LII Simpósio Brasileiro de Pesquisa Operacional
João Pessoa - PB, 3 a 5 de novembro de 2020

UMA ABORDAGEM HÍBRIDA BASEADA EM OTIMIZAÇÃO PARA GERAÇÃO DE *LICKS* DE GUITARRA

Anderson Rufino dos Santos Silva, Anand Subramanian
Universidade Federal da Paraíba
R. dos Escoteiros, s/n - Mangabeira, João Pessoa - PB, 58058-600
andersonrufino2007@gmail.com, anand@ci.ufpb.br

Nailson dos Santos Cunha
Universidade Federal de Campina Grande
R. Aprígio Veloso, 882 - Universitário, Campina Grande - PB, 58428-830
nailsoncunha@gmail.com

RESUMO

O presente trabalho descreve uma abordagem computacional que, mediante o uso de métodos exatos, heurísticos e estocásticos, realiza a construção de fragmentos melódicos descritos como *licks* de guitarra, no estilo *Blues*. Este tipo de aplicação integra-se no campo da Composição Algorítmica, que é a criação de música com o auxílio de algoritmos. Regras de transição entre fragmentos menores chamados *sublicks* são utilizadas para guiar o processo. Um modelo probabilístico baseado em Cadeias de Markov fez-se necessário na construção de um tipo diferente de *lick*, chamado de *lick turnaround*, devido às suas características peculiares. Para validar a ferramenta proposta, melodias geradas pelo algoritmo, bem como composições criadas por humanos, foram submetidas à um teste de audição com um grupo de 110 pessoas com diferentes níveis de conhecimento musical, a fim de observar possíveis diferenças estatísticas.

PALAVRAS CHAVE. Otimização Combinatória, Programação Inteira, Solos de Guitarra.

OC – Otimização Combinatória, OA – Outras aplicações em PO.

ABSTRACT

The present work describes a computational approach that, using exact, heuristic and stochastic methods, performs the construction of melodic fragments described as guitar licks. This type of application is integrated in the field of Algorithmic Composition, which is the creation of music with the aid of algorithms. Transition rules between smaller fragments called sublicks are used to guide the process. A probabilistic model based on Markov Chains was necessary in the construction of a different type of lick, called turnaround lick, due to its peculiar characteristics. To validate the proposed tool, melodies generated by the algorithm, as well as compositions created by humans, were submitted to a hearing test with a group of 110 people with different levels of musical knowledge, in order to observe possible statistical differences.

KEYWORDS. Combinatorial Optimization. Integer Programming, Guitar Solos.

CO - Combinatorial Optimization, OA - Other PO applications.

Galoá { Este trabalho foi publicado utilizando o **Galoá** proceedings

1. Introdução

Interpretado literalmente, o termo "Composição Algorítmica" é auto-explicativo: o uso de algoritmos para compor música. Essa é uma definição ampla, pois há séculos os músicos vêm propondo métodos que podem ser considerados como algorítmicos em algum sentido, apesar da criatividade humana ter um papel fundamental [Fernández e Vico, 2013].

[Pope, 1995] afirma que a CA pode assumir diversas formas, tais como procedural, estocástica ou sistemas baseados em conhecimento que são geradores no sentido de usarem dados de entrada para gerar dados musicais como saída. Geralmente, a literatura classifica os algoritmos de composição de acordo com a maneira que os mesmos processam os dados musicais, bem como suas estruturas. Diante disso, [Papadopoulos e Wiggins, 1999] classificam em seu trabalho os métodos computacionais para composição musical como: modelos matemáticos, sistemas baseados em conhecimento, gramáticas, métodos evolucionários, sistemas de aprendizagem, sistemas distribuídos e sistemas híbridos.

Um dos primeiros exemplos de composições geradas por computador está no trabalho de [Hiller e Isaacson, 1957], que utilizou o computador Illiac para gerar a peça musical Illiac Suite. A ideia principal foi usar regras de seleção para aceitar ou descartar aleatoriamente tons e ritmos gerados, através de um sistema de cadeias de Markov [Sandred et al., 2009].

No estilo musical *rock*, um *lick* caracteriza-se como uma pequena frase melódica. Isto pode soar de forma suspeita próximo à definição de *riff*, porém existe uma pequena diferença. A maioria dos guitarristas concordam que um *riff* é uma figura repetida que pode ser usada para formar a base de uma música, enquanto que um *lick* está mais para um trecho melódico e, geralmente, não é usado para formar a base de uma figura rítmica [Corporation et al., 2014]. Sendo assim, neste trabalho, o termo *lick* é utilizado para representar pequenos fragmentos melódicos de guitarra, e o termo *sublick* representa porções melódicas ainda menores, permitindo definir um *lick* como uma combinação de *sublicks*.

[Cunha et al., 2016] acreditam que a ferramenta resultante do seu trabalho pode vir a ser utilizada como auxiliar no processo de composição, criação e aprendizagem de música no instrumento guitarra. Caso o usuário seja um estudante, por exemplo, ele teria a oportunidade de estudar e aprender os *licks* de guitarra gerados. Além disso, a ferramenta também pode ser útil como um recurso para compositores com bloqueio criativo. Dessa forma, esse tipo de usuário poderia utilizar os *licks* gerados automaticamente como ponto de partida para uma possível composição, ou até mesmo, em suas próprias composições de fato.

