

Colagem, recorte e erros em um processo composicional utilizando o Music21.

Guilherme Lunhani¹

¹Instituto de Artes e Design – Universidade Federal de Juiz de Fora
Juiz de Fora, MG

gcravista@gmail.com

Abstract. *This article describes a case study in Computer Generated Assistance, for music analysis and didactic composition. A Python command was programmed to automate routines based on [Music21 2015] library, such: selection, cluster and fragmentation from a document in J.S.Bach's corpus. Some compositional exercises were used to test a operation, called glitch, in this corpus. In addition, [MuseScore 2015] and [Lilypond 2015] were used to edit and diagram scores. At the end, comment on bugs, compositional problems and future plans of composition.*

Resumo. *Este artigo descreve um estudo de caso em Assitência Gerada por Computador para análise musical e composição didática. Um comando Python foi programado para automatizar rotinas da biblioteca [Music21 2015] como: seleção, agrupamento e fragmentação de um documento no corpus bachiano. Alguns exercícios composicionais foram usados para testar uma operação glitch, neste corpus. Adicionalmente, [MuseScore 2015] e [Lilypond 2015] foram utilizados para edição. Ao final comentarei bugs, problemas composicionais e planos futuros de composição.*

1. Introdução

Este artigo trata de um protótipo de ferramenta, *m21.py*, um binário *Python* para manipulação criativa e analítica do [Music21 2015]. Na seção 2, relacionamos alguns textos à elaboração do programa. Na seção 3 uma justificativa para o projeto.

Uma descrição dos métodos de desenvolvimento, e do método criativo (aplicado aos materiais pré-composicionais gerados), são explicados na seção 4. Na seção 5 são apresentados alguns detalhes da ferramenta, úteis no processo de geração do material. Na seção 6, é apresentado um exemplo prático.

Na seção 7 são apresentados problemas técnicos observados, e uma autocrítica. Na seção 8, os planos futuros de desenvolvimento.

2. Trabalhos relacionados

Este trabalho iniciou com a investigação do Music21 que, segundo [Soares 2015]:

É uma biblioteca projetada para trabalhar com manipulação e análise de corpus de arquivos partituráveis. Prepara a conversão entre diversos arquivos de dados musicais.(...) Music21 tem uma abordagem voltada para uma "musicologia assistida por computador" e já tem incorporada em

suas classes algumas ferramentas comuns a esta prática como: numeração de grau funcional de acorde, numeração de classes de altura usando a classificação de Allen Forte : a implementação dos algoritmos de detecção de tonalidade elaborado por Krumhansl (1990) e aperfeiçoada por Temperley (2001), busca de padrões como transposições e inversões e outros.[Soares 2015, p. 71-72]

Por um lado, o desenvolvimento do *m21* busca reforçar a utilização de métodos estatísticos aplicados à musicologia¹. Por outro encoraja a transposição de técnicas, nas palavras de [Cascone 2000], pós-digitais, do universo eletroacústico para o universo da notação em partitura. No caso do universo eletroacústico, Cascone esclarece:

A estética pós-digital foi desenvolvida em parte como resultado de uma experiência imersiva, de um trabalho em ambientes repletos de tecnologia digital: computadores, zumbido de *fans*², impressoras a laser produzindo documentos, sonificação de interfaces de usuário, e o barulho abafado de discos rígidos. Mas mais especificamente, surgia das falhas da tecnologia digital que surgiram neste novo trabalho: falhas, *bugs*, aplicação de erros, quebras de sistema, clíping, serrilhamento, distorção, quantização, e mesmo o ruído de fundo de placas de som são materiais crus que compositores procuram incorporar em sua música.[Cascone 2000, p. 393]³

Neste sentido, foi prototipada uma tradução da citação acima, para um universo de peças do corpus do Music21. Com a aplicação de um único algoritmo de erro, em diferentes peças do corpus (como por exemplo as de J.S.Bach), são geradas harmonias características de um período que, que segundo Soares, existia:

(...) antes da preocupação imediata com os timbres ou da era das manipulações de amostras sonoras - e de certa maneira ainda proto-serialista. Uma música por vezes chamada politonal, polimodal ou usando o termo de Straus (2004): pós-tonal.”.[Soares 2015, p. 18]

Realizamos portanto, um cruzamento de espaços conceituais da composição pós-tonal com procedimentos de falhas. Não sendo um tipo composição nova, é relevante do ponto de vista didático. Neste sentido busquei elaborar uma ferramenta para ser usada em processos criativos musicais, considerando os fatores pedagógico, histórico e analítico.

