



**Universidade do Estado do Rio de Janeiro**

Centro de Tecnologia e Ciências

Instituto de Matemática e Estatística

Natan Figueiredo

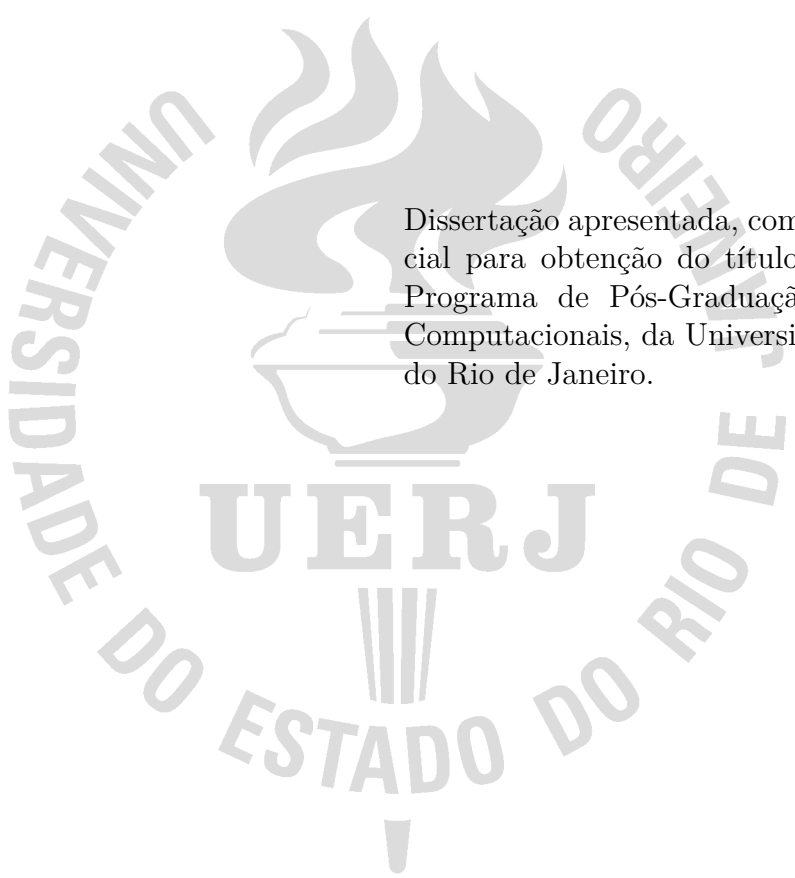
**Problemas Computacionais de Arranjo Musical**

Rio de Janeiro

2024

Natan Figueiredo

## Problemas Computacionais de Arranjo Musical



Dissertação apresentada, como requisito parcial para obtenção do título de Mestre, ao Programa de Pós-Graduação em Ciências Computacionais, da Universidade do Estado do Rio de Janeiro.

Orientadores: Prof. Dr. Luerbio Faria  
Prof. Dr. Vinicius Fernandes dos Santos

Rio de Janeiro

2024

CATALOGAÇÃO NA FONTE  
UERJ/REDE SIRIUS/BIBLIOTECA CTC/A

F475      Figueiredo, Natan.  
Problemas computacionais de arranjo musical/ Natan Figueiredo . -2024.  
88 f.: il.

Orientadores: Luerbio Faria, Vinicius Fernandes dos Santos  
Dissertação (Mestrado em Ciências Computacionais) - Universidade do Estado  
do Rio de Janeiro, Instituto de Matemática e Estatística.

1. Complexidade computacional - Teses. 2. Arranjo(Música) - Teses. I. Faria,  
Luerbio. II. Santos, Vinicius Fernandes dos. III. Universidade do Estado do Rio de  
Janeiro. Instituto de Matemática e Estatística. IV. Título.

CDU 004

Patricia Bello Meijinhos CRB7/5217 - Bibliotecária responsável pela elaboração da ficha catalográfica

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta  
dissertação, desde que citada a fonte.

---

Assinatura

---

Data

Natan Figueiredo

## **Problemas Computacionais de Arranjo Musical**

Dissertação apresentada, como requisito parcial para obtenção do título de Mestre, ao Programa de Pós-Graduação em Ciências Computacionais, da Universidade do Estado do Rio de Janeiro.

Aprovada em 26 de Setembro de 2024.

Banca Examinadora:

---

Prof. Dr. Luerbio Faria (Orientador)  
Instituto de Matemática e Estatística - UERJ

---

Prof. Dr. Vinicius Fernandes dos Santos (Orientador)  
Universidade Federal de Minas Gerais

---

Prof. Dr. Bruno Masquio  
Instituto de Matemática e Estatística - UERJ

---

Prof. Dr. Fernando Henrique de Oliveira Iazzetta  
Universidade de São Paulo

---

Prof. Dr. Uéverton dos santos Sousa  
Universidade Federal Fluminense

Rio de Janeiro

2024

## **AGRADECIMENTOS**

Ao Luerbio, por ser mais do que um orientador presente e proativo, sendo também um grande companheiro, sempre com boas conversas e caronas.

À Heloize, por me ajudar a aprimorar minha escrita, minhas apresentações, mas principalmente por estar ao meu lado.

## RESUMO

FIGUEIREDO, Natan. *Problemas Computacionais de Arranjo Musical*. 2024. 88 f. Dissertação (Mestrado em Ciências Computacionais) – Instituto de Matemática e Estatística, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2024.

Arranjos musicais têm sido tratados recentemente do ponto de vista combinatório. Demaine e Moses em 2017 [1], consideraram três problemas de decisão associados a arranjos musicais. Esses problemas consideram uma partitura com um número dado de pautas correspondentes aos instrumentos participantes. O objetivo é selecionar um certo número de pautas que atendam a algumas propriedades musicais específicas. Nessa dissertação, definimos versões mais gerais dos três problemas originais de Demaine e Moses, as quais chamamos ARRANJO-CONSONANTE GERAL,  $j$ -NOTAS SIMULTÂNEAS GERAL e VELOCIDADE DE TRANSIÇÃO LIMITADA GERAL. Demonstramos a NP-completude para os novos problemas a partir de  $\text{MAX3SAT}_3$  e GRAU MÁXIMO 4 CONJUNTO INDEPENDENTE, problemas cujas versões de otimização são MAXSNP-completos. A relação entre os ótimos nessas provas permitiram mostrar que as versões de otimização correspondentes dos 3 problemas são MAXSNP-difíceis e que portanto, não admitem Esquema de Aproximação Tempo Polinomial a menos que  $P=NP$ . Provamos que o problema  $j$ -NOTAS SIMULTÂNEAS é não aproximável por uma razão de aproximação inferior a  $n^{1-\epsilon}$ , para nenhum  $\epsilon > 0$ . Além disso, demonstramos que, quando no máximo 4 instrumentos tocam simultaneamente em cada tempo, o problema se torna FPT (tratável por parâmetro fixo) em relação ao tamanho  $k$  da solução. Dois novos problemas de decisão foram introduzidos ARRANJO PARA  $k$  INSTRUMENTOS e PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS. Os dois novos problemas foram provados serem NP-completos, respectivamente, através de reduções de  $k$ -COLORAÇÃO e COBERTURA DE VÉRTICES. O problema PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS também foi demonstrado ser FPT no parâmetro  $\tau$  – o número máximo de pautas que toca em cada tempo.

Palavras-chave: arranjo musical. complexidade. NP-completo. aproximação.

## ABSTRACT

FIGUEIREDO, Natan. *Musical Arrangement Computational Problems*. 2024. 88 f.  
Dissertação (Mestrado em Ciências Computacionais) – Instituto de Matemática e Estatística, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2024.

Musical arrangements have recently been considered from a combinatorial point of view. Demaine and Moses in 2017 [1] stated three decision problems associated with musical arrangements. These problems input a score with a given number of staves corresponding to the participating instruments. The goal is to select a certain number of staves such that they meet some specific musical properties. In this dissertation, we define more general versions of Demaine and Moses' three original problems, which we call GENERAL CONSONANT ARRANGEMENT, GENERAL  $j$ -SIMULTANEOUS NOTES and GENERAL LIMITED TRANSITION SPEED. We prove the NP-completeness for the new problems from MAX3SAT<sub>3</sub> and MAXIMUM DEGREE 4 INDEPENDENT SET, problems whose optimization versions are MAXSNP-complete. The relationship between the optima in these problems allowed us to show that the corresponding optimization versions of the 3 problems are MAXSNP-hard and therefore do not allow Polynomial Time Approximation Scheme unless P=NP. We prove that the problem GENERAL  $j$ -SIMULTANEOUS NOTES is not approximable by an approximation ratio lower than  $n^{1-\varepsilon}$ , for any  $\varepsilon > 0$ , unless P=NP and when at each time a maximum of 4 instruments play, then GENERAL  $j$ -SIMULTANEOUS NOTES is FPT (Fixed Parameter Tractable) with respect to the size  $k$  of the solution. Two new decision problems are introduced here: the ARRANGEMENT FOR  $k$  INSTRUMENTS and the TIME FILLING BY INSTRUMENTS. The two new problems are proven to be NP-complete through, respectively, reductions from  $k$ -COLORING and from VERTEX COVER. The problem TIME FILLING BY INSTRUMENTS is proven to be FPT with respect to  $\tau$  – the maximum number of staves playing each time.

Keywords: musical arrangement. complexity. NP-complete. approximation.

## LISTA DE FIGURAS

|           |                                                                                                                                      |    |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1  | - Pauta e Nomes das notas . . . . .                                                                                                  | 11 |
| Figura 2  | - Figuras Rítmicas Musicais . . . . .                                                                                                | 11 |
| Figura 3  | - Exemplo de partitura. . . . .                                                                                                      | 13 |
| Figura 4  | - Exemplo de arranjo. . . . .                                                                                                        | 14 |
| Figura 5  | - Exemplo de instâncias SIM e NÃO de CICLO HAMILTONIANO . . . . .                                                                    | 15 |
| Figura 6  | - Exemplo de instâncias SIM de CICLO HAMILTONIANO GERAL para $k = 5$ . . . . .                                                       | 15 |
| Figura 7  | - Exemplo de instâncias de CICLO MÁXIMO. . . . .                                                                                     | 16 |
| Figura 8  | - Exemplo de uma partitura com quatro pautas . . . . .                                                                               | 22 |
| Figura 9  | - Claves Musicais . . . . .                                                                                                          | 22 |
| Figura 10 | - Nomes das figuras rítmicas que representam as notas . . . . .                                                                      | 23 |
| Figura 11 | - Nomes das figuras rítmicas que representam as pausas . . . . .                                                                     | 24 |
| Figura 12 | - Exemplo de duração de compassos e figuras rítmicas. . . . .                                                                        | 24 |
| Figura 13 | - Escalas Cromáticas em sequência . . . . .                                                                                          | 25 |
| Figura 14 | - Tabela e grafo da escala de notas musicais . . . . .                                                                               | 25 |
| Figura 15 | - Relações intervalares . . . . .                                                                                                    | 28 |
| Figura 16 | - Trecho exemplo de dissonância . . . . .                                                                                            | 28 |
| Figura 17 | - Exemplo de representação de um grafo . . . . .                                                                                     | 29 |
| Figura 18 | - Exemplo de ciclo Hamiltoniano . . . . .                                                                                            | 30 |
| Figura 19 | - NP-Compleitude . . . . .                                                                                                           | 32 |
| Figura 20 | - Portas Lógicas . . . . .                                                                                                           | 34 |
| Figura 21 | - Exemplo de Circuito Booleano . . . . .                                                                                             | 35 |
| Figura 22 | - Exemplo de Circuito Booleano associado ao problema Conjunto Domi-<br>nante . . . . .                                               | 36 |
| Figura 23 | - Ostinatos . . . . .                                                                                                                | 38 |
| Figura 24 | - Trecho de arranjo da música Carinhoso como exemplo de instância de<br>ARRANJO CONSONANTE . . . . .                                 | 40 |
| Figura 25 | - Arranjo para a instância $M$ de ARRANJO CONSONANTE . . . . .                                                                       | 41 |
| Figura 26 | - Trecho de arranjo da música Carinhoso como exemplo de instância de<br>LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS . . . . .           | 42 |
| Figura 27 | - Arranjo para a instância $M$ de LIMITAÇÃO NO NÚMERO DE NOTAS<br>SIMULTÂNEAS . . . . .                                              | 43 |
| Figura 28 | - Trecho de arranjo da música “Trenzinho do Caipira”, como exemplo<br>de instância de LIMITAÇÃO NA VELOCIDADE DE TRANSIÇÃO . . . . . | 44 |
| Figura 29 | - Demonstração de conflito entre pautas do problema LIMITAÇÃO NA<br>VELOCIDADE DE TRANSIÇÃO com $t = \frac{1}{2}$ . . . . .          | 45 |

|                                                                                                                                                                                                                                                       |    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 30 - Arranjo para a instância $M$ de LIMITAÇÃO NA VELOCIDADE DE TRANSIÇÃO . . . . .                                                                                                                                                            | 45 |
| Figura 31 - Divisão de melodias em pautas . . . . .                                                                                                                                                                                                   | 46 |
| Figura 32 - Construção do Definidor de Verdade $T_i$ em (a) e do Testador de Satisfação $S_j$ em (b). . . . .                                                                                                                                         | 50 |
| Figura 33 - Exemplo de partitura $\mathcal{H}$ da instância de ARRANJO CONSONANTE, construída a partir de uma instância $I$ de MAX3SAT <sub>3</sub> . . . . .                                                                                         | 51 |
| Figura 34 - Exemplo de certificado $\mathcal{H}'$ para a resposta SIM da instância $(\mathcal{H}, {}^3/_{10}, 3, 3+2)$ de ARRANJO CONSONANTE GERAL . . . . .                                                                                          | 52 |
| Figura 35 - Instância de CONJUNTO INDEPENDENTE e ARR-CON. . . . .                                                                                                                                                                                     | 57 |
| Figura 36 - CONJUNTO INDEPENDENTE $\propto$ ARR-CON. . . . .                                                                                                                                                                                          | 57 |
| Figura 37 - Instância de CONJUNTO INDEPENDENTE transformada em instância de SIM- $j$ -NOTAS. . . . .                                                                                                                                                  | 59 |
| Figura 38 - Algoritmo FPT em $\Delta$ e em $k$ para o problema SIM- $j$ -NOTAS. . . . .                                                                                                                                                               | 64 |
| Figura 39 - Construção de definidor de verdade $T_i$ (a) e testador de satisfação $S_j$ (b). . . . .                                                                                                                                                  | 66 |
| Figura 40 - Exemplo de partitura $\mathcal{H}$ para uma Instância $M = (\mathcal{H}, p, t, s, k) = (\mathcal{H}, {}^3/_{10}, 2, n, n+k)$ de VTL, construída a partir de $I = ((U, C), k)$ de MAX3SAT <sub>3</sub> com $ U  = n$ e $ C  = m$ . . . . . | 67 |
| Figura 41 - Exemplo de certificados para a resposta SIM de VTL e de MAX3SAT <sub>3</sub> para a resposta SIM. . . . .                                                                                                                                 | 68 |
| Figura 42 - Instância de CONJUNTO INDEPENDENTE e VTL. . . . .                                                                                                                                                                                         | 70 |
| Figura 43 - Instâncias de CONJUNTO INDEPENDENTE e de VTL. . . . .                                                                                                                                                                                     | 71 |
| Figura 44 - Trecho da sinfonia número 5 de Beethoven. . . . .                                                                                                                                                                                         | 74 |
| Figura 45 - Arranjo obtido do trecho da sinfonia número 5 de Beethoven. . . . .                                                                                                                                                                       | 75 |
| Figura 46 - Instâncias de $k$ -COLORAÇÃO e $k$ -ARRANJO. . . . .                                                                                                                                                                                      | 76 |
| Figura 47 - $k$ -COLORAÇÃO $\propto$ ( $k$ -ARRANJO) . . . . .                                                                                                                                                                                        | 77 |
| Figura 48 - Arranjo obtido do trecho da sinfonia número 5 de Beethoven. . . . .                                                                                                                                                                       | 79 |
| Figura 49 - COBERTURA POR VÉRTICES $\propto$ PRE-INST . . . . .                                                                                                                                                                                       | 81 |
| Figura 50 - Algoritmo FPT no tamanho da solução $k$ para o problema PRE-INST <sub>2</sub> . . . . .                                                                                                                                                   | 83 |

## LISTA DE TABELAS

|          |                                                                         |    |
|----------|-------------------------------------------------------------------------|----|
| Tabela 1 | - Sumário dos resultados da literatura e da dissertação . . . . .       | 20 |
| Tabela 2 | - Tabela com notas musicais e seus intervalos. . . . .                  | 27 |
| Tabela 3 | - Tabela com restrições sobre a complexidade de organizar música. . . . | 47 |

## SUMÁRIO

|       |                                                                                      |    |
|-------|--------------------------------------------------------------------------------------|----|
|       | <b>INTRODUÇÃO</b>                                                                    | 10 |
| 1     | <b>DEFINIÇÕES</b>                                                                    | 21 |
| 1.1   | Definições Musicais                                                                  | 21 |
| 1.2   | Grafos                                                                               | 29 |
| 1.3   | Complexidade                                                                         | 30 |
| 1.4   | Aproximação                                                                          | 31 |
| 1.5   | Complexidade Parametrizada                                                           | 33 |
| 2     | <b>TRABALHOS RELACIONADOS</b>                                                        | 37 |
| 2.1   | Contextualização de estudos matemáticos de música                                    | 37 |
| 2.2   | A complexidade de tempo de arranjo musical                                           | 39 |
| 2.2.1 | <u>Problema do Arranjo Consonante</u>                                                | 39 |
| 2.2.2 | <u>Problema da Limitação no Número de <math>j</math> Notas Simultâneas</u>           | 41 |
| 2.2.3 | <u>Problema da Velocidade de Transição Limitada</u>                                  | 43 |
| 3     | <b>PROBLEMAS TRABALHADOS</b>                                                         | 48 |
| 3.1   | <b>Problema do Máximo Arranjo Consonante</b>                                         | 48 |
| 3.1.1 | <u>Uma versão de maximização para arranjo consonante</u>                             | 53 |
| 3.1.2 | <u>Complexidade parametrizada de ARR-CON</u>                                         | 55 |
| 3.2   | <b>Problema de decisão das pautas com número limitado de notas</b>                   | 58 |
| 3.2.1 | <u>MAXSIM-<math>j</math>-NOTAS<math>\Delta_4</math></u>                              | 61 |
| 3.2.2 | <u>Um resultado de complexidade parametrizada</u>                                    | 63 |
| 3.3   | <b>Problema das pautas com limitação de velocidade de transição</b>                  | 64 |
| 3.3.1 | <u>Uma versão de maximização para VTL</u>                                            | 69 |
| 3.3.2 | <u>Complexidade parametrizada de VTL</u>                                             | 70 |
| 4     | <b>DOIS NOVOS PROBLEMAS</b>                                                          | 73 |
| 4.1   | <b>Problema de ARRANJO PARA <math>k</math> INSTRUMENTOS (<math>k</math>-ARRANJO)</b> | 73 |
| 4.2   | <b>Problema do PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS</b>                          | 79 |
| 4.2.1 | <u>Um resultado parametrizado para PREENCHIMENTO POR INSTRUMENTOS</u>                | 82 |
|       | <b>CONCLUSÃO</b>                                                                     | 84 |
|       | <b>REFERÊNCIAS</b>                                                                   | 86 |

## INTRODUÇÃO

Essa Dissertação estuda a computação de alguns problemas sobre a produção de arranjos musicais.

Os arranjos estudados são bastante específicos. Nossos problemas tem como entrada uma partitura formada por um conjunto de pautas ou instrumentos. Quer-se responder se existe um arranjo formado por um conjunto dessas pautas que tenha alguma propriedade.

Nós vamos analisar esses problemas sob a ótica de complexidade computacional e determinar quando eles possuem uma solução algorítmica eficiente e quando possivelmente, não tem.

Vamos observar que para a maioria deles não há uma solução eficiente aparente. A demonstração dessa dificuldade é muito recente [1]. E não é conhecida ainda na literatura nenhuma tentativa de solucionar o problema de forma não ótima, mas eficiente em termos de tempo. Em nosso trabalho, vamos explorar algumas possibilidades de contornar essa dificuldade obtendo soluções não ótimas em tempo polinomial, porém com algum grau de confiança, por meio de algoritmos de aproximação.

Começamos a apresentar esses problemas falando do histórico do estudo dos algoritmos aplicados à música, para depois explicar a estratégia algorítmica da abordagem de problemas difíceis. Enfim, estaremos em condições de enunciar os nossos problemas de arranjos musicais para o leitor e explicar o quê e como vamos fazer para provar nossos resultados.

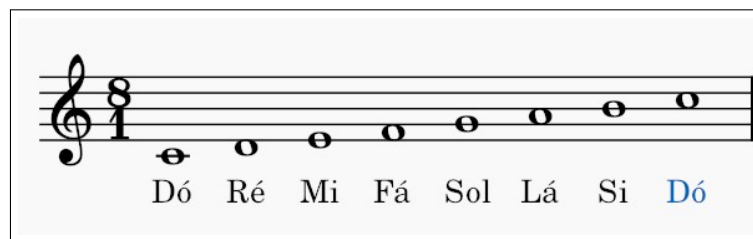
### Os algoritmos nos arranjos musicais

Do ponto de vista matemático, a música tem sido estudada em associação com instrumentos desde os tempos da escola Pitagórica. A ideia de utilizar instruções e processos formais para criar música remonta à Grécia Antiga. Pitágoras acreditava em uma conexão direta entre as leis da natureza e a harmonia dos sons, como expressa na música. De acordo com Boécio e outros [2], os Pitagóricos perceberam a relação consonante entre sons de martelos em forjas de ferreiros e descobriram que as massas dos martelos tinham uma razão de inteiros simples. A partir dessa observação, eles teriam concluído que sons consonantes e proporções simples estão relacionados. A partir dessa descoberta, foram criadas a “espiral das quintas pitagóricas” e a “escala pitagórica” que culminaram no sistema tonal que é utilizado até os dias de hoje e que usaremos como base para as definições musicais deste trabalho.