O trabalho está organizado como segue: a seção 2 apresenta uma breve revisão da literatura, descrevendo trabalhos que utilizam diversas abordagens computacionais para geração de música; a seção 3 apresenta a metodologia adotada, juntamente com a modelagem do problema abordado como um problema de otimização combinatória; a seção 4 apresenta os resultados computacionais obtidos com o experimento; e, por último, a seção 5 conclui o trabalho.

2. Trabalhos Relacionados

[Mistler, 2017] propôs um sistema capaz de transcrever músicas escritas em notação de partituras para tablaturas, além de prever uma tablatura ótima para qualquer melodia dada como entrada. Duas abordagens foram apresentadas, ambas utilizando Aprendizado de Máquina. Na primeira abordagem, os acordes são previstos baseando-se em acordes tocados anteriormente. Isto foi possível através de uma *Long Short-Term Memory Recurrent Neural Network*, um tipo de rede neural recorrente que utiliza a ideia de curto e longo prazo. Já na segunda abordagem, a dificuldade de um acorde é prevista em termos de uma função de custo, ao invés de prever o acorde ideal diretamente. Além disso, a função de custo é baseada em probabilidades condicionais nos dados de

treinamento e estimada através de uma *Feed-Forward Neural Network*, outro tipo de rede neural a qual não possui cliclos ou *loops* em sua arquitetura.

[Hall e Smith, 1996] apresentam um algoritmo baseado em cadeias de Markov que compõe melodias de *Blues* dada uma sequência de acordes. Ele é composto por duas fases: a fase de *análise* e a fase de *síntese*. A fase de *análise* produz um modelo de Markov composto por tabelas de transição de zero, primeira e segunda ordem, que cobrem tanto a parte de melodias quanto a parte de ritmos. Um conjunto de tabelas de transição é produzido para cada acorde nos sons analisados, com o intuito de capturar a relação entre harmonia e melodia. Já a fase de *síntese* utiliza as tabelas criadas na fase de *análise* para gerar novas melodias. A avaliação do modelo deu-se através da interação com usuários, os quais não foram capazes de destinguir composições geradas por humanos das geradas por computador.

[Tanaka e Fujii, 2015] propuseram um modelo baseado em programação linear inteira que pode prover especificações de características de estruturas musicais globais no contexto de composição musical, as quais foram descritas como equações e inequações lineares. Os autores endossam que a música pode ser definida como sequências de padões musicais como acordes, ritmos e melodias. Diante disso, quando uma nova peça é composta, é necessário escolher uma sequência dentre todas as combinações de padrões, fato que justifica a proposta do trabalho. Além disso, usuários podem utilizar a ferramenta para obter ajuda na composição de estruturas musicais desejadas.

[Cunha et al., 2018] desenvolveram uma ferramenta capaz de gerar solos de guitarra no estilo *Blues* utilizando uma abordagem baseada em programação linear inteira. O *framework* realiza o sequenciamento de *licks* de guitarra retirados da literatura especializada ([Fritts, 2012] [Marshall, 1999]) para compor o solo completo através de uma matriz de penalidades que armazena o custo de transição de um *lick* para outro. Os autores empregaram uma estratégia de avaliação interativa para validar os solos gerados, intitulados como "otimizados", de forma que estes, de certa forma, competiam com outros solos, denominados de "aleatórios", os quais foram gerados de maneira aleatória, ou seja, os *licks* eram concatenados de forma randômica. Os resultados foram bem sucedidos diante do fato de que os solos otimizados tiveram uma aprovação maior relacionada aos solos aleatórios pelos avaliadores.

3. Definição e Formulação do Problema

Antes de apresentar com detalhes os métodos computacionais utilizados no processo de construção dos *licks*, é importante descrever certas particularidades que são levadas em conta neste processo, como a classificação dos *licks* gerados pela ferramenta e a definição de outras terminologias musicais.

- ***Blues***: O *Blues* é um gênero musical nascido no sul dos Estados Unidos, sem data precisa. Porém, sabe-se que os primeiros traços que o definiram surgiram por volta de 1890 e consolidaram-se durante as décadas de 1920 e 1930, tendo seu apogeu durante a primeira metade da década de 1950 [Carrijo, 2014]. A palavra *Blues* possui uma concepção de tristeza, sofrimento, melancolia, pois desde seu início, serviu como forma de expressão artística desses temas [Assis e Macedo, 2008].

- ***12-Bar-Blues***: *12-Bar Blues* é um modelo do estilo musical *Blues* que representa uma progressão dos acordes I, IV e V referentes à escala da tonalidade da música que está sendo tocada, ao longo de 12 compassos [Ganapes, 2015].

- ***Blue note***: A chamada *blue note* é considerada como um tipo de elemento básico pertencente à estilos afro-americanos como *Jazz*, *Blues*, *Soul* e *Rock*, e possui características únicas

quando executadas em harmonias nestes estilos. Em termos de altura, a *blue note* equivale à quinta bemol (movimento de recuar a nota musical em meio tom) da escala natural da tonalidade em que se encontra a harmonia, acrescentada à escala pentatônica menor. Por exemplo, a *blue note* referente à escala pentatônica menor de Dó é G♭ (lê-se sol bemol). O trabalho de [Weisethaunet, 2001] aborda a *blue note* a partir de uma prespectiva histórica, assim como a sua utilização.

