# Implementação de um sistema composicional semiaberto a partir da similaridade entre conjuntos de classes de notas

Vinicius Braga
João Penchel
Igor Chagas
Rodrigo Furman
Pedro Proença
Liduino Pitombeira
Universidade Federal do Rio de Janeiro, Rio de Janeiro-RJ

**Resumo:** Este trabalho descreve a formalização e aplicação de um sistema composicional, denominado Similare, que articula micro e macroestrutura tomando como base as relações de similaridade entre conjuntos de classes de notas. Após o exame detalhado dos dois referenciais que dão sustentação teórica ao trabalho (sistemas composicionais e similaridade), o sistema foi definido e, a partir dele, duas obras foram planejadas: uma para quarteto de cordas e outra para piano solo. Durante o processo de pesquisa, aplicativos em Python foram elaborados para auxiliar os compositores na manipulação dos conjuntos.

**Palavras-chave:** Sistemas Composicionais. Planejamento Composicional. Similaridade entre conjuntos de classes de notas.

## Implementation of a semi-open compositional system based on the similarity between pitch-class sets

**Abstract:** *This paper describes the formalization and application of a compositional system, called SIMILARE, which articulates microstructure and macrostructure based on similarity relations among pitch-class sets. After a detailed examination of the two references that give theoretical support to the work (compositional systems and similarity), the system was defined and, from it, two works were planned: one for string quartet and other for solo piano. During the research process, Python applications were designed to assist composers in manipulating sets.*

**Keywords:** *Compositional Systems. Compositional Planning. Similarity among Pitch-class Sets.*

BRAGA, Vinicius; CHAGAS, Igor; PENCHEL, João; FURMAN, Rodrigo; PROENÇA, Pedro; PITOMBEIRA, Liduino. Implementação de um sistema composicional semiaberto a partir da similaridade entre conjuntos de classes de notas. *Opus*, v. 26 n. 2, p. 1-40, maio/ago. 2020. http://dx.doi.org/10.20504/opus2020b2611

Recebido em 30/3/2020, aprovado em 30/6/2020

Neste artigo descrevemos uma metodologia de formalização pré-composicional que, partindo de diretrizes de manipulação paramétrica, possibilita a geração de repositórios composicionais e o planejamento estrutural desses repositórios com o objetivo de gerar obras musicais organicamente coerentes. Primeiramente, definimos os referenciais teóricos utilizados neste trabalho, ou seja, os conceitos de sistema composicional (seção 1) e similaridade entre conjuntos de classes de notas (seção 2), exemplificando-os com a elaboração de pequenos trechos musicais. Em seguida, propomos o sistema composicional denominado Similare, que gerencia a forma de uma obra musical a partir das relações de similaridade entre conjuntos de classes de notas. Tomando como ponto de partida esse sistema, duas obras foram planejadas e compostas.

## Sistemas composicionais

A Teoria dos Sistemas Composicionais[1] se inspirou inicialmente na Teoria Geral dos Sistemas (BERTALANFFY, 1968) e no trabalho de Klir (1991), *Facets of Systems Science.* O primeiro posiciona a música, as artes em geral e os sistemas linguísticos em uma categoria denominada **Sistemas Simbólicos**, os quais são definidos por "regras do jogo" (BERTALANFFY, 1968: 29). Tais regras são as estruturas que dão sentido relacional aos objetos de um campo específico. Por exemplo, a sintaxe coordena a utilização de vocábulos em uma língua; as regras escolásticas de harmonia (de um determinado estilo e época) coordenam a construção horizontal (melódica) e vertical (harmônica) da música etc. Adicionalmente, Bertalanffy define um sistema como um "complexo de elementos em interação" (BERTALANFFY, 1968: 38). Klir (1991: 4), por sua vez, propõe uma definição formal para um sistema genérico, $S = (O,R)$, ou seja, um sistema (S) é o resultado de objetos (O) e relações (R).

Embora o termo sistema composicional seja utilizado na literatura (significando informalmente uma maneira de compor, bem como associado a sistemas algorítmicos computacionais)[2], uma definição formal para o termo só apareceu em 2011[3], na dissertação de Flávio Lima (2011), orientada por Liduino Pitombeira, na Universidade Federal da Paraíba: um sistema composicional é "um conjunto de diretrizes, formando um todo coerente, que coordena a utilização e interconexão de parâmetros musicais, com o propósito de produzir obras musicais". Posteriormente, uma atualização do termo foi realizada por Pitombeira (2015), incluindo não só parâmetros, mas também materiais em sua integridade, um fator que pode ser fundamental em sistemas intertextuais.

É importante salientar que, embora a Teoria dos Sistemas Composicionais, desenvolvida pelo autor que orienta este trabalho em parceria com seus orientandos, seja relativamente nova, tendo em vista que uma definição do conceito de sistema composicional só surge por volta de 2011, a literatura que a inspirou (prioritariamente os textos de Bertalanffy e Klir supracitados) não é recente. Fora do âmbito desta pesquisa, o trabalho mais atualizado que associa a Teoria Geral dos Sistemas com sistemas composicionais (embora sem fornecer uma definição para

[1] Este trabalho integra as pesquisas desenvolvidas no grupo de pesquisas MusMat, ligado ao Programa de Pós-Graduação em Música da Universidade Federal do Rio de Janeiro (PPGM/UFRJ), no âmbito do projeto de pesquisa Desenvolvimento de Sistemas Composicionais. A Teoria dos Sistemas Composicionais é uma concepção original do coordenador desse projeto, Prof. Liduino Pitombeira, que envolve a participação ativa de alunos de iniciação científica, mestrado e doutorado. Essa teoria, em constante desenvolvimento e aperfeiçoamento desde 2009, é o núcleo desse projeto de pesquisa. Diversos artigos produzidos anualmente entre 2010 e 2020 podem ser acessados gratuitamente no repositório Academia do autor que orienta este trabalho. O *link* é https://ufrj.academia.edu/LiduinoPitombeira.

[2] Citamos, por exemplo, os trabalhos de Pressing (1988), Didkovsky (1997) e Di Scipio (1995).

[3] Mesmo o artigo referencial de Lerdahl (1988), "Cognitive Constraints on Compositional Systems", no qual o termo aparece explicitamente no título, não fornece uma definição para o conceito de sistema composicional.

esse conceito) parece ser o de Georgescu e Georgescu, escrito em 1990[4]. Evidentemente, trabalhos recentes que produzem música algoritmicamente a partir de algum tipo de sistema computacional estão disponíveis na esfera da pesquisa acadêmica. Citamos, por exemplo, os trabalhos de Bell e Gabora (2016), que introduzem o sistema interativo de geração musical denominado NetWorks. Esse sistema, baseado na Teoria da Criatividade Honing[5] e na premissa de que a mente humana é um **sistema adaptativo complexo**, é autônomo quanto à geração de dados Midi (partindo de uma seleção da arquitetura e das regras de interação por um usuário-artista), mas o resultado é orquestrado por um compositor[6], que é visto como um colaborador do projeto. A maioria dos referenciais teóricos que dão suporte ao NetWorks datam da última década do século XX, com exceção da Teoria Honing, proposta em 2010. Outro trabalho, ainda mais recente, é o de Navarro *et al.* (2020), que utilizam um sistema multiagente, a Pangea (Platform for Automatic coNstruction of orGanizations of IntElligents Agents), para produzir sistemas composicionais (termo não definido no artigo) que geram repositórios harmônicos, partindo de seis regras básicas.

Embora os sistemas algorítmicos estejam incluídos em nossa pesquisa, no caso específico deste artigo, nosso foco se concentra em sistemas composicionais projetados manualmente, sendo o trabalho subsequente de planejamento composicional desenvolvido pelo compositor tanto de forma manual (na primeira obra, seção 5) como de forma híbrida (na segunda obra, seção 6). Vale mencionar que os trabalhos supramencionados apresentam elos com nossa pesquisa, especialmente na esfera dos sistemas retroalimentados que utilizam mapas caóticos, como veremos adiante. Desta forma, para o desenvolvimento dos projetos composicionais descritos neste artigo, continua sendo essencial a definição de Klir (1991: 4): um sistema é um conjunto de objetos e relações.

Na Teoria dos Sistemas Composicionais, os objetos são genéricos, o que significa que a eles não são atribuídos valores específicos. Portanto, pode-se exemplificar um sistema mínimo (denominemo-lo **Sistema Tricordal**), no qual dois conjuntos de classes de notas, A = {a1, a2, a3} e B = {b1, b2, b3}, se sucedam alternadamente e se relacionem entre si por uma operação de transposição (Tk), onde k pode ser um dos valores de A, ou seja, k = {a1 ∧ a2 ∧ a3}.

Partindo desse sistema mínimo, planejamos uma obra musical como exemplo. Como o **Sistema Tricordal** trata unicamente de conjuntos de classes de notas que se relacionam por transposição, a primeira fase do planejamento composicional consiste em particularizar valores para esses conjuntos que satisfaçam as diretrizes do sistema. Na Fig. 1 iniciamos o planejamento com um valor arbitrário para o conjunto A ({Ré, Fá e Dó#}[7], ou, em termos de classes de notas, {2,5,1}). O próximo conjunto é gerado a partir de uma transposição de A, por um fator de transposição cujo valor é igual a uma das classes de notas de A. Escolhemos k = 1, gerando o conjunto {3,6,2}, ao qual se aplica uma transposição $T_2$ (uma vez que 2 é uma das classes de notas desse conjunto), e assim por diante.

Na segunda fase do planejamento composicional, decidimos acerca dos demais parâmetros não declarados no sistema composicional: instrumentação, andamento, caráter, dinâmicas, articulações e ritmo. A Fig. 2 mostra uma possível realização do planejamento realizado[8].

[4] Uma visão geral sobre as perspectivas desse trabalho pode ser consultada em Pitombeira (2020).

[5] Segundo essa teoria, a mente criativa se adapta ao ambiente para minimizar entropia (GABORA, 2017).

[6] Um resultado pode ser escutado em Bell (2020). *Link* de acesso disponível na bibliografia deste artigo.

[7] Neste trabalho, classes de alturas desordenadas são indicadas entre chaves e separadas por vírgulas (Ex.: {0,1,2}, formas normais entre parênteses, sem separações por vírgulas (Ex.: (123)) e formas primas entre colchetes, também sem separação por vírgulas (Ex.: [016]). Classes de conjuntos (*set classes*) serão escritas diretamente (Ex.: 014). As classes de alturas 10 e 11 serão representadas por A e B.

[8] O trecho integra uma obra para fagote e quarteto de cordas do compositor Liduino Pitombeira intitulada *Brazilian Landscapes No. 20*.

Baseando-se em Durand (1996), Pitombeira (2018) propõe uma taxonomia para sistemas composicionais: sistema aberto, semiaberto e retroalimentado (Fig. 3). Os sistemas semiabertos não possuem entrada de materiais, ou seja, possuem apenas saída. Desta forma, produzem materiais musicais sem a necessidade de inserção de materiais da mesma natureza. O **Sistema Tricordal**, descrito anteriormente, é um exemplo de sistema semiaberto, uma vez que suas próprias definições internas nos permitiram planejar e compor o trecho musical mostrado na Fig. 2. Outro exemplo de sistema semiaberto é a geração aleatória de dados, os quais podem ser mapeados em parâmetros musicais.

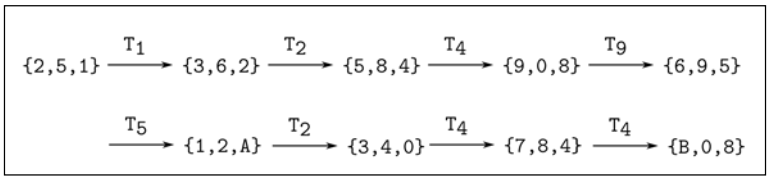



**Fig. 1:** Planejamento composicional dos tricordes com base no Sistema Tricordal

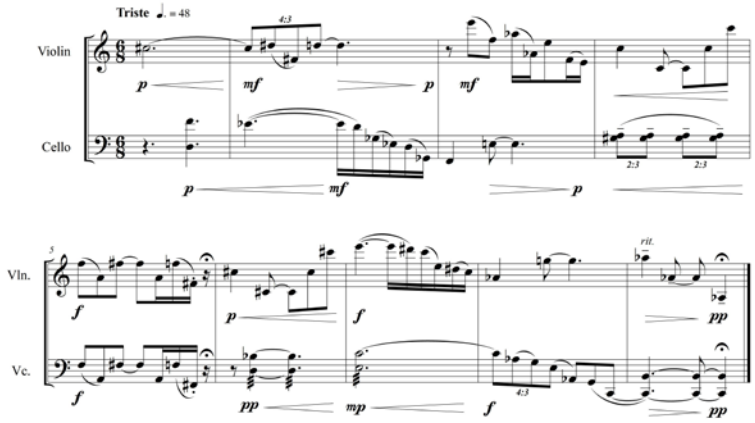



**Fig. 2:** Trecho gerado a partir do planejamento composicional dos tricordes com base no **Sistema Tricordal**

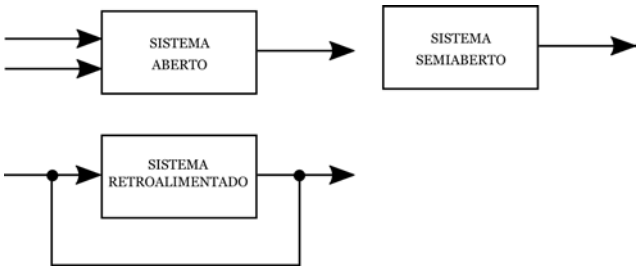



**Fig. 3:** Tipos de sistemas (PITOMBEIRA, 2018)

Por sua vez, os sistemas abertos permitem a entrada e saída de materiais (em sua totalidade ou apenas de alguns de seus parâmetros). O sistema mostrado na Fig. 4 é um exemplo de sistema aberto. Nesse sistema, arquivos no formato Midi são submetidos a três módulos modificadores: um transpositor, cujo fator de transposição k é inserido pelo usuário; um módulo expansor/contrator de durações, cujo fator de expansão/contração é também inserido pelo usuário; e um

módulo generalizador[9], que reordena randomicamente os eventos. Na Fig. 4, temos em A um fragmento original, com suas especificações de altura, duração e ordenação, que será inserido no sistema, cujo diagrama interno é mostrado em B, com seus três módulos (transpositor, expansor/contrator, generalizador). Consideremos, como exemplo, os fatores k=2 e d=2. As notas são identificadas pelo valor Midi (no qual o Dó central do piano equivale a 60). Para as durações, convencionou-se que uma semínima vale 12 unidades, para facilitar as divisões por três, no caso de quiálteras. Ao serem inseridos no sistema, os eventos são primeiramente alterados em termos de altura e duração e, em seguida, embaralhados pelo generalizador (cujo núcleo operacional é a função randômica *shuffle* do Python). O evento 1, por exemplo, com altura 60 é transposto para o valor 62 e sua duração (18) é multiplicada por 2, resultando no valor 36. Esse evento, após a generalização, passa a ser o segundo no novo fragmento (trecho C da Fig. 4). Em D o fragmento C é reescrito em compasso composto e tem, portanto, o andamento ajustado[10].

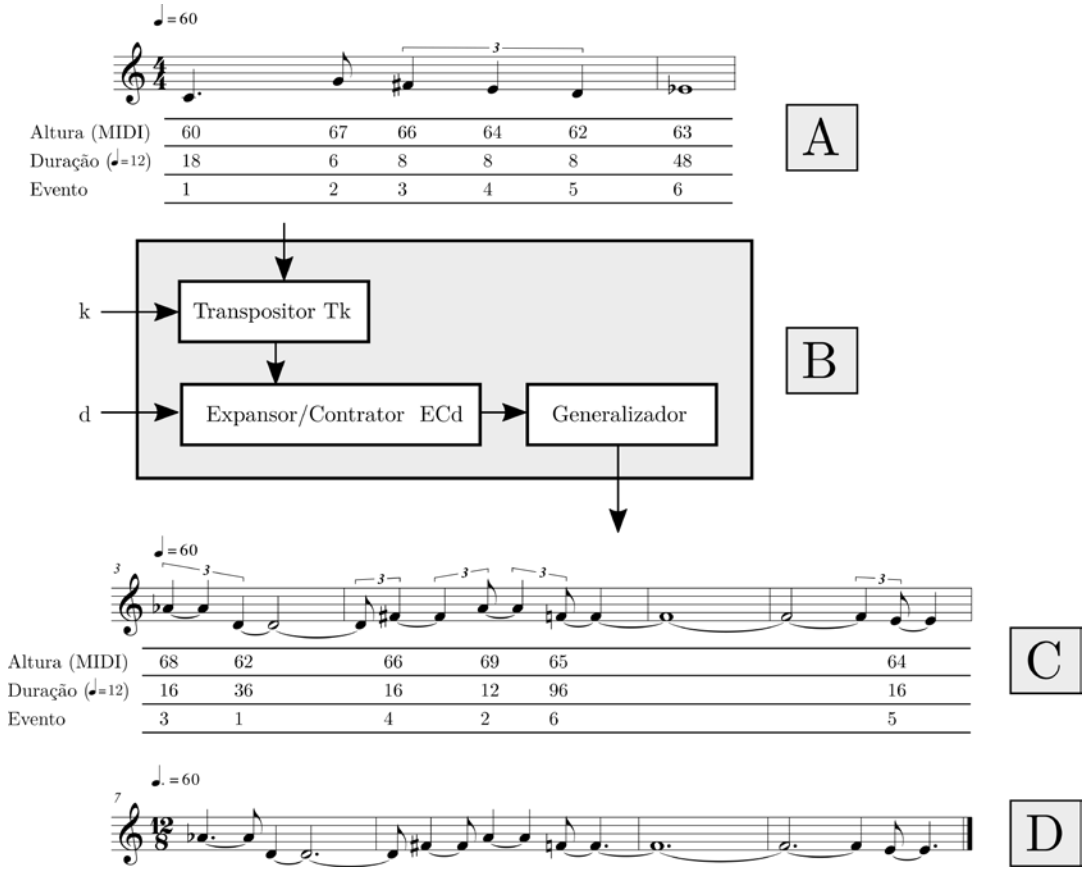



**Fig. 4:** Exemplo de sistema aberto. Em A temos um fragmento original, com suas especificações de altura, duração e ordenação, que será inserido no sistema, cujo diagrama é mostrado em B, com seus três módulos (transpositor, expansor/contrator, generalizador); em C temos um novo fragmento resultante das alterações realizadas pelo sistema; em D o fragmento C é reescrito de maneira mais amigável ao intérprete.

