação. Assim, organizados de cima para baixo e da esquerda para a direita, ‘como se lê’, podemos inferir, por exemplo, que o primeiro dos Blocos 1 teria nível de Similaridade High (SH) máximo, assim como o último dos Blocos 3 teria o nível de Contraste High (CH), ou seja, máximo.

Aqui surgindo novamente a necessidade de aprofundamento para uma maior acurácia na determinação de ações de segmentação e transformação que conduzam aos níveis de *S&C* desejáveis. Sobre esta questão Whitby (1953, p. 62)⁵⁹, nos alerta, já que desejamos estabelecer ‘pesos’ sobre as características melódicas a serem alteradas por transformadores durante as

⁵⁹ *I.A. Inteligência Artificial: Um guia para Iniciantes*

interações com nosso programa: “O problema é que quando há mais de um *Percéptron*⁶⁰, não sabemos qual das conexões precisam de ajustes. Todos os *percéptrons* são conectados e influenciam uns aos outros.” Nosso projeto trabalha com a ideia de que, sendo orientado para análise de características de segmentos melódicos e sua comparação com possíveis INputs já transformados, o programa ‘decidiria’ sobre os possíveis OUTputs sob orientação da régua de *similaridade e contraste*; esta, construída em correlação com transformadores e Blocos de Transformadores já orientados para determinados níveis, uma vez testados em grupo e/ ou separadamente e averiguada sua eficácia quanto aos níveis alcançados. Daí a necessidade de mais testes para implementação de partes do sistema de gerenciamento e calibragem de pesos das variantes melódicas em relação aos níveis desejados de *S&C* para os OUTputs da interação.

3. Resultados & Conclusão

Ao propormos um sistema de interação musical instantânea entre músico e computador, realizamos uma Revisão de Literatura. Esta, por sua vez, deu-se nas áreas de Sistemas interativos músico-computador, Improvisação e Criação musical coletiva. Construído em linguagem C++ e baseado na biblioteca MuMRT (Musical Material Real Time), o programa que aqui propomos deve receber INputs melódicos via MIDI, segmentá-los, transformá-los e compará-los, tomando por base critérios de análise, para avaliar graus de *S&C* entre os segmentos transformados e os segmentos do INput e ‘decidir’ sobre os possíveis OUTputs da interação. Nesse contexto lembramos que apenas os parâmetros de altura e duração fizeram parte da análise. Ao realizarmos diversos testes e práticas, a saber, entre outros: Disciplina da graduação/ UnB - “Oficina de construção e improvisação com instrumentos eletrônicos” e “Oficina de Construção

⁶⁰ a) An IBM 704 – a 5-ton computer the size of a room – was fed a series of punch cards. After 50 trials, the computer taught itself to distinguish cards marked on the left from cards marked on the right. It was a demonstration of the “perceptron” – “the first machine which is capable of having an original idea”, according to its creator, Frank Rosenblatt. (Melanie Lefkowitz) - <https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon>

b) Segundo Blay Whitby, em seu livro I.A. Inteligência Artificial (1959), *Percéptron*, [...] “é um dispositivo primário no desenvolvimento de redes neurais. Com o objetivo de fazer com que computadores funcionem como o cérebro humano, ou seja, treinando e fazendo escolhas; vários percéptrons são associados por sistemas de ‘pesos’”. Tendo por meta ‘evoluir’ na direção de uma adaptação, busca-se gerar mais *outcomes* desejáveis a partir de uma dada tarefa de análise, transformação, classificação, entre outras tantas.

de Flautas de PVC” - IV Flautas Gerais/ Diamantina-MG, procuramos aprimorar e validar algumas das ferramentas e condicionantes sintáticas do sistema. Em ambas oportunidades foi aplicada a metodologia e conceitos desenvolvidos para as oficinas e que se baseiam na criação musical coletiva e instantânea por meio da improvisação. Na ocasião foram testados e aprimorados conceitos e práticas relacionados à proposição do sistema interativo músico-computador por meio da manipulação de material melódico. Portanto, com o desenvolvimento do programa e buscando aprofundamento em nossas indagações procuramos responder, ainda que parcialmente, à questão de nossa pesquisa: *Tomando por base a interação músico-computador, como a modelagem desses ‘Princípios de criação musical’ pode contribuir para a compreensão de contextos composicionais coletivos, Instantâneos e improvisatórios?*

3.1 Discussão e Análise de Dados

Nesta seção, apresentamos alguns resultados dos testes do sistema interativo proposto, assim como uma análise dos dados levantados. A seguir exemplificamos com um segmento do INput, alguns resultados válidos de Transformações realizadas pelo programa em diversas fases de teste. Outros resultados (ver apêndice 6):

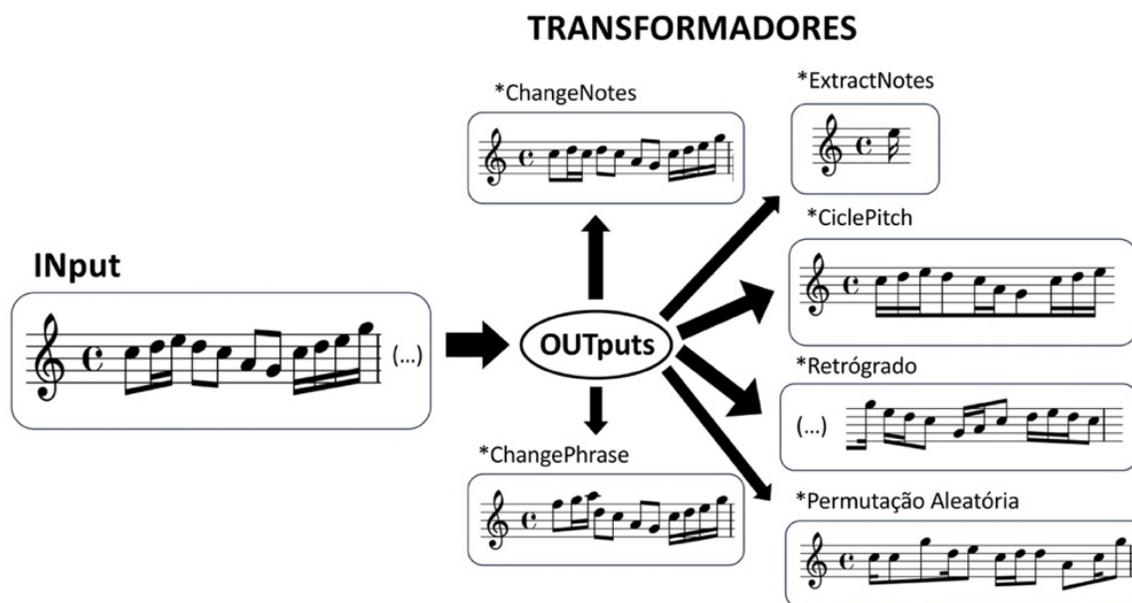


Figura 6 – OUTputs válidos para um dado INput alterado por diversos Transformadores

No quadro acima observamos o segmento do INput e os possíveis OUTputs após serem submetidos aos diversos *Transformadores*. Por meio dos testes, nas diferentes etapas de desenvolvimento do programa, pudemos observar o grau de alteração do material pelo uso de transformadores específicos.