- ***Licks* de repetição**: Um importante aspecto do *Blues*, é a repetição. Tal ideia envolve uma pequena frase musical, que pode repetir-se uma ou várias vezes [Phillips e Chappell, 2012]. É bastante útil quando há intenção de destacar ou aumentar a intensidade de um determinado momento durante a música [Fritts, 2012]. *Licks* de repetição possuem tais características.

- ***Licks turnaround***: Um *lick turnaround* é uma frase musical que ocorre nos dois últimos compassos de uma progressão em *blues*, usualmente conectando os acordes I e V (Dó e Sol) [Riker, 1994]. Exceto quando utilizado como uma finalização, o *lick turnaround* substitui os compassos 11 e 12 e "vira você" (*"turns you around"*) de volta para o primeiro compasso. [Chipkin, 1998].

- ***Licks* regulares**: Diferente dos *licks* de repetição e dos *licks turnaround*, os *licks* regulares são definidos neste trabalho como aqueles que não possuem nenhum aspecto especial semelhante à repetições ou sensações de término. Contudo, ainda sim, possuem sua importância pelo fato de representarem boa parte da composição da progressão *12-Bar Blues*.

3.1. Modelagem

O problema de geração de *licks* de guitarra pode ser visto como uma variante do Problema do Caixeiro Viajante (PCV). Dado um conjunto de cidades e o custo de deslocamento entre cada par delas, o PCV visa encontrar a maneira mais barata de visitar todas as cidades e retornar ao ponto de origem [Applegate et al., 2006]. O PCV pertence à classe de problemas NP-Difícil [Garey e Johnson, 2002], ou seja, o tempo gasto para resolvê-lo pode ser exponencial em relação ao tamanho da entrada [Buriol et al., 2000].

A metodologia desenvolvida para este trabalho é uma extensão do trabalho de [Cunha et al., 2018] e consiste na utilização de um modelo matemático para gerar *licks* de guitarra no estilo *Blues*. A divergência entre as abordagens ocorre devido os autores terem utilizado um banco de *licks* preestabelecidos a fim de gerar solos completos de guitarra (uma progressão *12-Bar Blues* completa) de modo que, elementos musicais como duração das notas e técnicas exclusivas do instrumento guitarra (*bend*, *slide*, *hammer-on*, *pull-off*) já encontravam-se encapsulados. Neste trabalho, como a intenção não é gerar solos de guitarra, e sim os próprios *licks*, a geração de blocos melódicos é realizada a nível elementar, através do sequenciamento de notas musicais.

Por conseguinte, um banco de *sublicks* é criado para ser utilizado no processo de sequenciamento. Os *sublicks* representam notas musicais em sua forma natural, bem como sob o efeito da aplicação de diversas técnicas, ambas as formas com durações variadas. Uma matriz armazena pontuações que representam custos de transição entre os *sublicks*, que podem ser penalidades ou bonificações, e são baseados em um conjunto de regras estabelecidas que abrangem conceitos de teoria musical e métodos de execução do instrumento guitarra, ambos voltados para o estilo *Blues*. Se uma determinada transição entre um *sublick* i e um *sublick* j for compatível com alguma regra de construção definida, ela será bonificada.

Com o intuito de diminuir o determinismo no processo de geração de *licks*, um subconjunto de *sublicks* é sorteado a partir dos *sublicks* contidos no banco. Também é possível passar

parâmetros de entrada para o modelo matemático, como quantidade de notas e técnicas contidas por *lick*, assim como a quantidade de *sublicks* desejada no sorteio para ser utilizada na composição.

Com o propósito de capturar as características relacionadas ao estilo e ao instrumento em questão, regras foram estabelecidas através da colaboração com músicos profissionais, como também pela consulta da literatura especializada em *Blues* e guitarra [Corporation et al., 2014; Phillips e Chappell, 2012; Fritts, 2012; McCabe, 2002; Weissman, 2004; Chipkin, 1998; Riker, 1994; Ganapes, 2015; Schonbrun, 2003]. A tabela 1 descreve todas as regras e pontuações utilizadas na construção das matrizes de custo dos *licks* regulares, dos *licks* de repetição e dos *licks turnaround*, enquanto que a tabela 2 identifica quais regras foram empregadas em determinado método de geração.

Tabela 1: Visão geral de todas as regras empregadas

Regra	**Descrição**	**Pont.**
1	Se a distância entre a nota i e a nota j é > 2	+250
2	Se a distância entre a nota i e a nota j é ≤ 2	-150
3	Se ambas as notas i e j possuem quiáltera e têm duração equivalente à Semínima	-100
4	Se ambas as notas i e j possuem quiáltera	+125
5	Se ambas as notas i e j possuem quiáltera e têm a mesma duração	+175
6	Se a nota i possui quiáltera e a nota j não, ou vice-versa	+100
7	Se a diferença de cordas onde estão localizadas as notas i e j é > 1	+150
8	Se a diferença de cordas onde estão localizadas as notas i e j é ≤ 1	-250
9	Se a nota i é igual à nota j	+75
10	Se a nota i é igual à nota j e a nota i possui bend e a nota j não, ou vice-versa	-225
11	Se a nota i possui bend atingindo a altura da nota j e j encontra-se na corda abaixo	-175
12	Se a nota j possui bend atingindo a altura da nota i e i encontra-se na corda acima	-175
13	Se a nota j localizar-se uma posição anterior ou posterior à nota i na escala pentatônica	-250
14	Se a soma das durações de i e j em caso de pausa é $>$ 1/2 de um compasso comleto	+75
15	Se a soma das durações de i e j em caso de pausa é $\leq$ 1/4 de um compasso completo	-75
16	Se a nota i é *blue note* (G♭) e a nota j é pausa	+100
17	Se a nota i é F e a nota j é *blue note* (G♭)	-100
18	Se a nota i é *blue note* (G♭) e a nota j é G	-100
19	Se a nota i é E♭, F ou C e a nota j é *blue note* (G♭)	-150
20	Se a nota i é *blue note* (G♭) e possui *bend* e a nota j é F ou G	-150
21	Se a nota i é diferente de E♭, F ou G e a nota j é *blue note* (G♭) e possui *bend*	+150
22	Se a nota i é *blue note* (G♭) e possui *bend* e a nota j é diferente de G♭ ou G	+150