[9] Inspirado na ferramenta intertextual de Straus (1990), denominada generalização, que altera a ordem dos eventos de um intertexto. Operacionalmente, a generalização é realizada com uma função em Python denominada *shuffle*. Assim, uma lista a = [1,2,3,4,5,6], ao ser submetida à função *shuffle*, pode se transformar em a = [3,1,4,2,6,5].

[10] Esses ajustes finos de editoração têm por objetivo melhorar a legibilidade da obra, mas não são essenciais à metodologia ou associados ao contexto do sistema aberto.

No terceiro tipo de sistema, o retroalimentado, os valores de saída são reintroduzidos na entrada. Tais sistemas são utilizados, por exemplo, nos processos caóticos. Para ilustrar o sistema retroalimentado, utilizaremos o sistema caótico conhecido como mapa de Hénon, que é um sistema dinâmico de tempo discreto, ou seja, um sistema que mapeia pontos no espaço em um tempo não contínuo. Esse sistema bidimensional, detalhadamente examinado por Bidlack (1992), é descrito pela equação 1.

$$x_{n+1} = 1 - \alpha x_n^2 + y_n$$
$$x_{n+1} = b x_n \qquad \text{Eq. 1}$$

Os resultados obtidos pela iteração da Eq. 1 serão mapeados para as notas e para as durações, tendo em vista que esse sistema caótico produz duas dimensões (x,y). Um programa em Python, mostrado no Quadro 1, nos permitirá realizar tal mapeamento. Os valores de a e b, 1.4 e 0.3, respectivamente, são constantes propostas no modelo clássico do mapa de Hénon. Esse programa também produz uma representação gráfica para o mapa de Hénon (Fig. 5). Na Fig. 6 temos um fragmento do trecho gerado pelo programa traduzido para notação musical a partir de restrições que nos permitiram ter notas exclusivamente no âmbito da escala cromática e com múltiplos de semicolcheias.

```
from matplotlib import *
from music21 import *

def HenonMap(a,b,x,y):
            return y + 1.0 - a *x*x, b * x

a = 1.4
b = 0.3
iterates = 100000

#  Initial Condition
xtemp = 0.1
ytemp = 0.3

x = [xtemp]
y = [ytemp]
melodia = stream.Stream()

for n in range(0,iterates):
  xtemp, ytemp = HenonMap(a,b,xtemp,ytemp)
  x.append( xtemp )
  y.append( ytemp )
  nota = note.Note(int(x[n]*10)+60)
  nota.quarterLength = (((abs(int(y[n]*10)))%12)+1)*0.25
  melodia.append(nota)

pyplot.plot(x,y,'b,')
pyplot.savefig('henon.jpg', format='jpeg', dpi=600)
pyplot.show()

melodia.write('midi','henonexample.mid')
```

**Quadro 1:** Programa em Python para gerar representação gráfica do mapa de Hénon e mapear x nas notas e y nas durações, produzindo um arquivo Midi.

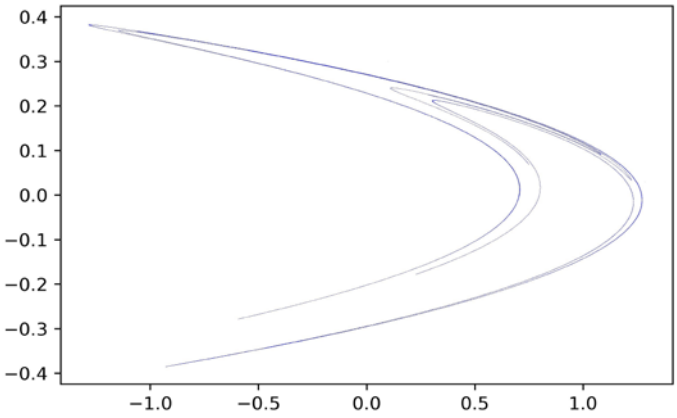



**Fig. 5:** Representação gráfica para o mapa de Hénon produzida pelo programa do Quadro 1

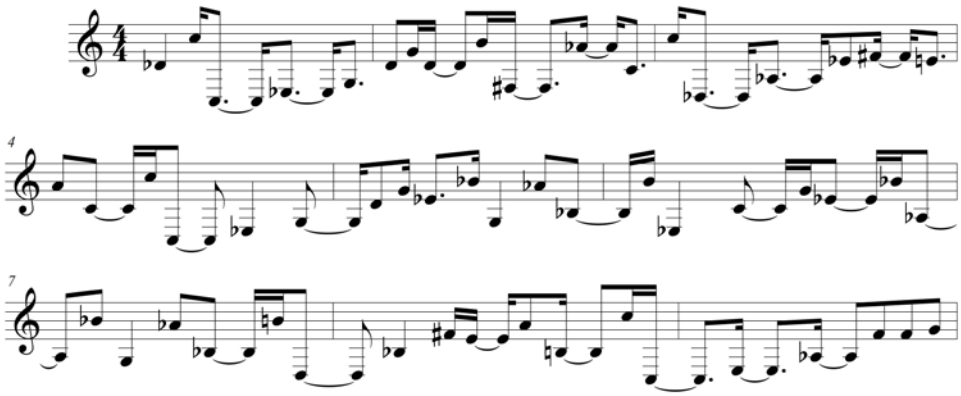


**Fig. 6:** Nove primeiros compassos (de 1588) gerados pelo programa do Quadro 1

## Grau de similaridade entre conjuntos de classes de notas

Com a utilização da teoria de classes de notas no repertório da música atonal, diversos autores se interessaram pelas relações intervalares entre classes de conjuntos distintas, a fim de examinar ou criar coerência no discurso musical com base no grau de similaridade. Isaacson (1990)[11] comparou alguns modelos apresentados por diferentes autores (Forte, Lord e Morris, Rahn, Lewin e Teitelbaum) e propôs um novo modelo baseado em Teitelbaum. Para Isaacson (1990: 2),

[11] A bibliografia que trata da similaridade entre conjuntos de classes de notas começa a ser publicada a partir da segunda metade do século XX. Dois dos artigos mais recentes sobre esse tópico são o de Samplaski (2005) e o de Tymoczko (2008). Samplaski compara os diversos métodos utilizados para o cálculo de similaridade através de uma técnica denominada *multidimensional scaling* (MDS). Tymoczko, por sua vez, examina a situação sobre duas perspectivas: a condução de vozes e o conteúdo intervalar. Sobre a condução de vozes, ele afirma que "compositores, sentados em um piano, julgariam que acordes são semelhantes quando esses podem ser ligados por pequenos movimentos físicos" (Original: "[...] composers, sitting at a piano keyboard, would judge chords to be similar when they can be linked by small physical motions"). Com relação ao conteúdo de intervalos, ele examina o método desenvolvido por Iann Quinn (2006, 2007), que se baseia no que Lewin (2001) denominou de "cinco propriedades de Fourier". As mais recentes publicações sobre similaridade entre conjuntos de classes de notas são: Honingh *et al.* (2009) e Ockelfor (2017). Contudo, esta última obra foi publicada originalmente em 2005, o que torna a primeira a mais nova publicação sobre o tema. Acrescente-se, no entanto, que o orientador deste artigo tem um trabalho publicado em 2012 sobre similaridade de classes de notas aplicada ao planejamento composicional, que todavia se restringe às sonoridades tricordais (DANTAS; PITOMBEIRA, 2012). Vale mencionar também o livro de Schuijer (2008), cujo capítulo 5, com 48 páginas, é integralmente dedicado ao conceito de similaridade, incluindo a similaridade entre conjuntos de classes de notas.

uma função de cálculo de similaridade eficiente deve atender aos seguintes critérios: "1) fornecer um valor diferenciado para cada par de conjuntos; 2) ser utilizada para conjuntos de qualquer cardinalidade; e 3) fornecer uma faixa considerável de valores discretos"[12]. Esses modelos utilizam como critério a diferença das qualidades intervalares dos vetores de cada conjunto, embora Allen Forte (1973: 50) considere também que o conteúdo das classes de notas seja um fator importante para determinar a máxima similaridade. O método de medição de Forte é restrito a conjuntos de classes de notas de mesma cardinalidade. Ele distingue três tipos de relações de similaridade e observa que a máxima correspondência entre dois vetores intervalares no âmbito de uma mesma cardinalidade é de quatro elementos (aqui excetuam-se evidentemente as relações Z). No primeiro tipo de relação, $R_0$, não há correspondência entre nenhum elemento dos vetores intervalares dos conjuntos que estão sendo comparados; no segundo tipo de relação, $R_1$, quatro elementos são correspondentes, e os dois que não correspondem apresentam similaridade diagonal; quando essa similaridade diagonal não ocorre, a relação é do tipo $R_2$. A Tab. 1 mostra exemplo das três relações ($R_0$, $R_1$ e $R_2$) entre duas classes de conjuntos ($S_1$ e $S_2$).

| | $S_1$ | $S_2$ | |
|---|---|---|---|
| $R_0$ | 0123 | 0157 | 2 1 1 1 1 0<br>2 1 0 1 1 1 |
| | <321000> | <110121> | |
| $R_1$ | 0125 | 0126 | |
| | <211110> | <210111> | |
| $R_2$ | 0123 | 0124 | |
| | <321000> | <221100> | |

**Tab. 1:** Exemplificação dos tipos de relações de similaridade entre classes de conjuntos, segundo Forte (1973).

No primeiro caso, os vetores correspondentes a classes de conjuntos 0123 e 0157 não apresentam nenhuma correspondência entre os componentes e, portanto, se configura a relação do tipo $R_0$. No segundo caso, os dois componentes dos vetores intervalares de 0125 e 0126 que não se correspondem apresentam similaridade diagonal entre si (esse tipo de similaridade é mostrado no diagrama situado no lado direito da Tab. 1). Isaacson aponta três problemas com o sistema de medição de similaridade de Forte: 1) os conjuntos precisam ter a mesma cardinalidade; 2) muitos pares de classes de conjuntos não se enquadram em nenhuma relação; e 3) baixo poder de discriminação em virtude do reduzido número de tipos de relações.

Por sua vez, Charles Lord e Robert Morris desenvolveram na mesma época (início da década de 1980), de maneira independente, funções para cálculo de similaridade entre conjuntos de classes de notas. A função de Morris (Eq. 2), calcula a similaridade (*SIM*) entre dois conjuntos de classes de notas (*X,Y*). O cálculo é feito pela soma dos valores absolutos das diferenças de todos os componentes dos vetores intervalares de cada conjunto. Desta forma, a similaridade entre as classes de conjuntos 012 e 013, cujos vetores são, respectivamente, <210000> e <111000>, é: $|1 - 2| + |1 - 1| + |1 - 0| + |0 - 0| + |0 - 0| + |0 - 0| = 2$.

[12] Original: "1) provide a distinct value for every pair of sets; 2) be useful (not just usable) for sets of any size; and 3) provide a wide range of discrete values".

$$SIM\ (X,Y) = \Sigma\ |\ x_{i-}\ y_i\ |$$ Eq. 2

A função *SIM* de Morris pode ser utilizada para comparar conjuntos de diferentes cardinalidades, porém produz somente 15 resultados comparativos. Por exemplo, a classe de conjuntos 012 apresenta o mesmo grau de similaridade (4) com as classes 014, 015, 016, 024, 025, 026 e 027. À medida que a diferença entre as cardinalidades aumenta, a função de Morris enxerga menos dissimilaridades entre os conjuntos. A função de Lord denominada *sf* é similar à de Morris; de fato, é a mesma função apenas dividida pelo fator 2. Essa função é aplicada a conjuntos de mesma cardinalidade.

John Rahn, também na mesma época (início da década de 1980), propõe uma função, denominada MEMB (Eq. 2), que computa valores dos vetores intervalares somente quando as classes intervalares estão presentes em ambos os conjuntos[13]. Assim, por exemplo, a similaridade entre as classes de conjuntos 012 e 013, cujos vetores são, respectivamente, <210000> e <111000>, considera apenas o primeiro e o segundo componentes, uma vez que as demais classes intervalares não estão presentes no primeiro ou no segundo vetores. O resultado é (2 + 1) + (1 + 1) = 5. A função MEMB produz 18 resultados comparativos[14].

$$MEMB_2\ (X,Y) = \Sigma\ |\ x_i + y_i\ |$$ Eq. 3

O cálculo de similaridade entre dois conjuntos, pelo método de Lewin, segundo a descrição de Isaacson (1990), é mostrado na equação 4[15]. No lado esquerdo da equação indica-se o que se quer calcular: a relação de similaridade (*REL*) entre dois conjuntos de classes de notas (*X, Y*). No numerador dispõe-se o somatório das raízes quadradas dos produtos dos componentes dos vetores intervalares de cada conjunto, multiplicando-se o resultado por 2. No denominador, temos a raiz quadrada dos produtos das cardinalidades pelas cardinalidades subtraídas de um, para cada conjunto.

$$REL\ (X,Y) = \frac{(2\Sigma\ \sqrt{(x_i, y_i)}}{\sqrt{(\#X(\#X\text{-}1)\#Y(\#Y\text{-}1)}}$$ Eq. 4

Calculemos, por exemplo (Eq. 5 e Eq. 6), a similaridade entre os conjuntos (016) e (0167). Primeiramente, devemos determinar os vetores intervalares de cada um:

[13] Esta função é, na verdade, mais genérica. Seu formato genérico não é discutido em Isaacson por não ser pertinente ao cálculo de similaridade. O índice 2 enfatiza que apenas intervalos (díades) estão sendo considerados.

[14] Um ajuste nessa função, realizada por Isaacson, inclui o fator $\frac{\#X(\#X\text{-}1)+\#Y(\#Y\text{-}1)}{2}$, onde #X e #Y são as cardinalidades dos conjuntos comparados (X, Y). Esse ajuste amplia o resultado para 34.

[15] Assim como a fórmula de Rahn, a fórmula original é mais genérica (LEWIN, 1979-1980). Isaacson (1990: 27) relata na nota 9 de seu artigo que a fórmula original apresentada por Lewin continha incorreções. A versão correta foi passada a Isaacson pelo próprio Lewin. A fórmula mostrada aqui (Eq. 4) é um caso particular da função geral de Lewin usada apenas para cálculo de similaridade intervalar. Kuuse (2001) acrescenta que, ao propor essa função em um nível muito genérico, Lewin (1979-1980) inspirou a formatação de três diferentes versões de REL (a mostrada na Eq. 4 e duas outras versões elaboradas pelo próprio Isaacson, em 1992, e posteriormente [1994] por Cástren, em suas teses de doutorado). Considerando que a similaridade entre conjuntos de classes de notas é um fator secundário no desenvolvimento deste artigo, não apresentaremos as demais versões de REL. Hovinen e Tenkanen (2007: 209) discutem uma dessas formas de REL em detalhes.

<100011> e <200022>, respectivamente. A quantidade de valores de similaridade resultantes dessa fórmula é extremamente superior aos das fórmulas de Morris e Lord. Isaacson (1990: 13) observa, no entanto, que, à medida que a cardinalidade de um ou de ambos os conjuntos aumenta, a faixa de valores fornecidos decresce. Assim, ao compararmos tricordes entre si, os valores de similaridade se situam entre 0 e 1; ao compararmos tricordes com nonacordes, a faixa se reduz para valores entre 0,5690 e 0,6395; e, ao compararmos nonacordes entre si, os valores ficam próximos do máximo (1). Adicionalmente, esse autor também afirma que o valor 1 (máxima similaridade) só é obtido quando os vetores são idênticos, o que não se confirma pelo cálculo mostrado nas Equações 5 e 6.

$$Rel_{016,0167} = \frac{2.(\sqrt{1.2} + \sqrt{0.0} + \sqrt{0.0} + \sqrt{0.0} + \sqrt{1.2} + \sqrt{1.2})}{\sqrt{3.(3\text{-}1).4.(4\text{-}1)}} \quad \text{Eq. 5}$$

$$Rel_{016,0167} = \frac{6.\sqrt{2}}{6.\sqrt{2}} = 1 \quad \text{Eq. 6}$$

O último autor examinado por Isaacson é Richard Teitelbaum, o qual, em um artigo de 1965, propôs um método de cálculo de similaridades que, apesar de ser mais antigo, fornece uma melhor discriminação (ou seja, uma maior quantidade de valores) do que as propostas anteriores (Forte, Lord, Morris), ao mesmo tempo em que não produz valores próximos da máxima similaridade quando as cardinalidades aumentam. O índice de similaridade (*s.i.*) entre dois conjuntos (*X,Y*) é dado pela raiz quadrada do somatório dos quadrados das diferenças entre os termos dos vetores intervalares de X e Y (Eq. 7). Como enfatiza Isaacson (1990: 16), "Teitelbaum aplica sua função somente a conjuntos de mesma cardinalidade"[16].

$$s.i.(X,Y) = \sqrt{\Sigma\,(x_i - y_i)^2} \quad \text{Eq. 7}$$

Finalmente, Isaacson apresenta sua proposta de cálculo de similaridade, denominada IcVSIM, que combina o grau de distinção de Teitelbaum e pode ser aplicada a conjuntos de cardinalidades diferentes. A fórmula de Isaacson calcula inicialmente a diferença entre as entradas dos vetores intervalares, produzindo o que ele denomina IdV (*interval-difference vector*) e calcula seu desvio padrão (σ). Esse é o valor de IcVSIM, cuja fórmula é mostrada na Equação 8. O exemplo abaixo tornará clara a aplicação dessa fórmula.

$$IcVSIM\,(X,Y) = \sqrt{\frac{\Sigma\,(IdV_i - \overline{IdV})^2}{6}} \quad \text{Eq. 8}$$

Como exemplo, calculemos o IcVSIM entre as classes de conjuntos 012 e 012345. Primeiramente calculamos o IdV pela subtração dos componentes dos vetores intervalares de

[16] Original: "Teitelbaum applies the function only to sets of the same size".

cada uma das classes de conjuntos. O vetor intervalar de 012 é <210000>, e o vetor intervalar de 012345 é <543210>. Subtraindo-se seus componentes, teremos que o IdV é <333210>. A média (***IdV***) dos termos de IdV é 2 (obtida pela soma dos componentes, 12, e a subsequente divisão pelo número de componentes, 6). De posse desses valores, é possível então calcular a similaridade entre as classes de conjuntos 012 e 012345 (Eq. 9 e 10)

$$IcVSIM\ (X,Y)=\sqrt{\frac{(3-2)^2 + (3-2)^2 + (3-2)^2 + (2-2)^2 + (1-2)^2 + (0-2)^2}{6}} \qquad \text{Eq. 9}$$

$$IcVSIM\ (X,Y) = \sqrt{\frac{1+1+1+0+1+4}{6}} = 1{,}15 \qquad \text{Eq. 10}$$

Temos então uma série de possibilidades para o cálculo da similaridade entre conjuntos de classes de notas. Embora a função IcVSIM de Isaacson pareça ser a solução definitiva para melhor efetivar tal cálculo, tendo em vista que resolve os pontos frágeis associados aos modelos anteriores, segundo os pré-requisitos definidos pelo próprio Isaacson, observamos que todas as propostas indicam algum tipo de similaridade. Como observado por Honingh *et al.* (2009),

> À primeira vista, essas medidas de similaridade parecem não estar tão relacionadas entre si. Eles diferem em alcance, intenção, forma de cálculo e muito mais. No entanto, Quinn [2006] argumenta que essas medidas de similaridade têm realmente muito em comum: elas tendem a agrupar os IcVs [vetores intervalares] em seis categorias diferentes, cada uma das quais pode-se dizer que corresponde a um ciclo de uma das seis classes de intervalo[17].