Apesar de todos esses formalismos gregos antigos baseados na observação da na-

tureza e conjecturas matemáticas, a música grega da época não era notada [3], mas tinha uma fixação que vinha de uma tradição oral. Guido d’Arezzo [4], um monge e professor italiano do século XI, criou um método para finalmente descrever as notas precisamente. Os nomes das notas: dó, ré, mi, fá, sol, lá, si foram baseados no começo das frases de um hino de louvor a São João Batista e são mostradas com sua notação na Figura 1. Antes de Guido d’Arezzo, a música era transmitida principalmente de forma oral, e a notação musical era rudimentar e imprecisa. A notação de Guido teve um impacto profundo na formalização da teoria musical, pois proporcionou uma linguagem comum para a escrita e leitura da música, permitindo maior precisão na composição e interpretação.

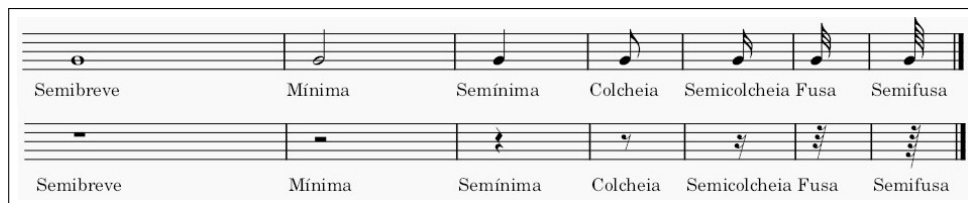
Figura 1 - Pauta e Nomes das notas



Legenda: Pauta com os nomes das notas nas suas respectivas notações.

Fonte: O autor, 2024.

Figura 2 - Figuras Rítmicas Musicais



Legenda: Duas pautas, a primeira (de cima) com uma nota em cada compasso, a segunda (de baixo) com uma pausa em cada compasso. As notas e pausas estão em ordem decrescente, cada uma com metade da duração da anterior (por exemplo, a Mínima dura metade do tempo da Semibreve).

Fonte: O autor, 2024.

Após a criação de uma escala musical e uma notação precisa, a primeira coisa mais próxima da definição de um algoritmo para a composição musical é dada com o nascimento da composição “canônica” no final do século XV [3]. A adoção de métodos combinatórios por Mersenne refletia sua *expertise* matemática e era motivada pelo desejo de demonstrar, de maneira objetiva, o vasto, porém calculável, número de possibilidades na construção melódica. O termo “cânion” deriva da palavra grega antiga “kanon”, que significa regra ou lei. Este método consiste em definir uma melodia principal, que é então acompanhada por uma série de imitações ou variações dessa parte, chamadas de

seguidores. Nesse caso, compositores de grande habilidade geram uma automatização de grande parte do processo composicional. A criatividade do compositor é concentrada em uma única melodia ou seção - a parte líder - a partir da qual uma composição inteira é obtida pela aplicação de diferentes transformações, como inversão, ou pela variação de parâmetros, como o número de seguidores e os atrasos entre eles.

Uma das manifestações mais notáveis de novos modelos matemáticos na teoria musical foi a adoção, pelo renomado matemático e teórico musical Marin Mersenne (1588-1648), de processos matemáticos baseados na combinatória em seu influente livro “Harmonie Universelle” [5], onde aplica fórmulas combinatórias, agora clássicas, para realizar operações como a tabulação de todas as 720 ( $= 6!$ ) permutações de um hexacorde (uma coleção de seis notas) [6]. A adoção de métodos combinatórios por Mersenne refletia seu desejo de demonstrar, de maneira objetiva, o vasto, porém calculável, número de possibilidades na construção melódica. Athanasius Kircher (1602–1680), um dos maiores polímatas do período barroco, introduziu em seu livro “Musurgia Universalis” um dispositivo de composição musical automática chamado *Arca Musarithmica*. Esse dispositivo permitia que não-músicos compusessem música polifônica por meio de um método sistemático, demonstrando a aplicação de princípios combinatórios na composição musical [7].

O próprio Mozart (1756-1791), criou um método chamado de Musikalisches Würfelspiel (Jogo de Dados Musical), que é uma forma de “composição aleatória”. Para cada um dos 16 compassos de um minueto vienense, o jogador poderia “gerar” aleatoriamente uma grande variedade de melodias por conta própria, selecionando opções aleatórias para cada compasso. Todas as escolhas fornecidas por Mozart foram criadas de modo que qualquer combinação resultasse em um minueto agradável, atendendo aos requisitos harmônicos e composicionais dos minuetos vienenses daquela época [8].

Lejaren Hiller foi o primeiro a usar um computador, o Illiac, para compor música. Com a ajuda de Leonard Isaacson, ele escreveu, em 1956, a primeira composição musical algorítmica feita por computador, a “Illiac Suite for String Quartet” [9]. O resultado do algoritmo foi então transposto para a notação musical tradicional, para ser executado por um quarteto de cordas.

O algoritmo desenvolvido por Hiller e Isaacson baseia-se em uma abordagem de gerador-modificador-seletor para simular o processo de composição. Primeiro, o algoritmo gera algum material musical bruto (o núcleo da composição), que é então modificado pela aplicação de várias técnicas e, finalmente, o melhor resultado é selecionado de acordo com algumas regras predeterminadas [10]. Embora a composição final seja produzida por um algoritmo, ela ainda opera dentro de certos limites e critérios definidos pelo compositor.

Iannis Xenakis é uma das principais referências na área da música computacional. Ele aplicou métodos estatísticos e probabilísticos, com o auxílio de computadores, para compor partituras. Xenakis utilizou algoritmos estocásticos para criar composições que

exploram sequências de “nuvens de som” — eventos sonoros compostos por milhares de sons pontuais isolados. Seu livro *Formalized Music* é considerado uma das obras mais importantes na bibliografia da música computacional [3].

Guido d’Arezzo, além de nomear as notas musicais, desenvolveu o conceito do tetragrama (pauta com 4 linhas), precursor do pentagrama moderno (pauta com 5 linhas) [4], o que possibilitou uma notação mais precisa e consistente das melodias. A padronização dos nomes das notas também facilitou o ensino e a propagação da música, estabelecendo as bases para a teoria musical ocidental como a conhecemos hoje. Nessa dissertação utilizamos a notação da *pauta* ou *pauta musical* com 5 linhas indicando a altura do som e diferentes desenhos das notas e das pausas que indicam sua duração, como podemos ver nas Figuras 1 e 2. As pautas, notas e pausas são divididas em *compassos*, que são uma unidade métrica demarcada por barras verticais na pauta, formada de tempos agrupados em porções iguais. Com exceção da Seção 3.3 e de figuras de exemplos de partituras, cada tempo musical dura um compasso inteiro neste trabalho, para mais fácil discernimento. Uma *partitura* é definida como uma reunião de pautas, cujos instrumentos correspondentes tocam simultaneamente. Um *arranjo* é uma versão de uma música já existente adaptada para um conjunto de instrumentos.

Figura 3 - Exemplo de partitura.

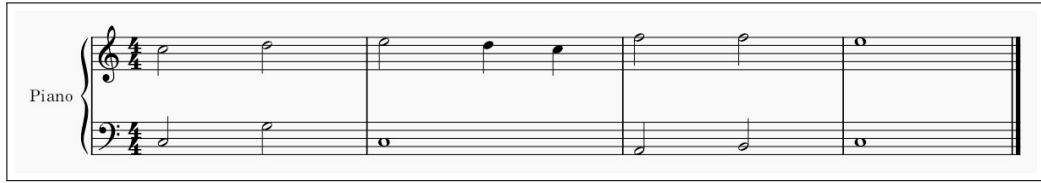


Legenda: Exemplo de partitura com 4 pautas tocando simultaneamente, uma para o violino, uma para o violoncelo e duas para o piano.

Fonte: ‘O autor, 2024’.

Podemos observar então, que a música pode gerar motivação para pesquisas em diversas áreas diferentes. Recentemente, observamos uma ampla gama de aplicações da inteligência artificial na música [11], como diferentes algoritmos para separar músicas em um sistema de classificação baseados em reconhecimento de emoções humanas [12]. Grafos tem sido usados para uma leitura de acordes mais intuitiva [13], o desenvolvimento de heurísticas voltadas à produção de música de forma automática, bem como a criação de

Figura 4 - Exemplo de arranjo.



Legenda: Exemplo de arranjo da música da Figura 3, com 2 pautas para piano. O piano toca as notas do violino e violoncelo da composição original garantindo que o arranjo seja reconhecido como uma versão da música original.

Fonte: ‘O autor, 2024’.

metaheurísticas para otimizar a harmonia musical [14].

## Tratando um problema combinatório difícil

Em combinatória temos alguns problemas que tratam de propriedades que ocorrem ou que não ocorrem em certas instâncias. Por exemplo, um grafo  $G = (V, E)$  é uma dupla, onde  $V$  consiste em um conjunto finito não-vazio de *vértices* e  $E$  é um conjunto de *arestas* formado por pares não ordenados de distintos elementos de  $V$ . Um problema clássico nesse contexto é determinar se um grafo contém ou não um ciclo Hamiltoniano.

CICLO HAMILTONIANO (CH)

**Instância:** Grafo  $G = (V, E)$ .

**Pergunta:** Existe um ciclo em  $G$  que passa por todos os vértices?

CICLO HAMILTONIANO foi provado ser NP-completo em 1972 [15]. Se a conjectura  $P \neq NP$  estiver correta significa que CICLO HAMILTONIANO é um problema que não pode ser resolvido em tempo polinomial, o que descarta a existência de algoritmos eficientes para CICLO HAMILTONIANO.

Na teoria da complexidade quando um problema é NP-completo e importante, nós temos algumas estratégias que permitem não propriamente resolver o problema em tempo polinomial, mas encontrar em tempo polinomial uma solução que aproxime de alguma forma a solução. No caso de CICLO HAMILTONIANO foi necessário [16] em 1979 considerar um problema de decisão CICLO HAMILTONIANO GERAL com um parâmetro  $k$  associado que identifica o tamanho de um ciclo do grafo  $G$ .

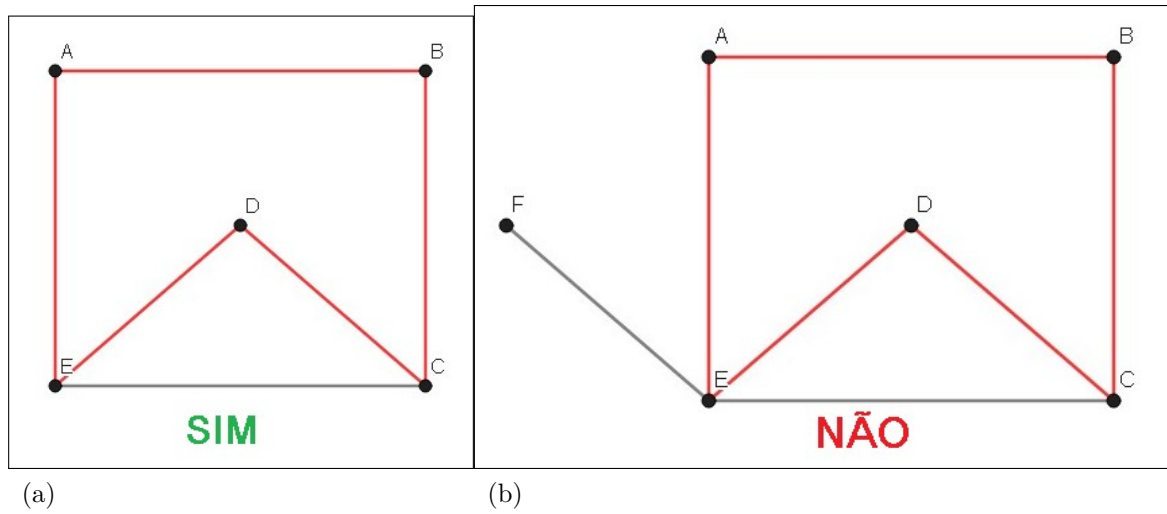
CICLO HAMILTONIANO GERAL

**Instância:** Grafo  $G = (V, E)$  e inteiro  $k > 0$ .

**Pergunta:** Existe um ciclo com pelo menos  $k$  vértices em  $G$ ?

CICLO HAMILTONIANO GERAL se estende naturalmente para o problema de otimi-

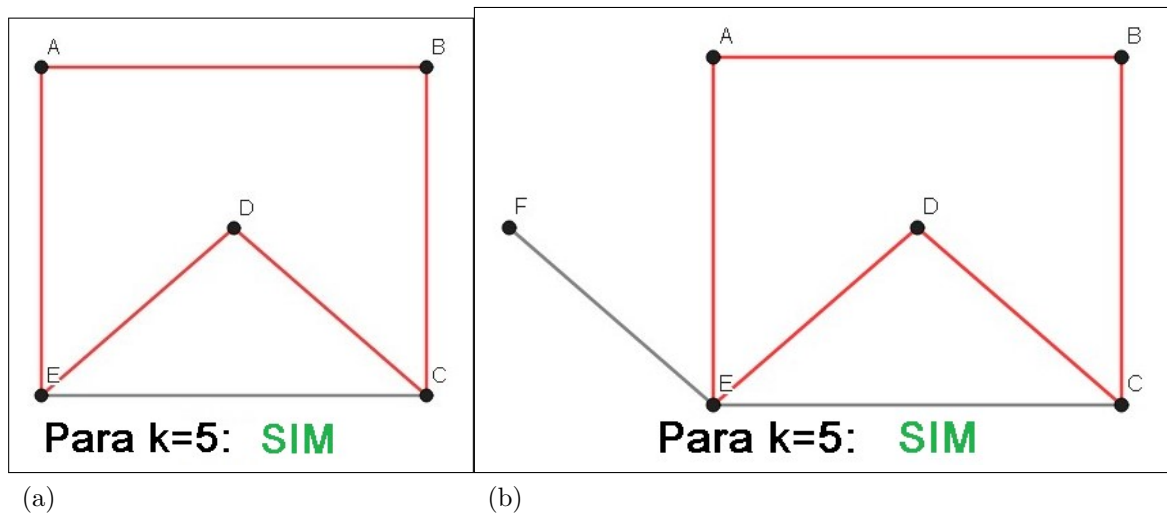
Figura 5 - Exemplo de instâncias SIM e NÃO de CICLO HAMILTONIANO



Legenda: Dois grafos como exemplos de instâncias SIM e NÃO de CICLO HAMILTONIANO. O segundo grafo 7b não pode ter um ciclo Hamiltoniano pois o vértice  $F$  tem apenas um vizinho, então não pode pertencer a nenhum ciclo.

Fonte: O autor, 2024.

Figura 6 - Exemplo de instâncias SIM de CICLO HAMILTONIANO GERAL para  $k = 5$ .



Legenda: Dois grafos como exemplos de instâncias de CICLO HAMILTONIANO GERAL. Graças ao parâmetro  $k = 5$  as duas instâncias são instâncias sim.

Fonte: O autor, 2024.

zação:

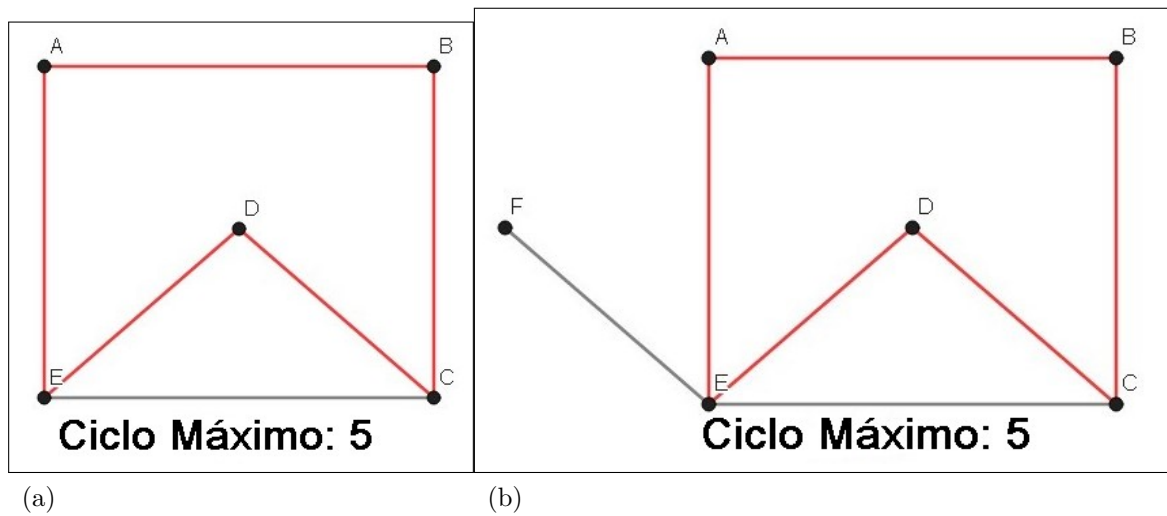
CICLO MÁXIMO

**Instância:** Grafo  $G = (V, E)$ .

**Objetivo:** Encontre o ciclo máximo de  $G$ .

Nesse problema de otimização consideramos  $OPT_{CH}(G)$  o tamanho máximo de um ciclo subgrafo de  $G$ .

Figura 7 - Exemplo de instâncias de CICLO MÁXIMO.



Legenda: Dois grafos como exemplos de instâncias de CICLO MÁXIMO. O número abaixo de cada grafo indica a quantidade de vértices presentes no maior ciclo do grafo.

Fonte: O autor, 2024.

Observe que os três problemas estão relacionados.

Do fato que CICLO HAMILTONIANO é NP-completo, uma primeira tentativa para tentar solucionar o problema seria determinar se existe um algoritmo aproximativo para CICLO MÁXIMO que encontre em tempo polinomial um ciclo  $C$  subgrafo de  $G$  que tenha um tamanho percentual  $\varepsilon \cdot OPT_{CH}(G)$  fixado  $0 < \varepsilon < 1$  do tamanho máximo  $OPT_{CH}(G)$  de um ciclo subgrafo de  $G$ . Infelizmente, existe um resultado de que a menos que  $P=NP$  não existe um tal algoritmo aproximativo para CICLO HAMILTONIANO GERAL.

Uma segunda possibilidade, para lidar com o problema NP-completo seria de encontrar  $A$  um algoritmo polinomial tratável por parâmetro fixo (FPT) no valor de  $k$ , isto é, que  $A$  roda em tempo polinomial para um valor de  $k$  fixo.

## A complexidade computacional dos arranjos musicais

Além dos estudos de Mersenne e Kircher, que remontam ao século XVII, problemas combinatórios musicais têm sido discutidos também em tempos mais recentes [17, 18]. Em 2017, Demaine e Moses [1, 19] fizeram a única contribuição conhecida, até onde vai nosso conhecimento, em problemas de decisão associados à organização de instrumentos e arranjos musicais. Os autores definiram 3 problemas de arranjos musicais.

Esses 3 problemas consideravam arranjos consonantes, limite nos números de notas no tempos do arranjo e arranjos sem notas simultâneas em partituras com tempo fixado de continuidade das notas.

Demaine e Moses provaram que todos esses problemas de decisão eram geralmente NP-completos. Que foi uma notícia bem ruim considerando que se  $P \neq NP$  não haveria esperança para uma solução ótima em tempo polinomial para esses três problemas.

Porém, no artigo original, nenhuma notícia foi dada a respeito dos problemas de otimização correspondentes visando soluções boas que não fossem ótimas obtidas em tempo polinomial. Nem mesmo estes problemas de otimização foram definidos.

Quando os problemas não podem ser resolvidos em tempo polinomial com base no tamanho  $n$  da entrada, e exigem tempo exponencial ou até fatorial, o valor de  $n$  não precisa ser muito elevado para tornar inviável o tempo de execução do algoritmo. Por exemplo, se um algoritmo requer  $n!$  passos para resolver um problema, considerando  $n = 25$ , mesmo com um computador capaz de realizar 1 bilhão de operações por segundo, ainda levaria milhões de anos para chegar à solução. Muitas composições do período romântico, predominante no século XIX, incluíam orquestras com até 20 tipos diferentes de instrumentos, totalizando cerca de uma centena de instrumentos ao todo. Ainda antes desse período, a composição *Spem In Alium* de Thomas Tallis (Pode ser escutada no Youtube), escrita no século XVI, foi criada para um coro com 40 vozes distintas [20].

Nessa dissertação, semelhantemente ao que foi motivado pelo problema do ciclo Hamiltoniano, nós analisamos os problemas de decisão de Demaine e Moses, obtendo versões de decisão gerais correspondentes que se estendessem naturalmente para as versões de otimização e parametrizadas dos problemas originais. Essas versões de decisão gerais, ampliaram os conjuntos de instâncias dos problemas originais, as quais também fornecemos provas de NP-completude. Essas provas são justificadas porque os problemas NP-completos que escolhemos transformar, tem propriedades algorítmicas fundamentais que permitiram descrever os comportamentos de aproximação ou de não aproximação e de parametrização para os problemas objeto considerados.

As relações entre os ótimos das instâncias dessas transformações permitiram provar que os três problemas de otimização são MAXSNP-difíceis e que portanto não admitem um algoritmo de aproximação para todo  $\varepsilon > 0$ , a menos que  $P=NP$ . Foi também mostrado que os problemas ARRANJO CONSONANTE GERAL e VELOCIDADE DE TRANSIÇÃO LIMITADA

GERAL são  $W[1]$  difíceis, o que significa que não admitem nenhum algoritmo tratável por parâmetro fixo. Um dos problemas – o de limitação de notas simultâneas não admite nenhum algoritmo aproximativo  $A$  com razão de aproximação  $R_A \leq n^{1-\epsilon}$ , porém, quando esse problema é restrito às instâncias onde cada pauta só pode tocar simultaneamente com no máximo outras 4, uma boa notícia é dada: de existir um algoritmo tratável por parâmetro fixo no tamanho  $k$  da solução.

Nosso trabalho também propõe dois novos problemas de decisão que infelizmente também provamos serem NP-completos:  $k$ -ARRANJO e PREENCHIMENTO POR INSTRUMENTOS. A segunda má notícia é que o problema  $k$ -ARRANJO é paraNP-completo, isto é, é NP-completo mesmo quando o parâmetro  $k$  do tamanho da solução é fixado em  $k \geq 3$ . Nós damos uma boa notícia para o problema PREENCHIMENTO POR INSTRUMENTOS exibindo um algoritmo tratável por parâmetro fixo no parâmetro  $\tau$  – o número máximo de instrumentos que toca em cada tempo.