Tabela 2: Correspondência entre as regras descritas e os métodos de geração

Regra	**Tipos de *licks***
1 a 15	regulares, repetição, *turnaround*
16	regulares, *turnaround*
17 e 18	*turnaround*
19 a 22	regulares

3.2. Formulação Matemática

Fazendo uma analogia entre o problema de geração de *licks* de guitarra e o PCV, pode-se perceber que as cidades correspondem aos *sublicks* e os deslocamentos entre elas equivalentes aos custos de transição. O modelo matemático desenvolvido é aplicado na geração dos *licks* regulares, *licks* de repetição e no segundo compasso dos *licks turnaround*, com pequenas alterações para estes dois últimos tipos.

Dado um grafo $G = (S, A)$, onde S representa o conjunto de *sublicks* e A o conjunto de arcos conectando dois *sublicks* tal que $A = \{(i, j) \mid i \neq j\}$, seja p_{ij} a pontuação de transição entre

o *sublick* i e o *sublick* j, d_i a duração total do *sublick* i, q_i a quantidade de notas que compõem o *sublick* i, N uma constante que define a quantidade exata de notas que o *lick* resultante deve conter, B o conjunto de *sublicks* que possuem a técnica *bend*, SB uma constante que define a quantidade exata de *bends* que o *lick* resultante deve conter, S' o conjunto de *sublicks* que possuem a técnica *slide*, SS uma constante que define a quantidade exata de *slides* que o *lick* resultante deve conter, Q o conjunto de *sublicks* que possuem quiáltera, SQ uma constante que define a quantidade exata de quiálteras que o *lick* resultante deve conter, BN o conjunto de *sublicks* que possuem *blue note*, SBN uma constante que define a quantidade exata de *blue notes* que o *lick* resultante deve conter, P o conjunto de *sublicks* que possuem pausa e SP uma constante que define a quantidade exata de pausas que o *lick* resultante deve conter. Seja x_{ij} a variável binária que assume valor 1 se o arco (i, j) for utilizado e 0, caso contrário e y_i uma variável auxiliar que assume o valor 1 caso o *sublick* i tenha sido utilizado e 0, caso contrário. A formulação matemática para o problema é a seguinte:

$$\min \sum_{i \in S} \sum_{j \in S} p_{ij} x_{ij} \tag{1}$$

$$\text{s.a.} \quad \sum_{j \in S} x_{ij} = y_i \qquad \forall i \in S \tag{2}$$

$$\sum_{i \in S} x_{ij} = y_j \qquad \forall j \in S \tag{3}$$

$$\sum_{i \in S} d_i y_i = 1 \tag{4}$$

$$\sum_{i \in S} q_i y_i = N \tag{5}$$

$$\sum_{i \in B} y_i = SB \tag{6}$$

$$\sum_{i \in S'} y_i = SS \tag{7}$$

$$\sum_{i \in Q} y_i = SQ \tag{8}$$

$$\sum_{i \in BN} y_i = SBN \tag{9}$$

$$\sum_{i \in P} y_i = SP \tag{10}$$

$$x_{ij} + x_{ji} \leq 1 \qquad \forall i \in S,\ \forall j \in S,\ i \leq j \tag{11}$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \geq 1, \qquad \forall S \subseteq V', |S| > 2 \tag{12}$$

$$x_{ij} \in \{0, 1\} \qquad \forall i \in S,\ \forall j \in S \tag{13}$$

$$y_i \in \{0, 1\} \qquad \forall i \in S \tag{14}$$

A função objetivo (1) minimiza o custo de transição entre os *sublicks*. As restrições (2) e (3) estabelecem que caso o *sublick* tenha sido utilizado, nele chega apenas um arco e dele sai apenas um arco, respectivamente. As restrições (4) definem que o total da soma das durações dos *sublicks* utilizados deve ser igual à 1, ou seja, equivalente ao tamanho de 1 *lick* (1 compasso). As

restrições (5) garantem que o número total de notas que compõem o *lick* resultante deve ser igual à constante N. As restrições (6), (7), (8), (9) e (10) garantem que, respectivamente, a quantidade de *bends*, *slides*, quiálteras, *blue notes* e pausas que aparece no *lick* resultante seja equivalente à quantidade definida pela entrada. As restrições (11) e (12) eliminam a ocorrência de *subtours* (ciclos) e as restrições (13) e (14) são restrições de integralidade que permitem que as variáveis assumam apenas os valores 0 ou 1.