Em nosso sistema aplicaremos o modelo de Lewin, uma vez que ele produz uma quantidade suficiente de classes de similaridade e permite a comparação entre conjuntos de cardinalidades diferentes, mesmo com os problemas que podem surgir nas situações apontadas por Isaacson, já mencionadas acima. O mesmo sistema, como veremos adiante, suporta outras funções para o cálculo de similaridade. Assim sendo, encorajamos outros compositores a realizarem experimentos nessa perspectiva, tendo em vista que nosso foco, neste artigo, é a formalização de um sistema composicional.

## Módulos auxiliares em Python

A fim de otimizar o trabalho, foram desenvolvidos três aplicativos em Python. O primeiro calcula o vetor intervalar do conjunto de classes de notas inserido. O segundo calcula o grau de similaridade entre dois conjuntos de classes de notas, com base na fórmula de Lewin. O terceiro aplicativo nos retorna à paleta de um determinado conjunto. Os códigos-fonte dos aplicativos são mostrados nos Quadros 2, 3 e 4.

[17] Original: "[...] at first sight, these similarity measures seem to be not so much related to each other. They differ in range, intention, way of calculation and more. However, Quinn argues that those similarity measures have actually a lot in common: they tend to group the IcV's in six different categories, each of which can be said to correspond to a cycle of one of the six interval classes".

```
print('')
print('==========Calculo do vetor intervalar==========')
print('')
print('Entre com um conjunto no formato xyz...na base 12')
A = list(input ('Conjunto:'))

#Função que converte a entrada em lista de inteiros
def convert (conjunto):
    for i in range(len(conjunto)):
        if conjunto[i]=='a':
            conjunto[i]=10
        if conjunto[i]=='b':
            conjunto[i]=11
        else: conjunto[i]=int(conjunto[i])
    return(conjunto)

#função que calcula o vetor intervalar
def vetor_intervalar (A):

    B = [0,0,0,0,0,0]

    for i in range(len(A)-1):
        for k in range (i+1,len(A)):
            intervalo=abs(A[i]-A[k])

            if intervalo==1 or intervalo==11:B[0]=B[0]+1
            if intervalo==2 or intervalo==10:B[1]=B[1]+1
            if intervalo==3 or intervalo==9:B[2]=B[2]+1
            if intervalo==4 or intervalo==8:B[3]=B[3]+1
            if intervalo==5 or intervalo==7:B[4]=B[4]+1
            if intervalo==6:B[5]=B[5]+1

    return(B)

print('Vetor intervalar:', vetor_intervalar(convert(A)))
```

**Quadro 2:** Programa em Python para cálculo do vetor intervalar

```
from math import *
from modulo_vetor_intervalar import *

def similaridade (A,B):
    Va = vetor_intervalar(convert(A))
    Vb = vetor_intervalar(convert(B))

    #inicializa o numerador da função de Lewin e calcula a cardinalidade dos conjuntos
    m = 0
    cA = len(A)
    cB = len(B)

    #calcula o denominador da função de Lewin
    d=sqrt(cA*(cA-1)*cB*(cB-1))

    #Calcula o somatório dos componentes dos vetores no numerador
    for i in range(6):
        m = m + sqrt(Va[i]*Vb[i])

    #calcula a similaridade entre os dois conjuntos
    s = 2*m/d

    return(s)
```

**Quadro 3:** Função para cálculo da similaridade entre dois conjuntos. Essa função utiliza o módulo vetor_intervalar (Quadro 2).

```
import itertools
from modulo_vetor_intervalar import *

conjunto = input('Entre com a classe de conjuntos:')
conjunto = convert(conjunto)

teste = []; #matriz final
transp=[]
inv=[]
matT = []; #matriz de transposição
matI = []; #matriz de inversão

for i in range(0,12):
    matT = [((x+i)%12) for x in conjunto]
    matT.sort()
    matI = [((12-(x+i))%12) for x in conjunto]
    matI.sort()
    if matT not in teste:
        teste.append(matT)
        transp.append(matT)
    if matI not in teste:
        teste.append(matI)
        inv.append(matI)

print("Tn" + " ------|--- ","TnI")
for a,b in itertools.zip_longest(transp,inv,fillvalue = ''):
    print(a,b)
```

**Quadro 4:** Gerador de paletas

## Sistema composicional Similare

Nesta seção, definiremos um sistema composicional semiaberto, denominado Similare, com foco na similaridade entre conjuntos de classes de notas. O fator de similaridade será decisivo na articulação formal da obra, uma vez que o sistema composicional seciona trechos pelo grau de similaridade entre classes de conjuntos de classes de notas. As seguintes diretrizes compõem o sistema:

1. A macroestrutura da obra alterna dois trechos.
2. O material do trecho A consistirá unicamente de três classes de conjuntos (*set classes*) distintas, de qualquer cardinalidade, com similaridade superior a 0,8. Serão construídas variantes de A, em termos de altura, textura e ritmo.
3. O material do trecho B consistirá unicamente de três classes de conjuntos distintas, de qualquer cardinalidade, com similaridade inferior a 0,5. Serão construídas variantes de B, em termos de altura, textura e ritmo.
4. Trechos da mesma família podem ser conectados livremente entre si (Ex.: A1-A4-A3 etc.). No entanto, as conexões entre trechos de famílias diferentes, ou seja, entre qualquer variante de A com qualquer variante B, será efetivada por um trecho transicional. Esses trechos transicionais serão elaborados de acordo com os seguintes critérios: se a transição for de A para B, esse trecho transicional deve ter sua similaridade

gradativamente reduzida. Inversamente, a conexão entre os trechos B e A é efetivada por um trecho transicional no qual a similaridade é gradativamente ampliada. A extensão temporal e a quantidade de classes de conjuntos desses trechos transicionais são determinadas na fase de planejamento composicional.

5. Trechos livres podem ser inseridos em qualquer trecho da obra, contanto que se restrinjam a no máximo 20% do total de trechos. O material para esses trechos é produzido pelo recorte aleatório de dois trechos contíguos quaisquer (por exemplo, final de A1 e início de A2). O trecho recortado será transposto e/ou invertido e/ou retrogradado. O fator de transposição será determinado na fase de planejamento composicional.

## Planejamento composicional de *Sistêmica*

O objetivo desta seção é a descrição detalhada do planejamento composicional de *Semiaberto*, primeiro movimento de uma obra para quarteto de cordas, intitulada *Sistêmica*, de Liduino Pitombeira, com base no sistema elaborado na seção anterior. O planejamento composicional se inicia pela geração de repositórios, tomando como base as definições do sistema Similare. O sistema indica que a obra se constrói com base na determinação de dois trechos que se alternam: um trecho A, com alta similaridade entre três classes de conjuntos, e um trecho B, com baixa similaridade, também entre três classes de conjuntos. Para alta similaridade, atribuíram-se valores acima de 0,8 e, para baixa similaridade, valores abaixo de 0,5. Desta forma, escolhemos para A1 (o primeiro trecho de A) as classes de conjuntos 016, 0167 e 0156, as quais têm similaridades que vão de 0,9 a 1 (Tabela 2)[18].

| | 016 | 0167 | 0156 |
|---|---|---|---|
| 016 | 1 | 1 | 0,9 |
| 0167 | 1 | 1 | 0,9 |
| 0156 | 0.9 | 0.9 | 1 |

**Tab. 2:** Classes de conjuntos do trecho A1 com suas respectivas intersimilaridades

Tais classes serão distribuídas texturalmente de acordo com um planejamento matricial, similar de certa forma ao ***design* composicional** de Morris (1987). A similaridade entre o planejamento matricial empregado neste trabalho e a proposta de Morris se dá unicamente pelo aspecto gráfico das matrizes, em especial com relação à distribuição defasada das classes de notas, o que indica já uma ordenação temporal, ainda que sem uma especificação rítmica precisa. A diferença com a proposta de Morris se dá principalmente em termos de conteúdo: enquanto em Morris há

[18] É interessante constatar que a classe de conjuntos 0167 apresenta quatro subconjuntos tricordais, cujas formas normais são (016), (701), (670) e (167). Todos esses conjuntos têm forma prima [016]. Desta forma, processa-se no interior do tetracorde 0167 uma autossimilaridade profunda. É daí que decorre a similaridade tão intensa entre essas duas classes de conjuntos (016 e 0167), que pode ser experimentada em termos de percepção. Por sua vez, os quatro subconjuntos tricordais da classe de conjuntos 0156 são, em termos de formais normais: (015) e (156), pertencentes à classe 015, e (016) e (056), pertencentes à classe 016. Uma observação dos vetores intervalares também pode revelar por que as classes 016 e 0167 apresentam essa similaridade auditiva. O vetor do primeiro é <100011>, e o vetor do segundo é <200022>, ou seja, ambos têm as mesmas classes intervalares (1, 5 e 6). O que muda é tão somente a quantidade dessas classes no interior dos conjuntos.

um forte inter-relacionamento entre os componentes[19], em nosso planejamento, as classes de conjuntos derivam unicamente das diretrizes sistêmicas. A Tab. 3 demonstra a distribuição das classes de conjuntos do trecho A1 entre os quatro instrumentos do quarteto. Como essa tabela mostra as classes de conjuntos, para a determinação do conjunto específico, precisaremos gerar as paletas de cada classe. As paletas para as três classes de conjuntos são mostradas na Tab. 4. A Fig. 7 mostra o trecho da Tab. 3 realizado com ritmos, articulações, dinâmicas. Do lado direito do trecho realizado musicalmente, temos duas tabelas: a superior é a própria Tab. 3; a inferior mostra as formas normais escolhidas (Tab. 4) e distribuídas desordenadamente.

| | | | |
|---|---|---|---|
| Violino 1 | 56 | | 0167 |
| Violino 2 | 01 | 01 | 016 |
| Viola | | 6 | 01 |
| Violoncelo | | | 56 |

**Tab. 3:** Classes de conjuntos do trecho A1 distribuídas no quarteto de cordas

| 0156 | 016 | | 0167 |
|---|---|---|---|
| Tn/TnI | Tn | TnI | Tn/TnI |
| (0 1 5 6) | (0 1 6) | (6 B 0) | (0 1 6 7) |
| (1 2 6 7) | (1 2 7) | (5 A B) | (1 2 7 8) |
| (2 3 7 8) | (2 3 8) | (4 9 A) | (2 3 8 9) |
| (3 4 8 9) | (3 4 9) | (3 8 9) | (3 4 9 A) |
| (4 5 9 A) | (4 5 A) | (2 7 8) | (4 5 A B) |
| (5 6 A B) | (5 6 B) | (1 6 7) | (5 6 B 0) |
| (6 7 B 0) | (6 7 0) | (0 5 6) | |
| (7 8 0 1) | (7 8 1) | (B 4 5) | |
| (8 9 1 2) | (8 9 2) | (A 3 4) | |
| (9 A 2 3) | (9 A 3) | (9 2 3) | |
| (A B 3 4) | (A B 4) | (8 1 2) | |
| (B 0 4 5) | (B 0 5) | (7 0 1) | |

**Tab. 4:** Paletas das classes de conjuntos do trecho A1

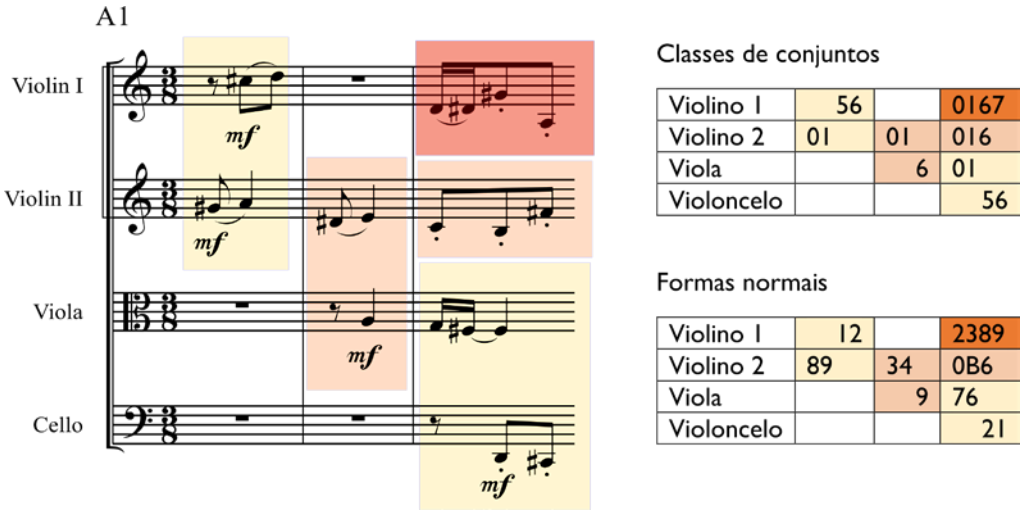



**Fig. 7:** Trecho A1

[19] Observe-se, por exemplo, o *design* matricial de Morris (1987: 229), no qual todas as classes de alturas derivam unicamente no conjunto {0,1,2,6,8}.

Da mesma forma, proporemos o primeiro trecho B, com características predominantemente verticais (harmônicas). As sonoridades desse trecho devem ter, segundo determinação do próprio sistema, similaridades inferiores a 0,5. Escolhemos as classes de conjuntos 037, 036 e 0246, cujas intersimilaridades são mostradas na Tab. 5. A Tab. 6 mostra a disposição dos conjuntos, e a Tab. 7 mostra a paleta das classes de conjuntos. A representação musical de B1 é mostrada na Fig. 8.

| | 037 | 036 | 0246 |
|---|---|---|---|
| 037 | — | 0,4714 | 0,3651 |
| 036 | 0,4714 | — | 0,2357 |
| 0246 | 0,3333 | 0,2357 | — |

**Tab. 5:** Classes de conjuntos do trecho B1

| | | | |
|---|---|---|---|
| Violino I | 0 | 0 | 0 |
| Violino 2 | 3 | 2 | 3 |
| Viola | 7 | 4 | 6 |
| Violoncelo | | 6 | |

**Tab. 6:** Classes de conjuntos do trecho B1 distribuídas no quarteto de cordas

| 037 | | 036 | 0246 |
|---|---|---|---|
| Tn | TnI | Tn/TnI | Tn/TnI |
| (0 3 7) | (5 9 0) | (0 3 6) | (0 2 4 6) |
| (1 4 8) | (4 8 B) | (1 4 7) | (1 3 5 7) |
| (2 5 9) | (3 7 A) | (2 5 8) | (2 4 6 8) |
| (3 6 A) | (2 6 9) | (3 6 9) | (3 5 7 9) |
| (4 7 B) | (1 5 8) | (4 7 A) | (4 6 8 A) |
| (5 8 0) | (0 4 7) | (5 8 B) | (5 7 9 B) |
| (6 9 1) | (B 3 6) | (6 9 0) | (6 8 A 0) |
| (7 A 2) | (A 2 5) | (7 A 1) | (7 9 B 1) |
| (8 B 3) | (9 1 4) | (8 B 2) | (8 A 0 2) |
| (9 0 4) | (8 0 3) | (9 0 3) | (9 B 1 3) |
| (A 1 5) | (7 B 2) | (A 1 4) | (A 0 2 4) |
| (B 2 6) | (6 A 1) | (B 2 5) | (B 1 3 5) |

**Tab. 7:** Paletas das classes de conjuntos do trecho B1

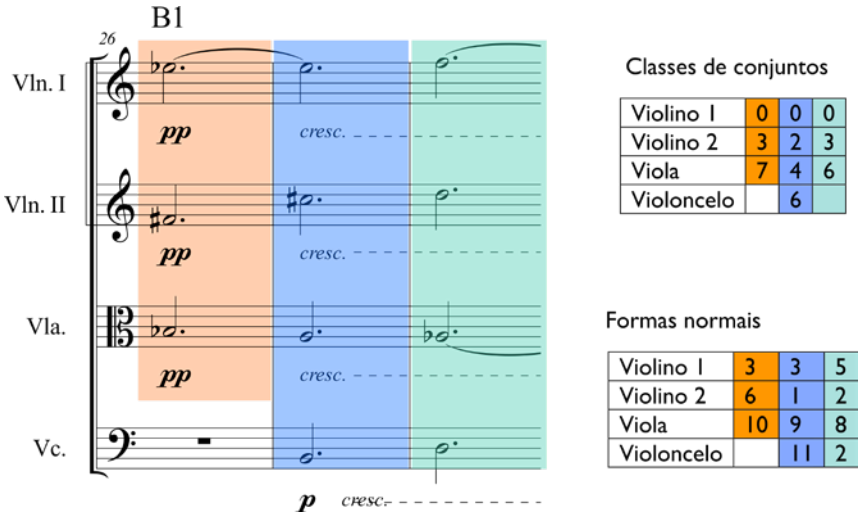



**Fig. 8:** Trecho B1

Determinados os trechos A1 e B1, estes serão submetidos a variações em termos de altura, textura e ritmo, de acordo com a segunda e terceira diretrizes do sistema Similare. Na Fig. 9 demonstramos a criação do trecho A2 pela inversão/transposição e a alteração rítmica e textural do trecho A1. O fator de inversão/transposição foi 0. O primeiro componente de A1, (8912), é submetido à operação $T_0I$, mas se mantém com o mesmo ritmo e na mesma distribuição instrumental (Violinos 1 e II). O segundo componente, (349), é submetido a T01 e é deslocado do Violino 2 e Viola para os Violinos 1 e II. No terceiro componente, os conteúdos dos Violinos I e II são submetidos à operação $T_01$, mantendo o mesmo ritmo e mesma distribuição instrumental, mas o conjunto de classes de notas, além de ser submetido à mesma operação, tem alterações rítmicas e se concentra unicamente na viola, ficando o violoncelo em silêncio. Tal alteração rítmica também provoca alteração textural: em termos particionais[20], a textura do terceiro componente de A1 (correspondente ao terceiro compasso) passa de 1.2 – $1^3$– 1.3 para $1^3$ – 1.2. Foram criados seis repositórios de A (A1-A6) e cinco de B (B1-B5), utilizando o mesmo procedimento de construção.