## O roteiro da dissertação

No primeiro capítulo deste trabalho, temos as definições necessárias para o entendimento de teoria musical básica que será usada nos problemas computacionais trabalhados, definições de grafos que serão utilizadas para trabalharmos nos problemas estudados e definições sobre complexidade de problemas de decisão polinomiais, não-determinístico polinomiais, problemas de otimização, algoritmos de aproximação e problemas parametrizados.

No segundo capítulo, faremos uma breve revisão bibliográfica com trabalhos sobre problemas combinatórios na música. Apresentamos também uma revisão mais aprofundada do capítulo de livro de Demaine e Moses [1] que apresentaram os três problemas originais do ARRANJO CONSONANTE, LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS e LIMITAÇÃO NA VELOCIDADE DE TRANSIÇÃO.

No terceiro capítulo nós formalizamos as versões generalizadas dos três problemas originais de Demaine e Moses [1] e introduzimos as seguintes versões de maximização:

- MÁXIMO ARRANJO CONSONANTE (MAXARRCON) que tem o objetivo de determinar um arranjo com o maior número de compassos sem nenhum intervalo dissonante e com no mínimo  $p\%$  das notas em cada tempo de uma partitura de entrada. Provamos que MAXARRCON é MAXSNP-difícil a partir de uma L-redução de MAX3SAT<sub>3</sub> para MAXARRCON.
- MÁXIMO NÚMERO DE PAUTAS COM NÚMERO LIMITADO DE NOTAS SIMULTÂNEAS (MAX $j$ SIM) que tem o objetivo de determinar a maior quantidade de pautas que podemos ter em um arranjo com no mínimo  $p\%$  das notas originais em cada tempo

e no máximo  $j$  notas simultâneas. Provamos que  $\text{MAX}j\text{SIM}$  é não aproximável por um algoritmo polinomial  $A$  com  $R_A \leq n^\varepsilon$  para todo  $\varepsilon > 0$  e que  $\text{MAX}j\text{SIM}\Delta 4$  a versão de  $\text{MAX}j\text{SIM}$  cujas instâncias satisfazem que cada pauta toca no máximo com 4 outras pautas, é  $\text{MAXSNP}$ -difícil com uma L-redução a partir de MÁXIMO GRAU 4 CONJUNTO INDEPENDENTE. Nós provamos que  $\text{MAX}j\text{SIM}\Delta 4$  é FPT em  $\Delta$  e em  $k$  da solução.

- MÁXIMA VELOCIDADE DE TRANSIÇÃO LIMITADA ( $\text{MAXVTL}$ ) que tem o objetivo de determinar o maior número de compassos em um arranjo com no mínimo  $p\%$  das notas da música original em cada tempo e com nenhuma transição entre nota ou acorde com menos de  $t$  tempos. Provamos que  $\text{MAXVTL}$  é  $\text{MAXSNP}$ -difícil a partir de uma L-redução a partir de  $\text{MAX3SAT}_3$ .

No quarto capítulo introduzimos dois novos problemas de decisão de arranjo musical, considerando outros exemplos práticos de arranjos específicos:

- ARRANJO PARA  $k$  INSTRUMENTOS que dada uma partitura  $\mathcal{H}$  com  $n$  instrumentos, pergunta se existe uma partição  $\mathcal{H} = \{A_1, \dots, A_k\}$ , tal que se  $H_i$  e  $H_j \in A_i$  tocam ao mesmo tempo, então nesse tempo ou as notas de  $H_i$  estão contidas nas notas de  $H_j$  ou as notas de  $H_j$  estão contidas nas notas de  $H_i$ .
- PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS - ( $\text{PRE-INST}$ ) que dada uma partitura  $\mathcal{H}$  com  $n$  instrumentos, pergunta se existe um arranjo  $\mathcal{H}' \subset \mathcal{H}$  com no máximo  $|\mathcal{H}'| = k$  instrumentos, tal que em cada tempo não silencioso de  $\mathcal{H}$  existe uma nota que é tocada por algum instrumento de  $\mathcal{H}'$ .

Provamos que os problemas  $k$ -ARRANJO e  $\text{PRE-INST}$  são NP-completos. Além disso, mostramos que  $k$ -ARRANJO é paraNP-completo, ou seja, permanece NP-completo mesmo quando o parâmetro  $k$  do tamanho da solução, é fixado em  $k \geq 3$ . Também exibimos um algoritmo de complexidade tratável à parâmetro fixo para  $\text{PRE-INST}$  com respeito ao parâmetro  $\tau$  – o número máximo de instrumentos que toca em cada tempo.

Na Tabela 1 apresentamos um sumário de nossos resultados, cujos espaços em azul são resultados de Demaine e Moses [1] e em preto os dessa dissertação. Alguns resultados desse trabalho (Teorema 4.1.1 e Teorema 4.2.1) já foram publicados [21] nos *Anais do Encontro de Teoria da Computação'2024*.

No capítulo final nós apresentamos nossas conclusões com todos os resultados obtidos, com a previsão de trabalhos futuros.

Tabela 1 - Sumário dos resultados da literatura e da dissertação

| Problema                   | $P$                                                                          | NP-completo                                   | Aproximação                                                                          | Parametrização                                                                                                         |
|----------------------------|------------------------------------------------------------------------------|-----------------------------------------------|--------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| ARR-CON                    | Se $p = 0, 1$ e $ \mathcal{H}  = 1$ [1]                                      | $0 < p < 1$ [1]                               | $\nexists$ EATP a menos que $P=NP$ . (Teo 3.1.4)                                     | W[1]-difícil no tamanho $k$ da solução (Teo 3.1.6)                                                                     |
| SIM- $j$ -NOTAS            | Se $p = 0, 1$ , $(p > \frac{j}{j+1})$ e $(p > \frac{1}{3})$ para $j = 1$ [1] | $0 < p \leq \frac{j}{j+2}$ [1]                | Não aproximável com $R_A \leq n^{1-\varepsilon}$ a menos que $P=NP$ (Teo 3.2.7)      | FPT em $\Delta$ e em $k$ para instâncias $M = (\mathcal{H}, 0, 1, k)$ . (Teo 3.2.8)<br>W[1]-difícil em $k$ (Cor 3.2.1) |
| SIM- $j$ -NOTAS $\Delta 4$ | Se $p = 0, 1$ , $(p > \frac{j}{j+1})$ e $(p > \frac{1}{3})$ para $j = 1$     | $0 < p \leq \frac{j}{j+2}$                    | 5-aproximativo (Teo 3.2.5)<br>$\nexists$ EATP a menos que $P=NP$ . (Teo 3.2.4)       | FPT em $\Delta$ e em $k$ para instâncias $M = (\mathcal{H}, 0, 1, k)$ . (Teo 3.2.8)                                    |
| VTL                        | $p = 0, 1$ [1]                                                               | $0 < p < 1$ [1]                               | $\nexists$ EATP a menos que $P=NP$ . (Teo 3.3.2)                                     | W[1]-difícil no tamanho $k$ da solução (Teo 3.3.4)                                                                     |
| $k$ -ARRANJO               | $k = 1, 2$ (Teo 4.1.4)                                                       | $k > 2$ (Teo 4.1.1)                           | Não aproximável com $R_A \leq n^{1-\varepsilon}$ a menos que $P=NP$ (Teo 4.1.3)      | paraNP-difícil no tamanho $k, k \geq 3$ da solução (Cor 4.1.1)                                                         |
| PRE-INST                   | $k$ constante ou em cada tempo no máximo 1 instrumento                       | No mínimo 2 instrumentos tocando. (Teo 4.2.1) | $O(\log T)$ , onde $T$ é o número de tempos não silenciosos da entrada. (Teo. 4.2.2) | FPT no parâmetro $\tau$ e no tamanho $k$ da Solução (Teo 4.2.3)                                                        |

Legenda: Tabela com o estado-da-arte da complexidade de tempo de arranjos musicais. Representamos em azul os resultados da literatura, em preto os resultados da dissertação.

## 1 DEFINIÇÕES

Nesse capítulo apresentamos as definições, conceitos utilizados bem como os problemas de decisão e otimização estudados.

### 1.1 Definições Musicais

Nesta seção, vamos passar por definições importantes para a compreensão dos problemas trabalhados.

A partitura, embora seja fundamental para a notação e preservação da música no Ocidente, não é capaz de capturar a totalidade da experiência musical. Ela registra elementos técnicos como altura, ritmo e dinâmica, mas não consegue representar plenamente a expressividade, o timbre e as nuances interpretativas que são essenciais para a performance musical. Existem outras maneiras de transmissão musical que a notação tradicional ocidental não abrange completamente. Assim, a partitura é uma ferramenta que reflete uma abordagem cultural específica.

Algumas definições costumam ser mais abstratas, mas vamos considerar versões simplificadas para focar em alguns aspectos específicos.

**Definição 1.1.1** (Dicionário Aurélio [22]). *Uma pauta musical na notação ocidental, é um sistema de cinco linhas horizontais, paralelas e equidistantes sobre e entre as quais se escrevem as notas musicais. Uma partitura musical é uma coleção de pautas musicais.*

Uma partitura é formada por pautas que representam os diferentes instrumentos que tocam uma música.

A posição vertical de uma nota na pauta denota o quão grave ou aguda (baixa ou alta respectivamente) e a posição horizontal representa a ordem em que as notas são tocadas da esquerda para a direita. Na Figura 8 temos um exemplo de partitura com quatro pautas musicais.

A posição vertical de uma nota denota o quão grave ou aguda ela é, mas as pautas não tem uma posição fixa universal para cada nota, para isso servem as claves, elas definem um parâmetro apontando onde uma nota específica é posicionada na pauta e a partir dela as outras notas vão estar a uma distância específica. As ainda mais usadas hoje em dia, são a Clave de Sol () , que é usada para instrumentos ou vozes mais agudas, a Clave de

Dó () e a Clave de Fá () , que é usada para instrumentos ou vozes mais graves.

**Definição 1.1.2** (Dicionário Aurélio [22]). *Uma clave musical é um sinal no princípio da pauta, para determinar o nome das notas e o grau de elevação delas na escala dos sons.*

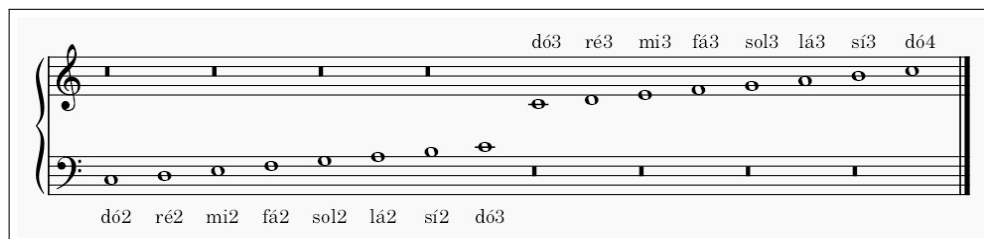
Figura 8 - Exemplo de uma partitura com quatro pautas



Legenda: Partitura com um violino, um violoncelo, e um piano, (repare que o piano é representado por duas pautas, porque pode tocar muitas notas simultaneamente e com intervalos maiores que apenas 5 linhas da pauta de distância). No piano, além das notas, temos também os símbolos das diferentes durações de silêncio (pausas).

Fonte: O autor, 2024.

Figura 9 - Claves Musicais



Legenda: Duas pautas musicais, a primeira com a *clave* de Sol, indicando que a nota Sol 3 se posiciona na quarta linha (de cima para baixo) e *clave* de Fá, que indica o Fá 2 na segunda linha.

Fonte: O autor, 2024.

**Definição 1.1.3.** [23] *O tempo na música é o pulso básico subjacente à música mensural, ou seja, a unidade temporal de uma composição. É o tempo regular ou padrão de tempo que serve como referência para a duração das notas e pausas.*

**Definição 1.1.4** (Dicionário Aurélio [22]). *O compasso é uma unidade métrica formada de tempos agrupados em porções iguais.*

O compasso na música serve para organizar o tempo em unidades regulares, facilitando a leitura e execução da música. Ele também estrutura o ritmo ao definir quantos tempos cada compasso contém, conforme indicado pela assinatura de tempo. Isso permite que músicos toquem juntos de forma sincronizada e que ouvintes acompanhem a música mais facilmente.

**Definição 1.1.5.** *As figuras rítmicas na música são cada um dos sinais gráficos que indicam a duração de uma nota ou de uma pausa. Na Figura 10, cada um dos símbolos com o nome abaixo (Semibreve, Mínima, Semínima, Colcheia, Semicolcheia, Fusa e Semifusa) é uma figura rítmica que representa uma duração de nota diferente. Da mesma maneira, também existem figuras rítmicas que representam diferentes durações para as pausas, conforme exemplificado na Figura 11.*

Uma figura rítmica não define a duração em segundos de uma nota, ela tem uma duração de tempo relativa. O que define a quantidade de tempos por minuto, ou BPM (battidas por minuto), é o andamento da música, como Adagio(60BPM), ou Allegro(120BPM), mas não serão usados andamentos neste trabalho.

Figura 10 - Nomes das figuras rítmicas que representam as notas



Legenda: Figuras rítmicas das pausas e seus respectivos nomes em ordem de duração decrescente: Semibreve, Mínima, Semínima, Colcheia, Semicolcheia, Fusa e Semifusa. Cada figura rítmica possui metade da duração da figura rítmica anterior (por exemplo, a Semínima dura metade de uma Mínima, que, por sua vez, dura metade de uma Semibreve).

Fonte: O autor, 2024.

**Definição 1.1.6.** *Uma fórmula de compasso é composta por dois valores, representada da seguinte maneira:  $\frac{a}{b}$ . O valor  $a$  (valor superior) apresenta quantos tempos o compasso dura, o valor inferior  $b$  (valor inferior) apresenta qual a figura rítmica é usada para representar o tempo, sendo 1=Semibreve, 2=Mínima, 4=Semínima, 8=Colcheia, 16=Semicolcheia, 32=Fusa, 64=Semifusa.*

Figura 11 - Nomes das figuras rítmicas que representam as pausas



Legenda: Figuras rítmicas das pausas e seus respectivos nomes (Homônimos aos nomes das notas) em ordem de duração decrescente: Semibreve, Mínima, Semínima, Colcheia, Semicolcheia, Fusa e Semifusa. Cada figura rítmica possui metade da duração da figura rítmica anterior.

Fonte: O autor, 2024.

Figura 12 - Exemplo de duração de compassos e figuras rítmicas.



Legenda: Quatro compassos totalmente preenchidos por notas. Repare que no compasso  $\frac{4}{4}$  cabe uma Semibreve, duas Mínimas, quatro Semínimas e oito Colcheias.

Fonte: O autor, 2024.

Dessa maneira, o compasso  $\frac{3}{4}$  é formado por três tempos e cada tempo dura uma semínima.

Nessa dissertação, o problema VTL é o único que utiliza a definição de compasso, portanto, com a exceção do problema VTL, cada compasso terá um único tempo. Por isso, não usaremos a palavra compasso, mas sim a palavra tempo com o mesmo sentido.

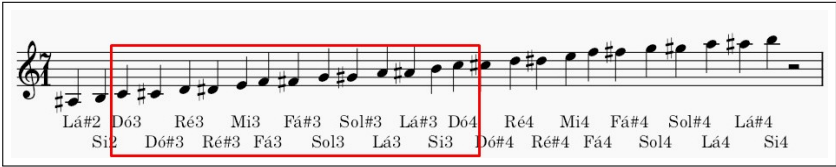
**Definição 1.1.7.** As notas musicais representam convencionadamente as alturas relativas dos sons produzidos por alguma fonte.

**Definição 1.1.8.** Uma escala é uma sequência de notas com distâncias relativas definidas.

As alturas das notas formam um espectro sequencial, as quais podem ser selecionadas a partir de suas frequências, que por sua vez se encontram, em um domínio contínuo. As escalas musicais permitem que as notas e as distâncias entre elas sejam tratadas em uma convenção de maneira discreta, simplificando a reprodução musical e facilitando a prática em conjunto.

Na música ocidental convencionou-se ao longo do tempo o uso de 12 notas: (dó, dó<sup>♯</sup> (dó sustenido) ou ré<sup>♭</sup> (ré bemol), ré, ré<sup>♯</sup> (ré sustenido) ou mi<sup>♭</sup> (mi bemol), mi, fá, fá<sup>♯</sup> (fá sustenido) ou sol<sup>♭</sup> (sol bemol), sol, sol<sup>♯</sup> (sol sustenido) ou lá<sup>♭</sup> (lá bemol), lá, lá<sup>♯</sup> (lá sustenido) ou si<sup>♭</sup>, si), cuja sequência chamamos *escala cromática*. Representamos essas notas na pauta da Figura 13. As sete notas naturais, sem bemol ou sustenido, tiveram seu uso estabelecido no período medieval (entre os séculos IX e XII) e as notas alteradas (bemol, sustenido) tiveram maior uso a partir do período barroco (do século XVII até metade de XVIII) [24]. Convenção essa que iremos usar ao longo dessa dissertação.

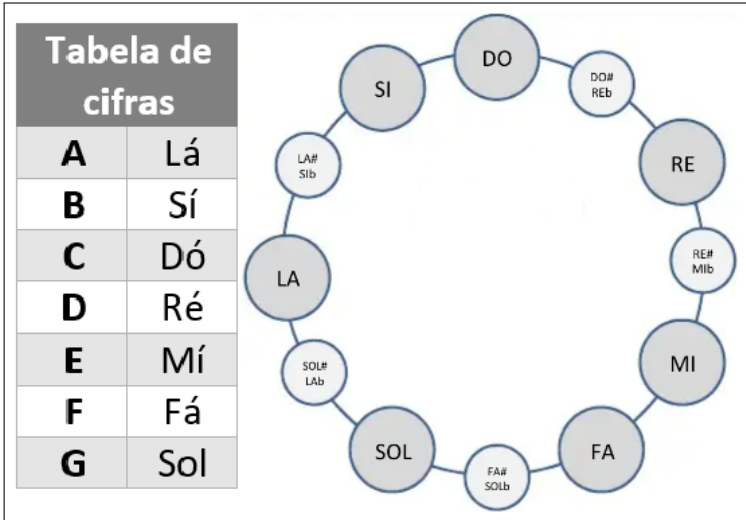
Figura 13 - Escalas Cromáticas em sequência



Legenda: Exemplo de pauta com sequência de 26 notas, semitom a semitom, começando do lá#2 e terminando no si4, totalizando duas escalas cromáticas mais duas notas. Um quadrado vermelho destaca um exemplo de escala cromática completa.

Fonte: O autor, 2024.

Figura 14 - Tabela e grafo da escala de notas musicais



Legenda: Tabela com cifras na coluna esquerda e as respectivas notas que elas representam na coluna direita. Junto a isso, no lado direito da figura, temos o grafo da escala musical com 12 vértices representando as notas, onde os vértices que são adjacentes estão a um ou onze semitons de distância (dependendo da oitava, por exemplo, si2-dó3 é 1 semitom, si3-do3 são 11).

Fonte: O autor, 2024.

**Definição 1.1.9.** *Um semitom, é a menor<sup>1</sup> distância entre duas notas nas escalas tonais e cromáticas.*

Por exemplo, as notas mi3 e fá3 estão a distância de um semitom, assim como dó3 e dó#3 (dó sustenido 3).

**Definição 1.1.10.** *Um intervalo é a distância entre duas notas simultâneas<sup>2</sup> e é contada em semitons.*

Por exemplo, o intervalo entre dó e mi é de 4 semitons, o intervalo entre ré e sol é 5.

As primeiras escalas musicais datam da Grécia Antiga e são necessárias para tocar múltiplos instrumentos simultaneamente [25]. Elas são construídas para maximizar o número de combinações de tons que, tocados simultaneamente ou em sequência, soam harmoniosos [26]. A escala que é mais amplamente utilizada até hoje é a *escala tonal*, em seus modos maior e menor. Não iremos nos aprofundar em definições da escala tonal para fins de objetividade, uma vez que ela não será usada neste trabalho. Para mais fácil enumeração das notas e formalização matemática, trabalharemos com a escala cromática, como a demonstrada na Figura 13.

**Definição 1.1.11** (Dicionário Aurélio [22]). *Consonância é o conjunto agradável de sons ou afinidade entre os sons.*

Em 2003, Bibby [2] observou que a consonância é tanto um critério psicológico e subjetivo quanto físico. Em termos físicos, quanto mais baixos forem os inteiros da razão da frequência do intervalo de um par de notas, mais consonante é o som. Por exemplo, um intervalo de quinta (duas notas com 7 semitons de distância entre elas, como dó3 e sol3) é considerado fortemente consonante já que a razão entre as frequências das notas é de aproximadamente 2 : 3. Um intervalo ainda mais consonante é o da oitava (com duas notas com 12 semitons de distância, por exemplo sol3 e sol4), com razão de 1 : 2. Por último, o intervalo mais consonante possível é o *uníssono*, com duas notas iguais tocando simultaneamente com a menor razão de inteiros possível 1 : 1.

Definimos como  $i$  o intervalo entre duas diferentes notas, um intervalo de quinta (7 semitons) é  $i^7$ , um intervalo de quarta (5 semitons) é  $i^5$ , um intervalo de segunda maior (2 semitons) é  $i^2$ , e assim por diante. Existem discussões sobre a consonância dos intervalos [27], mas neste trabalho consideramos o conjunto dos intervalos consonantes: O uníssono (0), a terça menor (3), terça maior (4), quarta justa (5), quinta justa (7), sexta menor (8), sexta maior (9) e oitava (12),  $\{i^0, i^3, i^4, i^5, i^7, i^8, i^9\} + 12n, n \in \mathbb{Z}$ . O conjunto

<sup>1</sup> Existem músicas que trabalham com microtons, que são distâncias menores que um semitom, mas não trabalharemos com microtons.