Para gerar os *licks* de repetição, uma abordagem quase idêntica à utilizada na geração dos *licks* regulares é empregada. A única diferença existente está relacionada ao tamanho do trecho melódico resultante que, neste caso, é equivalente à 1/4 de um compasso completo. Esta restrição é consequência do modo como os *licks* de repetição são tratados neste trabalho. Um *lick* de repetição é caracterizado como tal se ocorrerem determinados padrões de repetição ao longo da melodia.

Neste contexto, a ideia para construir *licks* de repetição foi justamente a de gerar vários fragmentos melódicos de tamanhos iguais a 1/4 de compasso, a fim de concatená-los para formar estes padrões. Desta forma, comparado ao modelo matemático utilizado para gerar os *licks* regulares, o modelo exato dos *licks* de repetição difere em apenas uma restrição, a qual refere-se ao tamanho da melodia resultante, que passa a ser 1/4 de um compasso completo. A equação 15 descreve a restrição adaptada.

$$\sum_{i \in S} d_i y_i = 1/4 \tag{15}$$

Para construir o primeiro compasso de um *lick turnaround*, uma abordagem baseada no método Cadeias de Markov foi elaborada. Os atributos modelados pelo esquema definido são a altura e a duração das notas musicais. Apesar de fazer parte do escopo do trabalho, técnicas de guitarra não estão incluídas na geração deste tipo de *lick*. Consequentemente, duas matrizes que representam a contagem das transições observadas de ambos os atributos são utilizadas.

Por conseguinte, um banco de *licks turnaround* foi criado englobando *licks* retirados de [McCabe, 2002]. Visando uma maior nitidez relacionada à essência do estilo dos *licks turnaround*, o primeiro compasso de cada *lick* do banco é dividido em quatro subcompassos, cada um contendo uma quantidade equivalente de notas musicais. Este artifício facilita o processo de predição de notas ao longo do compasso que está sendo construído, pois lidar localmente com uma porção do problema é mais trivial do que lidar com o problema como um todo. Ao final do processo, a concatenação dos quatro subcompassos preditos é realizada, formando um compasso completo.

A partir deste ponto, a nota de partida que iniciará o processo de predição e também a primeira nota do *lick* a ser obtido, é proveniente de um sorteio isolado, o qual é baseado na quantidade de ocorrências referente à primeira nota musical de cada *lick turnaround* do banco. Definida esta nota, o próximo passo é realizar sorteios sucessivos a fim de descobrir quais notas musicais farão parte do primeiro compasso do *lick turnaround* em construção. O processo termina quando um compasso completo é obtido através da concatenação de todas as notas musicais sorteadas. Vale salientar que o aprendizado em tempo de execução faz-se presente neste modelo.

Para construir o segundo compasso de um *lick turnaround*, a mesma abordagem matemática utilizada na construção dos *licks* regulares e dos *licks* de repetição é empregada, com ressalva para algumas diferenças. A primeira delas, é a de que as notas musicais utilizadas como entrada para gerá-lo não advêm da escala pentatônica menor de Dó, como nos *licks* regulares e nos *licks* de repetição, mas sim de notas musicais utilizadas na composição do segundo compasso dos *licks turnaround* contidos no banco.

A segunda diferença está relacionada ao tamanho do trecho melódico resultante que, neste caso, é equivalente a 1/2 de um compasso completo. Foi decidido que esta restrição teria este comportamento observando a forma como os *licks turnaround* do banco são finalizados. Em grande maioria dos casos, aproximadamente a primeira metade do segundo compasso destes *licks* é composta por notas musicais com durações que vão de semicolcheias à semínimas. Já a segunda metade é representada por notas com durações maiores, como mínimas, fazendo com que estas notas musicas soem durante todo o resto do compasso remanescente.

Desta maneira, com a finalidade de reproduzir a mesma idiossincrasia encontrada nestes *licks*, após a criação de um fragmento proporcional à 1/2 de um compasso completo, a última nota musical petencente a este fragmento é estendida de forma a completar o *lick*, produzindo o efeito de soamento desta nota. A equação (16) descreve a restrição adaptada dos modelos descritos anteriormente para o método de que se trata.

$$\sum_{i \in S} d_i y_i = 1/2 \tag{16}$$

4. Resultados

O algoritmo foi implementado nas linguagens C++ e *Python* versão 3.6, utilizando a biblioteca de código aberto *lxml* versão 4.5 para manipular arquivos do tipo MusicXML e o *solver* IBM CPLEX versão 12.6. Os testes foram executados em uma máquina com processador Intel Core i7-4500u de 1.8 GHz, 8 GB de memória RAM e sistema operacional Ubuntu 16.04 LTS.

Para validar os *licks* gerados, um teste de audição baseado na abordagem de [Hall e Smith, 1996] foi realizado com o propósito de verificar se os ouvintes conseguiriam diferenciá-los de *licks* criados por humanos. Caso o modelo tenha realmente conseguido capturar a natureza do estilo *Blues*, os participantes teriam dificuldade em distinguir a categoria das melodias apresentadas. O framework desenvolvido por [Cunha et al., 2018] foi utilizado para concatenar os *licks* em ambas as situações.