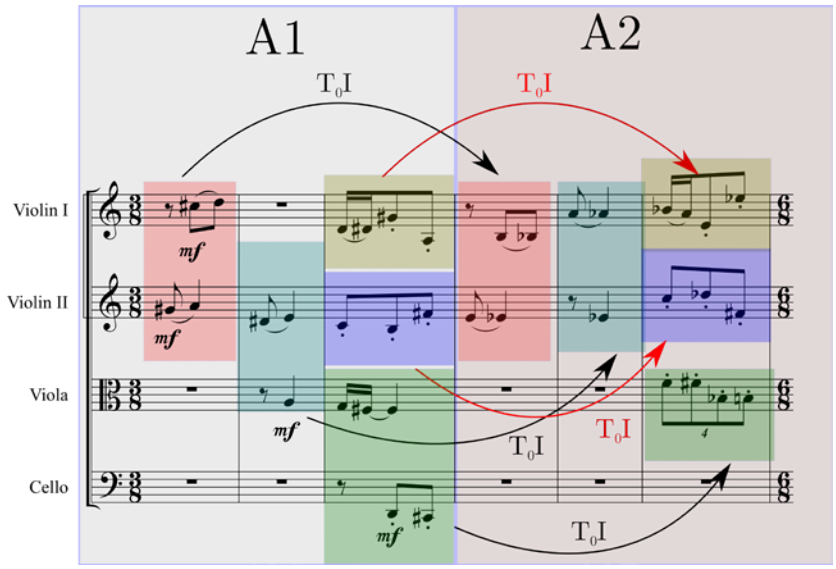



**Fig. 9:** Criação do trecho A2 pela inversão/transposição e alteração rítmica e textural do trecho A1

A quarta diretriz do sistema Similare trata da conexão entre os trechos A e B, em qualquer uma de suas configurações. Essa conexão ocorre de maneira transicional, obtida por meio de uma gradação entre as similaridades, utilizando qualquer quantidade de classes de conjuntos. No Quadro 5, planejamos a transição entre A e B através de quatro classes de conjuntos: as duas primeiras, X e Y, têm similaridade entre 0,8 e 0,7 e são sobrepostas, isto é, são organizadas de maneira simultânea; as duas últimas, ZW, têm similaridade entre 0, 4714 e 0,6 e são justapostas, ou seja, organizadas de maneira linear.

Se escolhermos como X a classe 0137 e como Y a classe 02479, teremos satisfeitas as condições de similaridade do primeiro par de classes de conjuntos, uma vez que a similaridade entre ambos é de 0,79. Da mesma forma, se Z for 0146 e W for 027, com similaridade igual a 0,57, teremos satisfeitas as condições do segundo par. Foram criadas três transições de A para

[20] Para maiores informações sobre esse método de análise textural, ver Gentil-Nunes (2009).

B (AB1, AB2, AB3) e três transições de B para A (BA1, BA2, BA3). As paletas das classes de conjuntos são mostradas na Tab. 8. A Fig. 10 mostra, como exemplo, a transição AB1.

| A | X<br>Y | ZW | B |
|---|---|---|---|
| 1 - 0,9 | 0,8 - 0,7 | 0,6 - 0,4714 | 0,4717 - 0,2357 |

**Quadro 5:** Plano de transição entre os trechos A e B, através das sonoridades XY dispostas verticalmente e ZW dispostas horizontalmente. A transição entre B e A será efetuada de forma reversa.

| 0137 | | 02479 | | 0146 | | 027 | |
|---|---|---|---|---|---|---|---|
| Tn | Tn/I | Tn | TnI | Tn | TnI | Tn | TnI |
| (0 1 3 7) | (0 B95) | (0 2 4 7 9) | (0A853) | (0 1 4 6) | (0B86) | (0 2 7) | (0A5) |
| (1 2 4 8) | (10A6) | (1 3 5 8 A) | (1B964) | (1 2 5 7) | (1097) | (1 3 8) | (1B6) |
| (2 3 5 9) | (21B7) | (2 4 6 9 B) | (20A75) | (2 3 6 8) | (21A8) | (2 4 9) | (207) |
| (3 4 6 A) | (3208) | (3 5 7 A 0) | (31B86) | (3 4 7 9) | (32B9) | (3 5 A) | (318) |
| (4 5 7 B) | (4319) | (4 6 8 B 1) | (42097) | (4 5 8 A) | (430A) | (4 6 B) | (429) |
| (5 6 8 0) | (542A) | (5 7 9 0 2) | (531A8) | (5 6 9 B) | (541B) | (5 7 0) | (53A) |
| (6 7 9 1) | (653B) | (6 8 A 1 3) | (642B9) | (6 7 A 0) | (6520) | (6 8 1) | (64B) |
| (7 8 A 2) | (7640) | (7 9 B 2 4) | (7530A) | (7 8 B 1) | (7631) | (7 9 2) | (750) |
| (8 9 B 3) | (8751) | (8 A 0 3 5) | (8641B) | (8 9 0 2) | (8742) | (8 A 3) | (861) |
| (9 A 0 4) | (9862) | (9 B 1 4 6) | (97520) | (9 A 1 3) | (9853) | (9 B 4) | (972) |
| (A B 1 5) | (A973) | (A 0 2 5 7) | (A8631) | (A B 2 4) | (A964) | (A 0 5) | (A83) |
| (B 0 2 6) | (BA84) | (B 1 3 6 8) | (B9742) | (B 0 3 5) | (BA75) | (B 1 6) | (B94) |

| | |
|---|---|
| AB1 | |
| AB2 | |
| AB3 | |

| | |
|---|---|
| BA1 | |
| BA2 | |
| BA3 | |

**Tab. 8:** Paletas das classes de conjuntos das transições A-B e B-A: 0137,02479,0146 e 027. A legenda de cores abaixo dessa tabela indica quais os conjuntos utilizados em cada uma das transições.

Por último, foram criados trechos livres, segundo a quinta definição do sistema. Exemplificaremos o procedimento de criação do trecho livre L1, pela concatenação do último compasso de A5 (c. 4) com o primeiro compasso de AB1 (c. 5). Cada segmento foi submetido à inversão $T_0I$. O procedimento é ilustrado na Fig. 11: do lado esquerdo se encontra o recorte dos trechos originais e do lado direito o novo trecho que deu origem ao trecho livre L1 (c. 58).

De posse do repositório, iniciamos a justaposição dos trechos de forma empírica. O plano estrutural da obra é mostrado na Tab. 9. Durante a composição da obra, ajustes foram realizados na agógica, na dinâmica e na articulação, quando necessário. Um compasso de silêncio, não previsto no sistema, foi introduzido em um dado momento da obra para rearticular a recorrência de um trecho. A partitura integral da obra se encontra no Apêndice 1 deste artigo.

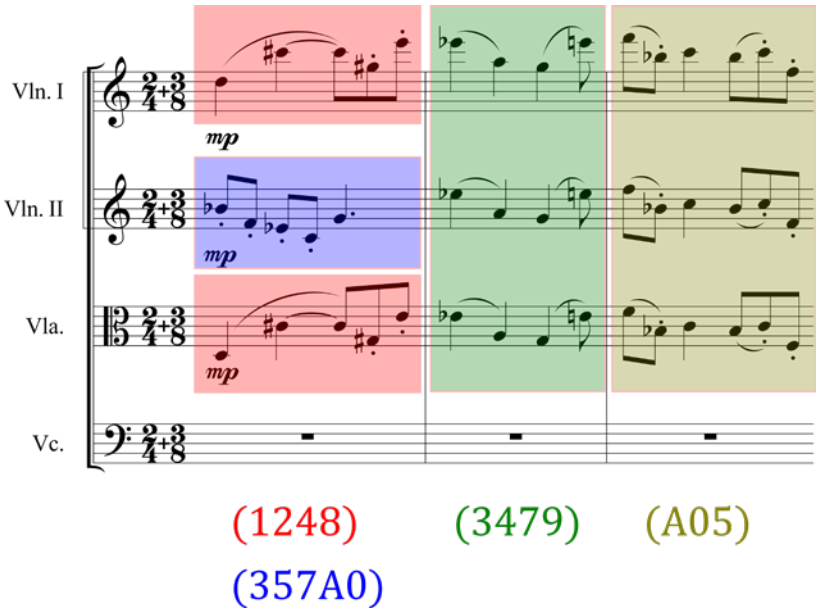



**Fig. 10:** Transição AB1: seguindo as formas normais especificadas na Tab. 7, temos a sobreposição de (1248) e (357A0), seguida pela justaposição de (3479) e (A05).

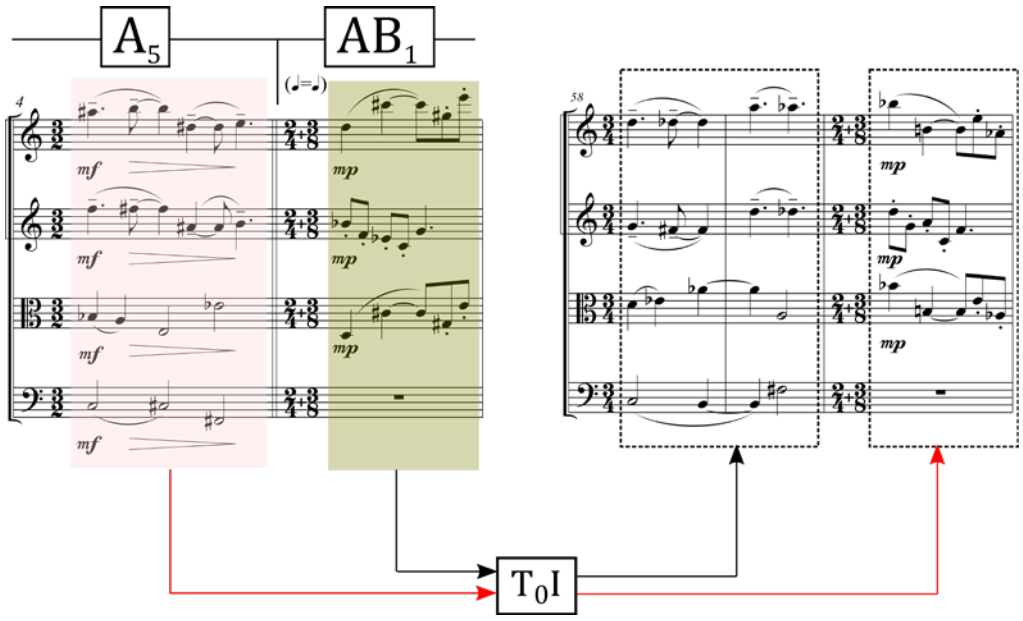



**Fig. 11:** Criação do trecho livre L1, pela concatenação do final de A5 com o início de AB2 e subsequente inversão ($T_0I$)

| Trecho | Compasso |
|---|---|
| A5 | 01 – 04 |
| AB1 | 05 – 07 |
| B3 | 08 – 10 |
| B2 | 11 |

**Tab. 9:** Plano estrutural da distribuição dos trechos | Continua...

| Trecho | Compasso |
|---|---|
| BA1 | 12 – 13 |
| A4 | 14 – 21 |
| A6 | 22 – 33 |
| AB2 | 34 – 36 |
| B1 | 37 – 39 |
| B4 | 40 – 46 |
| B5 | 47 – 55 |
| BA2 | 56 – 58 |
| L3 | 59 – 60 |
| A3 | 61 – 63 |
| L1 | 64 – 66 |
| A1 | 67 – 69 |
| A2 | 70 – 72 |
| L2 | 73 – 74 |
| A4 | 75 – 82 |
| A6 | 84 – 95 |

**Tab. 9:** Plano estrutural da distribuição dos trechos

## Planejamento composicional de *SIM(ulation)*

Nesta seção será detalhada a composição de *SIM(ulation)*, para piano, de Pedro Proença, que foi planejada com a ajuda de dois programas escritos em Python. O primeiro programa, denominado *sistema1*, tem a função de gerar as classes de conjuntos para os trechos A e B. O segundo programa, denominado *transitions*, gera as classes de conjuntos para as transições. Dado um valor inicial para a classe de conjunto do trecho A (com similaridade superior a 0,8) e outro para a classe de conjunto do trecho B (com similaridade ou inferior a 0,5), as demais classes de conjuntos foram determinadas aleatoriamente pelo *sistema1*. Ao confeccionar as paletas, foi decidido que a representação dos conjuntos de classes de notas não se daria por suas formas normais, mas seriam arranjados por suas transposições, sempre com o primeiro elemento igual a 0, e cada conjunto representado em ordem crescente.

Para o trecho A, escolhemos a classe 0158. Lançando esse valor no *sistema1*, encontramos as classes de conjuntos 014568 e 0147, que se adéquam ao sistema Similare. Essas três classes de conjuntos são denominadas A1, A2 e A3, respectivamente. Essas classes de conjuntos foram distribuídas na Tab. 10 com seus graus de similaridade. Similarmente, tendo como classe de conjunto inicial do trecho B o 0246, geramos através do *sistema1* as classes de conjuntos 015 e 0123, que satisfazem às condições do sistema (Tab. 12). Essas três classes de conjuntos são denominadas B1, B2 e B3, respectivamente. Em seguida, construímos as paletas para cada uma dessas classes de conjuntos (Tab. 11 e 13).

| | 0158 | 0147 | 014568 |
|---|---|---|---|
| 0158 | — | 0,8738 | 0,888 |
| 0147 | 0,8738 | — | 0,8922 |
| 014568 | 0,888 | 0,8922 | — |

**Tab. 10:** Classes de conjuntos do trecho A

| A1 = 0158 | A2 = 0147 | | A3 = 014568 | |
|---|---|---|---|---|
| Tn/TnI | Tn | TnI | Tn | TnI |
| (0 1 5 8) | (0 1 4 7) | (0 5 8 B) | (0 1 4 5 6 8) | (0 4 6 7 8 B) |
| (1 2 6 9) | (1 2 5 8) | (1 6 9 0) | (1 2 5 6 7 9) | (1 5 7 8 9 0) |
| (2 3 7 A) | (2 3 6 9) | (2 7 A 1) | (2 3 6 7 8 A) | (2 6 8 9 A 1) |
| (3 4 8 B) | (3 4 7 A) | (3 8 B 2) | (3 4 7 8 9 B) | (3 7 9 A B 2) |
| (4 5 9 0) | (4 5 8 B) | (4 9 0 3) | (4 5 8 9 A 0) | (4 8 A B 0 3) |
| (5 6 A 1) | (5 6 9 0) | (5 A 1 4) | (5 6 9 A B 1) | (5 9 B 0 1 4) |
| (6 7 B 2) | (6 7 A 1) | (6 B 2 5) | (6 7 A B 0 2) | (6 A 0 1 2 5) |
| (7 8 0 3) | (7 8 B 2) | (7 0 3 6) | (7 8 B 0 1 3) | (7 B 1 2 3 6) |
| (8 9 1 4) | (8 9 0 3) | (8 1 4 7) | (8 9 0 1 2 4) | (8 0 2 3 4 7) |
| (9 A 2 5) | (9 A 1 4) | (9 2 5 8) | (9 A 1 2 3 5) | (9 1 3 4 5 8) |
| (A B 3 6) | (A B 2 5) | (A 3 6 9) | (A B 2 3 4 6) | (A 2 4 5 6 9) |
| (B 0 4 7) | (B 0 3 6) | (B 4 7 A) | (B 0 3 4 5 7) | (B 3 5 6 7 A) |