¹Como por exemplo, aqueles propostos por [?].

²“Ventiladores” do *cooler*.

³Tradução de *The “post-digital” aesthetic was developed in part as a result of the immersive experience of working in environments suffused with digital technology: computer fans whirring, laser printers churning out documents, the sonification of user-interfaces, and the muffled noise of hard drives. But more specifically, it is from the “failure” of digital technology that this new work has emerged: glitches, bugs, application errors, system crashes, clipping, aliasing, distortion, quantization noise, and even the noise floor of computer sound cards are the raw materials composers seek to incorporate into their music.*

Questões diversas

Comentamos antigas conversas com o compositor Franscisco Zmekhol Nascimento de Oliveira, que sugeria compor peças com um mesmo processo, mas que segundo a contingência do momento, seriam diferenciadas por seus eventos musicais.

Esta abordagem encontrou respaldo nas estratégias composicionais sugeridas por partituras-planimétricas, propostas por [Koellreutter 1987]. Esta abordagem exercitar a improvisação discursos musicais, ao articular sons e silêncios.

A articulação de sons e silêncios, por sua vez, é semelhante aos fraseados dos corais de J.S. Bach.

3. Justificativa

Este trabalho capacitou a produção de um número considerável de exercícios criativos, para piano solo. A intenção é oferecer uma ferramenta para geração de materiais pré-composicionais, de maneira quase imediatista. Pode ser útil em cursos de Composição Assistida por Computador, em universidades ou em oficinas de arte.

4. Metodologia

Organização dos códigos

O programa foi separado em três arquivos: *i*) um binário em *Python* que realiza tarefas gerais da linha de comando (*m2l*); *ii*) rotinas do Music21 (*m2lutils.py*); e *iii*) um para rotinas externas (*tools.py*)⁴.

Categorização do software

Nas palavras de [Cope 2008, p. x-xiii], o *m2l* pode ser classificado como uma ferramenta para uma Assistência Gerada por Computador (*Computer Generated Assistance* ou CGA). Dentro das sub-categorias de CGA propostas por Cope, o *m2l* pode ser incluído nos três modos abaixo: 1) uso de uma Linguagem de Programação em texto (PLs)(*Programming Languages*) ao invés de uma linguagem de programação visual (VPL); 2) o material partitural é gerado para performance humana ao invés de uma performance eletroacústica; 3) utilização de regras diversas (*Rules Based*) e composição dirigida por dados (*Data-Driven*).

Método de composição

Em geral, o procedimento de composição se deu a partir da execução de um comando *m2l*, como explicitado nas seções 5 e 6.

O passos executados pelo comando, no exemplo apresentado na seção 6, foram :x *i*) subtração de compassos, aleatoriamente; *ii*) dos compassos restantes, notas pertencentes a este compasso foram suprimidas em um bloco harmônico; e *iii*) destes blocos, as oitavas serão embaralhadas para gerar simultanéides [Koellreutter 1987];

O material resultante do processo acima foi editado no [MuseScore 2015]. Algumas interferências não previstas foram incluídas. Seguimos com a observação de “organici-dades” próprias do material pré-composicional: fraseados, pontos culminantes e pontos de relaxamento, criando um pequeno discurso musical.

Por último, a peça foi diagramada no [Lilypond 2015].

⁴Todos códigos, exemplos e documentação estão disponíveis <https://www.github.com/jahpd/m2l>.