Sendo assim, para conduzir o experimento, um sistema *web* foi desenvolvido a fim de submeter músicos e apreciadores à análise de seis solos de guitarra baseados no modelo *12-Bar Blues*, três deles compostos por *licks* gerados pela ferramenta e outros três compostos por *licks* criados por humanos, de maneira que as progressões eram dispostas aleatoriamente e sem que o usuário tivesse ciência desta circunstância. Uma escala de modelo Likert [Aguiar et al., 2011] variando de 1 (péssimo) a 7 (excelente) serviu para mensurar o nível de satisfação dos avaliadores, os quais deveriam informar também se o solo poderia ser utilizado em uma composição musical ou apresentação ao vivo. No total, 110 pessoas participaram do experimento enquadrando-se em 3 grupos de conhecimento musical distintos, sendo 28 de nível iniciante, 52 de nível intermediário e 30 de nível profissional. Para fins de conveniência, os solos baseados em *licks* criados por humanos serão mencionados pelo termo “solos humanos” (H), enquanto que os solos baseados em *licks* criados pelo computador serão mencionados como “solos de máquina” (M).

Na figura 1, a parcela das avaliações referente aos solos humanos que classificam-se entre péssimo e moderado corresponde a 38,79% do total, enquanto que a mesma parcela referente aos solos de máquina, para o mesmo intervalo, corresponde a 40,00%. Analogamente, as parcelas complementares das avaliações que classificam-se entre bom e excelente correspondem a 61,21% e 60,00% do total, respectivamente.

Para verificar a evidência de que não exista diferença estatística entre as amostras, o *Wilcoxon signed test rank* foi aplicado, o qual adequa-se à análise de dados pareados de forma combinada através da diferença entre médias. Em termos estatísticos, o teste é utilizado para verificar

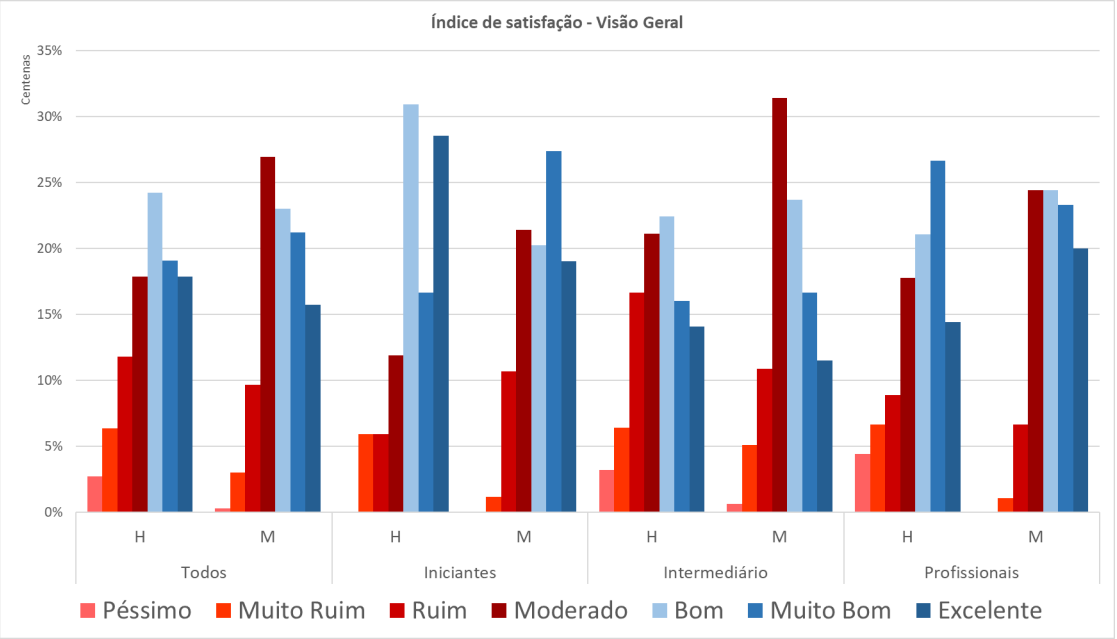

Figura 1: *Contabilização geral dos índices de satisfação*

se a distribuição de probabilidade de uma amostra pertencente à uma determinada classe é igual à distribuição de probabilidade de outra amostra pertentente à uma classe distinta [Woolson, 2007].

A figura 2 apresenta a diferença média de satisfação entre os tipos de solo de acordo com cada grupo de participantes, os quais são representados pelas cores na legenda. Nestas circunstâncias, com um intervalo de confiança de 95%, pode-se notar que a diferença geral das médias (cor vermelha) para essa amostra é muito pequena e favorável aos solos de máquina, havendo até mesmo a probabilidade de não existir, considerando a total extensão do intervalo de confiança. A mesma situação ocorre para os grupos de participantes de nível iniciante (cor azul) e intermediário (cor verde).

Por outro lado, identificou-se um cenário distinto para o grupo de participantes de nível profissional (cor amarela), onde a diferença média de satisfação entre ambos os tipos de solo pode chegar a 0,88 no pior caso e 0,05 no melhor caso, favorável aos solos de máquina. Este fato é passível de discussão e pode estar atrelado ao tamanho dos fragmentos melódicos o qual se é trabalhado em cada abordagem. Como descrito anteriormente, o banco de *licks* utilizados para gerar os solos humanos baseou-se em *licks* consolidados com o tamanho padrão de 1 compasso. Isto significa que características como a interdependência entre notas musicais estavam previamente encapsuladas, havendo uma limitação na flexibilidade criativa em comparação aos solos de máquina, os quais foram baseados em *licks* gerados mediante o sequenciamento a nível de nota musical. Uma amostra maior de participantes contribuiria para tirar melhores conclusões.