**Tab. 11:** Paletas das classes de conjuntos do trecho A

| | 0246 | 015 | 0123 |
|---|---|---|---|
| 0246 | — | 0,3333 | 0,4083 |
| 015 | 0,3333 | — | 0,4083 |
| 0123 | 0,4083 | 0,4083 | — |

**Tab. 12:** Classes de conjuntos do trecho B

| B1 = 0246 | B2 = 015 | | B3 = 0123 |
|---|---|---|---|
| Tn/TnI | Tn | TnI | Tn/TnI |
| (0 2 4 6) | (0 1 5) | (0 7 B) | (0 1 2 3) |
| (1 3 5 7) | (1 2 6) | (1 8 0) | (1 2 3 4) |
| (2 4 6 8) | (2 3 7) | (2 9 1) | (2 3 4 5) |
| (3 5 7 9) | (3 4 8) | (3 A 2) | (3 4 5 6) |
| (4 6 8 A) | (4 5 9) | (4 B 3) | (4 5 6 7) |
| (5 7 9 B) | (5 6 A) | (5 0 4) | (5 6 7 8) |
| (6 8 A 0) | (6 7 B) | (6 1 5) | (6 7 8 9) |
| (7 9 B 1) | (7 8 0) | (7 2 6) | (7 8 9 A) |
| (8 A 0 2) | (8 9 1) | (8 3 7) | (8 9 A B) |
| (9 B 1 3) | (9 A 2) | (9 4 8) | (9 A B 0) |
| (A 0 2 4) | (A B 3) | (A 5 9) | (A B 0 1) |
| (B 1 3 5) | (B 0 4) | (B 6 A) | (B 0 1 2) |

**Tab. 13:** Paletas das classes de conjuntos do trecho B

Decidimos que seriam duas transições entre as seções A e B (cada uma com três elementos) e três transições entre B e A (cada uma com dois elementos). Os conjuntos das transições foram escolhidos aleatoriamente, de acordo com os seguintes critérios: 1) A primeira transição de A para B (Tr1AB) consiste em três conjuntos (Tr1AB1, Tr1AB2, Tr1AB3), cujas similaridades são maiores que 0,65 e menores que 0,8; 2) A segunda transição de A para B (Tr2AB) consiste em dois conjuntos (Tr2AB1, Tr2AB2), cujas similaridades são maiores que 0,5 e menores que 0,65; a primeira transição de B para A (Tr1BA) consiste em dois conjuntos (Tr1BA1, Tr1BA2)

com similaridades entre 0,5 e 0,6; 3) A segunda transição de B para A (Tr2BA) consiste em dois conjuntos (Tr2BA1, Tr2BA2) com similaridades entre 0,6 e 0,7; 4) A terceira transição de B para A (Tr3BA) consiste em dois conjuntos (Tr3BA1, Tr3BA2) com similaridades entre 0,7 e 0,8. Os conjuntos escolhidos pelo programa *transitions* foram: Tr1AB1 = 01248, Tr1AB2 = 0235, Tr1AB3 = 0127, Tr2AB1 = 01246, Tr2AB2 = 037, Tr1BA1 = 02358, Tr1BA2 = 012, Tr2BA1 = 0258, Tr2BA2 = 0145, Tr3BA1 = 013457 e Tr3BA2 = 015. Esses critérios são resumidos na Tab. 14. As Tab. 15-19 contêm as paletas de todos esses conjuntos. As Tab. 19-26 mostram o planejamento detalhado de todas as classes de conjuntos utilizadas em *SIM(ulation)*, compasso a compasso. Os demais parâmetros (ritmo, articulação, dinâmica etc.) foram livremente escolhidos pelo compositor. A partitura integral da obra se encontra no Apêndice 2 deste artigo.

| Transição | Nome abreviado | Abreviações dos conjuntos | Similaridades |
|---|---|---|---|
| Transição 1 de A para B | Tr1AB | Tr1AB1, Tr1AB2, Tr1AB3 | 0,65 < SIM < 0,8 |
| Transição 2 de A para B | Tr2AB | Tr2AB1, Tr2AB2 | 0,5 < SIM < 0,65 |
| Transição 1 de B para A | Tr1BA | Tr1BA1, Tr1BA2 | 0,5 < SIM < 0,6 |
| Transição 2 de B para A | Tr2BA | Tr2BA1, Tr2BA2 | 0,6 < SIM < 0,7 |
| Transição 3 de B para A | Tr3BA | Tr3BA1, Tr3BA2 | 0,7 < SIM < 0,8 |

**Tabela 14:** Critérios para a confecção dos conjuntos de transição

| Tr1AB1 = 01248 | | Tr1AB2 = 0235 | Tr1AB3 = 0127 |
|---|---|---|---|
| Tn | TnI | Tn/TnI | Tn/TnI |
| (0 1 2 4 8) | (0 4 8 A B) | (0 2 3 5) | (0 1 2 7) |
| (1 2 3 5 9) | (1 5 9 B 0) | (1 3 4 6) | (1 2 3 8) |
| (2 3 4 6 A) | (2 6 A 0 1) | (2 4 5 7) | (2 3 4 9) |
| (3 4 5 7 B) | (3 7 B 1 2) | (3 5 6 8) | (3 4 5 A) |
| (4 5 6 8 0) | (4 8 0 2 3) | (4 6 7 9) | (4 5 6 B) |
| (5 6 7 9 1) | (5 9 1 3 4) | (5 7 8 A) | (5 6 7 0) |
| (6 7 8 A 2) | (6 A 2 4 5) | (6 8 9 B) | (6 7 8 1) |
| (7 8 9 B 3) | (7 B 3 5 6) | (7 9 A 0) | (7 8 9 2) |
| (8 9 A 0 4) | (8 0 4 6 7) | (8 A B 1) | (8 9 A 3) |
| (9 A B 1 5) | (9 1 5 7 8) | (9 B 0 2) | (9 A B 4) |
| (A B 0 2 6) | (A 2 6 8 9) | (A 0 1 3) | (A B 0 5) |
| (B 0 1 3 7) | (B 3 7 9 A) | (B 1 2 4) | (B 0 1 6) |

**Tab. 15:** Paletas das classes de conjuntos da transição 1 entre A e B

| Tr2AB1 = 01246 | |
|---|---|
| Tn | TnI |
| (0 1 2 4 6) | (0 6 8 A B) |
| (1 2 3 5 7) | (1 7 9 B 0) |
| (2 3 4 6 8) | (2 8 A 0 1) |
| (3 4 5 7 9) | (3 9 B 1 2) |
| (4 5 6 8 A) | (4 A 0 2 3) |
| (5 6 7 9 B) | (5 B 1 3 4) |
| (6 7 8 A 0) | (6 0 2 4 5) |
| (7 8 9 B 1) | (7 1 3 5 6) |
| (8 9 A 0 2) | (8 2 4 6 7) |
| (9 A B 1 3) | (9 3 5 7 8) |
| (A B 0 2 4) | (A 4 6 8 9) |
| (B 0 1 3 5) | (B 5 7 9 A) |

| Tr2AB2 = 037 | |
|---|---|
| Tn | TnI |
| (0 3 7) | (0 5 9) |
| (1 4 8) | (1 6 A) |
| (2 5 9) | (2 7 B) |
| (3 6 A) | (3 8 0) |
| (4 7 B) | (4 9 1) |
| (5 8 0) | (5 A 2) |
| (6 9 1) | (6 B 3) |
| (7 A 2) | (7 0 4) |
| (8 B 3) | (8 1 5) |
| (9 0 4) | (9 2 6) |
| (A 1 5) | (A 3 7) |
| (B 2 6) | (B 4 8) |

**Tab. 16:** Paletas das classes de conjuntos da transição 2 entre A e B

| Tr1BA1 = 02358 | |
|---|---|
| Tn | TnI |
| (0 2 3 5 8) | (0 4 7 9 A) |
| (1 3 4 6 9) | (1 5 8 A B) |
| (2 4 5 7 A) | (2 6 9 B 0) |
| (3 5 6 8 B) | (3 7 A 0 1) |
| (4 6 7 9 0) | (4 8 B 1 2) |
| (5 7 8 A 1) | (5 9 0 2 3) |
| (6 8 9 B 2) | (6 A 1 3 4) |
| (7 9 A 0 3) | (7 B 2 4 5) |
| (8 A B 1 4) | (8 0 3 5 6) |
| (9 B 0 2 5) | (9 1 4 6 7) |
| (A 0 1 3 6) | (A 2 5 7 8) |
| (B 1 2 4 7) | (B 3 6 8 9) |

| Tr1BA2 = 012 |
|---|
| Tn/TnI |
| (0 1 2) |
| (1 2 3) |
| (2 3 4) |
| (3 4 5) |
| (4 5 6) |
| (5 6 7) |
| (6 7 8) |
| (7 8 9) |
| (8 9 A) |
| (9 A B) |
| (A B 0) |
| (B 0 1) |

**Tab. 17:** Paletas das classes de conjuntos da transição 1 entre B e A

| Tr2BA1 = 0258 | | Tr2BA2 = 0145 |
|---|---|---|
| Tn | TnI | Tn |
| (0 2 5 8) | (0 4 7 A) | (0 1 4 5) |
| (1 3 6 9) | (1 5 8 B) | (1 2 5 6) |
| (2 4 7 A) | (2 6 9 0) | (2 3 6 7) |
| (3 5 8 B) | (3 7 A 1) | (3 4 7 8) |
| (4 6 9 0) | (4 8 B 2) | (4 5 8 9) |
| (5 7 A 1) | (5 9 0 3) | (5 6 9 A) |
| (6 8 B 2) | (6 A 1 4) | (6 7 A B) |
| (7 9 0 3) | (7 B 2 5) | (7 8 B 0) |
| (8 A 1 4) | (8 0 3 6) | (8 9 0 1) |
| (9 B 2 5) | (9 1 4 7) | (9 A 1 2) |
| (A 0 3 6) | (A 2 5 8) | (A B 2 3) |
| (B 1 4 7) | (B 3 6 9) | (B 0 3 4) |

**Tab. 18:** Paletas das classes de conjuntos da transição 2 entre B e A

| Tr3BA1 = 013457 | | Tr3BA2 = 015 | |
|---|---|---|---|
| Tn | TnI | Tn | TnI |
| (0 1 3 4 5 7) | (0 5 7 8 9 B) | (0 1 5) | (0 7 B) |
| (1 2 4 5 6 8) | (1 6 8 9 A 0) | (1 2 6) | (1 8 0) |
| (2 3 5 6 7 9) | (2 7 9 A B 1) | (2 3 7) | (2 9 1) |
| (3 4 6 7 8 A) | (3 8 A B 0 2) | (3 4 8) | (3 A 2) |
| (4 5 7 8 9 B) | (4 9 B 0 1 3) | (4 5 9) | (4 B 3) |
| (5 6 8 9 A 0) | (5 A 0 1 2 4) | (5 6 A) | (5 0 4) |
| (6 7 9 A B 1) | (6 B 1 2 3 5) | (6 7 B) | (6 1 5) |
| (7 8 A B 0 2) | (7 0 2 3 4 6) | (7 8 0) | (7 2 6) |
| (8 9 B 0 1 3) | (8 1 3 4 5 7) | (8 9 1) | (8 3 7) |
| (9 A 0 1 2 4) | (9 2 4 5 6 8) | (9 A 2) | (9 4 8) |
| (A B 1 2 3 5) | (A 3 5 6 7 9) | (A B 3) | (A 5 9) |
| (B 0 2 3 4 6) | (B 4 6 7 8 A) | (B 0 4) | (1 1 6 A) |

**Tab. 19:** Paletas das classes de conjuntos da transição 3 entre B e A

| Compasso | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Mão direita - A1 | 0158 | 1269 | 56A1 | 8914 |
| Mão esquerda - A2 | ------- | 0147 | 1258 | 458B |
| **Compasso** | **5** | **6** | **7** | **8** |
| Mão direita - A2 | 8147 | 9258 | 058B | 38B2 |
| Mão esquerda - A2 | 78B2 | (A1)7803 | 8914 | 0158 |
| **Compasso** | **9** | **10** | **11** | **12** |
| Mão direita - A1 | 348B | 4590 | 8914 | B047 |
| Mão direita - A3 | 014568 | | 125679 | |
| Mão esquerda - A1 | 348B | (A2)347A | 458B | 78B2 |

**Tab. 20:** Seção A; 3/4 | Continua...

| Compasso | 13 | 14 | 15 | 16 |
|---|---|---|---|---|
| Mão direita - A2 | B47A | 058B | 38B2 | 6B25 |
| Mão direita - A3 | 4589AB | | 569AB0 | |
| Mão esquerda - A2 | AB25 | (A1)AB36 | B047 | 348B |

**Tab. 20:** Seção A; 3/4

| Compasso | 17 | 18 | 19 | 20 | |
|---|---|---|---|---|---|
| Mão direita - Tr1AB2 | 0235 | 2457 | 3568 | 578T | |
| Mão esquerda - A1 | 673B | (Tr1AB3)0127 | 1238 | 2349 | |
| **Compasso** | **21** | **22** | **23** | **24** | **25** |
| Mão direita - Tr1AB1 | 789B3 | 89A04 | 9AB15 | B0137 | 3457B |
| Mão direita - Tr1AB1(1) | - | 80467 | 91578 | B379A | 37B12 |
| Mão esquerda - Tr1AB3 | 7892 | (Tr1AB2)2457 | 4679 | 578A | 79A0 |
| Mão esquerda - Tr1AB3 | - | AB05 | B016 | 0127 | 5670 |

**Tab. 21:** Transição 1 de A para B; 2/4

| Compasso | 26 | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|
| Mão direita - Tr2AB2 | 037 | (Tr2AB2 1)380 | (Tr2AB2)7A2 | (Tr2AB2 1)A37 | (Tr2AB2)259 |
| Mão direita - Tr2AB1(1) | A4689 | (Tr2AB1)4568A | (Tr2AB1)60245 | (Tr2AB1)89A02 | (Tr2AB1)93578 |
| Mão esquerda - Tr2AB2 1 | 059 | (Tr2AB2)580 | (Tr2AB2 1) 926 | (Tr2AB2)259 | (Tr2AB2 1) 6B3 |
| Mão esquerda - Tr2AB 1 | 23468 | (Tr2AB1)39B12 | (Tr2AB1)4568T | (Tr2AB1)60245 | (Tr2AB1)89A02 |

**Tab. 22:** Transição 2 de A para B; 5/8

| Compasso | 31 | 32 | 33 | 34 |
|---|---|---|---|---|
| Mão direita - B1 | 0246 | 68A0 | 9B13 | 0246 |
| Mão direita - B2 | 237 | (B2)726 | (B2)AB3 | (B2)180 |
| Mão esquerda - B2 | 4B3 | (B2)891 | (B2)B67 | (B2)237 |
| Mão esquerda - B3 | 6789 | 9AB0 | 0123 | 3456 |
| **Compasso** | **35** | **36** | **37** | **38** |
| Mão direita - B3 | 1234 | 89AB | 9AB0 | 4567 |
| Mão direita - B2 | 891 | (B2I)948 | (B2)459 | (B2)504 |
| Mão esquerda - B1 | 0246 | 1357 | 8A02 | 9B13 |
| **Compasso** | **39** | **40** | **41** | **42** |
| Mão direita - B1 | 468A | (B3)4567 | AB01 | (B1)A024 |
| Mão direita - B2 | 504 | (B2)67B | (B2I)073 | (B2)B04 |
| Mão esquerda - B2 | 67B | (B2)837 | (B2)237 | (B2)07B |
| Mão esquerda - B3 | 789A | (B1)A24 | 468A | (B3)1234 |

**Tab. 23:** Seção B; 4/4

| Compasso | 43 | 44 | 45 |
|---|---|---|---|
| Mão direita - Tr1BA2 | 012 | 456 | 89A |
| Mão esquerda - Tr1BA1 | 3568B | (Tr1BA1(1))37A01 | (Tr1BA1)B1247 |

**Tab. 24:** Transição 1 de B para A; 5/8

| Compasso | 46 | 47 | 48 | 49 |
|---|---|---|---|---|
| Mão direita - Tr2BA2 | 0145 | 1256 | 4589 | 569A |
| Mão esquerda - Tr2BA1 | 68B2 | 7903 | A036 | B147 |
| Mão esquerda - Tr2BA1 | - | BA14 | 7B25 | A258 |

**Tab. 25:** Transição 2 de B para A; 5/8

| Compasso | 50 | 51 | 52 | 53 |
|---|---|---|---|---|
| Mão direita - Tr3BA1 | 013457 | 124568 | 34678A | 45789B |
| Mão direita - Tr3BA1 | - | 05789B | 5A0124 | 702346 |
| Mão esquerda - Tr2BA1 | B369 | (Tr3BA2)3A2 | 837 | A59 |
| Mão esquerda - Tr3BA2 | 9A2 | AB3 | 015 | 126 |
| **Compasso** | **54** | **55** | **56** | |
| Mão direita – Tr3BA1 | 5689A0 | 78AB02 | - | |
| Mão direita – Tr3BA1 | 813457 | 924568 | B4678A | |
| Mão esquerda – Tr3BA2l | B6A | 07B | 291 | |
| Mão esquerda – Tr3BA2 | 237 | 459 | - | |

**Tab. 26:** Transição 3 de B para A; 2/4

| Compasso | 57 | 58 | 59 | 60 |
|---|---|---|---|---|
| Mão direita - A3 | 014568 | 569AB1 | (A3)157890 | 802347 |
| Mão esquerda - A1 | 1269 | 67B2 | 56A1 | 9A25 |
| Mão esquerda - A2l | 4903 | 8147 | (A2)78B2 | 0147 |
| **Compasso** | **61** | **62** | **63** | **64** |
| Mão direita - A1 | 4590 | 56A1 | 9A25 | 0158 |
| Mão direita - A2 | 5690 | 67A1 | AB25 | 1258 |
| Mão esquerda - A2 | 9258 | A369 | 27A1 | 5A14 |
| Mão esquerda - A3 | 04678B | 157890 | 59B014 | 802347 |
| **Compasso** | **65** | **66** | **67** | **68** |
| Mão direita - A3 | 014568 | (A3)59014 | (A2)8147 | B47A |
| Mão esquerda - A1 | 1269 | 8914 | (A2)8903 | B036 |

**Tab. 27:** Seção A (reprise); 3/4

## Conclusões

Examinamos neste trabalho a implementação de um sistema composicional semiaberto cujas estruturas léxicas e sintáticas foram coordenadas pelo grau de similaridade entre classes de conjuntos de classes de notas. Vimos que há vários modelos de cálculo para a determinação da similaridade, todos válidos e com diferentes graus de perspectiva. Tomando como ponto de partida o sistema composicional implementado, demonstramos a possibilidade de gerar duas obras completamente distintas pela aplicação de dois planejamentos composicionais diferenciados: o primeiro realizado manualmente e o segundo assistido por computador, no tocante ao parâmetro altura (sendo os demais parâmetros livremente manipulados pelo compositor).