<sup>2</sup> Existem intervalos entre notas não simultâneas também, mas não serão considerados neste trabalho.

dos intervalos dissonantes é o complemento, a segunda menor, segunda maior, quarta aumentada, sétima menor e sétima maior, respectivamente  $\{i^1, i^2, i^6, i^{10}, i^{11}\} + 12n, n \in \mathbb{Z}$ . A Tabela 2 mostra os intervalos e consonâncias entre as notas. Nesse trabalho, atribuímos um valor inteiro a cada nota da seguinte forma: dó =  $0 + 12n$ , ré♭ =  $1 + 12n$ , ré =  $2 + 12n, \dots$ , si =  $11 + 12n$ , onde  $n$  é a oitava considerada. O par de notas  $\{x, y\}$  é *consonante* se  $|x - y| = 0, 3, 4, 5, 7, 8, 9(12)$ , caso contrário  $(x, y)$  é *dissonante*. Para a conveniência do leitor, nós oferecemos uma Tabela 2 com cada nota e os respectivos pares de notas dissonantes.

Tabela 2 - Tabela com notas musicais e seus intervalos.

|      | dó | ré♭ | ré | mi♭ | mi | fá | sol♭ | sol | lá♭ | lá | si♭ | si |
|------|----|-----|----|-----|----|----|------|-----|-----|----|-----|----|
| dó   | 0  | 1   | 2  | 3   | 4  | 5  | 6    | 7   | 8   | 9  | 10  | 11 |
| ré♭  | 1  | 0   | 1  | 2   | 3  | 4  | 5    | 6   | 7   | 8  | 9   | 10 |
| ré   | 2  | 1   | 0  | 1   | 2  | 3  | 4    | 5   | 6   | 7  | 8   | 9  |
| mi♭  | 3  | 2   | 1  | 0   | 1  | 2  | 3    | 4   | 5   | 6  | 7   | 8  |
| mi   | 4  | 3   | 2  | 1   | 0  | 1  | 2    | 3   | 4   | 5  | 6   | 7  |
| fá   | 5  | 4   | 3  | 2   | 1  | 0  | 1    | 2   | 3   | 4  | 5   | 6  |
| sol♭ | 6  | 5   | 4  | 3   | 2  | 1  | 0    | 1   | 2   | 3  | 4   | 5  |
| sol  | 7  | 6   | 5  | 4   | 3  | 2  | 1    | 0   | 1   | 2  | 3   | 4  |
| lá♭  | 8  | 7   | 6  | 5   | 4  | 3  | 2    | 1   | 0   | 1  | 2   | 3  |
| lá   | 9  | 8   | 7  | 6   | 5  | 4  | 3    | 2   | 1   | 0  | 1   | 2  |
| si♭  | 10 | 9   | 8  | 7   | 6  | 5  | 4    | 3   | 2   | 1  | 0   | 1  |
| si   | 11 | 10  | 9  | 8   | 7  | 6  | 5    | 4   | 3   | 2  | 1   | 0  |

Legenda: Tabela com os intervalos contados pela quantidade de semitons entre as notas. Cada elemento da tabela  $m_{a,b}$  é um número que representa  $m_{a,b} = |a - b| \bmod 12$ . Essa tabela pode ser usada da seguinte maneira, em sua primeira linha: a nota dó é consonante com as notas: dó, mi♭, mi, fá, sol lá♭ e lá, e a nota dó é dissonante com as notas: ré♭, ré, sol♭, si♭, e sí. Os números verdes representam os intervalos consonantes e os vermelhos, os dissonantes.

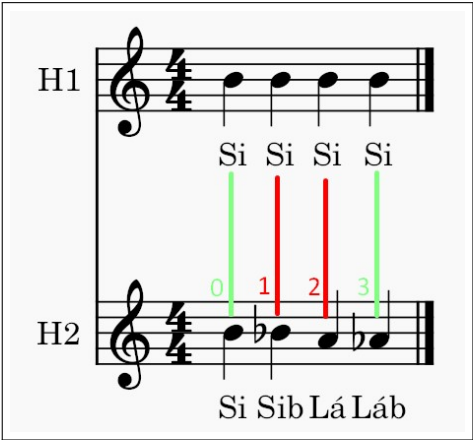
Fonte: O autor, 2021.

Mesmo músicas simples normalmente contêm alguns intervalos dissonantes, como na Figura 16, que exibe o trecho inicial de uma peça para coral de J.S. Bach.

**Definição 1.1.12** (Dicionário Aurélio [22]). *Um arranjo de uma música é uma modificação da música, quer seja na instrumentação, ou na distribuição dos instrumentos.*

Para os fins dessa dissertação, todos os arranjos considerados serão formados por um conjunto de pautas, subconjunto de uma partitura original.

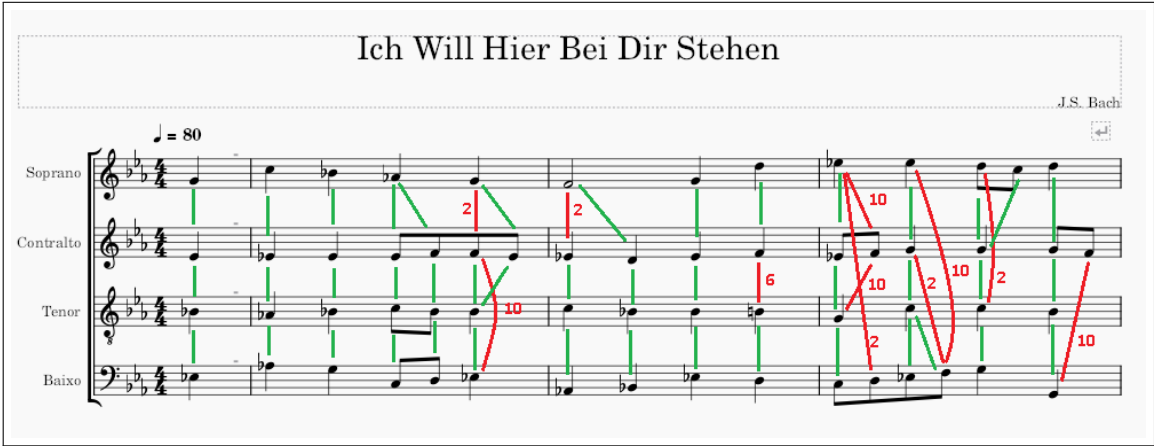
Figura 15 - Relações intervalares



Legenda: 2 Pautas com um compasso  $\frac{4}{4}$ , notas sendo tocadas simultaneamente nos 4 tempos, formando os intervalos 0, 1, 2 e 3 respectivamente. Os intervalos consonantes da cor verde (0 e 3) e os intervalos dissonantes da cor vermelha (1 e 2).

Fonte: O autor, 2024.

Figura 16 - Trecho exemplo de dissonância



Legenda: Primeiros 4 compassos de “Ich Will Hier Bei Dir Stehen”, da missa “Paixão segundo São Mateus, BWV 244” de Johann Sebastian Bach com linhas verdes indicando intervalos consonantes e linhas vermelhas indicando intervalos dissonantes. A distância em semitons de todos os pares de notas dissonantes (com linhas vermelhas), é indicada por números vermelhos, como os dispostos na Tabela 2.

Fonte: ‘O autor, 2024’.

## 1.2 Grafos

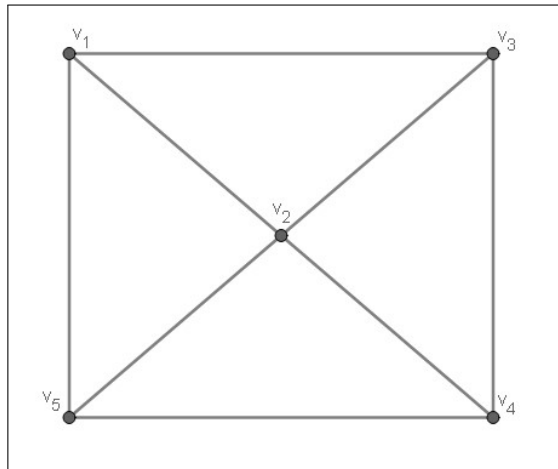
Ao longo da dissertação, utilizaremos grafos principalmente em reduções de dificuldade. Por conta disso, apresentamos nesta seção as definições pertinentes.

**Definição 1.2.1.** Um grafo  $G = (V, E)$  é um dupla, onde  $V$  consiste em um conjunto finito não-vazio de vértices e  $E$  é um conjunto de arestas formado por pares não ordenados de distintos elementos de  $V$ .

Utilizaremos a notação  $n = |V|$  e  $m = |E|$  e denotamos  $e = \{u, v\}$  por  $uv$ . Um grafo pode ser visualizado através de uma representação geométrica, na qual seus vértices correspondem a pontos distintos do plano em posições arbitrárias, enquanto que a cada aresta  $vw$  é associada uma curva arbitrária unindo os pontos correspondentes a  $v$  e  $w$  [28].

**Definição 1.2.2.** Se  $e = uv \in E$  é uma aresta, então  $e$  conecta  $u$  e  $v$ . Os vértices  $u$  e  $v$  são chamados de extremidades de  $e$ , então  $u$  e  $v$  são ditos serem vértices adjacentes. Também denotamos  $V(G)$  pelo conjunto de vértices de  $G$ .

Figura 17 - Exemplo de representação de um grafo



Legenda: Grafo  $G = (V, E)$ ,  $V = \{v_1, v_2, v_3, v_4, v_5\}$  e

$E = \{v_1v_2, v_1v_3, v_1v_5, v_2v_3, v_2v_4, v_2v_5, v_3v_4, v_4v_5\}$ .

Fonte: O autor, 2024.

**Definição 1.2.3.** Uma  $k$ -coloração de vértices de um grafo  $G = (V, E)$  é uma atribuição  $f: V \rightarrow \{1, \dots, k\}$  de  $k$  cores em  $\{1, \dots, k\}$  para os vértices de  $G$ . Uma  $k$ -coloração própria não tem 2 vértices adjacentes com a mesma cor.

**Definição 1.2.4.** Um hipergrafo é um par  $H = (V, E)$  onde  $V$  é um conjunto finito não vazio e  $E$  é uma coleção de conjuntos não vazios de  $V$ . O tamanho máximo de uma aresta  $\tau = \tau(H)$  de  $H = (V, E)$  é o maior número de vértices contido em uma aresta de  $H$ .

### 1.3 Complexidade

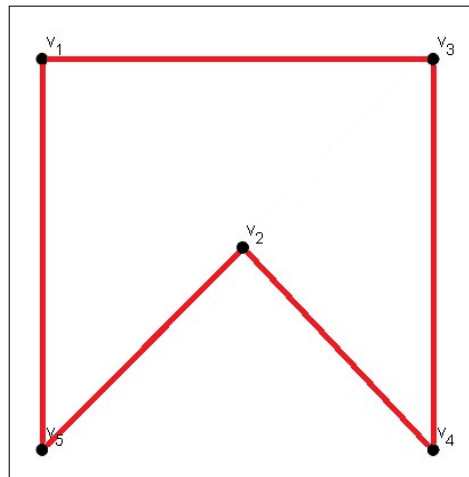
Nesta seção consideramos as classes de complexidade P e NP. Exploraremos a distinção entre problemas em P e NP. Essas definições e discussões servirão como base para a compreensão de questões mais avançadas relacionadas à complexidade computacional e à resolução eficiente de problemas computacionais.

**Definição 1.3.1.** Um problema de decisão  $\Pi = (D, Q)$  é formado por uma entrada  $D$  e uma pergunta  $Q$ , relacionada a  $D$ , cuja resposta é SIM ou NÃO.

**Definição 1.3.2.** Para um problema de decisão, um certificado para a resposta “SIM” é a prova que a resposta para a pergunta do problema é sim.

Por exemplo, o problema do CICLO HAMILTONIANO pergunta se dado um grafo  $G = (V, E)$  existe um ciclo que passa por todos os vértices do grafo. Um certificado para a resposta SIM do problema, é um subgrafo de  $G$  que é um grafo ciclo e contém todos os vértices de  $G$ . Na Figura 17 temos o exemplo de um grafo  $G$  e na Figura 18 temos o exemplo de um certificado para a resposta SIM de CICLO HAMILTONIANO considerando a instância  $G$ .

Figura 18 - Exemplo de ciclo Hamiltoniano



Legenda: Certificado para a resposta SIM de CICLO HAMILTONIANO considerando a instância  $G$  da Figura 17.

Fonte: O autor, 2024.

**Definição 1.3.3.** Seja  $A$  um algoritmo que resolve um problema de decisão. O tempo de execução de  $A$  é uma função  $f : \mathbb{N} \rightarrow \mathbb{N}$ , onde  $f(n)$  é o número máximo de passos que  $A$  usa em qualquer entrada de tamanho  $n$ .

**Definição 1.3.4.** *Seja  $t : \mathbb{N} \rightarrow \mathbb{N}$  uma função. A classe de complexidade de tempo,  $TEMPO(t(n))$ , é definida como  $TEMPO(t(n)) = \{\Pi \mid \Pi \text{ é um problema decidido por um algoritmo em tempo } O(t(n))\}$ .*

**Definição 1.3.5.** *A Classe Polinomial  $P$  é a classe de problemas de decisão decidíveis em tempo polinomial por um algoritmo. Em outras palavras,  $P = \bigcup_k TEMPO(n^k)$ .*

**Definição 1.3.6.** *A classe Não-determinística tempo Polinomial (NP) é composta por problemas de decisão para os quais, dado um certificado para a resposta SIM, é possível verificar sua validade em tempo polinomial por um algoritmo determinístico.*

**Definição 1.3.7** (Szwarcfiter [28]). *Dado um par de problemas de decisão  $\Pi_1 = (D_1, Q_1)$ ,  $\Pi_2 = (D_2, Q_2)$ , uma transformação polinomial de  $\Pi_1$  em  $\Pi_2$  é uma função  $f: D_1 \rightarrow D_2$ , tal que as seguintes condições são satisfeitas:*

1.  *$f$  é computada em tempo polinomial, e*
2. *Para toda instância  $I_1 \in D_1$  do problema  $\Pi_1$  tem-se:  $\Pi_1(I_1, Q_1)$  possui resposta SIM se e somente se  $\Pi_2(f(I_1), Q_2)$  possui resposta SIM.*

*Redução polinomial* é outro nome comumente usado para descrever uma transformação polinomial. Denota-se  $\Pi_1 \propto \Pi_2$  para indicar que existe uma transformação polinomial de  $\Pi_1$  em  $\Pi_2$ .

Transformações polinomiais são fundamentais para a compreensão da complexidade de problemas, pois permitem que sejam comparados em termos de NP-difícilidade. Isso ajuda a mapear a paisagem de complexidade, identificando problemas para os quais não existem algoritmos eficientes de resolução, a menos que  $P = NP$  [29], e orientando os esforços de pesquisa e otimização em problemas computacionais.

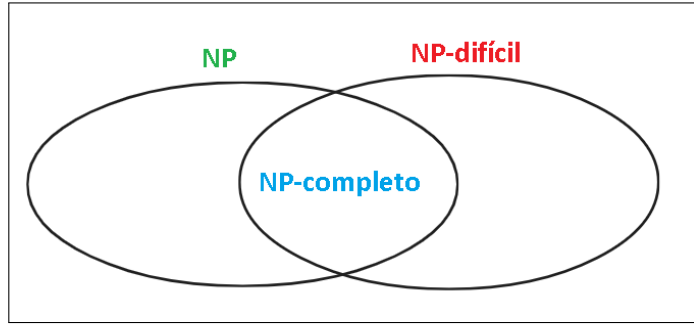
**Definição 1.3.8.** *Um problema de decisão  $\Pi = (D, Q)$  é NP-difícil se todo problema de decisão  $\Pi' \in NP$  satisfaz que  $\Pi' \propto \Pi$ . Um problema NP-difícil  $\Pi$  é NP-completo se  $\Pi \in NP$ .*

**Definição 1.3.9.** *Gadget é uma estrutura da linguagem que simula variáveis e cláusulas de formulas booleanas.*

## 1.4 Aproximação

Nem sempre é possível alcançar a melhor solução possível para um problema em tempo polinomial, devido a limitações de tempo, recursos ou complexidade. No entanto,

Figura 19 - NP-Compleitude



Legenda: Diagrama de Venn representando os conjuntos dos problemas NP no lado esquerdo, NP-difícil do lado direito, e NP-completo na interseção.

Fonte: O autor, 2024.

uma solução com custo próximo do custo do ótimo obtida em tempo polinomial pode ser de interesse. Além disso, buscar uma solução quase-ótima tende a ser significativamente mais fácil e menos dispendioso do que tentar obter uma solução ótima. Uma vez que a NP-completude de um problema é estabelecida, somos motivados a explorar possibilidades que sejam menos ambiciosas do que resolver o problema de forma exata, eficiente, em todas as ocasiões [30].

**Definição 1.4.1.** Dizemos que  $\Pi$  é um problema de otimização se para cada instância  $x$  de  $\Pi$  existe um conjunto de soluções viáveis  $F(x)$ , e para cada solução  $s \in F(x)$  temos o custo  $c(s)$ , um inteiro positivo. Seja  $\Pi$  um problema de otimização e  $x \in \Pi$  uma instância de  $\Pi$ . O problema  $\Pi$  pode ser de dois tipos: de minimização ou de maximização. A solução de custo ótimo de uma instância  $x$  de  $\Pi$  é definida como a solução  $x$  que tem custo  $OPT_{\Pi}(x) = \min_{s \in F(x)} c(s)$  se  $\Pi$  é de minimização ou  $OPT_{\Pi}(x) = \max_{s \in F(x)} c(s)$  se  $\Pi$  é um problema de maximização. A solução  $s \in F(x)$  com  $c(s) = OPT_{\Pi}(x)$  é a solução ótima de  $x$ . Um algoritmo de aproximação  $A$  para  $\Pi$  satisfaz que dada qualquer instância  $x \in \Pi$ , retorna uma solução viável  $A(x) \in F(x)$  de custo  $c(A(x))$ . Dizemos que  $A$  é um algoritmo exato para  $\Pi$  se para toda instância  $x$  de  $\Pi$ ,  $OPT_{\Pi}(x) = c(A(x))$ .

**Definição 1.4.2.** Dado  $\Pi$  um problema de otimização e  $\varepsilon > 0$ . Dizemos que  $A$  é um algoritmo  $\varepsilon$ -aproximativo para  $\Pi$ , ou que  $A$  é uma  $\varepsilon$ -aproximação para  $\Pi$  se para toda instância  $x \in \Pi$ , 
$$\frac{|c(A(x)) - OPT_{\Pi}(x)|}{\max\{OPT_{\Pi}(x), c(A(x))\}} \leq \varepsilon.$$

**Teorema 1.4.1.** Se  $A$  é um algoritmo  $\varepsilon$ -aproximativo para  $\Pi$  e  $x$  é uma instância de  $\Pi$ , então 
$$\frac{|OPT_{\Pi}(x) - c(A(x))|}{OPT_{\Pi}(x)} \leq \varepsilon.$$

*Demonstração.* Suponha que  $A$  é um algoritmo  $\varepsilon$ -aproximativo para  $\Pi$ . Então pela definição, para toda instância  $x \in \Pi$ , 
$$\frac{|c(A(x)) - OPT_{\Pi}(x)|}{\max\{OPT_{\Pi}(x), c(A(x))\}} \leq \varepsilon.$$
 Analisamos 2 casos.

1. Se  $\Pi$  é de maximização, então  $\frac{|OPT_{\Pi}(x) - c(A(x))|}{OPT_{\Pi}(x)} \leq \varepsilon$ .
2. Se  $\Pi$  é de minimização, então  $\frac{|OPT_{\Pi}(x) - c(A(x))|}{OPT_{\Pi}(x)} \leq \frac{|OPT_{\Pi}(x) - c(A(x))|}{c(A(x))} \leq \varepsilon$

□

**Definição 1.4.3.** *Dado um problema de otimização  $\Pi$ . Um esquema de aproximação tempo polinomial (EATP) para  $\Pi$  é um algoritmo  $A_{\varepsilon}$  que dada uma instância  $x$  de  $\Pi$  para cada  $\varepsilon > 0$  retorna uma  $\varepsilon$ -aproximação para  $\Pi$ .*

**Definição 1.4.4.** *Dados dois problemas de otimização  $A$  e  $B$ . Dizemos que  $A$   $L$ -reduz a  $B$  se existirem dois algoritmos em tempo polinomial  $f$  e  $g$  e constantes positivas  $\alpha$  e  $\beta$ , tal que para cada instância  $I_A$  de  $A$ , onde*

1. *O algoritmo  $f$  produz uma instância  $I_B = f(I_A)$  de  $B$ , tal que  $OPT_B(I_B) \leq \alpha OPT_A(I_A)$ ;*
2. *Dada qualquer solução viável  $s_B$  de  $I_B$  com custo  $c_B$ , o algoritmo  $g$  produz uma solução  $s_A$  de  $I_A$  com custo  $c_A$ , tal que  $|c_A - OPT_A(I)| \leq \beta |c_B - OPT_B(I_B)|$ .*

**Teorema 1.4.2.** *Se  $A$   $L$ -reduz para  $B$  e  $B$  tem um EATP então  $A$  tem um EATP.*

**Teorema 1.4.3.** *Se existe um EATP para um problema MAX-SNP-difícil, então  $P=NP$ .*

**Teorema 1.4.4.** *Se  $P \neq NP$ ,  $A$  é MAX-SNP-difícil e  $A$   $L$ -reduz para  $B$ , então não existe EATP para  $B$ .*

## 1.5 Complexidade Parametrizada

A Complexidade Parametrizada objetiva considerar problemas computacionais, analisando a complexidade não apenas em função do tamanho da entrada, mas também em função de outros parâmetros da instância. Nesta seção destacamos as definições fundamentais relacionadas à Complexidade Parametrizada.