Também foi computada a frequência com que a opção que define se o solo poderia ser usado em uma música ou em uma apresentação ao vivo foi marcada como “sim”. A figura 3 apresenta a diferença média das frequências de uso dos solos humanos e dos solos de máquina para todos os grupos de participantes, na qual é possível perceber a ocorrência de uma situação semelhante à da análise referente aos índices de satisfação, ou seja, uma diferença média de 0,73 no pior

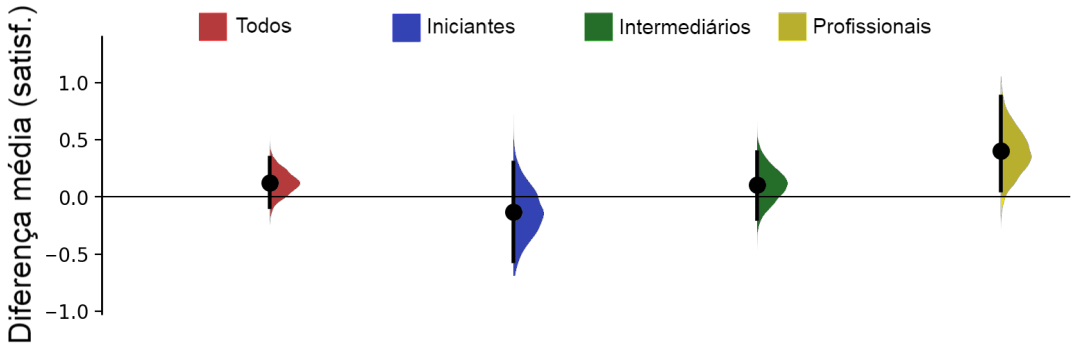

Figura 2: *Distribuição de diferença média de satisfação*

caso e 0,03 no melhor caso. Análogo ao estudo da satisfação, é plausível que essa condição seja resultante da carência de uma abordagem mais flexível no sentido de criatividade na composição dos solos humanos. A figura 4 reporta a frequência geral de uso dos solos para cada tipo de solo, de acordo com os grupos de participantes.

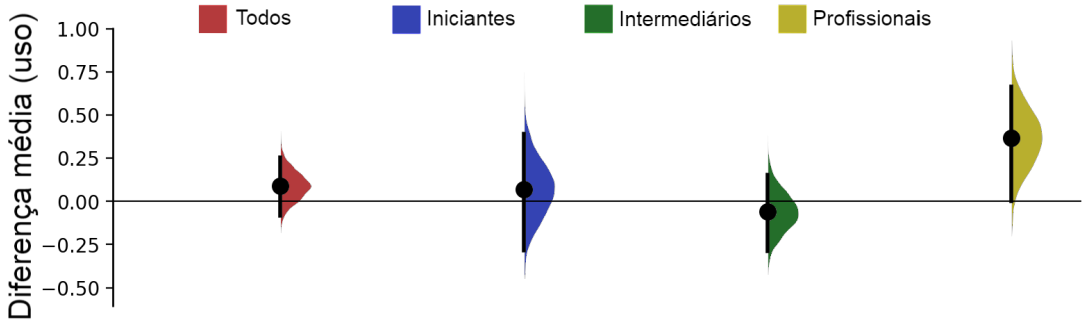

Figura 3: *Distribuição da diferença média do uso de solos*

5. Considerações Finais

Neste trabalho, propusemos um modelo híbrido baseado em Programação Linear Inteira e Cadeias de Markov para gerar *licks* de guitarra no estilo *Blues*. Os solos constituídos por estes *licks* soaram de forma agradável e o resultado do experimento concebeu a existência, ou não, de uma pequena diferença estatística favorável, em comparação com os solos baseados em licks criados por humanos. Como esta ferramenta é uma extensão do trabalho de [Cunha et al., 2018], o seu *framework* foi utilizado para construir os solos de guitarra em geral, na condição de produzir um ambiente em comum a todos.

Além deste aspecto, o *framework* conta com a flexibilidade da escolha de parâmetros que mostra-se bastante diversificada, sendo possível definir características pertencentes ao *lick* relacionadas à notas musicais e técnicas. Vale salientar que determinadas características como parametrização de entrada e regras de transição podem ser refinadas a fim de alcançar melhorias no sistema, ou até mesmo remodelar o seu conceito. Todos os solos submetidos à avaliação estão disponíveis no endereço https://bityli.com/2DJx2.