No nível das obras, a diferenciação é ainda mais marcante, pois é aí, nas microestruturas locais, que cada compositor imprime vigorosamente sua voz estética. Um exame comparativo na superfície das partituras (disponíveis nos Apêndices I e II deste artigo) nos revela, em um

primeiro contato, diferenças significativas com relação à textura e ao timbre. Com relação à textura, uma análise quantitativa das partições texturais[21] dos treze compassos iniciais de ambas as obras (Tab. 28) nos revela que a primeira obra já aponta para uma tendência às estruturas homorrítmicas, ou seja, uma maior convergência à aglomeração, enquanto a segunda se concentra na polifonia, ou seja, um movimento em direção à dispersão. Em seções posteriores da segunda obra (compassos 22 – 30), observa-se o emprego de partições texturais 2.1 e 2.2, revelando uma posterior tendência à aglomeração. Quanto ao timbre, as obras diferem drasticamente: a primeira é para quarteto de cordas, a segunda para piano.

As estruturas rítmicas foram livremente empregadas pelos compositores, uma vez que o sistema não prevê diretrizes nesse campo paramétrico. Na primeira obra, o léxico de figuras que compõem as estruturas rítmicas é quantitativamente menor (10) em comparação com a segunda obra (13). Qualitativamente, o léxico da segunda obra proporciona uma maior gradação rítmica: enquanto a semínima da primeira obra apresenta cinco subdivisões, na segunda obra, esse valor sobe para nove. O resultado é uma estrutura rítmica mais complexa na segunda obra. O léxico rítmico de ambas as obras é mostrado na Fig. 12.

| Compasso | Primeira obra | Segunda obra |
|---|---|---|
| | Partições | |
| 1 | $2 \rightarrow 2^2$ | 1 |
| 2 | $2 \rightarrow 2^2$ | $1^2$ |
| 3 | $2.1^2 \rightarrow 2^2$ | $1^2$ |
| 4 | $2.1^2 \rightarrow 2^2$ | $1^2$ |
| 5 | 2.1 | $1^2$ |
| 6 | 3 | $1^2$ |
| 7 | 3 | 2.1 |
| 8 | 3 | $1^2 \rightarrow 2$ |
| 9 | 3.1 | $1^2 \rightarrow 3.1 \rightarrow 3^2.1$ |
| 10 | 2.1 | $1^2 \rightarrow 3.1 \rightarrow 3^2.1$ |
| 11 | 4 | $1^2 \rightarrow 3.1 \rightarrow 3.2.1$ |
| 12 | 4 | $1^2 \rightarrow 3.1 \rightarrow 3.2$ |
| 13 | $2^2$ | $1^2$ |

**Tab. 28:** Partições texturais nos treze primeiros compassos das duas obras, revelando que no início há uma maior tendência homorrítmica na primeira e uma maior tendência à polifonia na segunda.

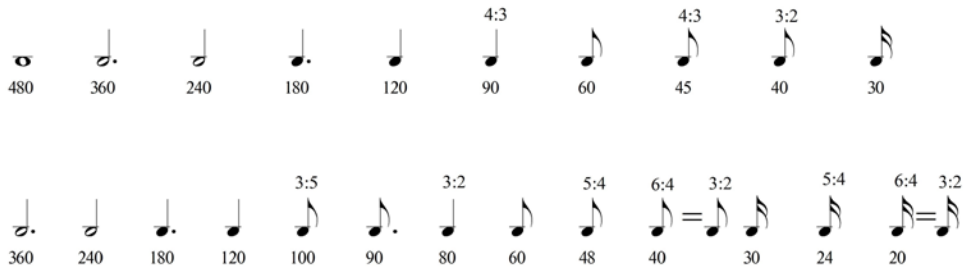



**Fig. 12:** Léxico rítmico para a primeira obra (linha superior) e segunda obra (linha inferior)

[21] Uma revisão bibliográfica e introdução básica aos conceitos fundamentais da Análise Particional (GENTIL-NUNES, 2009) foge aos objetivos do presente trabalho. O leitor pode consultar a bibliografia amplamente difundida desde 2009 sobre essa teoria já consolidada em nível nacional e internacional.

Dois outros fatores revelam características distintas das duas obras: estruturas imitativas e macro-harmonia. Na segunda obra, a estrutura canônica, logo no início, nos permite uma associação com as *Invenções a duas Vozes* de J. S. Bach, mas transpostas para um contexto atonal livre. Tomando como janela de observação os três compassos iniciais, verifica-se que a macro-harmonia é formada por nove classes de notas, cujo complemento cromático é um acorde aumentado, ou seja, faltam as classes de notas Ré♯ – Sol – Si. Por sua vez, a primeira obra apresenta poucas entradas imitativas (exceto entre os compassos 22—24). Em termos harmônicos, no entanto, detecta-se, entre os compassos 47–55, um ciclo claramente tonal de progressões por quartas. Contudo, paradoxalmente, a mesma janela inicial de observação tomada para a primeira obra (três compassos iniciais) revela na segunda obra uma macro-harmonia mais densa, formada por um agregado cromático. A atmosfera mais tradicional da segunda obra, nos momentos iniciais, mesmo com a presença de um agregado cromático, parece se estabelecer por questões texturais, com base nas partições texturais supramencionadas.

Com base na metodologia empregada nesta pesquisa e nos resultados obtidos, evidencia-se o potencial artístico e pedagógico da formalização exclusiva de relações em um primeiro estágio pré-composicional[22], permitindo o estabelecimento de uma estrutura (o sistema composicional) que pode ser revisitada diversas vezes por compositores diferentes, produzindo resultados coerentes e inusitados.

Este artigo é o primeiro de uma trilogia que se propõe a examinar cada um dos tipos de sistema (semiaberto, aberto e retroalimentado). Para os sistemas definidos, pelo menos duas novas obras serão planejadas e compostas, com a finalidade de avaliar suas características intrínsecas. Posteriormente, submeteremos cada um desses sistemas à apreciação de outros grupos de compositores (iniciantes e profissionais) para que produzam planejamentos e obras e nos relatem suas experiências em cada fase do processo. Estamos certos de que a contribuição desses experimentos e reflexões para o ensino e a pesquisa em composição serão benéficas.

## Referências

BELL, Shawn. NetWorks 1: Could-be Music. *Bandcamp*, 30 sept. 2014. Disponível em: https://shawnbell.bandcamp.com/album/networks-1-could-be-music. Acesso em: 14 jul. 2020.

BELL, Shawn; GABORA, Liane. A music-generating system inspired by the science of complex adaptive systems. *In*: INTERNATIONAL WORKSHOP ON MUSICAL META-CREATION, 4., 2016, Palo Alto. *Proceedings* [...]. Palo Alto: Association for the Advancement of Artificial Intelligence (AIII) Press, 2016.

BERTALANFFY, Ludwig von. *General System Theory*: foundation, development, application. New York: George Braziller, 1968.

BIDLACK, Richard. Chaotic systems as simple (but complex) compositional algorithms. *Computer Music Journal*, n. 16, p. 33-47, 1992.

[22] O que denominamos estágio pré-composicional em nossa pesquisa envolve duas fases. Na primeira fase, elabora-se um sistema composicional, que consiste em relações entre objetos genéricos (tais relações podem ser encapsuladas em um programa computacional, ser descritas de forma declarativa, ou ser expressas sob a forma de gráficos, tabelas e diagramas). Normalmente um sistema composicional trata de um número limitado de parâmetros musicais. Na segunda fase, se dá o planejamento composicional, no qual os objetos são especificados e os parâmetros não declarados no sistema são inseridos. De posse do planejamento composicional, inicia-se a fase composicional propriamente dita, uma atividade prática cujo produto final pode ser uma partitura, um arquivo de áudio (no caso da composição eletroacústica) ou partitura + áudio (nas produções mistas).

DANTAS, Weskley; PITOMBEIRA, Liduino. Definição de sintaxe composicional a partir do estabelecimento de hierarquias entre sonoridades tricordais relacionadas por similaridade. *Música Hodie*, Goiânia, v. 12, n. 2, p. 109-119, 2012.

DI SCIPIO, Agostino. Approaches to computer music as different models of compositional design. *Perspectives of New Music*, v. 33, n. 1/2, p. 360-402, Winter/Summer 1995.

DIDKOVSKY, Nick. Metamusic/Metatext: the blurry boundaries around distributed compositional systems. *Leonardo Music Journal*, v. 7, p. 73-76, 1997.

DURAND, Daniel. *La systémique*. Paris: Presses Universitaires de France, 1996.

FORTE, Allen. *The structure of atonal music.* New Haven: Yale University Press, 1973.

GABORA, Liane. Honing theory: a complex systems framework for creativity. *Nonlinear Dynamics, Psychology, and Life Sciences*, v. 21, n. 1, p. 35-88, 2017.

GENTIL-NUNES, Pauxy. *Análise particional*: uma mediação entre análise textural e a teoria das partições. 2009. 371 f. Tese (Doutorado em Música) – Centro de Letras e Artes, Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2009.

GEORGESCU, C.; GEORGESCU, M. A system approach to music. *Interface*, v. 19, n. 1, p. 15-52, 1990.

HONINGH, Aline; WEYDE, Tilmann; CONKLIN, Darrel. Sequential Association Rules in Atonal Music. *In*: MATHEMATICS AND COMPUTATION IN MUSIC - INTERNATIONAL CONFERENCE, 2., 2009, New Haven. *Proceedings* [...]. New Haven: John Clough Memorial Conference, june 19-22, 2009. p. 130-138.

HOVINEN, Erkki; TENKANEN, Atte. Bird's-Eye View of the Musical Surface: Methods for Systematic Pitch-Class Set Analysis. *Music Analysis*, v. 26, n. 1 e 2, p. 159-214, 2007.

ISAACSON, Eric J. Similarity of interval-class content between pitch-class sets: the IcVSIM relation. *Journal of Music Theory*, v. 34, n. 1, p. 1-28, Spring 1990.

KLIR, George. *Facets of Systems Science*. New York: Plenum, 1991.

KUUSI, Tuire. *Set-Class and Chord:* examining connection between theoretical resemblance and perceived closeness. Tese (Doutorado em Música) – Sibelius Academy, Helsinki, 2001.

LERDAHL, Fred. *Generative Processes in Music*: the psychology of performance, improvisation and composition. Oxford: Claredon Press, 1988.

LEWIN, David. Special cases of the interval function between pitch-class sets X and Y. *Journal of Music Theory*, n. 45, p. 1-29, 2001.

LIMA, Flávio. *Desenvolvimento de sistemas composicionais a partir da intertextualidade*. 2011. 239 f. Dissertação (Mestrado em Música) – UFPB, João Pessoa, 2011.

MORRIS, Robert. D. *Composition with Pitch-classes*: a theory of compositional design. New Haven: Yale University Press, 1987.

NAVARRO, Maria; CORCHADO, Juan Manuel; DEMAZEAU, Yves. 2020. A Musical Composition Application Based on a Multiagent System to Assist Novel Composers. *In*: INTERNATIONAL CONFERENCE OF COMPUTATIONAL CREATIVITY, 5., 2014, Ljubljana. *Proceedings* [...]. Ljubljana, 9-13 jun. 2014.

OCKELFORD, Adam. *Repetition in music*: theoretical and metatheoretical perspectives. London: Routledge, 2017.

PITOMBEIRA, Liduino. A produção de teoria composicional no Brasil. *In*: NOGUEIRA, Ilza; BORÉM, Fausto (org.). *O pensamento musical criativo*: teoria, análise e os desafios interpretativos da atualidade. Salvador: UFBA, 2015. p. 61-89. (Série Congressos da TEMA, v. 1).

PITOMBEIRA, Liduino. A systemic model for Debussy's Prelude No.1. *MusMat - Brazilian Journal of Music and Mathematics*, v. 2, n. 2, p. 37-57, 2018.

PITOMBEIRA, Liduino. Compositional systems: overview and applications. *MusMat - Brazilian Journal of Music and Mathematics*, v. 4, n. 1, p. 39-62, 2020.

PRESSING, Jeff. Nonlinear maps as generators of musical design. *Computer Music Journal*, v. 12, n. 2, p. 35-46, Summer 1988.

QUINN, Ian. Listening to similarity relations. *Perspectives of New Music*, v. 39, n. 2: p. 108-58, 2001.

QUINN, Ian. General Equal-Tempered Harmony (Introduction and Part I). *Perspectives of New Music*, v. 44, n. 2, p. 114-58, 2006.

QUINN, Ian. General Equal-Tempered Harmony (Parts II and III). *Perspectives of New Music*, v. 45, n. 1, p. 4-63, 2007.

SAMPLASKI, Art. Mapping the geometries of pitch-class set similarity measures via multidimensional scaling. *MTO*, v. 11, n. 2, 2005. Disponível em: https://mtosmt.org/issues/mto.05.11.2/mto.05.11.2.samplaski.html. Acesso em: 5 jul. 2020.

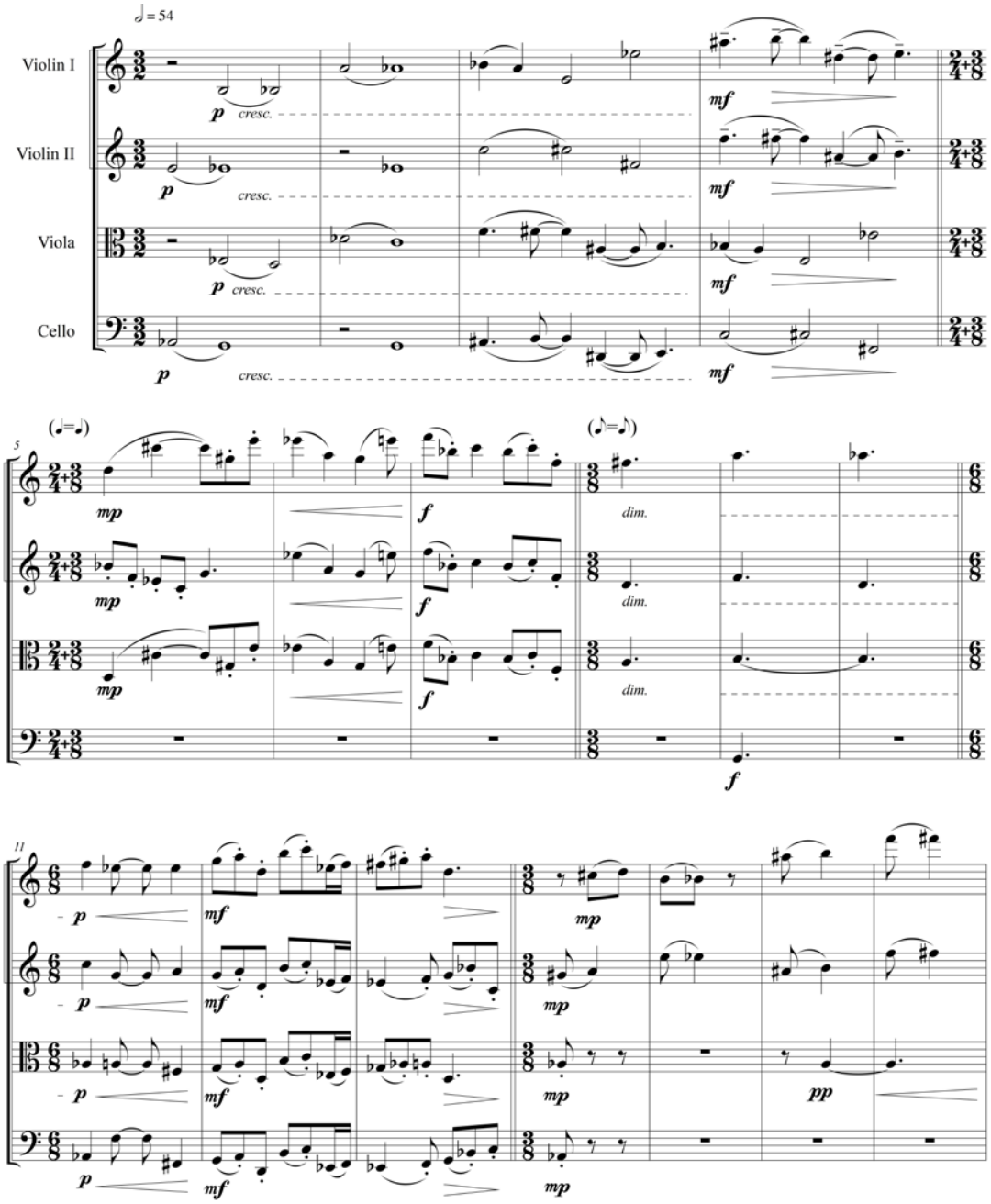
SCHUIJER, Michiel. *Analyzing atonal music*: pitch-class set theory and its contexts. Rochester: University of Rochester Press, 2008.