5. M21

Listamos, na tabela 1, algumas opções úteis do comando `./m21`. Foram usadas para a elaboração de materiais pré-composicionais, apresentados na seção 6.

Um manual de instalação e operação mais detalhado está disponível junto com o código-fonte⁵.

⁵Disponível em <https://www.github.com/jahpd/m21/doc/manual.pdf>.

Tabela 1: Tabela das opções utilizadas para produção e análise de um arquivo partiturável. Fonte: autor.

Nome	Comando	Abreviação	Execução
Compositor	<code>--composer</code>	-c	Campo de procura no corpus pelo nome de um compositor. Usado em conjunto com a opção “Index”.
Index	<code>--index</code>	-i	Indexação de uma peça (catálogo, p.e., bwv123). Usado em conjunto com a opção “Compositor”
Composição Assistida por Computador	<code>--CAC</code>	-C	<i>Flag</i> indicativa de uma operação de transformação em uma partitura. Usada em conjunto, as opções “Compositor” e “Index”.
Glitch	<code>--glitch</code>	-g	<i>Flag</i> indicativa do tipo de operação de transformação. A peça é desorganizada e verticalizada em blocos harmônicos, de maneira randômica, limitada apenas por regras de tessitura do piano.
Apresentar em um editor de partituras	<code>--Show</code>	-S	<i>Flag</i> indicativa que, o resultado obtido será executado em um editor de partituras apropriado, no caso deste trabalho, o [MuseScore 2015].
Plotar gráficos analíticos	<code>--plot-*</code>	-	<i>Flag</i> indicativa que um gráfico analítico será gerado. O Símbolo “*” representa a diversidade dos tipos de gráficos possíveis.

```
1 | ./m21 --show --composer bach --index bwv1 --plot-histogram-pitch-space
```

5.1. Análise orientada por gráficos

Histogramas e outros gráficos podem ser gerados para de diferentes aspectos de uma música. Por exemplo, na figura 1, apresentamos um histograma do espaço de alturas (*pitch-space*) do sexto movimento do BWV1 de J.S.Bach. O gráfico foi gerado com o auxílio do comando apresentado no código ??

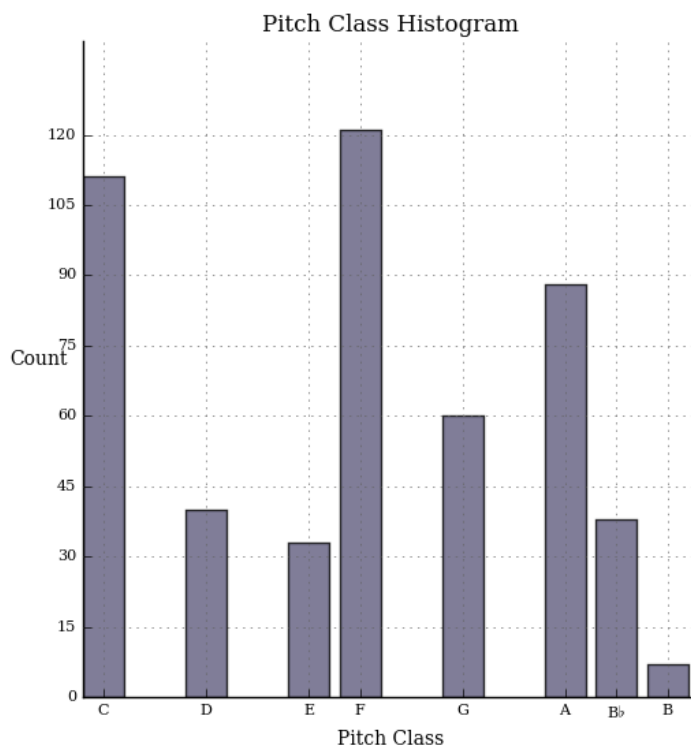


Figura 1: Histograma de *pitch-class* do BWV1.6, utilizando o comando Fonte: autor.