Sendo assim, numa régua contínua, partindo do nível de *Similaridade High (SH)*, passando por *SMedium* e *Slow*, *CLow*, *CMedium* até o nível de *Contraste High (CH)*, e tomando por base o INput e os OUTputs escolhidos no quadro acima, pudemos averiguar, por comparação entre si e com o INput, as correlações entre os segmentos transformados, transformações e suas tendências aos níveis específicos de *S&C* da seguinte maneira:

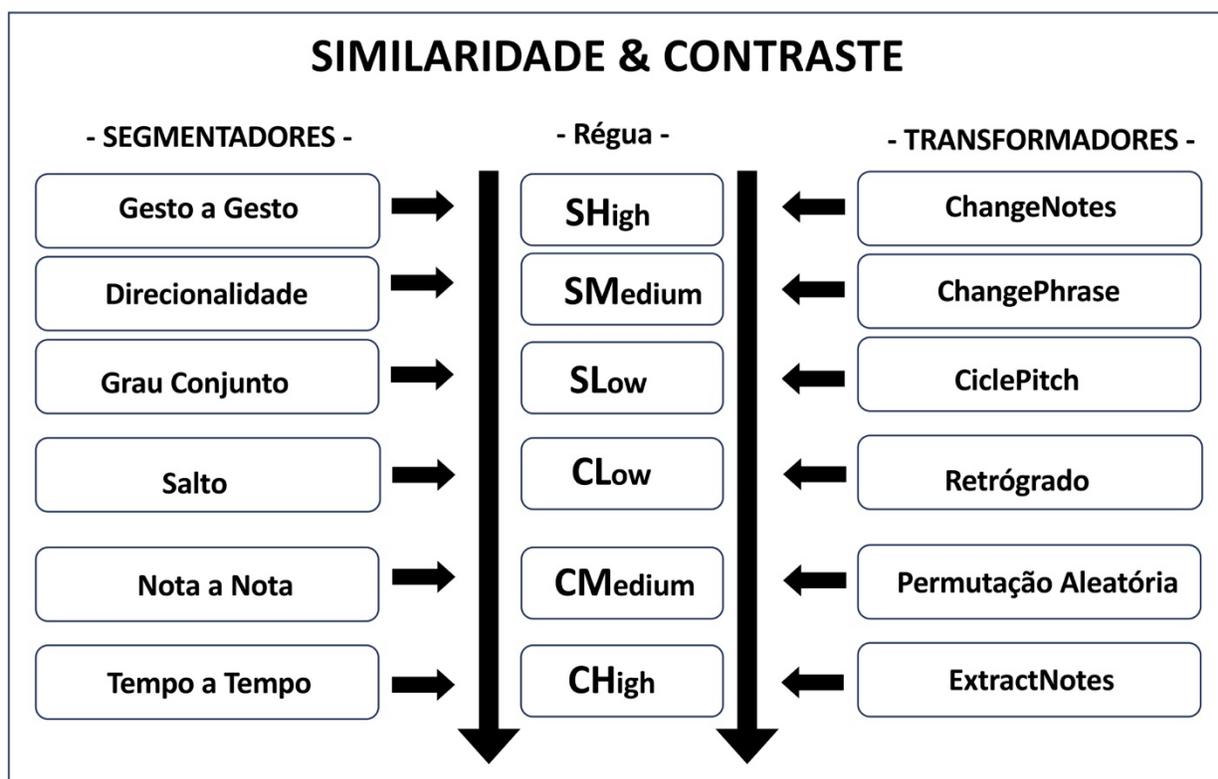


Figura 7 – Correspondência entre Segmentadores, Transformadores e a régua de *S&C*

Fonte: o Autor

No que se refere aos ‘pesos’ e/ ou número de vezes que determinadas *Segmentações* e *Transformações* sejam aplicadas sobre o material, é necessário que se faça uma regulação por meio de avaliações dos OUTputs. Daí também pudemos deduzir e confirmar nossas suspeitas

sobre a tendência que o grau de alteração do material produzido segue pela alteração de cada um dos *Critérios de Análise Melódica*, bem como a aplicação de cada *Segmentador*, *Transformador* e/ ou *Blocos de Transformações* sobre os segmentos do INput. Embora haja nuances, a semelhança visual entre os gráficos abaixo é, em parte, decorrente da proposição de uma régua gradual e contínua no que tange aos níveis de *S&C*, bem como do desenvolvimento de funções que correspondam à ela. A análise e regulagem de pesos das avaliações feitas pelo Gerenciador do sistema ainda precisam ser implementadas.

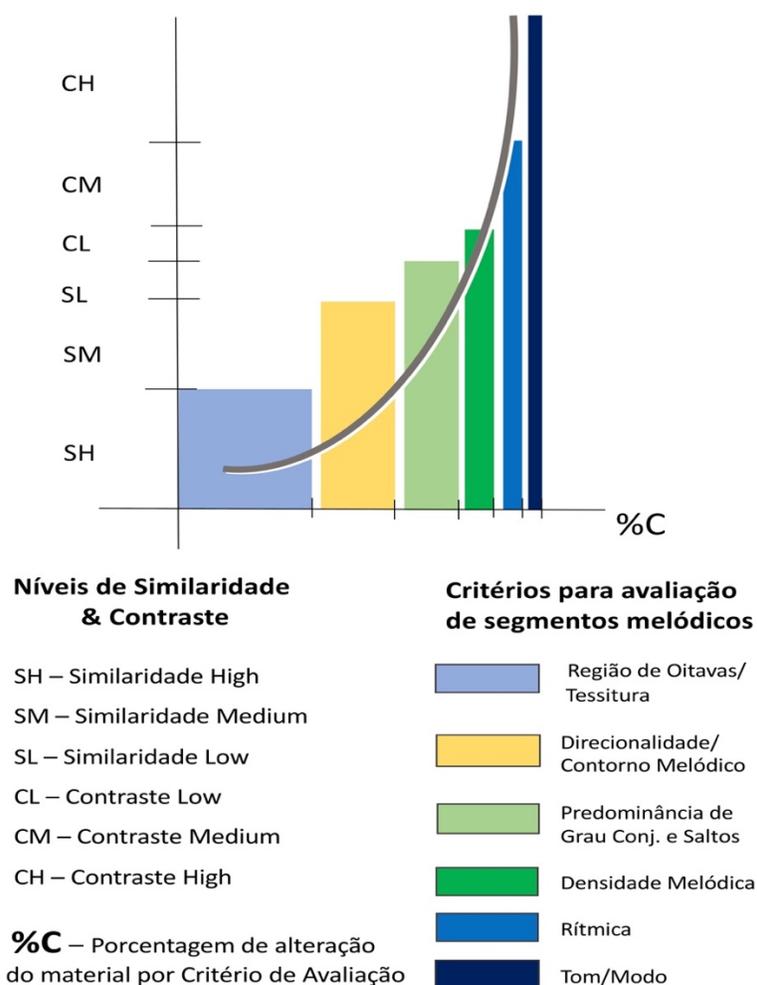


Gráfico 1 - *Critérios de Avaliação de Segmentos Melódicos e níveis de S&C*

Fonte: *o Autor*

No Gráfico 1 observamos, por exemplo, que alterações realizadas por *segmentadores* e *transformadores*, e que afetam o *critério de avaliação de segmentos melódicos* de *Região de*

Oitavas, contribuíram menos em termos de direcionamento do segmento para níveis de contraste. Isto acontecendo apenas se o critério fosse alterado por sucessivas transformações atuando sobre este critério. Já o critério *Tom/Modo*, por exemplo, se aplicado ao segmento melódico pelo transformador *ChangePhrase*, logo apresentaria tendências contrastantes, sendo desnecessária sua aplicação sucessiva para aquela intenção nesta interação. Este exemplo reforça a correlação entre determinados *critérios para avaliação de segmentos melódicos* e níveis específicos de *S&C*.