**Definição 1.5.1.** *Alfabeto é um conjunto finito não vazio.*

**Definição 1.5.2.** *Um problema de decisão parametrizado  $\Pi$  é um conjunto de instâncias contido em  $\Sigma^* \times \mathbb{N}$  para algum alfabeto finito não vazio  $\Sigma$  e  $(x, k)$  é uma instância de  $\Pi$ , onde  $x$  é a parte principal e  $k$  o parâmetro e o conjunto de instâncias é particionado no subconjunto de instâncias SIM e no subconjunto de instâncias NÃO para  $\Pi$ .*

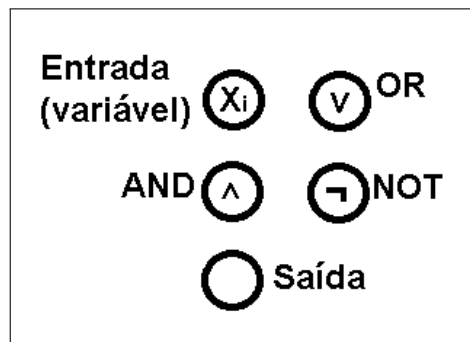
**Definição 1.5.3.** *Sejam  $A$  e  $B$  problemas parametrizados, dizemos que  $A$  reduz para  $B$  por uma redução parametrizada padrão se houver funções computáveis  $f, g$  e uma constante  $c$ , tal que  $g(k) \geq k'$ , a instância  $(x', k')$  é construída em tempo  $f(k) \cdot |x|^c$  e  $(x, k)$  é uma instância SIM de  $A$  se e somente se  $(x', k')$  é uma instância SIM de  $B$ .*

**Definição 1.5.4.** *Um problema parametrizado  $A$  é tratável por parâmetro fixo se existe um algoritmo que decida se a entrada  $(x, k)$  é uma instância SIM de  $A$  em tempo  $f(k)n^\alpha$ , onde  $n = |x|$ ,  $\alpha$  é uma constante (independente de  $n$  e de  $k$ ) e  $f$  é uma função computável arbitrária.*

**Definição 1.5.5.** *Um circuito booleano de decisão consiste em portas de entrada, portas de negação, portas AND, portas OR e uma única porta de saída. O peso de Hamming ou peso de uma atribuição é o número de valores verdadeiros nas portas de entrada.*

Quando utilizarmos portas lógicas no trabalho, elas serão representadas como na Figura 20.

Figura 20 - Portas Lógicas



Legenda: Desenhos de diferentes portas lógicas: A porta de entrada é a que possui a variável  $X_i$  escrita dentro do círculo. A porta OR tem  $\vee$  escrito dentro do círculo. A porta AND tem  $\wedge$  escrito e a porta NOT tem  $\neg$  escrito dentro do círculo.

Fonte: O autor, 2024.

O problema de decisão associando o peso de um circuito é definido da seguinte maneira:

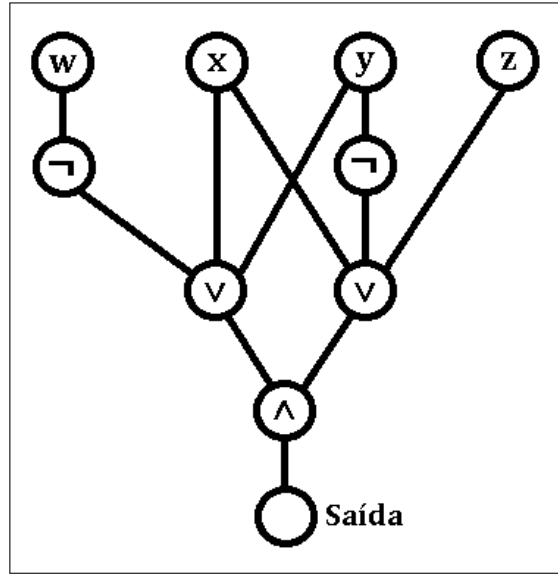
**PESO DA SATISFIABILIDADE DE UM CIRCUITO**

**Instância:** Um circuito Booleano  $C$  e um inteiro positivo  $k$ .

**Pergunta:** Existe uma atribuição de verdade de peso  $k$ , isto é com exatamente  $k$  valores de verdade para as portas de entrada, que torna a saída verdadeira?

**Definição 1.5.6.** *A profundidade de um circuito é o comprimento máximo de um caminho desde uma porta de entrada até a porta de saída.*

Figura 21 - Exemplo de Circuito Booleano



Legenda: Exemplo de circuito booleano associado à fórmula

$\phi = (\bar{w} \vee x \vee y) \wedge (x \vee \bar{y} \vee z)$ . Observe ue a fórmula tem uma atribuição de verdade satisfatível  $f: \{w, x, y, z\} \rightarrow \{V, F\}$  se e somente se  $F$  dá saída  $V$ .

Fonte: O autor, 2024.

Na Figura 21 o circuito tem profundidade 4, porque onde o caminho passa pela porta NOT, passando por quatro portas considerando a porta de saída. Na Figura 22b o circuito tem profundidade 3, todos os caminhos tem comprimento 3.

**Definição 1.5.7.** *Uma porta é grande se tiver mais de 2 entradas. O entrelaçamento de um circuito é o número máximo de portas grandes no caminho de uma porta de entrada até a porta da saída.*

Na Figura 21 temos um exemplo de circuito com entrelaçamento 1, pois qualquer caminho passa por exatamente uma porta grande (a porta OR). Na Figura 22b temos um exemplo onde o entrelaçamento é 2, pois todos os caminhos passam por 2 portas grandes.

Existe uma versão particular do problema PESO DA SATISFIABILIDADE DE UM CIRCUITO.

PESO DA SATISFIABILIDADE DE UM CIRCUITO COM ENTRELAÇAMENTO  $t$   
E PROFUNDIDADE  $h$

**Instância:** Um circuito Booleano  $C$  com entrelaçamento  $t$  e profundidade  $h$  e um inteiro positivo  $k$ .

**Pergunta:** Existe uma atribuição de verdade de peso  $k$ , isto é com exatamente  $k$  valores de verdade para as portas de entrada, que torna a saída verdadeira?

**Definição 1.5.8.** Um problema  $\Pi$  está na classe  $W[t]$  se houver uma constante  $h$  e uma redução parametrizada de  $\Pi$  para o problema PESO DA SATISFIABILIDADE DE UM CIRCUITO COM ENTRELACAMENTO  $t$  E PROFUNDIDADE  $h$ .

**Definição 1.5.9.** Um problema de decisão parametrizado  $\Pi$  é  $W[t]$ -difícil se para cada problema de decisão parametrizado  $\Pi_o$  em  $W[t]$ , há uma redução parametrizada de  $\Pi_o$  para  $\Pi$ .  $\Pi$  é  $W[t]$ -completo se  $\Pi \in W[t]$  e  $\Pi$  é  $W[t]$ -difícil.

Dado o problema de decisão CONJUNTO DOMINANTE.

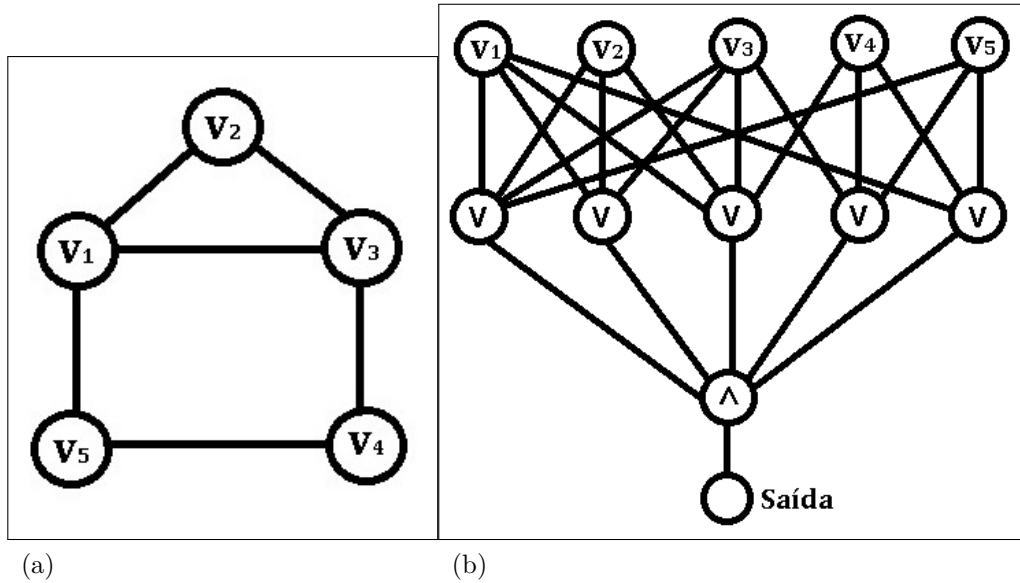
CONJUNTO DOMINANTE

**Instância:** Grafo  $G = (V, E)$  e inteiro positivo  $k$ .

**Pergunta:**  $G$  contém um conjunto dominante  $S \subset V$  de tamanho  $|S| \leq k$ , isto é para todo  $u \in V$  ou  $u \in S$  ou existe  $v \in S$ , tal que  $uv \in E$ ?

A Figura 22 descreve um circuito lógico que mostra que o problema CONJUNTO DOMINANTE está em  $W[2]$ .

Figura 22 - Exemplo de Circuito Booleano associado ao problema Conjunto Dominante



Legenda: A imagem (a) representa o grafo  $G = \{V, E\}$ , com  $V = (v_1, v_2, v_3, v_4, v_5)$  e  $E = \{(v_1, v_2), (v_1, v_3), (v_1, v_5), (v_2, v_3), (v_3, v_4), (v_4, v_5)\}$ . A imagem (b) representa o circuito lógico relativo ao problema CONJUNTO DOMINANTE para a instância  $G$ . Veja que  $S \subset V$  é um conjunto dominante com  $|S| \leq k$  para  $G$  se e somente se o circuito lógico tem saída verdadeira se a entrada tem uma atribuição com peso de Hamming no máximo  $k$  cuja porta de saída tem valor  $V$ . Em particular,  $S$  é um conjunto dominante com  $|S| \leq k$  se e somente se  $v_i = V$  dando saída  $V$ .

Fonte: O autor, 2024.

## 2 TRABALHOS RELACIONADOS

Neste capítulo, faremos uma revisão bibliográfica de trabalhos relacionados ao tema desta dissertação, sobretudo ao capítulo fundamental de Demaine e Moses [1], que motiva a criação dos novos problemas de decisão e aproximação, bem como dos resultados deles provenientes.

### 2.1 Contextualização de estudos matemáticos de música

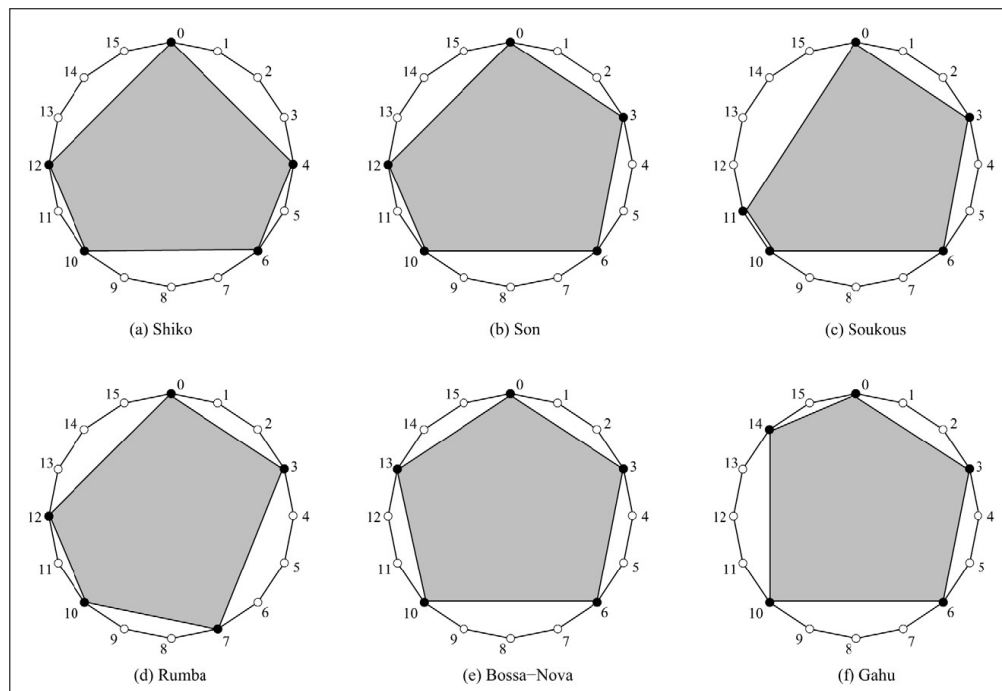
Fauvel et al. [31] investigam a relação entre música e matemática ao longo da história com foco em afinação e temperamento. Discutem tópicos como a ciência do som musical, a consonância e como e quando estas foram desenvolvidas. Exploram as estruturas matemáticas encontradas na música, apontando padrões em composições. Finalmente, dissertam, exemplificando com composições, sobre tópicos como microtons e composição com fractais.

Ferreira e Machado [3] descrevem algoritmos utilizados para a composição musical, desde os algoritmos criados antes do computador, como a utilização de “cânone” (que são composições com múltiplas vozes ou instrumentos em que uma melodia é imitada por um ou mais músicos, começando em diferentes momentos, de modo que a melodia original e suas imitações sejam audíveis simultaneamente criando uma harmonia), repetições com variações, dentre outras técnicas para auxiliar na criação musical, até os algoritmos que vieram depois do computador, usando métodos estocásticos, determinísticos, caóticos (que utilizam o princípio do caos e fractais) e de inteligência artificial (algoritmos genéticos, redes neurais e etc).

Toussaint [17] estudou técnicas e definiu uma nova medida para a igualdade entre ritmos. Toussaint discutiu as vantagens da igualdade entre ritmos sobre medidas antigas por discriminar entre linhas rítmicas. A igualdade entre os ritmos pode ser computada de forma mais eficiente. Vários problemas rítmicos em aberto são discutidos.

Demaine et al. [19] estudaram as relações entre o algoritmo Euclidiano clássico e muitas outras áreas de estudo, particularmente no contexto da música e da geometria de distância. Especificamente, os autores mostraram como a estrutura do algoritmo Euclidiano define uma família de ritmos que abrange mais de quarenta linhas de tempo (ostinatos) da música tradicional mundial.

Figura 23 - Ostinatos



Legenda: Seis dos ritmos fundamentais africanos e latino-americanos demonstrados no artigo, todos com a mesma soma de distâncias geodésicas em pares; no entanto, intuitivamente, o ritmo Bossa Nova é mais “uniforme” do que os demais, ou seja, o intervalo de tempo entre as articulações tem a mesma duração.

Fonte: ‘Demaine, 2009’.

## 2.2 A complexidade de tempo de arranjo musical

Demaine e Moses [1] consideraram o problema de decisão onde é dada uma partitura de entrada formada por uma coleção de pautas, indaga-se sobre a existência de um arranjo formado por um subconjunto de pautas sujeito a algumas restrições. Existem as restrições *universais*, que representam restrições sobre arranjos musicais em geral e existem as restrições *específicas*, as quais representam limitações para o artista que toca o arranjo e podem ser diferentes para instrumentos diferentes ou artistas diferentes.

São consideradas duas restrições universais:

1. Cada parte individual, pauta, é incluída ou excluída inteiramente.
2. Em qualquer tempo, pelo menos  $p\%$  (dado como entrada) das notas da partitura da música original precisam ser tocadas pelo arranjo formado.

Demaine e Moses trabalham com três restrições específicas em três problemas de decisão diferentes:

- Nenhum par de notas sendo tocadas simultaneamente está em dissonância.
- O número de notas simultâneas no arranjo não é maior que  $j$ , para uma constante inteira  $j \geq 1$  fixada.
- Cada nota da partitura é tocada continuamente por pelo menos  $r$  tempos,  $r \geq 1$  fixado. E em cada tempo do arranjo formado, nenhum par de notas em duas pautas do arranjo é tocado simultaneamente.

A primeira restrição universal, onde cada pauta é incluída ou excluída inteiramente, pode parecer muito limitante em adição com as restrições específicas. Mas a divisão de melodias em diferentes pautas torna os problemas menos restritivos [1] como é demonstrado na Figura 31.

Demaine e Moses [1] classificaram alguns casos nos quais os 3 problemas associados são polinomiais e NP-completos.

Descrevemos formalmente os problemas de Demaine e Moses a seguir.

### 2.2.1 Problema do Arranjo Consonante

A dissonância musical é estrutural e fundamental no estado-da-arte de muitas composições importantes. Dessa forma, a obtenção de uma partitura contendo pautas

sem dissonâncias não define, a princípio, um método eficaz de produção de uma música de excelência.

Recentemente, Demaine e Moses [1] em 2017 propuseram a construção de arranjos sem dissonâncias. Os autores mostraram que o problema geral é NP-completo e mostraram diversos casos em que o problema podia ser resolvido em tempo polinomial.

Além do incremento teórico ao estado-da-arte da literatura, o estudo da teoria dos arranjos sem dissonâncias, traz uma possível aplicação no paradigma de Demaine e Moses como uma ferramenta para os arranjadores com a obtenção de uma partitura sem dissonâncias como ponto de partida para obtenção de uma sequência de outras composições que introduzissem gradualmente as dissonâncias.

Como um exemplo do mundo real da aplicação do método apresentamos na Figura 24 uma partitura contendo um trecho de uma música de Pixinguinha e na Figura 25 um arranjo sem dissonâncias obtido a partir da partitura original.

Figura 24 - Trecho de arranjo da música Carinhoso como exemplo de instância de ARRANJO CONSONANTE

The image shows a musical score for the piece 'Carinhoso' by Pixinguinha. It consists of five staves: Flute (Fl.), Violin 1 (Vln. 1), Violin 2 (Vln. 2), Viola (Vla.), and Cello/Double Bass (Vc. Cb.). Red lines connect notes across different staves to indicate dissonances. Red numbers (1, 2, 6) are placed near the dissonant pairs, indicating the number of semitones between them. For example, in the first measure, there is a dissonance between the Flute and Violin 1 (1 semitone), and between the Violin 2 and Cello/Double Bass (2 semitones). In the second measure, there is a dissonance between the Flute and Violin 1 (6 semitones), and between the Violin 2 and Cello/Double Bass (2 semitones). In the third measure, there is a dissonance between the Flute and Violin 1 (1 semitone), and between the Violin 2 and Cello/Double Bass (6 semitones). In the fourth measure, there is a dissonance between the Flute and Violin 1 (6 semitones), and between the Violin 2 and Cello/Double Bass (2 semitones). In the fifth measure, there is a dissonance between the Flute and Violin 1 (1 semitone), and between the Violin 2 and Cello/Double Bass (6 semitones).

Legenda: Trecho de um arranjo da música “Carinhoso”, de Pixinguinha como instância  $M$  com  $p = 50\%$ . As linhas vermelhas indicam quais pautas tem pares de notas dissonantes, significando que não pode estar em um mesmo arranjo. Os números vermelhos indicam a quantidade de semitons de distância entre os pares de notas dissonantes, como dispostos na Tabela 2.

Fonte: Site Musica Brasilis, acessado em 2024, disponível em:

<<https://musicabrasilis.org.br/partituras/pixinguinha-carinhoso>>

No problema ARRANJO CONSONANTE [1] uma entrada consiste de uma partitura com  $n$  pautas e um número  $p, 0 \leq p \leq 100$ . A pergunta é se existe um arranjo formado por uma coleção de pautas com as seguintes propriedades:

1. Cada parte individual, pauta, é incluída ou excluída na íntegra no arranjo.

Figura 25 - Arranjo para a instância  $M$  de ARRANJO CONSONANTE

Legenda: O arranjo da instância  $M$  na Figura 24 com  $p = 50\%$  seleciona as pautas do violino 1, violino 2 e contrabaixo para o arranjo. Como todos os pares de notas são consonantes e pelo menos 50% das notas de  $M$  são tocadas em todos os tempos, o arranjo serve como um certificado para a resposta SIM da instância  $M$  de ARRANJO CONSONANTE

Fonte: Site Musica Brasilis, acessado em 2024, disponível em:

<<https://musicabrasilis.org.br/partituras/pixinguinha-carinhoso>>

2. Em qualquer tempo, pelo menos  $p\%$  notas da partitura da música original precisam ser tocadas.
3. Em qualquer tempo nenhum par de pautas tem notas dissonantes sendo tocadas.

A primeira propriedade estabelece que cada pauta deve ser inteiramente incluída ou excluída do arranjo, enquanto a terceira exige que não haja pares de notas dissonantes. Isso implica que, se duas pautas tocam um par de notas simultâneas e dissonantes, basta essa condição para garantir que o arranjo não possa conter ambas. Na Figura 24, é apresentada uma partitura como instância para o problema do ARRANJO CONSONANTE, com linhas vermelhas indicando os pares de notas dissonantes, o que também mostra quais pautas não podem ser incluídas juntas no arranjo. Já na Figura 25, é mostrado um arranjo dessa instância do ARRANJO CONSONANTE.

### 2.2.2 Problema da Limitação no Número de $j$ Notas Simultâneas

O termo *mixórdia* define uma mistura confusa de coisas variadas. A mixórdia musical pode então ser vista como uma mistura desorganizada de diferentes ritmos, notas, ou instrumentos.

Um princípio ingênuo para evitar a mixórdia musical é imaginar uma partitura que em cada tempo tocam sempre um número reduzido de notas.

Demaine e Moses [1] imaginaram a produção de arranjos na qual um conjunto de pautas era selecionado em uma partitura de modo que em cada tempo somente um número limitado de notas era tocado. Em sua definição, os autores exigiam que pelo menos um percentual das notas da partitura original fosse tocada em cada tempo de modo a garantir a verosimilhança com a partitura original. Os autores demonstraram que, apesar do resultado desfavorável, o problema geral é NP-completo, embora tenham mostrado alguns casos em que o problema pode ser resolvido em tempo polinomial.

Para a conveniência do leitor nós mostramos na Figura 26 uma aplicação do método com um trecho de uma partitura de uma música de Pixinguinha e na Figura 27 um arranjo obtido a partir da partitura original.

Figura 26 - Trecho de arranjo da música Carinhoso como exemplo de instância de LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS

Legenda: Trecho de um arranjo da música “Carinhoso”, de Pixinguinha como instância  $M$  com  $p = 30\%$  e  $j = 3$ .

Fonte: Site Musica Brasilis, acessado em 2024, disponível em:

<<https://musicabrasilis.org.br/partituras/pixinguinha-carinhoso>>

Na instância  $M$  da Figura 26, temos  $p = 30\%$  e  $j = 3$ . No primeiro tempo, como  $p = 30\%$ , pelo menos uma das duas pautas tocando deve estar presente no arranjo. No último tempo, onde são tocadas 8 notas simultâneas, o valor de  $j = 3$  implica que apenas algumas combinações de pautas podem ser escolhidas para o arranjo. A Figura 27 mostra um dos possíveis arranjos obtido a partir da partitura original da Figura 26.