6. Resultados

Comandos foram utilizados para a estruturação de “Corais”, pequenas peças didáticas feitas por colagem de materiais da bachianos, recorte por erros (*glitch*), e readequações do discurso harmônico pós-tonal.

Resultados foram possíveis com a implementação do comando explicitado no código 1.

```
1 | ./main.py --show --CAC --composer bach --index bwv1 --glitch 2
2 | ./main.py -S -C -c bach -i bwv1 -g 2
```

Listing 1: Código utilizado para composição do Coral #1. Na linha 1 apresentamos sua versão estendida, e na linha 2, sua versão comprimida. Fonte: Autor.

6.1. BWV1

O sexto movimento de *Wie schön leucht der Morgenstern* (Cantata para a festa da Anunciação, 1725, ver figura ??)⁶ foi utilizado segundo procedimentos explicados na seção 4. Foi gerado um conjunto de simultanóides⁷, apresentado na figura 2.



Figura 2: Sequência de simultanóides gerados. Fonte: Autor.

Cada bloco harmônico são notas de um determinado compasso, escolhido ao acaso pelo programa, comprimidas em um único evento. Algumas notas podem ter sido omitidas por erros de codificação no *script* Python. Mas isso é coerente com o princípio estético. É também interessante observar uma direcionalidade da tessitura, que vai da região média aos graves, percebida após repetidos usos do comando descrito.

Realizei algumas intervenções com este material:

- manutenção da ordem dos blocos;
- escolha de pontos que delimitam fraseados (silêncio como articulado de frases);
- observação de densidades na quantidade de notas como fator para o fim do fraseado;
- dinâmicas como um dispositivo de ênfase do fraseado harmônico;
- a modificação de uma oitava de dois si bemois para um intervalo de sétima ou nona de si bequadro e si bemol compasso, inserindo um intervalo de sétima com outro sib;
- deslocamento ou subtração de elementos da figura 2.

A peça finalizada está na figura 3.

As ferramentas analíticas auxiliaram a observação de semelhanças e diferenças entre uma peça original e a peça variada. Por exemplo, com BWV1 de J.S.Bach, o histograma da figura ?? revela fatos analíticos comuns, como uma ênfase do *pitch class* 5, Fá, I grau; e depois *pitch class* 0, Dó, V grau, seguido de outras classes, como Lá (III grau) e Sol (ii grau ou V/V); outros, como Ré (vi grau) e Si bemol, possuem a mesma quantidade. Por último Mi (vii grau) e si natural possuem menor número, talvez como dispositivos cadenciais (como por exemplo na “modulação” Fá⇒Dó do quinto compasso da figura ??).

A partir da artesanaria do material pré-composicional gerado por fragmentação do BWV1, notei, no histograma da figura 4 que algumas proporções foram mantidas de maneira aproximada. A quantidade de eventos diminui drasticamente. Mesmo com a segmentação da peça, um centro tonal ainda pode ser notado (embora de maneira bastante ambígua).

⁶Vídeo disponível em <https://www.youtube.com/watch?v=POe2fBjbswA>.

⁷Utilizo aqui a nomenclatura de [Koellreutter 1987] para identificar blocos harmônicos.

para Glerm Soares

Coral #1

Após uma extração de alturas do BWV1 em 08/7/2015

Bach / music21

Guilherme Lunhani

Largo

Piano

f *ff* *p* *f*

5

ff *sfz* *mp*

10

pp *sfz* *f* *mf*

Figura 3: Peça resultante das intervenções. Fonte: autor.