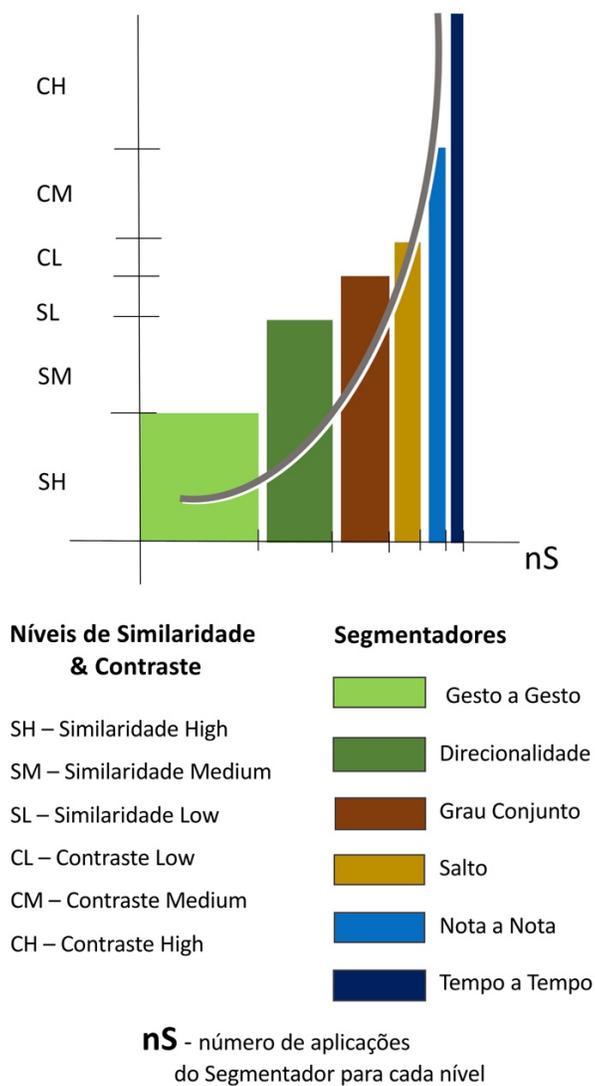


Gráfico 2 – Segmentadores e níveis de S&C

Fonte: o Autor

No Gráfico 2 observamos que, alterações realizadas por *segmentadores* específicos, como por exemplo o *Gesto a Gesto*, contribuiriam mais em termos de direcionamento do segmento do OUTPUT para níveis de similaridade. Sendo necessário pois que o segmento seja alterado por sucessivas segmentações para que atinja níveis de contraste, mesmo que baixos. Já o segmentador *Tempo a Tempo*, por exemplo, se aplicado ao segmento melódico, logo apresenta características contrastantes, sendo desnecessária sua aplicação sucessiva, a menos que a interação seja orientada para níveis altos de contraste. Este gráfico reforça a correlação entre determinados *segmentadores* e níveis específicos de *S&C*.

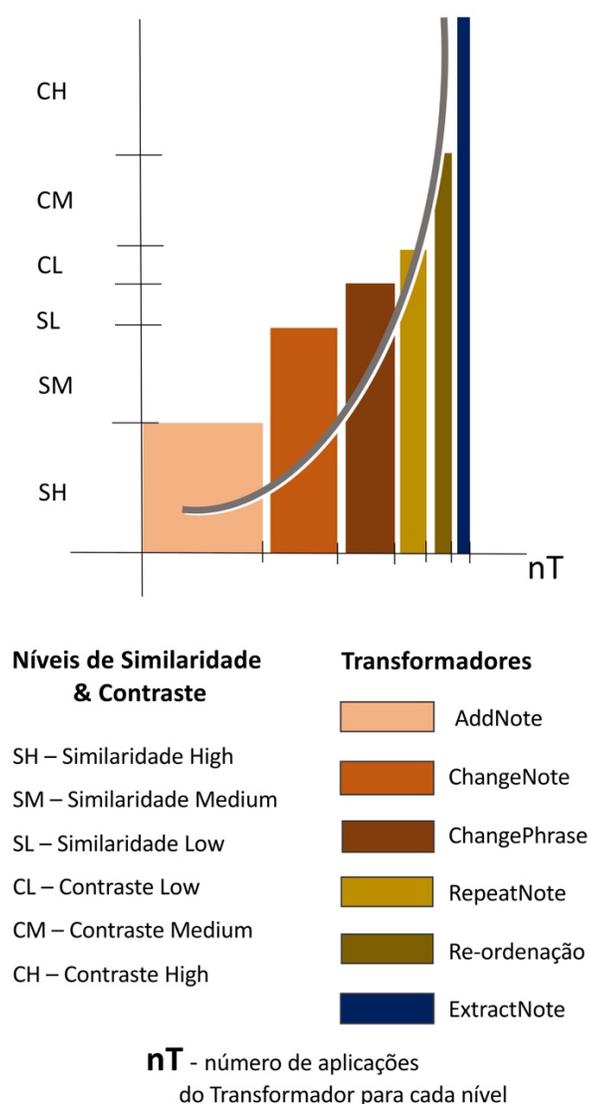


Gráfico 3 – Transformadores e níveis de *S&C*

Fonte: o Autor

No Gráfico 3 observamos que alterações realizadas por *transformadores* específicos, como por exemplo o *AddNote*, contribuiriam mais em termos de direcionamento do segmento para níveis de similaridade. Sendo necessário pois que o segmento seja alterado por sucessivas transformações para que atinja níveis de contraste, mesmo que baixos. Já os transformadores *ExtractNote* ou *Re-ordenação*, por exemplo, se aplicados ao segmento melódico, este logo apresenta características contrastantes, sendo desnecessária sua aplicação sucessiva, para níveis altos de contraste. Este gráfico reforça a correlação entre determinados *transformadores* e níveis específicos de *S&C*.

Numa fase subsequente, o Gerenciador, munido da prerrogativa de escolha do material do *OUTput* para um determinado nível, baseia-se na classificação dos segmentos transformados quanto à *S&C*, e pode ‘decidir’ por uma saída. Utilizando-se de *Blocos de Transformadores*, orientados para o nível específico desejado, a tendência à esta saída torna-se cada vez maior e mais objetiva. Os segmentos não escolhidos para este *OUTput* da interação são então avaliados/classificados e guardados na memória, “*Building Blocks*” para serem re-aproveitados em futuras interações. A seguir um exemplo de interação orientada para o nível de Contraste Low (*CLow*):

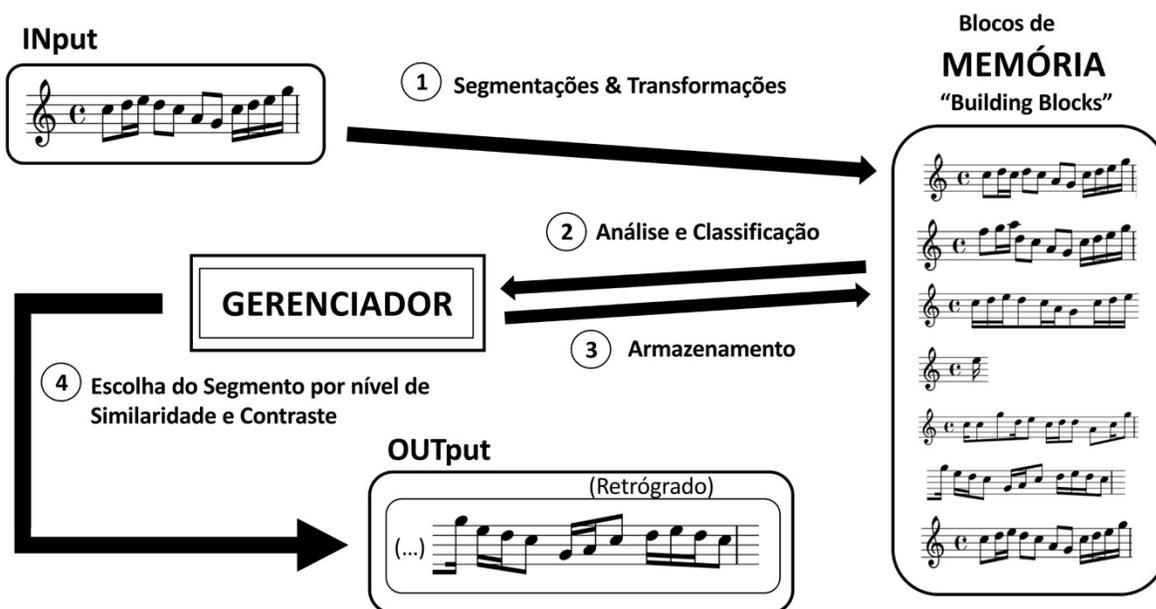


Figura 8 – Interação com nível de Contraste Low (*CLow*)