SCOTT, Damon; ISAACSON, Eric J. The interval angle: a similarity measure for pitch-class sets. *Perspectives of New Music*, v. 36, n. 2, p. 107-142, 1998.

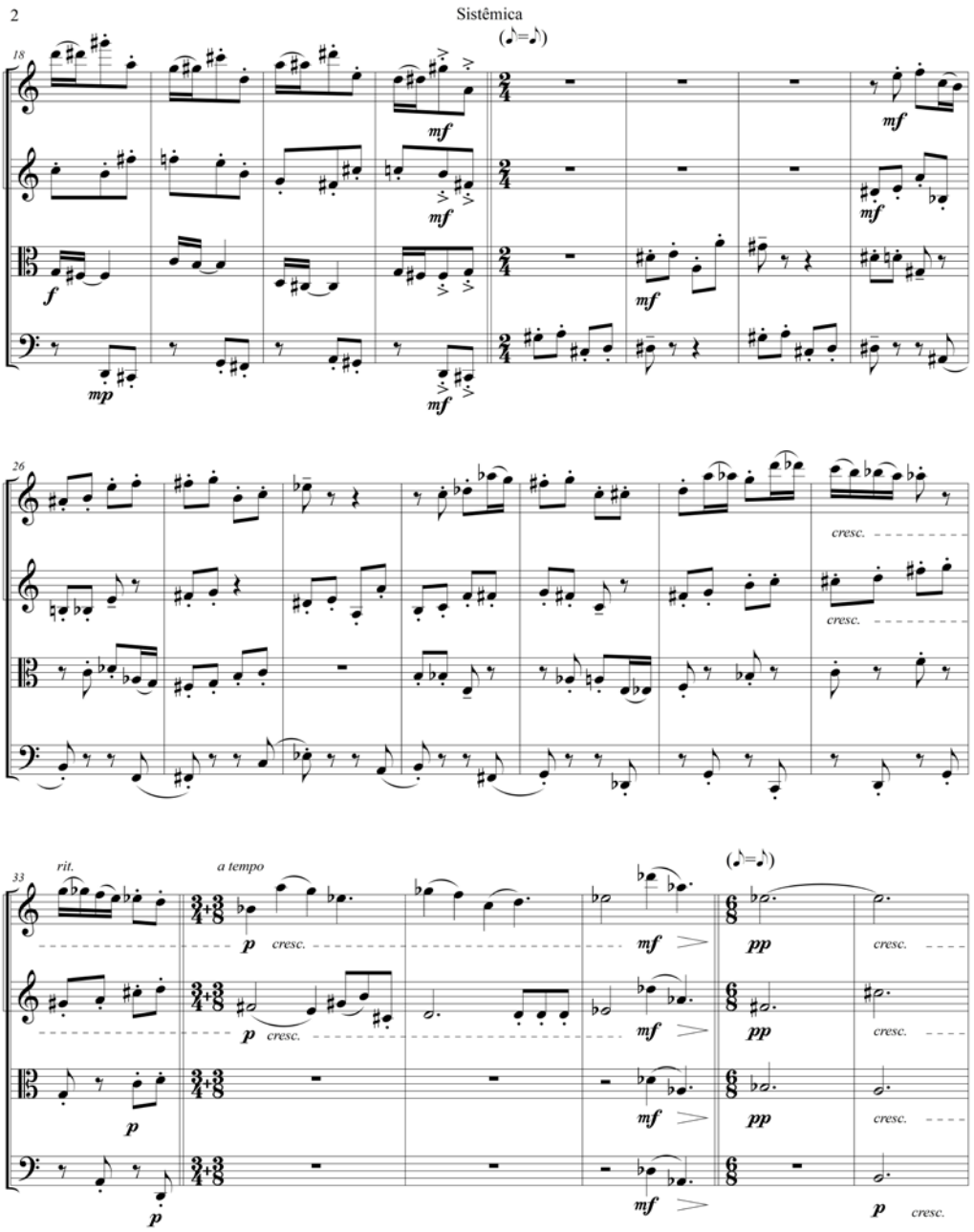
STRAUS, Joseph. *Remaking the past:* musical modernism and the influence of the tonal tradition. Cambridge: Harvard University Press, 1990.

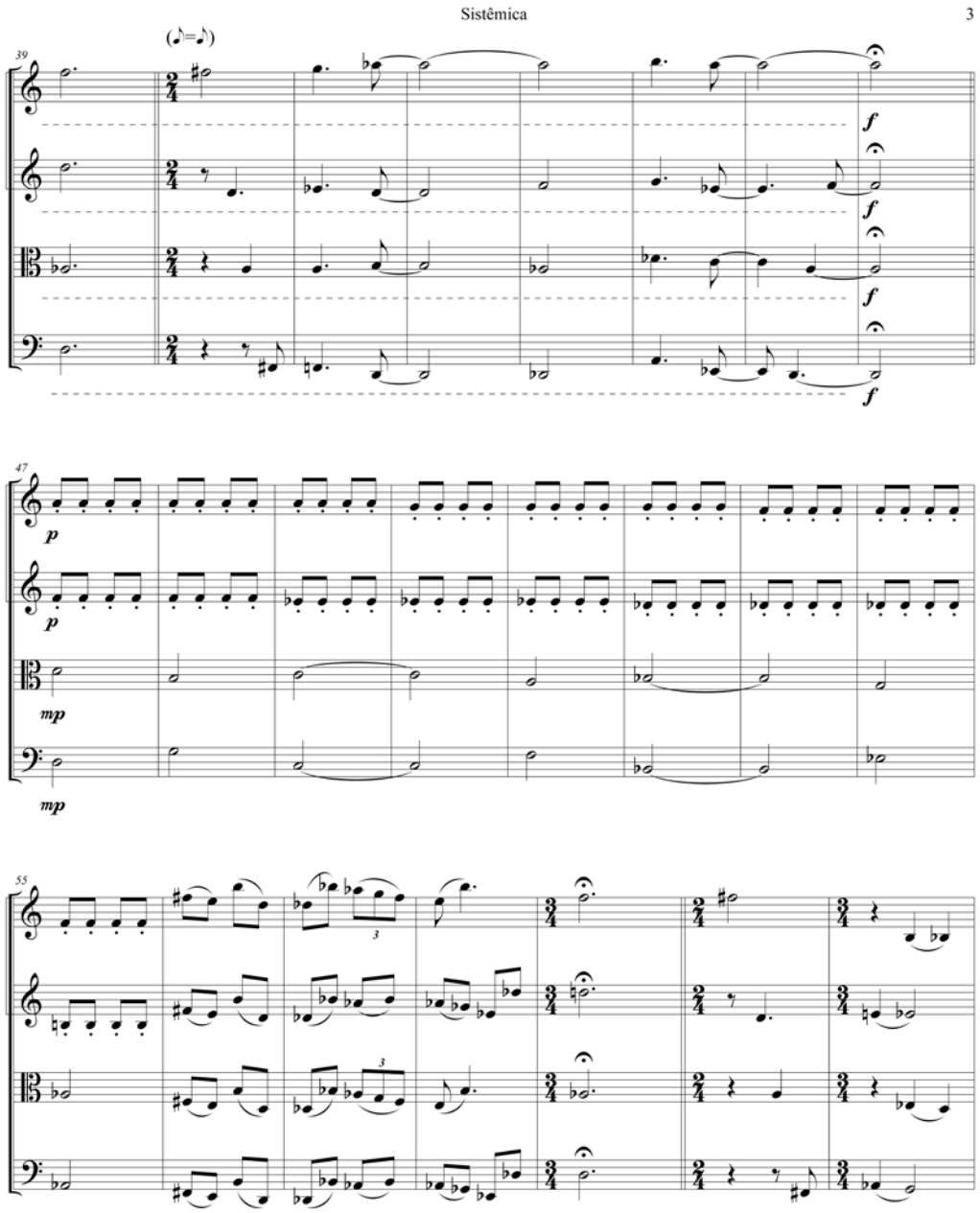
TYMOCZKO, Dmitri. Set-class similarity, voice leading, and the fourier transform. *Journal of Music Theory*, v. 52, n. 2, 2008.

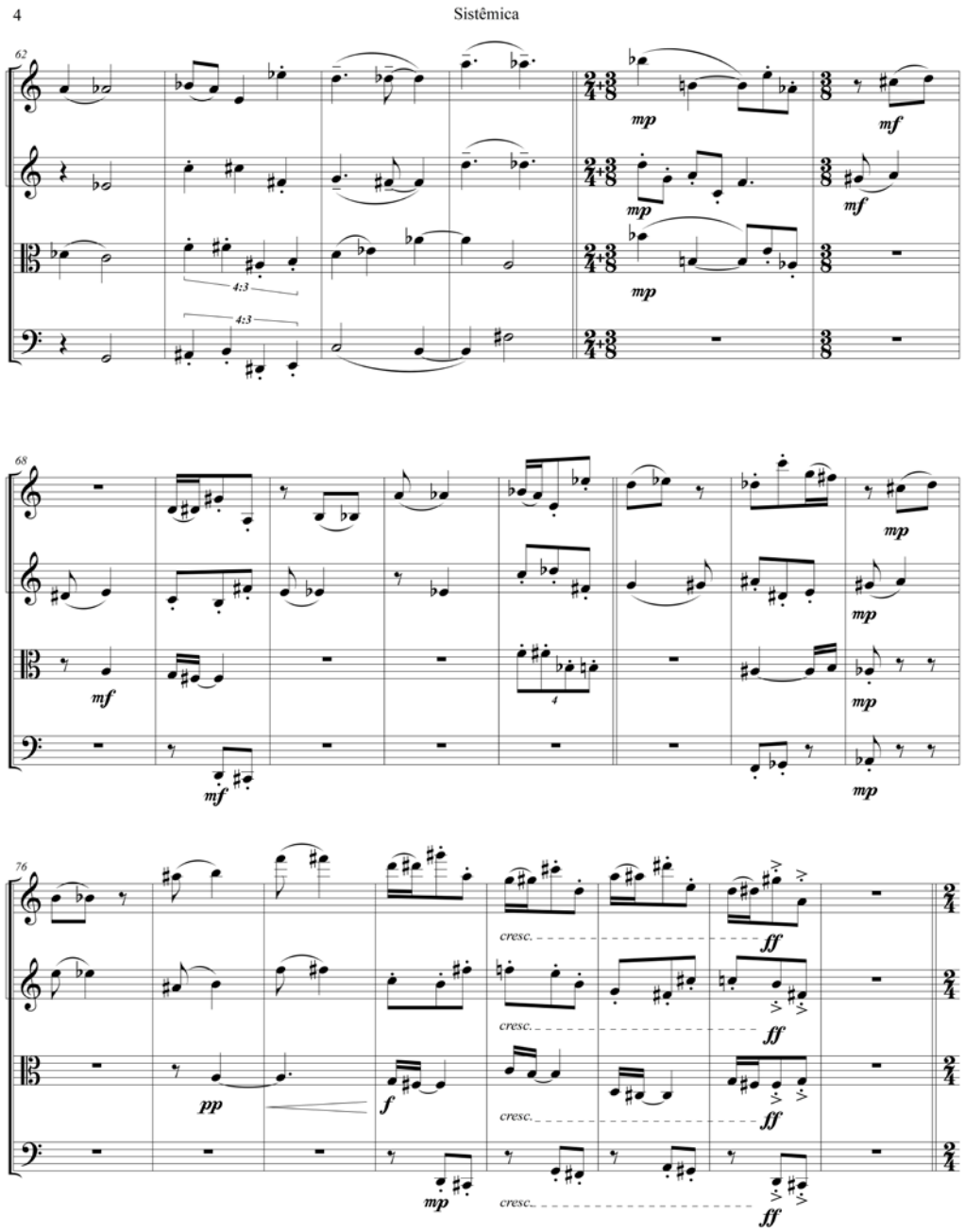
**Apêndice I:** *Semiaberto*, primeiro movimento de *Sistêmica*, de Liduino Pitombeira[23]