No problema LIMITAÇÃO NO NÚMERO DE  $j$  NOTAS SIMULTÂNEAS [1] uma entrada consiste de uma partitura com  $n$  pautas, um número  $p, 0 \leq p \leq 100$  e um inteiro  $j \geq 1$ . A pergunta é se existe um arranjo formado por uma coleção de pautas com as seguintes propriedades:

1. Cada parte individual, pauta, é incluída ou excluída na íntegra no arranjo.

Figura 27 - Arranjo para a instância  $M$  de LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS

Legenda: O arranjo da instância  $M$  na Figura 26 com  $p = 30\%$  e  $j = 3$  seleciona as pautas do violino 1 e violoncelo para o arranjo. Como em nenhum momento são tocadas mais do que  $j$  notas simultâneas e pelo menos 30% das notas de  $M$  são tocadas em todos os tempos, o arranjo serve como um certificado para a resposta SIM da instância  $M$  de LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS

Fonte: Site Musica Brasilis, acessado em 2024, disponível em:

<<https://musicabrasilis.org.br/partituras/pixinguinha-carinhoso>>

2. Em qualquer tempo, pelo menos  $p\%$  notas da partitura da música original precisam ser tocadas.
3. Em cada tempo o número de notas simultaneamente no arranjo não é maior que  $j$ .

### 2.2.3 Problema da Velocidade de Transição Limitada

Arranjadores frequentemente levam em consideração a praticidade de seus arranjos. A inclusão indiscriminada de muitas notas pode tornar o arranjo extremamente difícil ou até mesmo impossível de ser executado no instrumento escolhido.

O problema da limitação na velocidade de transição em arranjos musicais surge quando se busca controlar não apenas a inclusão de pautas, mas também a introdução das notas. Esse problema é pensado para um cenário onde a música original inclui várias pautas, representando diferentes instrumentos, e o arranjo é feito para apenas um instrumento (ainda que com múltiplas pautas). Isso pode gerar dificuldades, pois a adição de diversas pautas para um único instrumento pode tornar a execução inviável, devido à velocidade necessária para o instrumento tocar todas as notas do arranjo.

Demaine e Moses [1] propõem uma solução para evitar esse problema, que consiste em estabelecer, além de um percentual mínimo das notas da partitura original a serem tocadas, um valor mínimo  $t$  para a duração de uma nota ou acorde tocado isoladamente.

Esse valor  $t$  impede que muitas notas sejam tocadas em rápida sucessão no arranjo por um único instrumento, evitando assim que a execução se torne impraticável.

Para ilustrar o método, mostramos na Figura 28 um trecho de uma partitura de uma composição de Heitor Villa-Lobos. Na Figura 30, apresentamos um arranjo obtido a partir dessa partitura original.

No problema VELOCIDADE DE TRANSIÇÃO LIMITADA [1] uma entrada consiste de um inteiro positivo  $j \geq 1$  e uma partitura com  $n$  pautas, na qual cada nota é tocada continuamente em pelo menos  $j$  tempos e um percentual  $p$ ,  $0 \leq p\% \leq 1$ . A pergunta é se existe um arranjo formado por uma coleção de pautas com as seguintes propriedades:

1. Cada parte individual, pauta, é incluída ou excluída na íntegra no arranjo.
2. Em qualquer tempo, pelo menos  $p\%$  notas da partitura da música original precisam ser tocadas.
3. Nenhuma nota começa a ser tocada durante a duração  $t$  de outra nota.

Figura 28 - Trecho de arranjo da música ‘‘Trenzinho do Caipira’’, como exemplo de instância de LIMITAÇÃO NA VELOCIDADE DE TRANSIÇÃO



Legenda: Trecho de um arranjo da música ‘‘Trenzinho do Caipira’’, como instância  $M$  com  $p = 30\%$  e  $t = \frac{1}{2}$ .

Fonte: Site Musescore, acessado em 2024, disponível em:

<<https://musescore.com/user/53385732/scores/11406769>>

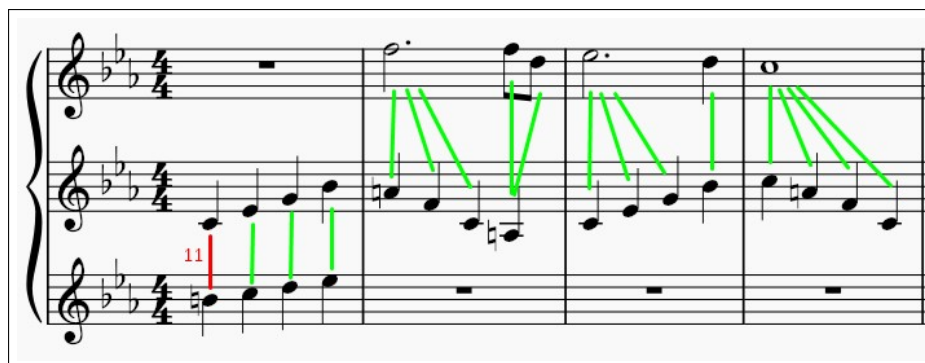
Na Figura 28, porque  $p = 30\%$ , pelo menos uma das duas pautas precisa ser escolhida. Considerando  $t = \frac{1}{2}$ , há um conflito no primeiro e terceiro compassos onde notas são tocadas em um intervalo de tempo menor que  $t$ . O conflito do terceiro compasso é demonstrado de forma visual para melhor entendimento do leitor na Figura 29. A Figura 30 mostra um arranjo que serve como um certificado para a resposta SIM da instância  $M$  de VELOCIDADE DE TRANSIÇÃO LIMITADA.



Figura 31 - Divisão de melodias em pautas



(a)



(b)

Legenda: Na entrada (31a) de ARRANJO CONSONANTE, um arranjo válido precisa descartar 50% das notas originais. Na entrada (31b), a pauta  $H_1$  de (31a) é dividida em  $H_1$  e  $H_3$ , possibilitando um arranjo consonante onde só é descartado um compasso da música original.

Fonte: O autor, 2024.

Demaine e Moses [1] estabeleceram a complexidade de tempo dos três problemas de decisão classificando diversos casos de cada um dos problemas em NP-completo ou polinomial de acordo com os valores de  $j$  e  $p$ .

No artigo original, Demaine e Moses produziram a seguinte tabela:

Tabela 3 - Tabela com restrições sobre a complexidade de organizar música.

| Variantes do problema                                                     | Complexidade |
|---------------------------------------------------------------------------|--------------|
| Para todo $p = 100$                                                       | P            |
| Para todo $p = 0$                                                         | P            |
| ARRANJO CONSONANTE ( $0 < p\% < 1$ )                                      | NPC          |
| VELOCIDADE DE TRANSIÇÃO LIMITADA ( $0 < p\% < 1$ )                        | NPC          |
| NÚMERO LIMITADO DE $j$ NOTAS SIMULTÂNEAS ( $0 < p\% \leq \frac{j}{j+2}$ ) | NPC          |
| NÚMERO LIMITADO DE $j$ NOTAS SIMULTÂNEAS ( $p\% > \frac{j}{j+1}$ )        | P            |
| NÚMERO LIMITADO DE 1 NOTA ( $p\% > \frac{1}{3}$ )                         | P            |

Legenda: Tabela resumando problemas propostos por Demaine e Moses [1]. P denota que o problema pode ser resolvido em tempo polinomial. NPC significa que o problema é NP-completo.

Fonte: Demaine e Moses [1], 2017.

As provas de NP-completude para ARRANJO CONSONANTE e VELOCIDADE DE TRANSIÇÃO LIMITADA são obtidas a partir de uma redução de 3SAT e para NÚMERO LIMITADO DE  $j$  NOTAS SIMULTÂNEAS é obtida a partir de uma redução de UM-EM-TRÊS 3SAT(X3SAT). Esses dois problemas de decisão tem a seguinte definição:

**Problema 2.2.1.** 3-SATISFABILIDADE (3SAT)

**Instância:** *Fórmula*  $(U, C)$ , onde  $U = \{u_1, u_2, \dots, u_n\}$  é um conjunto de variáveis lógicas e  $C = \{c_1, c_2, \dots, c_m\}$  é uma coleção de cláusulas disjuntivas sob  $U$ , tal que  $|c_i| = 3$ ,  $c_i \in C$ .

**Pergunta:** *Existe uma atribuição de verdade*  $\xi: U \rightarrow \{V, F\}$  para  $U$  que satisfaz cada cláusula  $c$  de  $C$ , ou seja, que para cada cláusula  $c \in C$  com  $c = (x_i \vee x_j \vee x_k)$  existe um  $\ell \in \{i, j, k\}$  tal que  $\xi(x_\ell) = V$ ?

**Problema 2.2.2.** EXATAMENTE 1 EM 3 SAT (X3SAT)

**Instância:** *Fórmula*  $(U, C)$ , onde  $U = \{u_1, u_2, \dots, u_n\}$  é um conjunto de variáveis lógicas e  $C = \{c_1, c_2, \dots, c_m\}$  é uma coleção de cláusulas disjuntivas sob  $U$ , tal que  $|c_i| = 3$ ,  $c_i \in C$ .

**Pergunta:** *Existe uma atribuição de verdade*  $\xi: U \rightarrow \{V, F\}$  para  $U$  onde cada cláusula  $c$  de  $C$  tem exatamente um literal verdadeiro?

### 3 PROBLEMAS TRABALHADOS

Os problemas de decisão sobre arranjo musical ARRANJO CONSONANTE, LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS e LIMITAÇÃO NA VELOCIDADE DE TRANSIÇÃO, foram introduzidos por Demaine e Moses [1]. Neste capítulo descrevemos generalizações dos 3 problemas originais, que vão nos permitir introduzir problemas de otimização desenvolvidos durante o trabalho de dissertação de mestrado, MAX ARRANJO CONSONANTE, MAXJSIM e MAX VELOCIDADE DE TRANSIÇÃO LIMITADA, bem como os resultados obtidos para os mesmos.

Também são propostos dois novos problemas: o problema ARRANJO PARA  $k$  INSTRUMENTOS e o problema PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS. Ambos são demonstrados como NP-completos por meio de reduções polinomiais para problemas já conhecidos como NP-completos.

#### 3.1 Problema do Máximo Arranjo Consonante

A generalização ARRANJO CONSONANTE GERAL que propomos nesta seção permite estender, de forma natural, o problema de decisão ARRANJO CONSONANTE para uma versão de otimização com o resultado de não existência de um esquema de aproximação tempo polinomial, assumindo que  $P \neq NP$ .

O problema de decisão ARRANJO CONSONANTE pergunta se dada uma música de entrada, podemos apenas com a remoção de pautas, criar um arranjo sem dissonâncias e com pelo menos  $p\%$  das notas da música original. Foi introduzido originalmente por Demaine e Moses [1] e descrito no capítulo anterior. A Figura 16 apresenta o exemplo de uma coleção de pautas e onde estão suas dissonâncias e consonâncias.

Mesmo as músicas mais simples (como na Figura 16) costumam ter algum intervalo dissonante, e a criação de um arranjo totalmente livre de intervalos dissonantes geralmente não é o objetivo de um arranjador. No entanto, a viabilidade de tal arranjo pode ser uma informação valiosa. Em uma versão generalizada do problema, é possível admitir uma quantidade limitada de tempos dissonantes no arranjo, o que torna a solução mais viável e desejável. Além disso, o problema generalizado permite a criação de uma versão de otimização que avalia o quão consonante um arranjo para a música dada como entrada pode ser.

A seguir, nós propomos uma versão generalizada de ARRANJO CONSONANTE, a qual permite utilizando seus parâmetros, a criação de uma versão de maximização natural do mesmo problema.

### ARRANJO CONSONANTE GERAL (ARR-CON)

**Instância:**  $M = (\mathcal{H}, p, k)$  onde  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  é uma coleção de pautas,  $0 \leq p\% \leq 1$  um percentual e um inteiro  $k \geq 0$ .

**Pergunta:** Existe um subcoleção  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq s$ , tal que as pautas de  $\mathcal{H}'$  não tem pares de notas dissonantes e em pelo menos  $k$  tempos, pelo menos  $p\%$  das notas de  $\mathcal{H}$  são tocadas?

Nós observamos que uma instância  $I = (\mathcal{H}, p, s, k)$  de ARRANJO CONSONANTE GERAL particular quando  $k$  é o número de tempos total de  $\mathcal{H}$  com  $s = 0$  é equivalente a uma instância  $(\mathcal{H}, p)$  do problema ARRANJO CONSONANTE de Demaine e Moses.

Demaine e Moses [1] demonstraram que o problema ARRANJO CONSONANTE é NP-completo. O problema ARRANJO CONSONANTE GERAL é uma generalização do problema original e, portanto, também é NP-completo. Ainda assim, faremos uma redução do problema generalizado, que será utilizada na subseção 3.1.1 para abordar a versão de otimização.

**Teorema 3.1.1.** ARRANJO CONSONANTE GERAL (ARR-CON) é NP-completo.

*Demonstração.* Dada uma instância  $(\mathcal{H}, p, s, k)$  de ARR-CON. ARR-CON está em NP, pois um certificado  $\mathcal{H}' \subset \mathcal{H}$  pode ser verificado em tempo polinomial checando se  $|\mathcal{H}'| \geq s$  e se pelo menos  $p\%$  das notas de  $\mathcal{H}$  são tocadas em pelo menos  $k$  tempos.

Vamos usar o problema NP-completo MAX3SAT<sub>3</sub> [30] para provar que MAX3SAT<sub>3</sub>  $\propto$  ARR-CON.

### MAX3SAT COM NO MÁXIMO TRÊS OCORRÊNCIAS POR VARIÁVEL (MAX3SAT<sub>3</sub>)

**Entrada:** Fórmula  $(U, C)$ , onde  $U = \{u_1, \dots, u_n\}$  é um conjunto de variáveis lógicas e  $C = \{c_1, \dots, c_m\}$  é uma coleção de cláusulas, tal que cada variável de  $U$  ocorre 2 ou 3 vezes nas cláusulas de  $C$  e para toda  $c \in C$ ,  $|c| = 3$  ou  $|c| = 2$  e um inteiro positivo  $k > 0$ .

**Pergunta:** Existe uma atribuição de verdade para  $U$  que satisfaz pelo menos  $k$  cláusulas de  $C$ ?

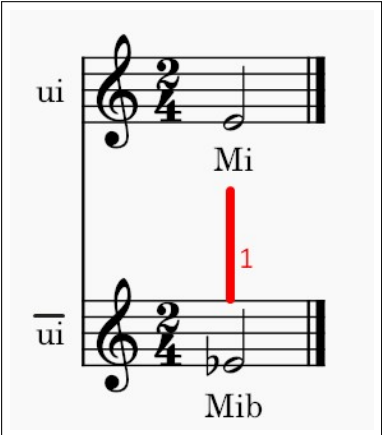
Dada uma instância  $I = ((U, C), k)$  de MAX3SAT<sub>3</sub> nós construímos uma instância  $(\mathcal{H}, p, s, k)$  de ARR-CON com  $p = 30\%$  usando a técnica “projeto de componente” de Karp [15].

Para os literais  $u_1, \bar{u}_1, u_2, \bar{u}_2, \dots, u_n, \bar{u}_n$  nós produzimos, respectivamente, as pautas  $H_1, \dots, H_{2n}$ , tal que para cada  $i \in \{1, \dots, n\}$ :

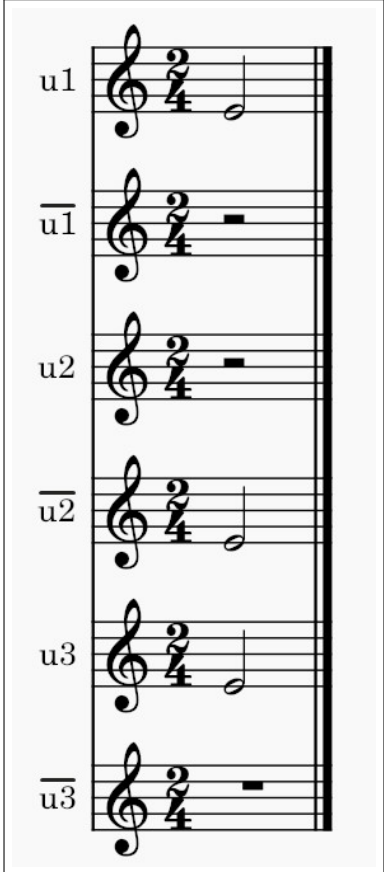
1. no tempo  $i$  de  $H_{2i-1}$  seja tocada a nota E (Mi).
2. no tempo  $i$  de  $H_{2i}$  seja tocada a nota Eb (Mi bemol).

Para cada  $j \in \{1, \dots, m\}$  com a cláusula  $c_j = (x \vee y \vee z) \in \{c_1, \dots, c_m\} = C$  as pautas  $H_x, H_y$  e  $H_z$ , respectivamente, correspondentes aos literais  $x, y, z$ , possuem a nota E no compasso  $n + j$ .

Figura 32 - Construção do Definidor de Verdade  $T_i$  em (a) e do Testador de Satisfação  $S_j$  em (b).



(a)



(b)

Legenda: Exemplo de construção de variável gadget  $T_i$  (a), onde são tocadas simultaneamente as notas E e E $\flat$ , criando um intervalo dissonante, impedindo que ambos os literais sejam verdadeiros ao mesmo tempo. E exemplo de cláusula gadget  $S_j$  obtido a partir de  $c_j = (u_1, \overline{u_2}, u_3)$  (b), onde  $u_1, \overline{u_2}$  e  $u_3$  tocam a nota E simultaneamente, permitindo que de zero até três literais sejam verdadeiros.

Fonte: O autor, 2024.

Nós lembramos que as notas  $E$  (Mi) e  $E^\flat$  (Mi bemol) são dissonantes. Assim, em um certificado positivo  $\mathcal{H}'$  para a instância  $(\mathcal{H}, p, s, k)$ , as pautas  $H_{2i-1}$  e  $H_{2i}$  não podem ser tocadas simultaneamente e pelo menos uma delas precisa ser tocada porque  $p = 30\%$ . Isto é, exatamente um deles ocorre: ou  $H_{2i-1} \in \mathcal{H}'$  ou  $H_{2i} \in \mathcal{H}'$ . Nós adiantamos para o leitor que a resposta SIM de  $(\mathcal{H}, p, s, k)$  implica a definição de uma atribuição de verdade satisfatível, onde  $u_i = V$  se e somente se  $H_{2i-1}$  pertence ao arranjo. A construção é concluída pela instância  $(\mathcal{H}, p, s, k') = (\{H_1, \dots, H_{2n}\}, 3/10, n, n+k)$ .

Figura 33 - Exemplo de partitura  $\mathcal{H}$  da instância de ARRANJO CONSONANTE, construída a partir de uma instância  $I$  de MAX3SAT<sub>3</sub>.

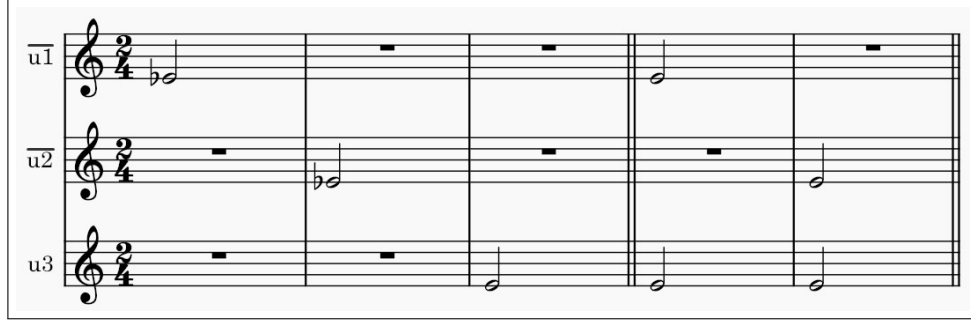
Legenda: Exemplo de partitura  $\mathcal{H} = \{H_1, H_2, H_3, H_4, H_5, H_6\}$  da instância  $(\mathcal{H}, 3/10, 3, 3+2) = (\mathcal{H}, p, n, n+k)$  de ARRANJO CONSONANTE GERAL, construída a partir da instância  $I = ((U, C), 2) = ((U, C), k)$ , com  $U = \{u_1, u_2, u_3\}$  e  $C = \{(\overline{u_1}, u_2, u_3), (u_1, \overline{u_2}, u_3)\}$  de MAX3SAT<sub>3</sub>. Os primeiros  $n = 3$  tempos de  $H_{2i-1}$  e  $H_{2i}$ ,  $i \in \{1, 2, 3\}$  tocam notas dissonantes, então não podem ambos pertencer a um certificado  $\mathcal{H}'$  para a resposta SIM. Mais ainda, como  $|\mathcal{H}'| \geq 3$ , temos que exatamente um de  $H_{2i-1}, H_{2i}$ ,  $i \in \{1, 2, 3\}$  estão em  $\mathcal{H}'$ . Para cada  $j \in \{1, 2\}$  com a cláusula  $c_j = (x \vee y \vee w) \in \{c_1, c_2\} = C$  as pautas  $H_x, H_y$  e  $H_z$ , respectivamente, correspondentes aos literais  $x, y, z$ , possuem a nota  $E$  no compasso  $n+j = 3+j$ .

Fonte: ‘O autor, 2024’.

Observe que, na partitura  $\mathcal{H} = \{H_1, \dots, H_{2n}\}$  temos um número total de tempos é igual a  $n + m$  e  $2n$  pautas.