Fonte: o Autor

Na discussão sobre o tempo de resposta ou “reatividade do sistema” (função ‘Cronos’), sobretudo no que tange às *Segmentações e Transformações* por métodos de ‘sorteio’ de variáveis definidas, procuramos dar ao programa um número de opções de segmentação do INput, para que pudesse gerar, inicialmente, mais OUTputs desejáveis quanto à sua *similaridade* em relação aos INputs, podendo averiguar assim, a aplicabilidade do sistema interativo proposto. Uma outra solução para esta questão foi também a aplicação das variáveis de ponto de corte e duração do trecho segmentado. A partir disso os segmentos continuam a ser sorteados randomicamente, porém agora, com alguns limites estabelecidos. Estes, inicialmente, atuando sobre as interações mais próximas, como por exemplo num sorteio com chances de 70% para escolhas a partir dos últimos 5 registros/ ou segundos (segmentos transformados ou não). Também, sendo aqui estabelecidos valores randômicos para o tempo de resposta de cada OUTput, com cifras entre 2 e 5 segundos a serem sorteados; o que nos justifica, mais uma vez, um aprofundamento ainda maior na elaboração da função ‘Cronos’ do Gerenciador (Inteligência) para o *timing* de saída dos OUTputs. No presente estágio de desenvolvimento do programa, caso os últimos 5 gestos segmentados não sejam válidos, por falta de dados, situação na qual o músico tenha parado de tocar por exemplo, o Gerenciador dispõe, a princípio, apenas dos 5 últimos OUTputs para serem processados, uma vez que estes seriam resgatados da memória recente ou “Building Blocks” do sistema.

Como nos contextos de improvisação sugeridos nas obras abertas de criação instantânea por meio da improvisação guiada dos *Game of pieces* de John Zorn, buscamos dar ao nosso programa alguma orientação no sentido de transformação do material musical, isso se confina inicialmente à régua de *Similaridade & Contraste* e ao nível determinado para a interação. Nesse sentido podemos ainda criar Blocos de níveis de *S&C* para interações, de maneira que o programa siga uma ordem na aplicação de *Segmentadores e Transformadores* orientados para uma ordem de saída em níveis de *S&C* determinados. No entanto, no que tange ao tempo de saída dos OUTputs bem como das relações de presença ou ausência de pulso ainda nos vemos nos desenvolvimentos de condicionantes à participação ou não de determinantes prévios à interação. No caso de OMAX por exemplo, observamos que as funções do sistema são disparadas por uma pessoa durante a interação da máquina com o músico. Esses disparos podem ser agendados ou acessados por algum “gatilho” da interação, podendo determinar mudanças de

abordagem e podendo ocasionar ainda mudanças de seção por contraste nas saídas, ao menos em termos da percepção.

3.2 Considerações Finais

Ao concluirmos esta etapa de nossa pesquisa, fazemos uma revisão de nossos objetivos e metodologia, bem como procuramos estabelecer, em que medida pudemos responder à nossa questão. Nesse sentido avaliamos ainda possíveis aprimoramentos e desenvolvimentos futuros das funções e condicionantes do sistema interativo proposto no cap. 2 desta dissertação. Abaixo incluímos uma tabela atualizada com as funções já aplicadas do programa, embora ainda em desenvolvimento:

PRERROGATIVAS E FUNÇÕES	OBSERVAÇÕES	Projeção para a próxima fase de desenvolvimento	STATUS - Nov. 2020 -
Critérios para Análise de segmentos melódicos	Podem ser acrescidos outros critérios posteriormente	Aguardando outros desenvolvimentos para refinamentos futuros	• APLICADOS
Régua de <i>Similaridade & Contraste</i>	Deve passar por refinamentos ao longo do desenvolvimento do programa	Aguardando outros desenvolvimentos para refinamentos futuros	• APLICADA
Segmentadores	Podem ser acrescidos outros posteriormente	Sugerimos aprimoramentos	• APLICADOS e FUNCIONAIS
Transformadores	Podem ser acrescidos outros posteriormente	Sugerimos aprimoramentos	• APLICADOS e FUNCIONAIS
Função <i>Cronos</i>	As diversas funções associadas ao tempo na música. Ex. Pulso, <i>timing</i> de saída, etc.	Etapa de desenvolvimento das Funções do Gerenciador	• APLICADA PARCIALMENTE/ ESTÁGIO INICIAL (Método de escolhas por sorteio)

Tabelas 4 – *Status do Desenvolvimento das Prerrogativas e Funções APLICADAS do Sistema*

Fonte: o Autor

Nossa atenção aqui volta-se para o fato de que, em grande medida, realizamos o que nos propusemos inicialmente. Após uma vasta pesquisa na literatura propusemos as bases para o

desenvolvimento de um sistema interativo músico-computador. Conceitualizamos uma *estrutura de fluxo de dados* (figura 3, p.18), na qual apresentamos o processo como um todo, em que *Critérios de Análise de segmentos melódicos, Segmentadores e Transformadores* realizam suas funções orientadas para uma régua contínua com níveis de *S&C*, a partir da qual o Gerenciador faria uma escolha para *OUTPUTS* da interação. Observamos aqui que, embora essas funções de segmentação e transformação já estejam implementadas em alguma medida, entendemos que ainda necessitam de refinamento, seja pelo aprimoramento dos critérios pelos quais se baseiam, seja pelo peso que a alteração de cada critério tem no sentido de direcionar as transformações e escolhas de *OUTPUTS* da interação para um determinado nível de *S&C*. Avaliamos ainda que novos testes seriam necessários para uma maior generalização no sentido das correlações entre *Critérios de Avaliação de segmentos, Segmentadores e Transformadores*. Ao observarmos atentamente os gráficos 1, 2 e 3, das págs. 32, 33 e 34, respectivamente, notamos o quão distantes encontram-se os extremos de *SH* e *CH*, enquanto que tornam-se quase indissociáveis os níveis de *SL* e *CL*, por estarem muito próximos, entre si, em termos de alteração do material original. Para a percepção do músico na interação com a máquina, mesmo os mais experientes, por exemplo, as nuances dos níveis intermediários, parecem menos relevantes. É provável, portanto, que as relações de maior ou menor contraste se sobreponham no sentido da percepção do músico. Testes com segmentos do *INPUT* que sejam bem contrastantes entre si, em termos das características melódicas resultantes da segmentação, por exemplo, conduziram nossa análise para uma maior acurácia na aplicação tanto de *Transformadores*, quanto especialmente dos *Blocos de Transformadores* específicos, a exemplo dos que sugerimos na Tabela 2 da pág. 28, ainda em fase de implementação.