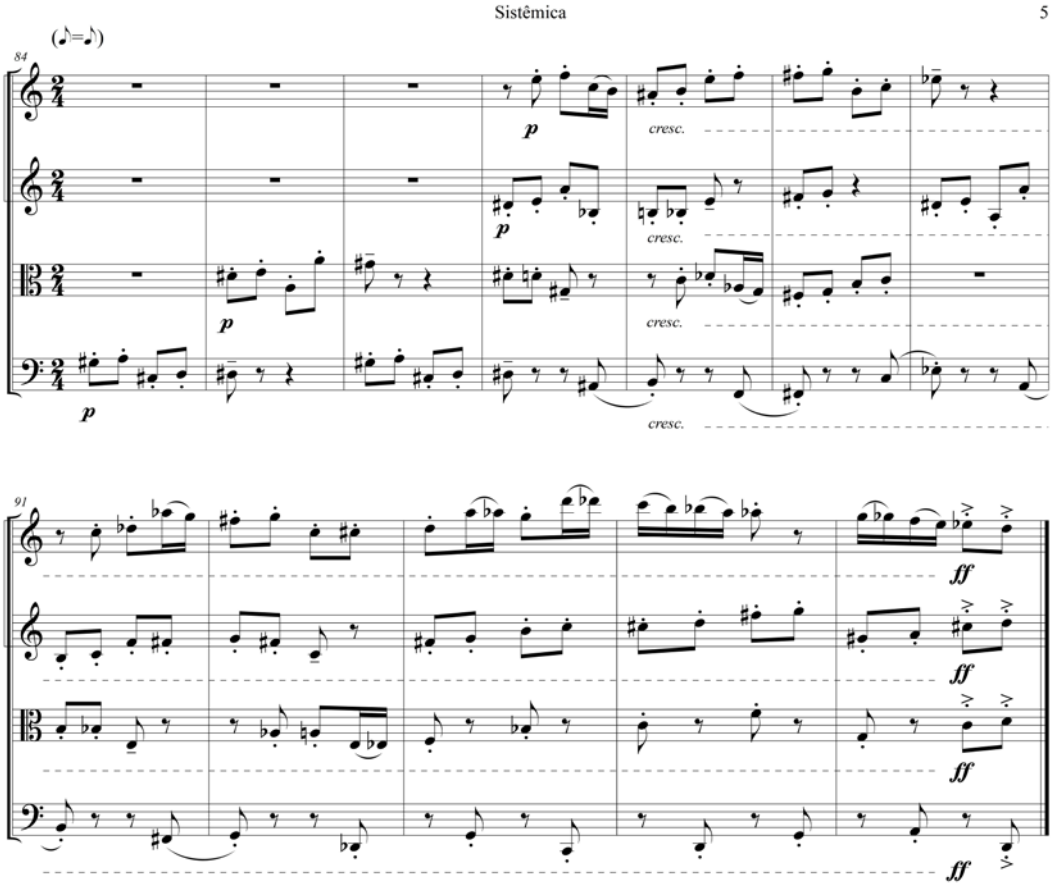



[23] *Link* para vídeo-partitura: https://www.youtube.com/watch?v=HsQUZzngz9M

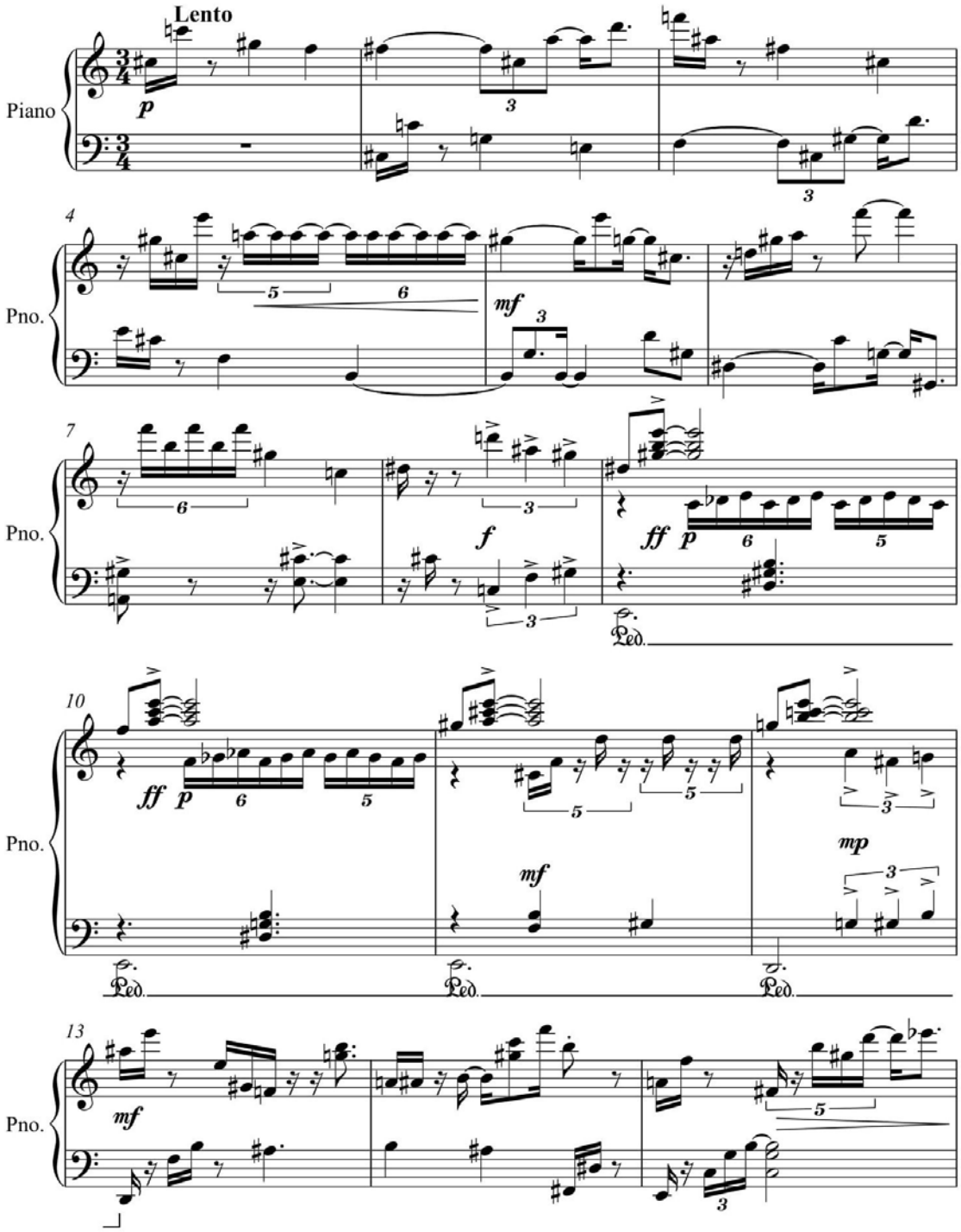

2
Sistêmica
18
mf
mp
26
cresc.
33
rit.
a tempo
p
cresc.
pp


Sistêmica
3
(♪=♪)
39
f
47
p
mp
55
3

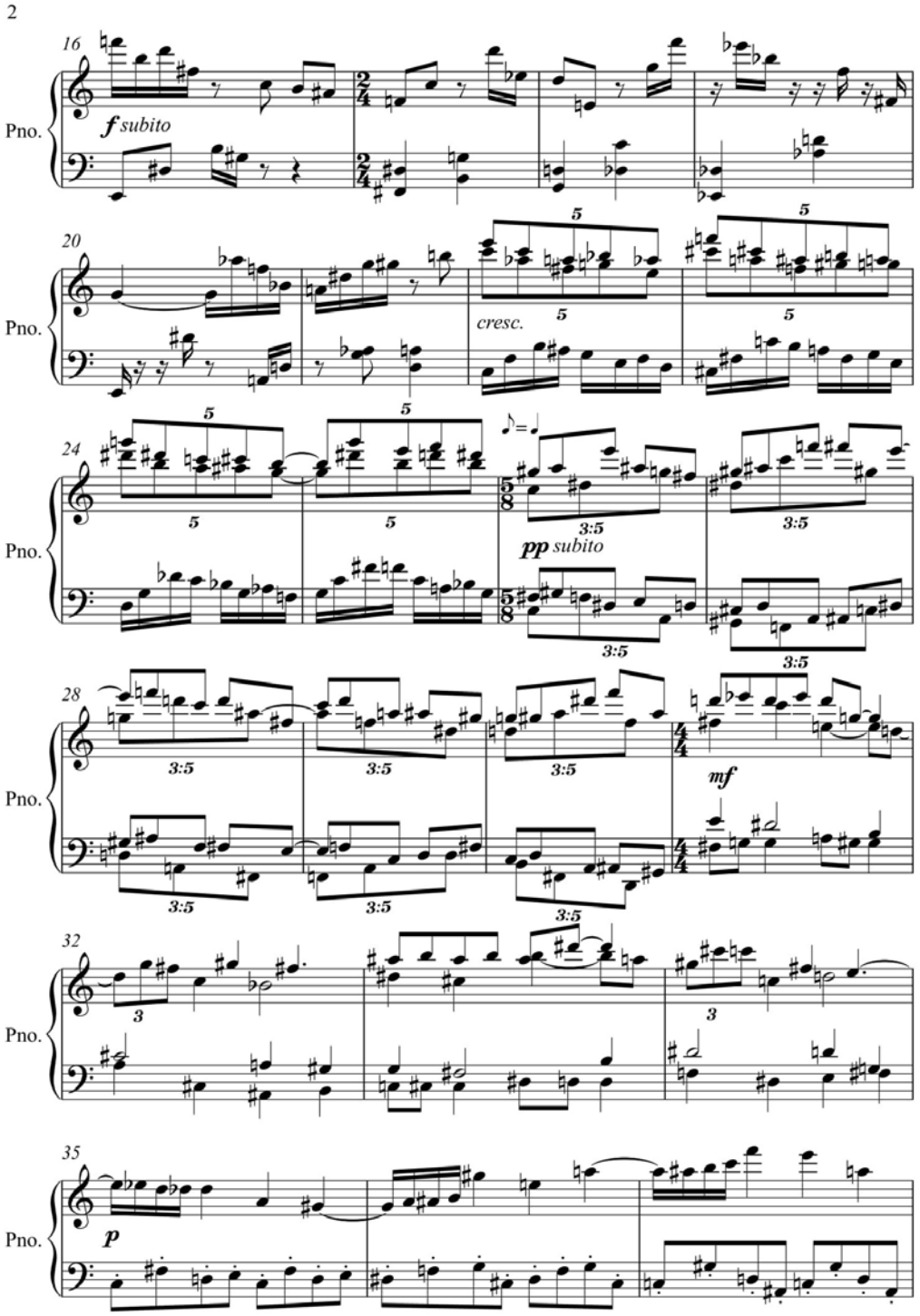

4
Sistêmica
62
mp
mf
4:3
68
mf
4
76
pp
f
cresc.
ff

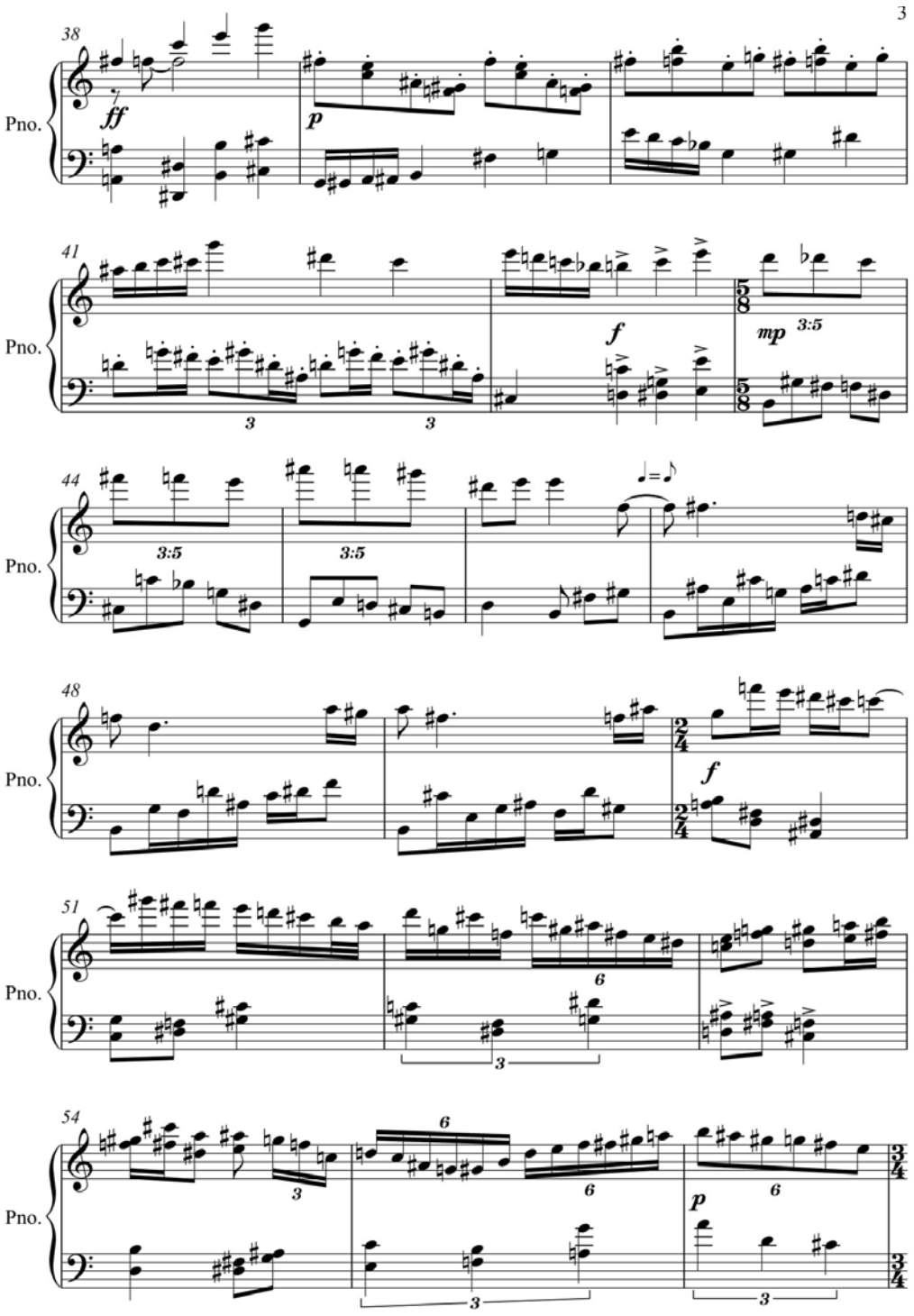

Sistêmica
5
84
91
p
cresc.
ff

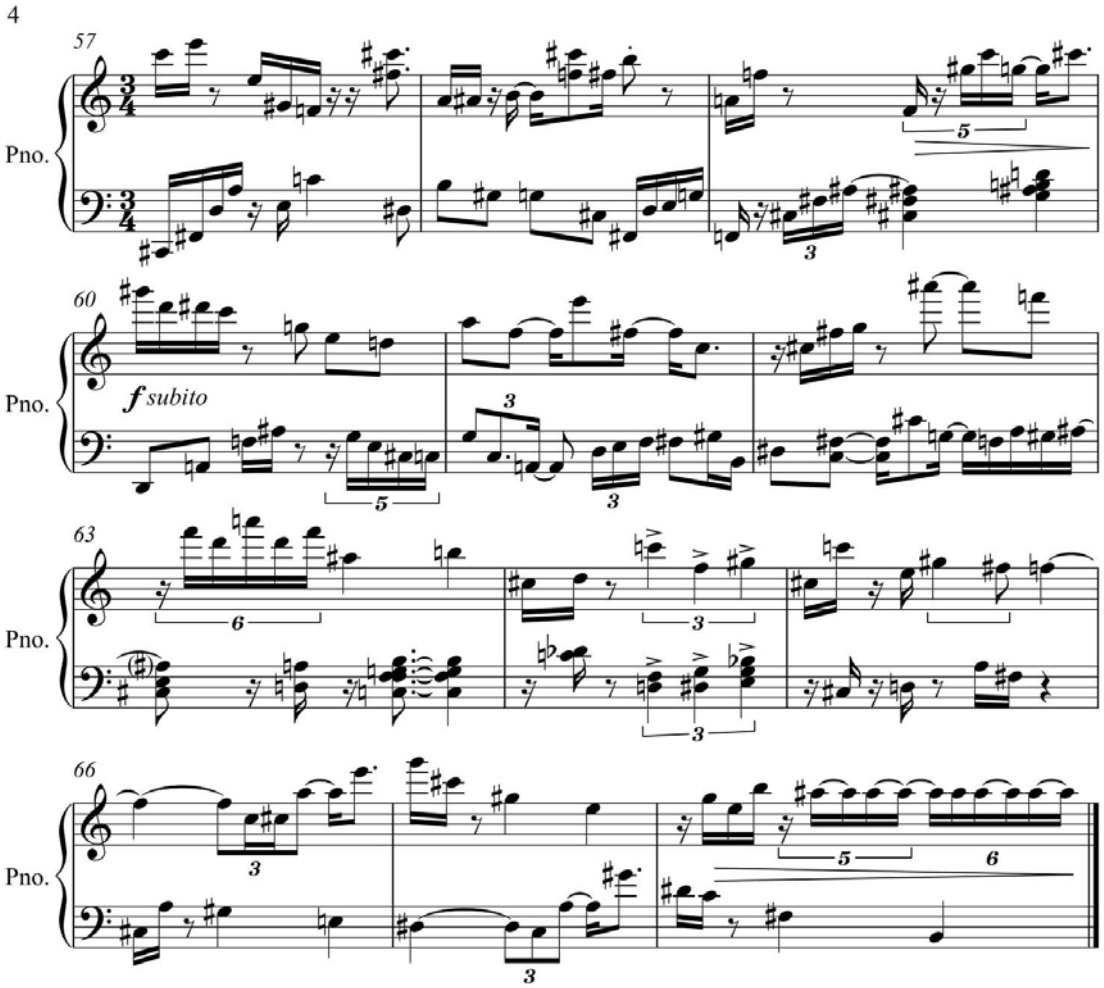
**Apêndice:** *SIM(ulation)*, de Pedro Proença[24]




[24] *Link* para a vídeo-partitura: https://www.youtube.com/watch?v=QYBXxxuRDDE


2
16
Pno.
f subito
20
Pno.
cresc.
24
Pno.
pp subito
28
Pno.
mf
32
Pno.
35
Pno.
p


3
38
Pno.
ff
p
41
Pno.
f
mp
3:5
44
Pno.
3:5
3:5
48
Pno.
f
51
Pno.
54
Pno.
p




**Vinicius Braga** é formado em Engenharia Elétrica pela Universidade Federal Fluminense (UFF) e atualmente cursa Música – Composição na Universidade Federal do Estado do Rio de Janeiro (UFRJ), onde também estudou Filosofia. Trabalha com composição de trilhas sonoras para *videogames* há mais de oito anos, incluindo trabalhos aclamados pela crítica, como *A Lenda do Herói*. Como compositor, transita do tonal ao pós-tonal e utiliza o conceito de áudio dinâmico na implementação de suas obras. Multi-instrumentista, hoje faz parte da equipe de pesquisa do grupo MusMat da UFRJ, onde contribui tanto em projetos de análise da música popular brasileira, orientado pelo Prof. Carlos Almada, quanto no desenvolvimento de sistemas composicionais, orientado pelo Prof. Liduino Pitombeira, como bolsista de iniciação científica. viniciusramosbraga@gmail.com

**João Travassos Penchel** (Brasil, 1974) formou-se em Violão pela Escola de Música da UFRJ em 2016. Guitarrista de formação popular autodidata, integrou bandas de blues, rock e reggae nos idos da década de 1990. Concluindo neste ano de 2020 a segunda graduação iniciada em 2016, estuda composição com o Prof. Liduino Pitombeira. Possui obras de caráter pós-tonal dentro do âmbito de formação da universidade, com influências fundamentais no jazz fusion norte-americano e na música popular brasileira, bem como de compositores modernos e contemporâneos, tais como Villa-Lobos e Ligetti. Teve obras de concerto executadas em estreias mundiais por grupos como Quarteto Kalimera, Quinteto Lorenzo Fernandes e Quinteto de Metais da UFRJ. Como coautor, participou da publicação de artigos e apresentações em congressos de pesquisa, teoria musical e composição em São Paulo e no Rio de Janeiro. Atualmente participa do grupo de pesquisa MusMat. joaopenchel@hotmail.com

**Igor Chagas, mestrando** na área de Poéticas da Criação Musical, sob orientação do Prof. Carlos Almada, na Escola de Música da UFRJ, onde se formou em Licenciatura em Música. Iniciou seus estudos musicais na Escola de Música Villa-Lobos, concluindo os cursos de guitarra e arranjo. Continuou seus estudos no Centro

Ian Guest de Aperfeiçoamento Musical, se formando em percepção musical, harmonia, improvisação e arranjo. Atualmente integra a Orquestra de Ukuleles da UFRJ e atua como guitarrista no trio instrumental 3.por.10. Em sua produção acadêmica, conta com diversos artigos publicados relacionados à música e à tecnologia em congressos como Anppom, Eitam e MusMat. chagasguitarr@gmail.com

**Rodrigo Furman** (Brasil, 1986) é compositor, regente e multi-instrumentista, com maior expertise em instrumentos de arco, principalmente a viola. Especialista em orquestração e música sinfônica moderna e contemporânea, estudou Música tanto em aulas particulares quanto nas principais instituições do Rio de Janeiro, como a Escola de Música Villa-Lobos, o Conservatório Brasileiro de Música e a Universidade Federal do Rio de Janeiro, onde está concluindo seu Mestrado em Poéticas da Criação Musical/Composição, sob orientação do Prof. Carlos Almada, com foco na ópera-rock *Metropolis Part 2* da banda Dream Theater e no gênero prog-metal, além de ser pós-graduando em Regência Coral pelo Conservatório Brasileiro de Música. Participa do grupo de pesquisas MusMat. Anteriormente, esteve pesquisando sobre música eletroacústica e computacional na mesma instituição. Em paralelo, é Bacharel em Sistemas de Informação pela Faculdade de Informática Lemos de Castro, instituição onde também cursou o ensino médio técnico em Informática. Atualmente, cursa Licenciatura em Física no IF-UFRJ. rodrigofurman@gmail.com

**Pedro Proença** (Brasil, 1989) é baixista e compositor, natural do Rio de Janeiro. Toca baixo elétrico desde os 11 anos. Formou-se Bacharel em Matemática na UERJ em 2013 e está iniciando o Mestrado em Música, na linha de Poéticas da Criação Musical, no âmbito do projeto Desenvolvimento de Sistemas Composicionais na UFRJ em 2020, orientado pelo compositor Liduino Pitombeira. Formado no curso técnico de composição e arranjo da Escola de Música Villa-Lobos, estudou também com Arthur Kampela na Unirio em 2018. pedrofaria1989@gmail.com

**Liduino Pitombeira** (Brasil, 1962) é professor de composição da Escola de Música da Universidade Federal do Rio de Janeiro (UFRJ). Suas obras têm sido executadas no Brasil e no exterior e têm recebido diversas premiações em concursos de composição no Brasil e nos Estados Unidos. Pitombeira recebeu seu PhD em Composição e Teoria pela Louisiana State University (EUA), onde estudou com Dinos Constantinides. Tem publicado diversos artigos científicos sobre composição e teoria e desenvolvido pesquisa como membro do grupo MusMat da UFRJ. Foi premiado em 2019 com a Medalha Villa-Lobos, concedida pela Academia Brasileira de Música, e homenageado pela vida e obra no VII Festival de Música Contemporânea Brasileira. pitombeira@musica.ufrj.br