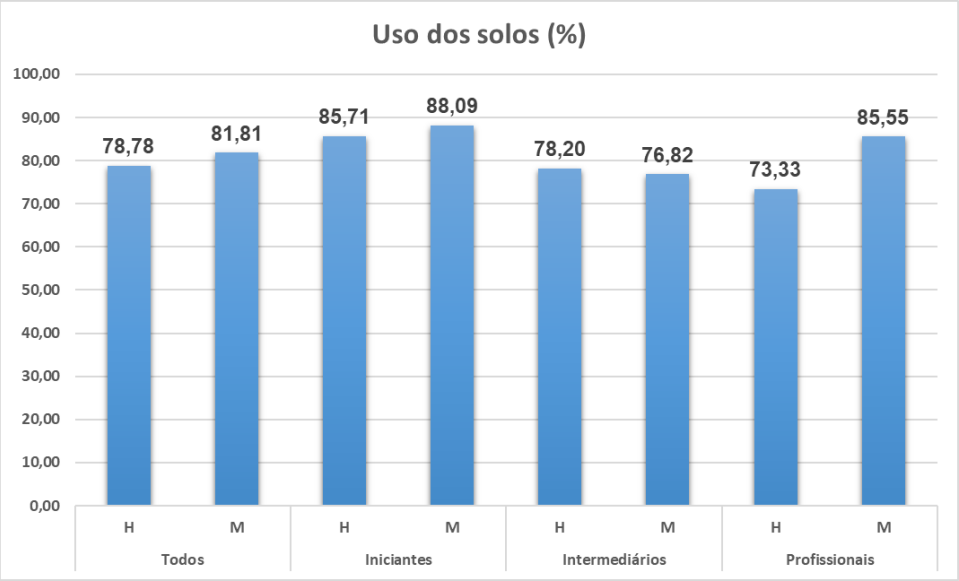

Figura 4: *Frequência de uso dos solos*

Referências

Aguiar, B., Correia, W., e Campos, F. (2011). Uso da escala likert na análise de jogos. *Anais do X Simpósio Brasileiro de Games e Entretenimento Digital, 07-09 de novembro de 2011 Salvador*.

Applegate, D. L., Bixby, R. E., Chvatal, V., e Cook, W. J. (2006). *The traveling salesman problem: a computational study*. Princeton university press.

Assis, D. T. F. d. e Macedo, K. B. (2008). Psicodinâmica do trabalho dos músicos de uma banda de blues. *Psicologia & sociedade*, 20(1):117–124.

Buriol, L. S. et al. (2000). Algoritmo memetico para o problema do caixeiro viajante assimetrico como parte de um framework para algoritmos evolutivos.

Carrijo, D. D. P. (2014). Cinema & blues: representações audiovisuais do gênero no século xxi.

Chipkin, K. (1998). *Real Blues Guitar*. Contemporary Guitar. Alfred Publishing Company, Incorporated. ISBN 9780898985795. URL `https://books.google.com.br/books?id=wPmuwEWoqyMC`.

Corporation, H., Chappell, J., Phillips, M., e Serna, D. (2014). *Guitar All-In-One For Dummies, Book + Online Video & Audio Instruction*. –For dummies. Wiley. ISBN 9781118872024. URL `https://books.google.com.br/books?id=sZAYBAAAQBAJ`.

Cunha, N. d. S. et al. (2016). Um framework para a geração semiautomática de solos de guitarra.

Cunha, N. d. S., Subramanian, A., e Herremans, D. (2018). Generating guitar solos by integer programming. *Journal of the Operational Research Society*, 69(6):971–985.

Fernández, J. D. e Vico, F. (2013). Ai methods in algorithmic composition: A comprehensive survey. *Journal of Artificial Intelligence Research*, 48:513–582.

Fritts, L. (2012). *2000 Blues: Licks that rock!* Centerstream Publishing.

Ganapes, J. (2015). *Blues you can use: a complete guide to learning blues guitar*. Hal Leonard Corporation.

Garey, M. R. e Johnson, D. S. (2002). *Computers and intractability*, volume 29. wh freeman New York.

Hall, M. A. e Smith, L. (1996). A computer model of blues music and its evaluation. *The Journal of the Acoustical Society of America*, 100(2):1163–1167.

Hiller, L. e Isaacson, L. M. (1957). *Illiac suite, for string quartet*, volume 30. New Music Edition.

Marshall, W. (1999). *101 must-know blues licks*. Hal Leonard.

McCabe, L. (2002). *101 Blues Guitar Turnaround Licks*. Mel Bay Publications Inc.

Mistler, E. (2017). Generating guitar tablatures with neural networks. *Master of Science Dissertation, The University of Edinburgh*.

Papadopoulos, G. e Wiggins, G. (1999). AI methods for algorithmic composition: A survey, a critical view and future prospects. *In AISB Symposium on Musical Creativity*.

Phillips, M. e Chappell, J. (2012). *Guitar For Dummies, with DVD*. John Wiley & Sons.

Pope, S. T. (1995). Fifteen years of computer-assisted composition. In *Proceedings of the 2nd Brazilian Symposium on Computer Music*.

Riker, W. (1994). *Complete Blues Guitar Method: Mastering Blues Guitar*. Complete Method. Alfred Music. ISBN 9781457439674. URL `https://books.google.com.br/books?id=jSNWkVYkt5AC`.

Sandred, O., Laurson, M., e Kuuskankare, M. (2009). Revisiting the illiac suite–a rule-based approach to stochastic processes. *Sonic Ideas/Ideas Sonicas*, 2:42–46.

Schonbrun, M. (2003). *The Everything Rock & Blues Guitar Book: From Chords to Scales and Licks to Tricks, All You Need to Play Like the Greats*. Everything®. Adams Media. ISBN 9781605505312. URL `https://books.google.com.br/books?id=epntDQAAQBAJ`.

Tanaka, T. e Fujii, K. (2015). Describing global musical structures by integer programming on musical patterns. In *Mathematics and Computation in Music*, p. 52–63. Springer.

Weisethaunet, H. (2001). Is there such a thing as the 'blue note'? *Popular Music*, 20(1):99–116.

Weissman, D. (2004). *Blues: the basics*. Routledge.

Woolson, R. (2007). Wilcoxon signed-rank test. *Wiley encyclopedia of clinical trials*, p. 1–3.

Galoá { Este trabalho foi publicado utilizando o **Galoá** proceedings