Ainda em fases preliminares, e buscando aprimorar as funções do Gerenciador do sistema relacionadas ao ‘*Cronos* na música’, realizamos testes com pedais para ‘*TAP Tempo*’, em que variações de andamento poderão ser assimiladas na avaliação da máquina. Esta classe é chamada *PedalTimer*⁶¹ no código e acreditamos que trará certa ‘organicidade’ para a interação, uma vez que, a ideia de pulso será finalmente introduzida no programa. Esta aplicação terá grande influência, naturalmente, com relação ao *timing* de saída do programa. Desejamos aqui aprimorar as condicionantes de escolha do Gerenciador para as decisões de momento das saídas. Por enquanto nos baseamos em escolhas de probabilidades com valores de tempo previamente

⁶¹ *PedalTimer*: O código desta Classe, entre outros ainda em desenvolvimento, encontram-se no apêndice 7 desta dissertação

sugeridos, como explicitado no parágrafo final do item 3.1 (pg. 36). A pesquisa sobre os *nonpreemptive scheduling processes* ainda fariam parte de nossos aprofundamentos futuros, portanto. Abaixo incluímos uma tabela atualizada com as funções ainda não aplicadas do programa e em diferentes estágios de desenvolvimento:

PRERROGATIVAS E FUNÇÕES	OBSERVAÇÕES	Projeção para a próxima fase de desenvolvimento	STATUS - Nov. 2020 -
Calibragem de ‘Pesos’ das alterações de Critérios para Análise e Comparação de segmentos melódicos	Pesos sugeridos em porcentagem de alteração por Critério de Análise e Comparação	Próxima etapa de testes do programa. (Prévio às funções do Gerenciador)	• NÃO APLICADO
Blocos de Transformadores	Podem ser acrescidos outros posteriormente	Etapa de desenvolvimento das Funções do Gerenciador	• NÃO APLICADO (Preparando fase de testes)
Análise e Comparação entre segmentos melódicos	Função Principal do Gerenciador	Etapa de desenvolvimento das Funções do Gerenciador	• NÃO APLICADO
Memória “Building Blocks”	Armazenamento de segmentos avaliados	Etapa de desenvolvimento das Funções do Gerenciador	• NÃO APLICADO
Linhas de Interação, Engajamento e Sintaxe	“Gramática”/ Blocos de Orientações prévias às Interações	Etapa de desenvolvimento das Prerrogativas do Gerenciador	• NÃO APLICADO
Gerenciador/ “Inteligência”	Análise, Comparação, gestão de Resgates da Memória e OUtputs	Aguarda o desenvolvimento de funções anteriores	• NÃO APLICADO

Tabelas 5 – *Status do Desenvolvimento das Prerrogativas e Funções NÃO APLICADAS do Sistema*

Fonte: o Autor

Com relação às questões de estrutura formal das interações, observamos que não pudemos nos aprofundar até o momento em termos de aplicação dessas propostas no código propriamente dito. No entanto entendemos que sejam relevantes para desenvolvimentos futuros

do sistema interativo. O que temos até aqui seriam esboços de estruturas pré-estabelecidas para a entrada do programa em determinada interação. Exemplo disso seriam formas tais como ABA, AB, etc. Seguimos, no entanto, realizando pesquisas para aprofundamentos desta função. Daí justificamos também que o que entendemos agora, como o desenvolvimento da pesquisa, e que denominamos, ao longo do trabalho, por *Sintaxes*, *Linhas de Interação*, “*Regras de Engajamento*” ou mesmo *Nuances interativas*, seriam ainda desenvolvimentos decorrentes do aprimoramento e refinamento de todas as funções do sistema, além da criação de novas funções, um nível de complexidade na interação acima do que nos encontramos no momento. Representariam pois, estas “Linhas”, uma abordagem, uma postura, um viés de comunicação previamente estabelecido para a interação como um todo ou mesmo uma seção específica da interação. Confirmamos aqui a explicação dada por Pressing (1998) em relação à “gramática”, não no sentido do desenvolvimento de uma nova, caso da abordagem feita pela Improvisação não-idiomática, mas no sentido mais tradicional; da forma e dos desenvolvimentos motivados por exemplo. Concluimos em certa medida, em concordância com Schaub *et al.* (sd), que ainda há um certo ‘obscurantismo’ nesta prática de interação com a máquina, não sendo apenas os OUTputs não-válidos um exemplo disso, mas também ‘aquela gramática’ mesmo que parece precisar ser reformulada constantemente à medida do desenvolvimento do programa. A ‘mão dupla’, na comunicação entre o músico e o computador, à qual Sundberg (2005) se refere, para nós ainda parece uma realidade um tanto distante pelos estágios atuais de desenvolvimento do sistema interativo. No entanto ela surge aos poucos pela aplicação de funções e surgimento de novas relações. Isso nos dá uma ideia do nível de complexidade que o código deve alcançar para que o programa atinja certa ‘organicidade’ de interação. O aprofundamento na pesquisa por novas funções, em algoritmos dos mais distintos, pode nos dar ainda algum *insight* nesse sentido, atentos às questões levantadas por Dannenberg (1991), com relação ao uso de algoritmos de outras áreas.

Embora tenhamos optado pela improvisação enquanto abordagem no “*Engajamento*” com a máquina, e tendo em nossa revisão de literatura nos baseado, também, em trabalhos de modelagem de estilo, reforçamos nossa busca por uma ‘abordagem neutra’, embora não nos reconheçamos totalmente, dentro do que seja compreendido por Improvisação Livre e/ ou não-idiomática. Daí conclusão de que precisamos aprofundar a pesquisa, também ampliando seu escopo para os elementos de Intensidade, Timbre, Harmonia, etc. Lembrando que desejamos utilizar os resultados desta em projetos nas áreas de produção musical, trilhas sonoras e arranjos,

bem como na exploração de novas possibilidades de performance, procuramos ampliar, em tempo, nossa proposta artística na flauta transversal a partir de um sistema interativo em estágio funcional.

Uma vez observados os aprofundamentos da pesquisa e resultados do desenvolvimento, do sistema interativo músico-computador proposto, reforçamos a validade deste trabalho enquanto metodologia de composição e musicalização coletivas por meio da improvisação. Pudemos observar a evolução interativa que os integrantes das “Oficinas de criação musical coletiva”, realizadas durante o período da pesquisa, experimentaram, tanto pela instrumentalização absorvida nas práticas de improvisação orientadas, quanto pelos esclarecimentos sobre as condicionantes de construção do programa. Voltando à nossa questão, averiguamos, também, que a abordagem de orientação e aprimoramento da percepção dos integrantes das interações para as relações e possíveis níveis de *Similaridade & Contraste*, favoreceram sua postura consciente no sentido da *Criação Coletiva Instantânea*, um dos nossos objetivos no desenvolvimento do programa.

Observamos que a busca pela similaridade mostra-se, em geral, uma tendência inicial das interações, por parte dos músicos, uma vez que os integrantes buscam situar-se no ‘terreno musical’ para então partirem em direção às propostas mais contrastantes. Este fato reforça em grande parte as afirmações de Stravinsky⁶² como citado na pág. 8 de nosso trabalho. Muito relevante também, é a percepção, embora pelo viés da Fenomenologia, de uma associação clara entre a similaridade e a coletividade, em oposição à associação contraste e individualidade também presente nas interações. Isto evidencia-se ainda mais no momento em que os integrantes passam a propor novos materiais na interação. Confirmando aqui, também, que a busca por similaridade tende à provocar reações durante a interação, já a proposição naturalmente apresenta-se com características orientadas para ações contrastantes.

Uma reflexão pois: Em música ou em se tratando do Som propriamente dito/ um matéria prima da outra, poderíamos dizer que a similaridade máxima ocorre com o Uníssono, ou até mesmo o Silêncio, invadindo os terrenos da Metafísica talvez. Já o momento em que o contraste

⁶² [...] tenho considerado sempre que, em geral, é mais satisfatório proceder por similaridade do que por contraste. [...] O contraste produz um efeito imediato. A similaridade nos satisfaz a longo prazo. O contraste é elemento de variedade, mas divide nossa atenção. A similaridade nasce de um esforço pela unidade. A necessidade de se buscar a variedade é perfeitamente legítima, mas não devemos esquecer que o Uno precede o múltiplo. (STRAVINSKY 1996, p. 38)

se apresenta, oposições tomam forma, no entanto elas não precisam estar em desacordo, são variações, evidenciam as diferentes realidades e possibilidades de um todo neste plano criativo, uma oportunidade para o diálogo talvez. Já o contraste máximo, por sua vez, seria determinado, no sistema proposto, a partir de um nível de alteração do material em que as características de avaliação sejam alteradas o suficiente para que não se reconheça seu parentesco com o segmento do INput original. No entanto concluímos que a partir de então, ultrapassando os limites de nossas análises, tudo seria classificado como contraste máximo, por estar para lá do que definimos por *CHigh* na régua de *S&C* do programa.