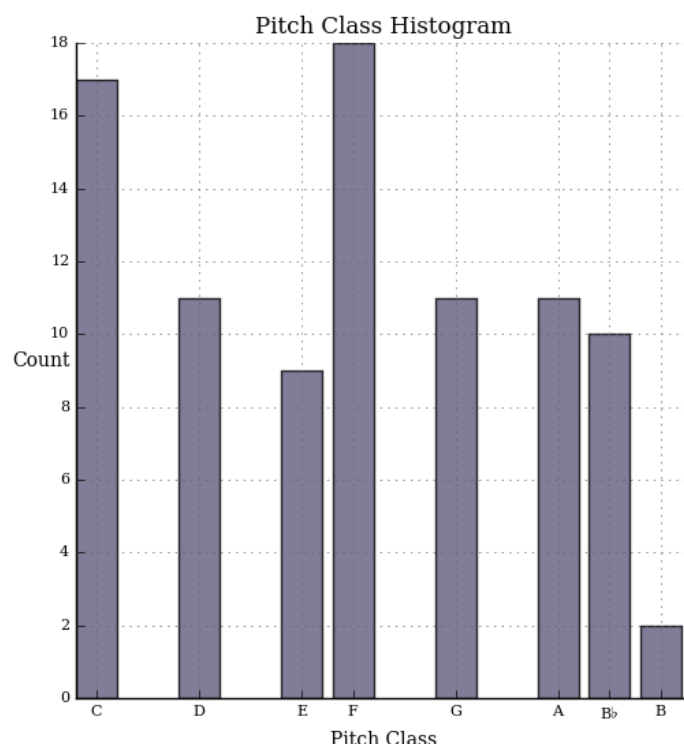


Figura 4: Histograma de *pitch-class* da peça feita, utilizando o comando `./main.py -x examples/bwv1/bwv1.xml --plot-histogram-pitch-class`

7. Conclusão

Duas funcionalidades do Music21 foram observadas: Musicologia Assistida por Computador e Composição Assistida por Computador (CAC). Da documentação e do corpus do *software*, programei rotinas que lidassem com aquelas tarefas que considerei demasiadamente laboriosas. Para isso proponho comandos que podem auxiliar em tarefas cotidianas, composicionais ou analíticas.

Da rotina para composição/análise, cito a busca no corpus, de uma obra específica ou obras pelo nome do compositor. Das rotinas analíticas, enumero *a*) identificação dos graus *b*) estruturação intervalar de blocos harmônicos *c*) plotamento de histogramas de classes de altura. Das rotinas composicionais, enumerei procedimentos para uma composição por fragmentação (*glitch*): *d*) colagem de fragmentos de uma partitura *e*) compressão de melodias em blocos harmônicos dos fragmentos resultantes *f*) troca de oitavas com as notas deste bloco *g*) possível fragmentação do bloco resultante em figuras ou arpejos. Existe adicionalmente um *bug* que ocorre por fatores lógicos, próprio da atividade de programação realizada, mas ainda não resolvido. Possivelmente o compositor pode lidar com dados nulos. Caso a peça de entrada seja muito grande, pode acontecer do programa não responder.

Foi observado uma versatilidade de materiais pré-composicionais gerados, bem como a possibilidade de improvisar com estes materiais. O exemplo apresentado na seção 6 foi feito em algumas horas. Com isso espero oferecer aos docentes e discentes em composição musical uma ferramenta alternativa para atividades diáticas em processos criativos. r

8. Planos Futuros

Correção do *bug* e implementação de um novo comando que fragmente várias obras do corpus em um único material pré-composicional (comando `--popp` ou `-p`). Continuação de novas composições para o ciclo.

9. Agradecimentos

Ao Guilherme Rafael Soares por apresentar a biblioteca *music 21*. Aos desenvolvedores do *muscore* e *lilypound*. FAPEMIG pelo financiamento da pesquisa.

Referências

- Cascone, K. (2000). The aesthetics of failure: 'post-digital' tendencies in contemporary computer music. *Computer Music Journal*.
- Cope, E. D. (2008). Prefacio a OM composer's book vol. 2. In *OM Composer's Book*, volume 2, pages ix–xv. Editions Delatour.
- Koellreutter, H. J. (1987). *Introdução à estética e à composição musical contemporânea*. Movimento, 2 edition.
- Lilypond (2015). Lilypond.
- MuseScore (2015). Musescore.
- Music21 (2015). Music21.
- Soares, G. R. (2015). Luteria composicional de algoritmos pós-tonais v1.1final.