Suponha que  $(I = (U, C), k)$  é uma instância SIM de MAX3SAT<sub>3</sub>. Então existe uma atribuição de verdade  $f: U \rightarrow \{V, F\}$ , tal que  $k$  ou mais cláusulas de  $C$  tem pelo menos um literal verdadeiro. Nós damos uma receita para mostrar que  $(\{H_1, \dots, H_{2n}\}, 3/10, n, n+k)$  é também uma instância SIM de ARRANJO CONSONANTE GERAL escolhendo exatamente

Figura 34 - Exemplo de certificado  $\mathcal{H}'$  para a resposta SIM da instância  $(\mathcal{H}, 3/10, 3, 3 + 2)$  de ARRANJO CONSONANTE GERAL



Legenda: Certificado  $\mathcal{H}'$  para a resposta SIM da instância  $(\mathcal{H}, 3/10, 3, 3 + 2)$  do exemplo da Figura 33 de ARRANJO CONSONANTE GERAL obtida a partir da atribuição de verdade satisfatível  $\overline{u}_1 = \overline{u}_2 = u_3 = V$  para a instância

$I = ((U, C) = (\{u_1, u_2, u_3\}, \{(\overline{u}_1, u_2, u_3), (u_1, \overline{u}_2, u_3)\}), 2)$  de  $\text{MAX3SAT}_3$ . Veja que tocam-se 50% das notas de  $\mathcal{H}$  nos tempos 1, 2 e 3 e tocam-se  $2/3$  das notas nos tempos 4 e 5. E  $1/2 \geq 3/10$  e  $2/3 \geq 3/10$ .

Fonte: ‘O autor, 2024’.

uma das pautas  $H_{2i-1}, H_{2i}, i \in \{1, \dots, n\}$  para um certificado  $\mathcal{H}'$  para a resposta SIM. Nós selecionamos a pauta  $H_{2i-1}$  para  $\mathcal{H}'$  se e somente se  $f(u_i) = V$ . Veja primeiro que  $|\mathcal{H}'| = n$ , portanto  $|\mathcal{H}'| \geq n$ . Nos primeiros  $n$  tempos são tocadas 50% das notas. Como existem  $k$  das  $m$  cláusulas satisfeitas, em  $k$  tempos dos tempos  $n + 1$  até  $n + m$ , pelo menos  $1/2 > 3/10$  das notas de um tempo de  $\mathcal{H}$  correspondente a uma cláusula satisfeita de tamanho 2 e pelo menos  $1/3 > 3/10$  das notas de um tempo de  $\mathcal{H}$  correspondente a uma cláusula satisfeita de tamanho 3 são tocadas. Nenhuma nota é tocada nos tempos correspondentes às cláusulas não satisfeitas. Assim, existe um subcoleção  $\mathcal{H}' \subset \mathcal{H}$ , tal que as pautas de  $\mathcal{H}'$  podem ser tocadas sem nenhum intervalo dissonante, tal que pelo menos  $3/10$  das notas em pelo menos  $n + k$  tempos de  $\mathcal{H}$  sejam tocadas. Assim,  $(\{H_1, \dots, H_{2n}\}, 3/10, n, n + k)$  é uma instância SIM de ARR-CON.

Suponha que  $(\{H_1, \dots, H_{2n}\}, 3/10, n, n + k)$  é uma instância SIM de ARR-CON. Seja a subcoleção  $\mathcal{H}' \subset \mathcal{H}$ , um certificado para a resposta SIM de ARR-CON, onde as pautas de  $\mathcal{H}'$  podem ser tocadas sem nenhum intervalo dissonante com pelo menos  $3/10$  das notas de pelo menos  $n + k$  tempos de  $\mathcal{H}$  sejam tocadas com  $|\mathcal{H}'| \geq n$ .

Sabemos que para todo  $i \in \{1, \dots, n\}$  nunca aparecem ao mesmo tempo  $H_{2i-1}$  e  $H_{2i}$  em  $\mathcal{H}'$ , pois no tempo  $i$  de  $H_{2i-1}$  e  $H_{2i}$  temos as notas  $E$  (Mi) e  $E_b$  (Mi bemol), que são dissonantes. Como  $|\mathcal{H}'| \geq n$ , temos que para cada  $i \in \{1, \dots, n\}$  aparece  $H_{2i-1}$  ou  $H_{2i}$  em  $\mathcal{H}'$ . Nesse caso, sabemos mais,  $|\mathcal{H}'| = n$ . Dessa forma, damos a seguinte receita para produzir uma atribuição de verdade  $f: U \rightarrow \{V, F\}$  para  $U$  que satisfaz pelo menos  $k$  cláusulas de  $C$ : fazemos  $f(u_i) = V$  se e somente se  $H_{2i-1}$  aparece em  $\mathcal{H}'$ . Vamos provar que  $f$  satisfaz pelo menos  $k$  cláusulas de  $C$ . Como o número de tempos em que pelo menos  $3/10$  das notas em cada tempo de  $\mathcal{H}$  são tocadas em pelo menos  $n + k$  tempos, temos que

os  $n$  primeiros tempos, onde são tocadas 50% das notas e mais  $k$  tempos adicionais que correspondem as cláusulas com pelo menos um literal verdadeiro. E assim, que  $f$  satisfaz pelo menos  $k$  cláusulas de  $C$ .

Dessa forma,  $I = (U, C)$  é uma instância SIM de  $\text{MAX3SAT}_{\frac{3}{10}}$  se e somente se  $(\{H_1, \dots, H_{2n}\}, \frac{3}{10}, n, n + k)$  é uma instância SIM de ARR-CON. Assim,  $\text{MAX3SAT}_{\frac{3}{10}} \propto \text{ARR-CON}$  e dessa forma, ARR-CON é NP-completo.  $\square$

Introduzimos agora MAXARR-CON uma versão de otimização de ARR-CON. MAXARR-CON tem como objetivo encontrar o maior número de tempos da música original (partitura) que pode ser tocado sem notas dissonantes, com pelo menos  $s$  pautas e  $p\%$  das notas originais em cada tempo.

### 3.1.1 Uma versão de maximização para arranjo consonante

MAX ARRANJO CONSONANTE [MAXARR-CON]

**Instância:** Coleção de pautas  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  uma percentagem  $0 < p\% \leq 1$  e um inteiro positivo  $s$ .

**Objetivo:** Encontrar o maior  $k$  para o qual existe um subconjunto  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq s$ , tal que  $\mathcal{H}'$  não tem par de notas dissonantes e  $p\%$  das notas em pelo menos  $k$  tempos de  $\mathcal{H}$  são tocadas.

Vamos provar que MAXARR-CON não possui um esquema de aproximação em tempo polinomial (EATP) a menos que  $P=NP$ . Para isso, usamos a versão de otimização MAXSAT de SAT a qual ao invés de verificar se  $I = (U, C)$ , com  $n = |U|$  variáveis e  $m = |C|$  cláusulas, é satisfatível queremos encontrar uma atribuição de verdade que satisfaça um número máximo  $OPT_{\text{MAXSAT}}(I)$  de cláusulas. Veja que  $I$  é satisfatível se e somente se  $OPT_{\text{MAXSAT}}(I) = m$ .

No caso da versão de otimização de  $\text{MAX3SAT}_{\frac{3}{10}}$ , vamos usar o mesmo nome para ambos os problemas de maximização e decisão, deixando para o contexto sua identificação. Lembramos que a versão de decisão de  $\text{MAX3SAT}_{\frac{3}{10}}$  tem uma instância correspondente a  $((U, C), k)$ , enquanto que a de maximização de  $\text{MAX3SAT}_{\frac{3}{10}}$  a instância é da forma  $(U, C)$ . E o objetivo é encontrar o  $k$  máximo tal que exista uma atribuição de verdade que satisfaça pelo menos  $k$  cláusulas de  $C$ . O contexto dirá se estamos falando do problema de decisão ou de otimização.

No livro de Eli Upfal e Michael Mitzenmacher [32] encontramos o seguinte teorema:

**Teorema 3.1.2** (Eli Upfal e Michael Mitzenmacher [32]). *Dado  $I = (U, C)$  uma instância de MAXSAT com  $|C| = m$  cláusulas, onde  $\ell_i$  é o número de literais na  $i$ -ésima cláusula*

para  $i = 1, \dots, m$ . Seja  $\ell = \min_{i=1}^m \ell_i$ . Então existe uma atribuição de verdade para  $U$  que satisfaz pelo menos  $\frac{(2^\ell-1)}{2^\ell}m$  das cláusulas de  $C$ , isto é

$$OPT_{maxsat}(I) \geq \sum_{i=1}^m (1 - 2^{-\ell_i}) \geq m(1 - 2^{-\ell}) = \frac{(2^\ell-1)}{2^\ell}m.$$

Se  $\ell$  é o tamanho da menor cláusula do problema e, em  $\text{MAX3SAT}_{\frac{2}{3}}$  o menor tamanho é 2, sabemos que

**Corolário 3.1.1.** *Se  $I = (U, C)$  é uma instância de  $\text{MAX3SAT}_{\frac{2}{3}}$  com  $|U| = n$  e  $|C| = m$ , então*

$$Opt_{\text{MAX3SAT}_{\frac{2}{3}}}(I) \geq \frac{3}{4}m.$$

**Lema 3.1.1.** *Se  $I = (U, C)$  é uma instância de  $\text{MAX3SAT}_{\frac{2}{3}}$  com  $|U| = n$  e  $|C| = m$ , então*

$$\frac{2}{3}n \leq m \leq \frac{3}{2}n.$$

*Demonstração.* Vamos usar o Princípio da Casa dos Pombos duas vezes. Sabemos que as cláusulas de  $C$  tem tamanho 2 ou 3 e as variáveis de  $U$  ocorrem 2 ou 3 vezes. Assim, o menor número de cláusulas é atingido quando temos que cada variável ocorre duas vezes e cada cláusula tem tamanho 3, nesse caso  $2n = 3m$ . Reciprocamente, o maior número de cláusulas é atingido quando temos que cada variável ocorre três vezes e cada cláusula tem tamanho 2, nesse caso  $3n = 2m$ . Dessa forma,  $\frac{2}{3}n \leq m \leq \frac{3}{2}n$ .  $\square$

Nós usamos o problema de maximização  $\text{MAX3SAT}_{\frac{2}{3}}$  para provar que  $\text{MAX ARRANJO CONSONANTE}$  é  $\text{MAXSNP}$ -difícil.

**Teorema 3.1.3** (Papadimitriou [30]).  $\text{MAX3SAT}_{\frac{2}{3}}$  é  $\text{MAXSNP}$ -completo.

**Teorema 3.1.4.**  $\text{MAXARR-CON}$  é  $\text{MAXSNP}$ -difícil.

*Demonstração.* Para provar que  $\text{MAXARR-CON}$  é  $\text{MAXSNP}$ -difícil, provamos que  $\text{MAX3SAT}_{\frac{2}{3}}$  L-reduz para  $\text{MAXARR-CON}$ .

Seja  $I = ((U, C), k)$  uma instância de  $\text{MAX3SAT}_{\frac{2}{3}}$ . Utilizando a mesma construção do Teorema 3.1.1, temos que  $I = ((U, C), k)$  é uma instância SIM de  $\text{MAX3SAT}_{\frac{2}{3}}$  se e somente se  $M = (\{H_1, \dots, H_{2n}\}, \frac{3}{10}, n, n+k)$  é uma instância SIM de  $\text{ARR-CON}$ . Em outras palavras, significa que quando  $k$  cláusulas são satisfeitas em  $I$ ,  $\mathcal{H}'$  possui  $n+k$  tempos sem nenhum par de notas dissonantes, onde são tocadas pelo menos  $\frac{3}{10}$  das notas em cada um desses  $n+k$  tempos. Assim,  $OPT_{\text{MAXARR-CON}}(M) = n + OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I)$ . Para a conveniência do leitor nós oferecemos na Figura ?? um exemplo da transformação, onde  $OPT_{\text{MAXARR-CON}}(M) = n + OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) = 4 + 5 = 9$

Vamos provar agora a primeira parte da L-redução, isto é, que existe  $\alpha > 0$ , tal que  $OPT_{\text{MAXARR-CON}}(M) \leq \alpha OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I)$ . A seguir, vamos usar o Teorema 3.1.1 e o Lema 3.1.1 para provar a sequência de desigualdades necessárias. Considere que  $OPT_{\text{MAXARR-CON}}(M) = n + OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) \leq \frac{3}{2}m + OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) \leq$

$\frac{3}{2} \left( \frac{4}{3} OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) \right) + OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) = 3 OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I)$ . O que mostra que  $\alpha = 3$  resolve.

Vamos provar a segunda parte da L-redução, isto é, que dada uma solução  $s_M$  de custo  $|s_M|$  para  $M$ , existe um algoritmo tempo polinomial  $g$  que obtém uma solução  $g(s_M) = s_I$  de custo  $|s_I|$  para  $I$ , tal que existe  $\beta > 0$  onde  $|OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) - |s_I|| \leq \beta |OPT_{\text{MAXARR-CON}}(M) - |s_M||$ .

Dada  $s_M$  uma solução viável de custo  $|s_M|$  para  $M$ . Porque  $s_M$  é viável, não existem tempos com notas dissonantes. Se para algum  $i \in \{1, \dots, n\}$  nem  $H_{2i-1}$ , nem  $H_{2i}$  pertencem a  $s_M$ , então nós modificamos  $s_M$ , ampliando  $s_M$ , inserindo  $H_{2i-1}$  em  $s_M$ , aumentando o número de tempos de  $s_M$  sem notas dissonantes. Lembre que na instância de MAXARR-CON não existem notas dissonantes nos tempos de  $n+1$  até  $n+m$ . Assim, podemos definir uma solução viável  $s_I$  para  $I$ , fazendo  $u_i = V$  se e somente se  $H_{2i-1}$  pertence a  $s_M$ .

Dessa forma, se  $k$  é o número de cláusulas satisfeitas por  $s_I$ , então  $|s_M| = n+k$ . Então temos que  $|OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) - |s_I|| = |OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) - k| = |n + OPT_{\text{MAX3SAT}_{\frac{2}{3}}}(I) - (n+k)| = |OPT_{\text{MAXARR-CON}}(M) - |s_M||$ . O que mostra que  $\beta = 1$  resolve. Portanto, MAXARR-CON é MAX-SNP-difícil.  $\square$

Considerando o Teorema 1.4.3, a menos que  $P=NP$ , não existe um esquema de aproximação em tempo polinomial para MAX ARRANJO CONSONANTE.

**Corolário 3.1.2.** *Se  $P \neq NP$ , então não existe um esquema de aproximação em tempo polinomial de MAX ARRANJO CONSONANTE.*

### 3.1.2 Complexidade parametrizada de ARR-CON

Nesta subseção, iremos demonstrar que a menos que  $P=NP$ , ARR-CON não é tratável por parâmetro fixo (FPT). Faremos isso através de uma redução CONJUNTO INDEPENDENTE  $\propto$  ARR-CON.

CONJUNTO INDEPENDENTE (CI)

**Instância:** Grafo  $G = (V, E)$  e inteiro positivo  $k$ .

**Pergunta:** O grafo  $G$  contém um conjunto independente  $S \subseteq V$  com  $|S| \geq k$ , ou seja, um conjunto  $S \subseteq V$ , tal que para cada  $uv \in E$  ou  $u \notin S$  ou  $v \notin S$ ?

**Teorema 3.1.5** (Downey e Fellows [33]). CONJUNTO INDEPENDENTE é  $W[1]$ -completo.

**Teorema 3.1.6.** ARR-CON é  $W[1]$ -difícil no parâmetro  $k$  da solução.

*Demonstração.* Para provar que ARR-CON é  $W[1]$  difícil, fornecemos uma redução parametrizada de CI parametrizado pelo tamanho do conjunto independente parametrizado.

Dado uma instância  $(G = (V, E), k)$  com  $|V| = n$  e  $|E| = m$  de CONJUNTO INDEPENDENTE nós construímos uma instância  $(\mathcal{H}, p, k) = (\mathcal{H}, 0, k)$  tal que  $G$  tem um conjunto independente  $S$  com  $|S| \geq k$  se e somente se  $(\mathcal{H}, 0, k)$  é uma instância SIM de ARR-CON.

Para cada vértice  $v_1, v_2, \dots, v_n \in V$  nós produzimos, respectivamente as pautas  $H_1, H_2, \dots, H_n \in \mathcal{H}$  e para cada aresta  $e_j = v_i v_\ell$ ,  $H_i$  toca a nota E (mi) e  $H_\ell$  toca a nota F (fá) simultaneamente no tempo  $t$ ,  $t \in \{1, \dots, m\}$ . Como as notas mi e fá são dissonantes, isso cria um conflito. Assim, para cada aresta  $v_i v_\ell \in E$ , como em cada tempo  $H_i$  e  $H_\ell$  tocam notas dissonantes, apenas uma das pautas,  $H_i$  ou  $H_\ell$  pode pertencer a solução viável  $\mathcal{H}$ . A construção é concluída com instância  $(\mathcal{H}, p, k') = (\{H_1, \dots, H_n\}, 0, k)$ .

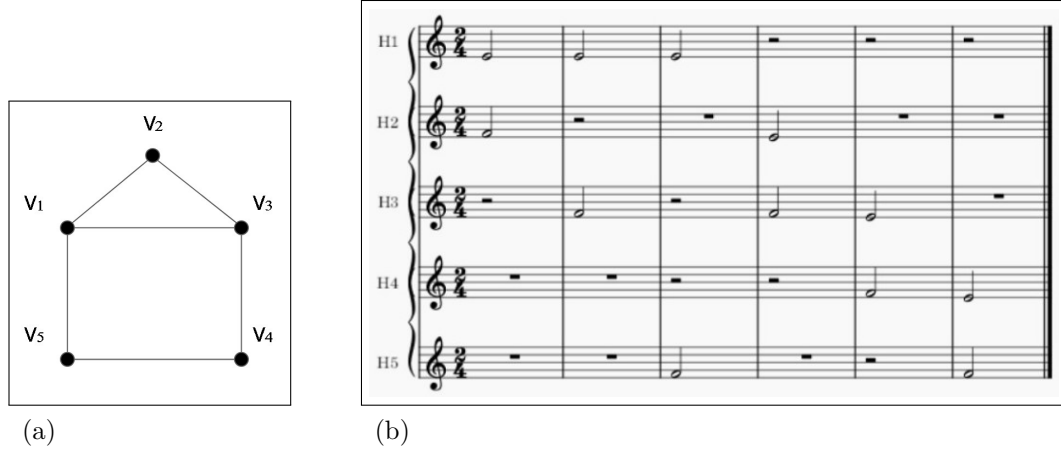
Suponha que  $(G = (V, E), k)$  é uma instância SIM de CONJUNTO INDEPENDENTE. Então existe um conjunto independente  $S$  em  $G$  de tamanho  $|S| \geq k$ . Nós damos uma receita para mostrar que  $(\{H_1, \dots, H_n\}, 0, k)$  é também uma instância SIM de ARR-CON escolhendo a pauta  $H_i$  para adicionar para a subcoleção  $\mathcal{H}'$  se e somente se  $v_i \in S$ . Para a conveniência do leitor, oferecemos na Figura 43 o certificado associado de uma instância de ARR-CON construída a partir de uma instância de CONJUNTO INDEPENDENTE. Observe que no  $x$ -ésimo tempo, a aresta  $e_x = v_i v_\ell$ , pelo menos uma das pautas  $H_i$  ou  $H_\ell$ , não pode estar em  $\mathcal{H}'$ , porque tocam um par de notas dissonantes. Assim como  $v_i$  e  $v_\ell$  extremidades da aresta  $v_i v_\ell$  não podem estar ambos em  $S$ . Assim, existe uma subcoleção  $\mathcal{H}' \subset \mathcal{H}$  onde nenhum par de notas simultâneas é dissonante e  $|\mathcal{H}'| \geq k$ .

Suponha que  $(\{H_1, \dots, H_n\}, 0, k)$  é uma instância SIM de ARR-CON. Então existe uma subcoleção  $\mathcal{H}' \subset \mathcal{H}$ , tal que nenhum par de notas simultâneas é dissonante e  $|\mathcal{H}'| \geq k$ . Sabemos que para cada aresta  $v_i v_\ell$ , as pautas  $H_i$  e  $H_\ell$  não podem tocar juntos simultaneamente pois tocam notas dissonantes e portanto não podem estar ambos em  $\mathcal{H}'$ . Dessa forma, fornecemos a seguinte receita para produzir um conjunto independente  $S \subseteq V$  com tamanho  $|S| \geq k$ . Fazemos  $v_i \in S$  se e somente se  $H_i$  aparece em  $\mathcal{H}'$ . Vamos provar que  $|S| \geq k$  e que nenhum par de vértices em  $S$  são adjacentes. Assim, uma aresta  $v_i v_\ell \in E$  implica em ou  $H_i \notin \mathcal{H}'$  ou  $H_\ell \notin \mathcal{H}'$ . Portanto, se  $v_i v_\ell \in E$  e  $v_i \in S$ , então  $v_\ell \notin S$ . Assim,  $S$  é independente. Como  $|\mathcal{H}'| \geq k$ , então  $|S| \geq k$ .

Ou seja,  $(G = (V, E), k)$  é uma instância SIM de CONJUNTO INDEPENDENTE se e somente se  $(\{H_1, \dots, H_n\}, 0, k)$  é uma instância SIM de ARR-CON. Assim, CONJUNTO INDEPENDENTE reduz parametricamente para ARR-CON. Então de acordo com a Definição 1.5.9, ARR-CON é W[1]-difícil.

□

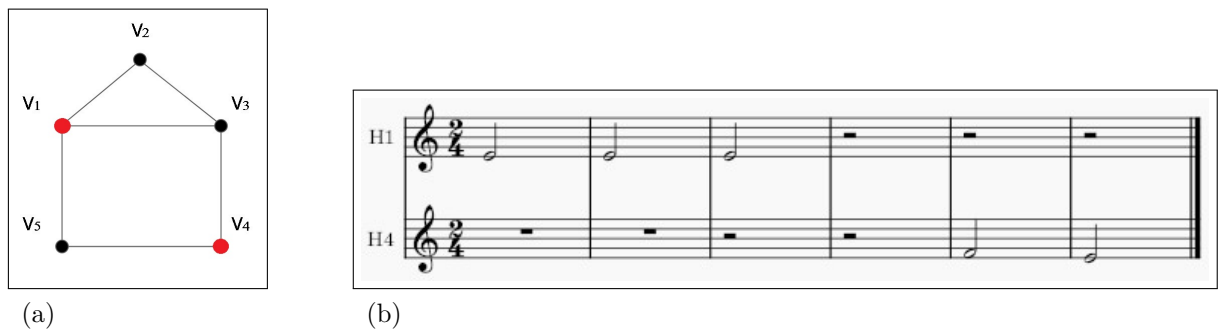
Figura 35 - Instância de CONJUNTO INDEPENDENTE e ARR-CON.