Indo além nos perguntamos então: Em termos do desenvolvimento de sistemas interativos entre músico e computador, e partindo de um esforço interativo guiado pelas relações de *S&C*, como podemos ainda favorecer uma maior proximidade, organicidade e fluidez na comunicação com a máquina? Que indícios nos conduziriam à confirmação dos alicerces da gramática que buscamos construir por meio desses princípios aqui descritos? Teria a linguagem surgido como sintoma da separação da Unidade, no entanto sendo um caminho para ela e a estratificação deste contraste em níveis graduais um caminho para aquela construção gramatical? E por fim ainda: O quê procura o Homem ao tentar desenvolver sistemas para comunicar-se com a máquina, senão a busca por mais um caminho no sentido da compreensão de si mesmo? A Lógica e a Matemática darão conta da complexidade desta tarefa? Que limiares aguardam uma Humanidade governada pela Inteligência Artificial? Que música será esta que faremos e ouviremos a seguir?

Referências Bibliográficas

ARIZA, C. *An Open Design for Computer-Aided Algorithmic Music Composition*: athenaCL. Boca Raton, FL: Dissertation.com, 2005.

ALPERN, A., *Technics for Algorithmic Composition*. Hampshire College, 1995.

ABREU, F. J. O., *O jogo musical Cobra (1984) de John Zorn: Reflexões sobre sua história e prática*. Universidade Federal de Minas Gerais - UFMG, 2013.

ASSAYAG, G. and BLOCH, G. and CHEMILLIER, M. and CONT, A. and DUBNOV, S., *Omax Brothers: a Dynamic Topology of Agents for Improvisation Learning*, Workshop on Audio and Music Computing for Multimedia, 2004. <<http://recherche.ircam.fr/equipements/repemus/OMax/>>

BEILEY, D. *Improvisation: its nature and practise in music*. Da Capo Press, London, 1992.

BILES, J. A., *GenJam: A Genetic Algorithm for Generating Jazz Solos*, in Proceedings of the ICMC, 1994.

CONT, A., *Antescofo: Anticipatory Synchronization and Control of Interactive Parameters in Computer Music*, International Computer Music Conference (ICMC), Belfast - Ireland. pp.33-40, 2008.

CONT, A., DUBNOV, S. and ASSAYAG, G., *Antecipatory Model of Musical Style Imitation Using Collaborative and Competitive Reinforcement Learning*, in Butz, M. V., SIGAUD, O.,

PEZZULO, G., and BALDASSARRE, G., Eds, *Anticipatory Behavior in Adaptive Learning Systems*, Springer-Verlag, Berlin, Heidelberg, p. 285–306, 2007.

COPE, D., *Virtual Music: Computer Synthesis of Musical Style*, MIT Press, 2004.
 _____ *Experiments in Music Intelligence*, A-R Editions, Inc. 1996.

COSTA, R. L. M. *O Músico enquanto meio e os territórios da livre Improvisação*. Tese de Doutorado, São Paulo: PUC, 2003.

CROCHEMORE, M., HANCART, C., and LECROQ, T., *Algorithms on strings*, p.56, 2007.

DANNENBERG, R., *Practical aspects of Midi conducting programs*, Proceedings of the International Computer Music Conference - ICMA, 1991.

DOMENICI, C. L., *A voz do Performer na Música e na Pesquisa, Anais do II Simpom*, Universidade Federal do Rio Grande do Sul, 2012.

DUARTE, M. L., *"A Contribuição de Max Mathews para a Música Computadorizada"*, Anais do III Simpom - Dissertação de Mestrado, USP, 2014.

FERREIRA, A. B. de H. e J.E.M.M., Editores LTDA., *Dicionário Aurélio Básico da Língua Portuguesa*, RJ: Nova Fronteira S/A, 1998. <<https://www.dicio.com.br/aurelio/>> (acessado em junho de 2020).

GRIFFITHS, P., *A guide to electronic music*. Bath: Thames and Hudson, p. 128, 1979.
 _____ *Encyclopaedia of 20th Century Music*, Thames and Hudson Inc., p.52, 1986.
 _____ *A Música Moderna: uma história concisa e ilustrada de Debussy a Boulez*, Ed. Jorge Zahar, p.156, 1987.

HILLER, L. A. and ISAACSON. L. M., *Experimental Music: Composition with an Electronic Computer*, McGraw-Hill Book Company, New York, 1959.

HUAISS, A., *Grande Dicionário Huiass da Língua Portuguesa*, Ed. Objetiva., 2009. <<https://www.dicio.com.br/houaiss/>> (acessado em junho de 2020).

HURON, D., *Sweet Anticipation: Music and the Psychology of Expectation*. MIT Press, Cambridge, 2006.

IRCAM - Institut de Recherche et Coordination Acoustique/ Musique <<http://repmus.ircam.fr/omax/home>> (acessado em junho de 2019).

LEWIS, G., “*Too Many Notes: Computers, Complexity and Culture in Voyager*”, *Leonardo Music Journal*, v.10, p. 33-39, 2000.

MELLO, C.E., *Musical Materials and Algorithmic Composition*, Anais da ANPPOM, 2009.

_____ *Biblioteca MuMRT (Musical Materials Real Time)* Disponível em: <<https://github.com/carlosemello/MuM>> e <<https://github.com/carlosemello/MuMDocs>> (último acesso em outubro de 2020).

MEYER, L. B., *Emotion and Meaning in Music*, Chicago University Press, Chicago, 1956.

PACHET, F., *The Continuator: Musical Interaction with Style*, in *Proceedings of the ICMC*, 2002.

POLASCHEGG, N. *Interweavings - towards a new view of the relation between composition and improvisation*. *Musik Texte*, [s.l.], v. 114, p. 1–18, 2007.

PRESSING, J., *Psychological Constraints on Improvisational Expertise and Communication*, in *In the Course of Performance: Studies in the World of Musical Improvisation*. Ed. Nettle B. and Russel, M. University of Chicago Press, Chicago, p. 53-74, 1998.

_____ *Improvisation: Methods and Models*, in: Sloboda, J. *Generative Processes in Music*, Clarendon, Oxford, p. 129-178, 1988.

ROWE, R., *Interactive Music Systems* (1993)

SCHAUB, Stéphan, TAVARES, T. F. & MONTEIRO, A. C., *Interactive Free Improvisation Using Time-domain Extrapolation of Textural Features*, NICS - Núcleo Interdisciplinar de Comunicação Sonora/ UNICAMP e FEEC - Faculdade de Engenharia Elétrica e de Computação/ UNICAMP.

SCHOENBERG, A., *Fundamentos da Composição Musical*. São Paulo: Editora da Universidade de São Paulo, 2012.

STRAVINSKY, I., *Poética Musical em 6 lições*. Rio de Janeiro: Jorge Zahar, 1996.

SUNDBERG, J., *Computer synthesis of music performance*. In: SLOBODA, J. A. (Org.). *Generative processes in music: the psychology of performance, improvisation and composition*. New York: Oxford University Press, p. 52–69, 2005.

WHITBY, B., *I.A. Inteligência Artificial: Um guia para Iniciantes*, 1953. Tradução: Claudio Blanc – São Paulo: Madras, 2004.

WIKIPEDIA: *Computer Music* (Música Gerativa - gráfico de CHRISTOPHER ARIZA) < <https://en.wikipedia.org/wiki/File:GenSystemVenn.png> > (Acesso em junho de 2019).

XENAKIS, I., *Formalized Music. Thought and Mathematics in Composition*. Indiana University Press, 1971.

