

# Composição Algorítmica em Redes Complexas

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

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

{vitorgr, lvelho}@impa.br

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

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

## 1. Introdução

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

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

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

## 2. Música em Forma de Rede

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

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

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

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

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

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

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

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

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

### 3. Composição Algorítmica em Redes Complexas

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

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

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

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

```

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

```

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

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

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

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

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

#### 4. Trabalhos Futuros

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

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

#### Referências

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