Apêndices

Apêndice 1 - Exemplos de possíveis Interações do Sistema

A

INput

OUTput

(Imitação - Repetição)

(Variação Rítmica - Notas Repetidas)

B

INput

OUTput

(Transposição)

(Variação - Retrogradação)

(Transposição - Retrogradação)

(Variação Rítmica - Notas Repetidas - Transposição)

(Modulação)

(Transposição)

(Imitação - Repetição)

C

INput

OUTput

(OSTINATO - Escolha de Notas)

Critérios para avaliação de Frases

1. ANÁLISE

1.1 Frase Isolada (Input ou Output)

1. Tom/ Modo
2. Rítmica: Diferentes durações
3. Direcionalidade do contorno Melódico (Paralelo, Oblíquo e Contrário)
4. Densidade: Número de notas por pulso
5. Predominância de Graus conjuntos e Saltos (tipos de Intervalos e ordem de aparecimento)
6. Região (oitavas)
7. Tessitura (Amplitude)
8. ...

1.2 Frase Relacionada (Input e Output) - Similaridade e Contraste

Níveis de 1 a 6 (SH a CH)

(Ordem de Importância - Pesos dos Critérios)

1.2.2 Similaridade

I. S1High

1. Mesmo Tom e Modo (Mudança de 0 a 1 acidente na armadura)
2. Mesma Rítmica do input
3. Movimento Paralelo (Predominância de 3as/ 6as, 4as/ 5as e 8as)
4. Mesmo Número de notas por Pulso
5. Preserva entre 81% e 100% de intervalos e ordem de aparecimento
6. Mesma Oitava

II. S2Medium

1. Tons e Modos (Mudança de 2 acidentes na armadura)
2. Preserva de 61% e 80% da Rítmica do Input
3. Movimento Paralelo (Predominância de 2as/ 7as)
4. Preserva de 61% e 80% do Número de notas por Pulso
5. Preserva entre 61% e 80% dos tipos de intervalos e ordem de aparecimento
6. Preserva de 61% e 80% de notas na mesma Oitava do Input

III. S3Low

1. Tons e Modos (Mudança de 3 acidentes na armadura)
2. Preserva de 41% e 60% da Rítmica do Input e ordem de aparecimento
3. Movimento Oblíquo
4. Preserva de 41% e 60% do Número de notas por Pulso
5. Preserva entre 41% e 60% dos tipos de intervalos
6. Preserva de 41% e 60% de notas na mesma Oitava do Input

1.2.3 Contraste**IV. C1Low**

1. Tons e Modos (Mudança de 4 acidentes na armadura)
2. Preserva de 21% e 40% da Rítmica do Input
3. Movimento Oblíquo
4. Preserva de 21% e 40% do Número de notas por Pulso
5. Preserva entre 21% e 40% dos tipos de intervalos e ordem de aparecimento
6. Preserva de 21% e 40% de notas na mesma Oitava do Input

V. C2Medium

1. Tons e Modos (Mudança de 5 acidentes na armadura)
2. Preserva de 11% e 20% da Rítmica do Input
3. Movimento Contrário
4. Preserva de 11% e 20% do Número de notas por Pulso
5. Preserva entre 11% e 20% dos tipos de intervalos e ordem de aparecimento
6. Preserva de 11% e 20% de notas na mesma Oitava do Input

VI. C3High

1. Tons e Modos (Mudança de 6 acidentes na armadura)
2. Preserva de 0% e 10% da Rítmica do Input
3. Movimento Contrário (com cruzamento de vozes)
4. Preserva de 0% e 10% do Número de notas por Pulso
5. Preserva entre 0% e 10% dos tipos de intervalos e ordem de aparecimento
6. Preserva de 0% e 10% de notas na mesma Oitava do Input

Apêndice 3 - Funções de Transformação do Sistema

INPUTs

- Nota
- Frase
- Gestos

I. Avaliação de CARACTERÍSTICAS Melódicas:

1. Tom/Modo (Escolha de notas) - GRAU DE IMPORTÂNCIA -
 2. Rítmica: Durações (organização & Padrões)
 3. Direcionalidade: (Ascendente/ Descendente & Contínua)
 4. Predominância de Graus Conjuntos e Intervalos (Mov. Escalar & Saltos)
 5. Densidade (Quantidade de Eventos ou notas/ Tempo)
 6. Tessitura (Pontos extremos do gesto)
 7. Região (Oitava)
- 

II. Definição de Níveis de SIMILARIDADE & CONTRASTE

(Porcentagem de TRANSFORMAÇÕES nas 7 características melódicas descritas)

- | | | |
|------------|------------|----------------------|
| 1. SLow | 4. CLow | - SISTEMA DE PESOS - |
| 2. SMedium | 5. CMedium | |
| 3. SHigh | 6. CHigh | |
- 

OUTPUTs

III. TRANSFORMAÇÕES

I. DEFINIÇÃO de GESTOS

- a) Por tempo (U.Ts ou segundos?)
- b) Silêncios (Maiores que um tempo?)

2. COPIAR

- a) Copiar (*Ipsis Literis*)
- b) Oitavar (Asc. e Desc: 1 ou 2 oitavas)
- c) Colar

3. COLAR

- a) Antes de um Gesto ou Nota
- b) Depois de um Gesto ou Nota

4. GRUPOS DE NOTAS “Aleatórias”**a) Escolha de notas e quantidades**

- b) Uma (1) nota
- c) Duas (2) notas
- d) Três (3) notas
- e) Quatro (4) notas
- f) Cinco (5) notas – podendo ser uma escala Pentatônica
- g) Seis (6) notas – podendo ser a as escalas de Tons Inteiros ou a Aumentada
- h) Sete notas – podendo ser qualquer Tom/ Modo

5. RECONHECIMENTO/ DEFINIÇÃO

- a) Notas (grupos de 1 a 7 notas)
- b) Ritmos (Padrões)
- c) Tons/ Modos (Escalas)

6. TRANSPOSIÇÃO

- a) Diatônica: Mesmo Tom/ Outro Modo (6 possibilidades/ Modos Gregos)
- b) Paralela: Outro Tom/ Mesmo Modo (Tons vizinhos – IV e V/ Asc. ou Desc.)
- c) Outras escalas: Pentatônicas e Menores Harmônicas...
- d) Colar**

7. CORTAR/ SEGMENTAR

- a) DIVIDIR em segmentos por critérios:
 - Densidade: Número de notas
 - Graus Conjuntos e Saltos
 - Direcionalidade: Asc. ou Desc.
 - Tessitura: pontos extremos

b) Escolher Notas**c) ORGANIZAR os segmentos ou Notas (Antes ou Depois)****d) Colar****8. ESCOLHER**

- a) As Notas ímpares, pares, padrões de escolha ou sorteio? (número mínimo e máximo de opções)
- b) Os Segmentos por ordem de eventos ou sorteio? (número mínimo e máximo de opções)
- c) Ritmos: Curtos e Longos
- d) RE-ORGANIZAR**
- e) Copiar e Colar**

9. GRAVAR: (“Building Blocks”)

- a) Escolher: Transformações, inputs ou outputs
- b) Copiar
- c) Guardar na Memória

10. RE-ORGANIZAR NOTAS/ Técnicas de Variação temática

- a) Retrógrada
- b) Inversão
- c) Retrógrado da Inversão
- d) Aumentação
- e) Diminuição

11. OSTINATOS

- a) Repetições de Gestos e/ ou Notas
- b) Rítmicos
- c) Escolher
- d) Copiar e Colar

12. ACRESCENTAR & EXTRAIR

- a) Notas
- b) Gestos

Apêndice 4 - Funções de Segmentadores do Sistema

Linhas gerais:

- Os 'sorteios' representam escolhas entre valores pré-estabelecidos

* DURAÇÕES

- U.T. - Semínima = 60bpm
- IN ou OFF beat - Durações entre 1s e 16s
- Acurácia da quadratura de 16 tempos: \underline{x} 64 possibilidades = 4x4 semicolcheias em 4 compassos de 4/4)
- Time Stamp ...

* ALTURAS

- Escopo limitado às alturas combinadas dos 7 modos gregos das 12 armaduras de clave tonal

• Critérios de Segmentação:

1. Tempo:

a) Gesto a Gesto

- Gestos com durações 'sorteadas', segmentados um após o outro a partir do primeiro INput. (1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 18, 24 e 30 – tamanhos dos gestos em segundos)

b) Randômico/ Aleatório

- O ponto de início e fim do gesto segmentado deverá ser 'sorteado' entre 1 e 64 possibilidades (16x4)

c) Tempo a Tempo

- Gestos segmentados a cada segundo, tendo tanto tamanho como durações 'sorteadas'

2. Silêncio:

- Segmentação a partir de 1s ou mais de silêncio entre notas do INput

3. Número de Notas

- Segmentando de 1 a 32 notas, ‘sorteadas’ e máximo de 16 tempos:
 - a) Gesto a Gesto
 - Gestos com números de notas ‘sorteadas’, segmentados um após o outro a partir do primeiro INput.
 - b) Randômico (Até 64 pontos de início e tamanho = 16x4)
 - O número da nota de início e fim do gesto segmentado deverá ser ‘sorteado’ entre 1 e 64 possibilidades (16x4)
 - c) Nota a Nota (até 16 tempos)
 - Gestos segmentados a cada nota, tendo a quantidade de notas ‘sorteadas’

4. Alturas

- a) Graus Conjuntos (Movimento Escalar – intervalos diatônicos de 2as M ou m)
- b) Saltos (Intervalos maiores que 3as M ou m)
- c) Direcionalidade – segmentos a partir da mudança de direção do movimento. Tipos:
 - Movimento Contínuo – Alturas repetidas ou longas (maiores que 4 tempos)
 - Movimento Ascendente
 - Movimento Descendente
 - Contorno Melódico (Conjuntos combinados dos Movimentos precedentes:
(1 a 1, 2 a 2, 3 a 3 e 4 a 4)
- d) Escala (Alterações de Armadura de Clave – A partir de 1 e limite de 2 acidentes por vez/ Tons vizinhos)
- e) Paralelismo
- f) Ostinatos melódicos – padrões de 1 a 12 alturas que se repitam ao menos uma vez.
Avaliado nota a nota.

5. Durações

- a) Conjuntos de mesma duração – Segmentação a partir da mudança rítmica do INput

- b) Contraste consistentes de duração – Segmentação a partir de durações 4x maiores ou menores que média das durações anteriores
- c) Ostinatos rítmicos – padrões de 1 a 12 durações avaliados nota a nota

Apêndice 5 - Alguns Testes do Segmentador (CSound - Partitura)

```

[Destination 0]: Barramento 1
[Destination 1]: Barramento IAC 2
[Destination 2]: USB Uno MIDI Interface

[Source 0]: Barramento 1
[Source 1]: Barramento IAC 2
[Source 2]: USB Uno MIDI Interface

Count: 0
while1 while1
while1 while1 while1 while1 while1 while1 while1 while1 while1 while1 while1 while1
while1 while1 while1 while1 while1 while1 while1 while1 while1 while1 while1 while1
while1 while1 while1 while1

escolhaTempoGestoAGesto: 24
escolhaNotaAleatoriaComeco: 34
escolhaTempoAleatoria: 8
escolhaTempo-TempoATempo: 3

; =====
; Function Tables:
; =====
f1 0 4096 10 1 .9 .1 .8 .2 .7 .3 .6 .4 .5
f2 0 4096 10 1 0 1 0 1 0 1 0 1
f3 0 4096 10 .1 .3 .5 .7 .5 .3 .1
f4 0 4096 10 .8 .6 .4 .2 .4 .6 .8

; =====

; =====
; VOICE: 0, Instr.: 0
; =====

i0      0.000      0.000      0.00      0.000
i1      0.000      1.042      8.00      0.609
i1      1.024      1.014      8.01      0.711
i1      2.056      1.012      8.02      0.445
i1      3.066      1.053      8.03      0.469
i1      4.124      0.494      8.04      0.422
i1      4.641      0.511      8.05      0.547
i1      5.119      0.528      8.06      0.727
i1      5.637      0.491      8.07      0.656
i1      6.109      0.506      8.08      0.555
i1      6.614      0.517      8.09      0.609
i1      7.097      0.510      8.10      0.531
i1      7.591      0.455      8.11      0.578
i1      8.058      0.254      9.00      0.727
i1      8.296      0.254      9.01      0.578
i1      8.564      0.249      9.02      0.570
i1      8.789      0.243      9.03      0.617
i1      9.043      0.230      9.04      0.695
i1      9.287      0.273      9.05      0.641
i1      9.531      0.230      9.06      0.633
i1      9.777      0.269      9.07      0.625
i1      10.040      0.232      9.08      0.656
i1      10.286      0.249      9.09      0.641
i1      10.533      0.258      9.10      0.578
i1      10.801      0.253      9.11      0.578
i1      11.061      0.528      10.00     0.781
i1      11.563      0.514      10.01     0.102
i1      12.074      0.530      10.02     0.727
i1      12.586      0.521      10.03     0.633
i1      13.113      0.471      10.04     0.609
i1      13.589      0.510      10.05     0.711
i1      14.096      0.496      10.06     0.633

```

i1	14.618	0.468	10.07	0.422
i1	15.074	0.525	10.08	0.586
i1	15.620	0.551	10.09	0.570
i1	16.137	0.526	10.10	0.586
i1	16.690	0.556	10.11	0.625
i1	17.233	0.572	11.00	0.625
i1	17.771	0.561	10.11	0.555
i1	18.304	0.537	10.10	0.711
i1	18.843	0.583	10.09	0.625
i1	19.392	1.024	10.08	0.781
i1	20.414	0.971	10.07	0.695
i1	21.359	0.982	10.06	0.781
i1	22.350	0.995	10.05	0.672
i1	23.354	2.006	10.04	0.609
i1	25.308	1.977	10.03	0.805

```

; =====
; END VOICE 0
; =====

```

```

; =====
; Function Tables:
; =====
f1 0 4096 10 1 .9 .1 .8 .2 .7 .3 .6 .4 .5
f2 0 4096 10 1 0 1 0 1 0 1 0 1
f3 0 4096 10 .1 .3 .5 .7 .5 .3 .1
f4 0 4096 10 .8 .6 .4 .2 .4 .6 .8

```

```

; =====
; VOICE: 0, Instr.: 0
; =====

```

i1	0.000	0.526	10.10	0.586
----	-------	-------	-------	-------

```

; =====
; END VOICE 0
; =====

```

```

; =====
; Function Tables:
; =====
f1 0 4096 10 1 .9 .1 .8 .2 .7 .3 .6 .4 .5
f2 0 4096 10 1 0 1 0 1 0 1 0 1
f3 0 4096 10 .1 .3 .5 .7 .5 .3 .1
f4 0 4096 10 .8 .6 .4 .2 .4 .6 .8

```

```

; =====
; VOICE: 0, Instr.: 0
; =====

```

i0	0.000	0.000	0.00	0.000
i1	0.000	1.042	8.00	0.609
i1	1.024	1.014	8.01	0.711
i1	2.056	1.012	8.02	0.445
i1	3.066	1.053	8.03	0.469

```

; =====
; END VOICE 0
; =====

```


The image displays a page of musical notation for guitar, organized into two columns of ten staves each. The notation is written in treble clef with a common time signature (C). The music consists of eighth and sixteenth notes, often beamed together in pairs or groups, and includes some slurs and accents. The overall style is that of a guitar exercise or a short piece of music.

The image displays a musical score for a single system, consisting of 12 staves. The music is written in a treble clef with a common time signature (C). The notation includes various rhythmic values such as eighth and sixteenth notes, often beamed together, and rests. The piece concludes with a final double bar line on the twelfth staff.

Anexos

Anexo 1 – *Lista de métodos da função principal da biblioteca MuMRT*

(arquivo em anexo e também disponível em: <<https://github.com/carlosemello/MuMDOcs>>)