Legenda: Exemplo para a transformação  $\text{CONJUNTO INDEPENDENTE} \propto \text{ARR-CON}$ , onde a instância  $M = (\mathcal{H}, 0, k)$  de ARR-CON (43b) é obtida a partir da instância  $(G)$  de CONJUNTO INDEPENDENTE(43a).

Fonte: O autor, 2024.

Figura 36 - CONJUNTO INDEPENDENTE  $\propto$  ARR-CON.



Legenda: Arranjo  $\mathcal{H}'$  de ARR-CON criado a partir de conjunto independente  $S$  de  $G$  de CONJUNTO INDEPENDENTE

Fonte: O autor, 2024.

### 3.2 Problema de decisão das pautas com número limitado de notas

O problema de decisão LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS foi introduzido por Demaine e Moses [1].

Nessa seção, nós propomos o problema NÚMERO LIMITADO DE  $j$  NOTAS SIMULTÂNEAS GERAL (neste trabalho vamos abreviar para SIM- $j$ -NOTAS que generaliza o problema de decisão LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS). Faremos neste capítulo, uma redução do problema SIM- $j$ -NOTAS que será utilizada na Seção 3.2.1 para a versão de otimização do mesmo. Nós consideramos uma versão natural de maximização MAXSIM- $j$ -NOTAS de SIM- $j$ -NOTAS e provamos que MAXSIM- $j$ -NOTAS não admite nenhum esquema de aproximação em tempo polinomial (EATP) a menos que  $P=NP$ .

O problema de decisão NÚMERO LIMITADO DE  $j$  NOTAS SIMULTÂNEAS GERAL tem a seguinte formulação.

NÚMERO LIMITADO DE  $j$  NOTAS SIMULTÂNEAS GERAL (SIM- $j$ -NOTAS)

**Instância:**  $M = (\mathcal{H}, p, j, k)$  onde  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  é uma coleção de pautas,  $0 \leq p\% \leq 1$  um percentual e um par de inteiros  $j, k \geq 0$ .

**Pergunta:** Existe uma subcoleção  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq k$ , tal que no máximo  $j$  notas são tocadas em cada tempo e pelo menos  $p\%$  das notas de  $\mathcal{H}$  são tocadas?

Nós observamos que uma instância  $(\mathcal{H}, p, j, k)$  do problema NÚMERO LIMITADO DE  $j$  NOTAS SIMULTÂNEAS GERAL com  $k = 0$  é equivalente a uma instância do problema de decisão LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS. Portanto, nosso problema é uma generalização do problema original.

No problema LIMITAÇÃO NO NÚMERO DE NOTAS SIMULTÂNEAS apresentado por Demaine, a complexidade do problema era sensível ao valor de  $p$  (o problema é polinomial se  $p = 0$ , ou se  $p = 1$ ). Mas com a adição de  $k \geq 1$  o problema continua NP-completo mesmo com  $p = 0$  e iremos provar isso na transformação do Teorema 3.2.1 a seguir.

**Teorema 3.2.1.** SIM- $j$ -NOTAS é NP-completo.

*Demonstração.* SIM- $j$ -NOTAS está em NP, pois um certificado pode ser verificado em tempo polinomial checando se  $|\mathcal{H}'| \geq k$  e se em cada tempo: no máximo  $j$  notas são tocadas e pelo menos  $p\%$  das notas de  $\mathcal{H}$  são tocadas.

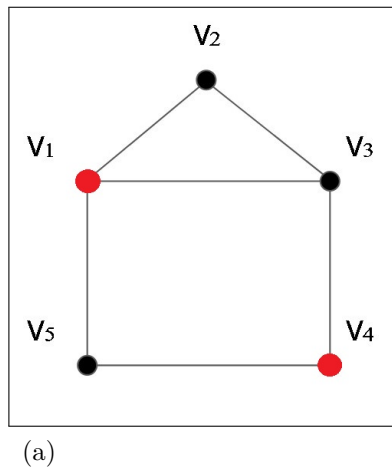
Vamos usar o problema NP-completo CONJUNTO INDEPENDENTE [16] para provar que SIM- $j$ -NOTAS é NP-completo, através de uma redução polinomial CONJUNTO INDEPENDENTE  $\propto$  SIM- $j$ -NOTAS.

Dado uma instância  $(G = (V, E), k)$  com  $|V| = n$  e  $|E| = m$  de CONJUNTO INDEPENDENTE nós construímos uma instância  $(\mathcal{H}, p, j, k) = (\mathcal{H}, 0, 1, k)$  tal que  $G$  tem um conjunto independente  $S$  com  $|S| \geq k$  se e somente se  $(\mathcal{H}, 0, 1, k)$  é uma instância SIM de SIM- $j$ -NOTAS.

Para cada vértice  $v_1, v_2, \dots, v_n \in V$  nós produzimos, respectivamente as pautas  $H_1, H_2, \dots, H_n \in \mathcal{H}$  e para cada aresta  $e_j = v_i v_\ell$ ,  $H_i$  e  $H_\ell$  tocam a nota E (mi) simultaneamente no tempo  $t$ ,  $t \in \{1, \dots, m\}$ .

Recordamos que  $j = 1$ , o que significa que só pode haver no máximo uma nota tocando em cada tempo. Assim, para cada aresta  $v_i v_\ell \in E$ , como em cada tempo  $H_i$  e  $H_\ell$  tocam notas simultâneas, apenas uma das pautas,  $H_i$  ou  $H_\ell$  pode pertencer a solução viável  $\mathcal{H}$ . A construção é concluída com instância  $(\mathcal{H}, p, j, k') = (\{H_1, \dots, H_n\}, 0, 1, k)$ .

Figura 37 - Instância de CONJUNTO INDEPENDENTE transformada em instância de SIM- $j$ -NOTAS.



$U = \{H1, H2, H3, H4, H5\}$   
 $C = \{(H1, H2), (H1, H3), (H1, H5), (H2, H3), (H3, H4), (H4, H5)\}$   
 $J=1$   
 $p=0$

(b)

Legenda: A Figura (a) representa o grafo  $G = (V, E)$ , com  $V = \{v_1, v_2, v_3, v_4, v_5\}$  e  $E = \{v_1 v_2, v_1 v_3, v_1 v_5, v_2 v_3, v_3 v_4, v_4 v_5\}$  parte da instância  $(G, 2)$  de CONJUNTO INDEPENDENTE. A Figura (b) representa a instância  $(\mathcal{H}, 0, 1, 2)$  de SIM- $j$ -NOTAS construída a partir da instância  $(G, 2)$ . Nas duas instâncias são mostrados em vermelho os certificados  $\{v_1, v_4\}$  e  $\{H_1, H_4\}$  para a resposta SIM de CONJUNTO INDEPENDENTE e SIM- $j$ -NOTAS respectivamente.

Fonte: O autor, 2024.

Suponha que  $(G = (V, E), k)$  é uma instância SIM de CONJUNTO INDEPENDENTE. Então existe um conjunto independente  $S$  em  $G$  de tamanho  $|S| \geq k$ . Nós damos uma receita para mostrar que  $(\{H_1, \dots, H_n\}, 0, 1, k)$  é também uma instância SIM de SIM- $j$ -NOTAS escolhendo a pauta  $H_i$  para adicionar para a subcoleção  $\mathcal{H}'$  se e somente se  $v_i \in S$ . Para a conveniência do leitor, oferecemos na Figura 37 o certificado associado de uma instância de SIM- $j$ -NOTAS construída a partir de uma instância de CONJUNTO INDEPENDENTE. Observe que no  $x$ -ésimo tempo, a aresta  $e_x = v_i v_\ell$ , pelo menos uma das pautas  $H_i$  ou  $H_\ell$ , não pode estar em  $\mathcal{H}'$ , porque  $j = 1$ . Assim como  $v_i$  e  $v_\ell$  extremidades da aresta  $v_i v_\ell$  não podem estar ambos em  $S$ . Assim, existe uma subcoleção  $\mathcal{H}' \subset \mathcal{H}$  tal que pelo menos  $p\%$  e no máximo  $j = 1$  das notas de  $\mathcal{H}$  são tocadas em cada tempo e  $|\mathcal{H}'| \geq k$ .

Suponha que  $(\{H_1, \dots, H_n\}, 0, 1, k)$  é uma instância SIM de SIM- $j$ -NOTAS. Então

existe uma subcoleção  $\mathcal{H}' \subset \mathcal{H}$ , tal que no máximo uma das notas de  $\mathcal{H}$  são tocadas em cada tempo e  $|\mathcal{H}'| \geq k$ . Sabemos que para cada aresta  $v_i v_\ell$ , as pautas  $H_i$  e  $H_\ell$  não podem tocar juntos simultaneamente pois  $j = 1$  e portanto não podem estar ambos em  $\mathcal{H}'$ . Dessa forma, fornecemos a seguinte receita para produzir um conjunto independente  $S \subseteq V$  com tamanho  $|S| \geq k$ . Fazemos  $v_i \in S$  se e somente se  $H_i$  aparece em  $\mathcal{H}'$ . Vamos provar que  $|S| \geq k$  e que nenhum par de vértices em  $S$  são adjacentes. Assim, como em uma aresta  $v_i v_\ell \in E$  com  $j = 1$  implica em ou  $H_i \notin \mathcal{H}'$  ou  $H_\ell \notin \mathcal{H}'$ . Portanto, se  $v_i v_\ell \in E$  e  $v_i \in S$ , então  $v_\ell \notin S$ . Assim,  $S$  é independente. Como  $|\mathcal{H}'| \geq k$ , então  $|S| \geq k$ .

Ou seja,  $(G = (V, E), k)$  é uma instância SIM de CONJUNTO INDEPENDENTE se e somente se  $(\{H_1, \dots, H_n\}, 0, 1, k)$  é uma instância SIM de SIM- $j$ -NOTAS. Assim, CONJUNTO INDEPENDENTE  $\propto$  SIM- $j$ -NOTAS e dessa forma, SIM- $j$ -NOTAS é NP-completo.  $\square$

Recorremos ao Teorema 3.3.3 de Downey e Fellows, que nos permite derivar o seguinte corolário:

**Corolário 3.2.1.** SIM- $j$ -NOTAS é  $W[1]$ -difícil no parâmetro  $k$  da solução.

A seguir, propomos uma versão de otimização para o problema de decisão SIM- $j$ -NOTAS que nós chamamos de MAXSIM- $j$ -NOTAS.

MÁXIMO NÚMERO LIMITADO DE NOTAS SIMULTÂNEAS (MAXSIM- $j$ -NOTAS)

**Instância:**  $M = (\mathcal{H}, p, j)$  onde  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  é uma coleção de pautas,  $0 \leq p\% \leq 1$  um percentual e um inteiro  $j \geq 0$ .

**Objetivo:** Determine  $k$  máximo tal que existe um subconjunto  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq k$ , tal que as pautas de  $\mathcal{H}'$  podem ser tocadas simultaneamente com pelo menos  $p\%$  das notas tocadas em cada tempo de  $\mathcal{H}$  e no máximo  $j$  das notas de  $\mathcal{H}'$  tocadas em cada tempo.

Agora nós consideramos a versão de otimização de CONJUNTO INDEPENDENTE que manteremos com o mesmo nome do problema de decisão. A entrada dada do problema de otimização é o mesmo grafo  $G = (V, E)$  da versão de decisão de CONJUNTO INDEPENDENTE. O objetivo é encontrar o  $k$  máximo tal que existe um conjunto independente  $S \subset V$  de  $G$  com tamanho  $|S| \geq k$ . O contexto dirá se estamos falando do problema de decisão ou de otimização.

**Teorema 3.2.2.** Se  $(\mathcal{H}, 0, 1, k)$  é a instância de SIM- $j$ -NOTAS obtida pela redução polinomial do Teorema 3.2.1 a partir da instância  $(G = (V, E), k)$  de CONJUNTO INDEPENDENTE, então

$$Opt_{\text{MAXSIM-}j\text{-NOTAS}}(\mathcal{H}, p, j) = Opt_{\text{MAXSIM-}j\text{-NOTAS}}(\mathcal{H}, 0, 1) = Opt_{\text{CI}}(G).$$

*Demonstração.* Dado  $I = (G = (V, E), k)$  uma instância de CONJUNTO INDEPENDENTE. A partir do Teorema 3.2.1 temos que  $(G = (V, E), k)$  é uma instância SIM de CONJUNTO

INDEPENDENTE se e somente se  $M = (\{H_1, \dots, H_n\}, 0, 1, k)$  é uma instância SIM de SIM- $j$ -NOTAS. Significando que existe um conjunto independente  $S$  com tamanho  $|S| = k$ , se e somente se existe uma coleção  $\mathcal{H}'$  com  $k$  pautas, onde são tocadas no máximo 1 nota por tempo ( $j = 1$ ). Dessa forma,  $OPT_{\text{MAXSIM-J-NOTAS}}(M) = OPT_{\text{CI}}(I)$ .  $\square$

### 3.2.1 MAXSIM- $j$ -NOTAS $\Delta 4$

Papadimitriou [30] considerou a versão GRAU 4-CONJUNTO INDEPENDENTE (CI $\Delta 4$ ) de conjunto independente onde a entrada é restrita a grafos com grau no máximo 4.

**Teorema 3.2.3** (Papadimitriou [30]). GRAU 4-CONJUNTO INDEPENDENTE é MAX-SNP-completo.

Nós consideramos agora MAXSIM- $j$ -NOTAS $\Delta 4$  uma versão diferente de MAXSIM- $j$ -NOTAS, onde as instâncias têm a propriedade de que cada pauta toca no máximo com 4 outras pautas simultaneamente. Nós observamos que quando a entrada é restrita a grafos com grau máximo 4 a transformação polinomial CONJUNTO INDEPENDENTE $\propto$ MAXSIM- $j$ -NOTAS possui a imagem exatamente nas instâncias de MAXSIM- $j$ -NOTAS $\Delta 4$ .

Nosso objetivo nessa seção é analisar a complexidade de MAXSIM- $j$ -NOTAS $\Delta 4$ .

**Teorema 3.2.4.** MAXSIM- $j$ -NOTAS $\Delta 4$  é MAXSNP-difícil, mesmo quando no máximo duas pautas tocam em cada tempo.

*Demonstração.* Dada  $M = (\{H_1, \dots, H_n\}, 0, 1)$  a instância de MAXSIM- $j$ -NOTAS $\Delta 4$  obtida a partir da instância  $G = (V, E)$  de CONJUNTO INDEPENDENTE com a transformação de NP-completude do Teorema 3.2.1.

Pelo Teorema 3.2,  $OPT_{\text{MAXSIM-}j\text{-NOTAS}\Delta 4}(M) = OPT_{\text{CI}\Delta 4}(I)$ . Então,  $\alpha = 1$  resolve para que  $OPT_{\text{MAXSIM-}j\text{-NOTAS}\Delta 4}(M) \leq \alpha OPT_{\text{CI}\Delta 4}(I)$ .

Vamos provar a segunda parte da L-redução, isto é, que dada uma solução  $s_M = \mathcal{H}'$  de custo  $|s_M|$  para  $M = (\{H_1, \dots, H_n\}, 0, 1, k)$ , existe um algoritmo tempo polinomial  $g$  que obtém uma solução  $g(s_M) = s_I$  de custo  $|s_I|$  para  $(G = (V, E), k)$ , tal que existe  $\beta > 0$ , onde  $|OPT_{\text{CI}\Delta 4}(I) - |s_I|| \leq \beta |OPT_{\text{MAXSIM-}j\text{-NOTAS}\Delta 4}(M) - |s_M||$ .

Seja  $s_M = \mathcal{H}' \subset \mathcal{H} = \{H_1, \dots, H_n\}$  uma solução viável de custo  $|s_M|$  para  $M$ . Porque  $s_M$  é viável, em nenhum momento, as pautas de  $\mathcal{H}'$  tocam mais do que 1 nota. Se para algum  $i \in \{1, \dots, n\}$ ,  $H_i \notin \mathcal{H}'$  e a adição de  $H_i$  para a subcoleção  $\mathcal{H}'$  não faz com que em nenhum tempo mais do que  $j$  notas sejam tocadas simultaneamente, então modificamos ampliando  $s_M$ , inserindo  $H_i$  em  $s_M$ , aumentando então o número de pautas de  $s_M$ . Assim, podemos definir uma solução viável  $S = s_I$  para  $(G = (V, E), k)$ , fazendo  $v_i \in s_I$  se e somente se  $H_i$  pertence a  $s_M$ .

Dessa forma, se  $k$  é o número de vértices em  $s_I$ , então  $|s_M| = k$ . Então temos que  $|OPT_{\text{CI}\Delta 4}(I) - |s_I|| = |OPT_{\text{CI}\Delta 4}(I) - k| = |OPT_{\text{MAXSIM-}j\text{-NOTAS}\Delta 4}(M) - |s_M||$ . O que

mostra que  $\beta = 1$  resolve para que  $|OPT_{CI\Delta 4}(I) - |s_I|| \leq \beta |OPT_{MAXSIM-j-NOTAS\Delta 4}(M) - |s_M||$ . Dessa forma, MAXSIM- $j$ -NOTAS $\Delta 4$  é MAXSNP-difícil.  $\square$

Mostramos à seguir que a menos que  $P = NP$  não existe algoritmo de aproximação em tempo polinomial  $A$  para uma instância  $M = (\{H_1, \dots, H_n\}, 0, 1, k)$  de MAXSIM- $j$ -NOTAS dentro de  $n^{1-\varepsilon}$ , para todo  $\varepsilon > 0$ . Para grafos com grau máximo 4, exibimos um algoritmo  $A$  de aproximação de tempo polinomial dentro de  $R_A = \frac{1}{5}$  para MAXSIM- $j$ -NOTAS $\Delta 4$ . Mostramos também que MAXSIM- $j$ -NOTAS $\Delta 4$  é FPT com respeito ao tamanho da solução.

Seja  $A$  um algoritmo para um problema de otimização  $\Pi$ . Seja  $D_\Pi$  o conjunto de todas as instâncias de  $\Pi$ ,  $x \in D_\Pi$  uma instância de  $\Pi$  e  $Opt_\Pi(x)$  o valor de uma solução ótima para a instância  $x$ . Seja  $y$  uma solução viável obtida a partir do algoritmo  $A$  para a instância  $x$ . Seja  $c(y)$  o valor da solução viável  $y$  obtida do algoritmo  $A$  a partir de  $x$ . O fator de aproximação do algoritmo  $A$  é  $R_A(x, y) = \max_{x \in D_\Pi} \left\{ \frac{Opt_\Pi(x)}{c(y)}, \frac{c(y)}{Opt_\Pi(x)} \right\}$ . Observe que  $R_A(x, y) \geq 1$ ; e  $R_A(x, y) = 1$  se e somente se o algoritmo  $A$  obtiver uma solução  $y$ , tal que  $Opt_\Pi(x) = c(y)$ . Além disso, se  $\Pi$  é um problema de maximização, então  $R_A(x, y) = Opt_\Pi(x)/c(y)$ . O fator de aproximação absoluta [16] de  $A$  é  $R_A = \inf\{r \geq 1 : R_A(x, y) \leq r, \text{ para todas as instâncias } x \in D_\Pi\}$ . Para os próximos resultados considere  $n$  o tamanho da entrada para uma instância de um problema. O Teorema de Zuckerman [34] estabelece um limite inferior de  $n^{1-\varepsilon}$  para o fator de aproximação absoluta de CONJUNTO INDEPENDENTE assumindo que  $P \neq NP$ .

**Teorema 3.2.5.** *Existe um algoritmo aproximativo  $A$  para MAXSIM- $j$ -NOTAS $\Delta 4$  com fator de aproximação  $R_A = 5$ .*

*Demonstração.* Dada  $M = (\{H_1, \dots, H_n\}, 0, 1)$  a instância de MAXSIM- $j$ -NOTAS $\Delta 4$  obtida a partir da instância  $G = (V, E)$  de CONJUNTO INDEPENDENTE com a transformação de NP-completude do Teorema 3.2.1. O algoritmo obtém uma partitura  $\mathcal{H}'$  considerando a cada turno, tomar uma pauta aleatória  $H$  de  $\mathcal{H}$  para  $\mathcal{H}'$ , removendo da partitura corrente  $\mathcal{H}$  a pauta  $H$  e todas as pautas que compartilhem notas com  $H$ . Até  $\mathcal{H}$  se tornar vazia. A solução  $\mathcal{H}'$  satisfaz que  $\frac{|\mathcal{H}|}{|\mathcal{H}'|} \geq 5$ . Como  $OPT_{MAXSIM-j-NOTAS\Delta 4}(M) \leq |\mathcal{H}|$ , temos que,  $R_A = \frac{OPT_{MAXSIM-j-NOTAS\Delta 4}(M)}{|\mathcal{H}'|} \leq \frac{|\mathcal{H}|}{|\mathcal{H}'|} \leq 5$   $\square$

**Teorema 3.2.6** (Zuckerman 2006 [34]). *Se  $P \neq NP$  e  $A$  é um algoritmo de aproximação em tempo polinomial para CONJUNTO INDEPENDENTE, então  $R_A > n^{1-\varepsilon}$  para todo  $\varepsilon > 0$ .*

**Teorema 3.2.7.** *Se  $P \neq NP$  e  $A$  é um algoritmo de aproximação em tempo polinomial para MAXSIM- $j$ -NOTAS, então  $R_A > n^{1-\varepsilon}$  para todo  $\varepsilon > 0$ .*

*Demonstração.* Suponha que  $P \neq NP$ . Dado um grafo  $G = (V, E)$  uma instância de CONJUNTO INDEPENDENTE. Seja  $M$  a instância para MAXSIM- $j$ -NOTAS obtida em tempo polinomial no Teorema 3.2.1. Dado  $\varepsilon > 0$  suponha, por absurdo, que exista um algoritmo de

aproximação em tempo polinomial  $A$  para MAXSIM- $j$ -NOTAS com  $R_A \leq n^{1-\varepsilon}$ . Considere  $M = (\{H_1, \dots, H_n\}, 0, 1)$ , a instância de MAXSIM- $j$ -NOTAS obtida a partir da instância  $G = (V, E)$  de CONJUNTO INDEPENDENTE com a transformação de NP-completude definida no Teorema 3.2.1. Seja  $y$  a solução viável para MAXSIM- $j$ -NOTAS em  $M = (\{H_1, \dots, H_n\}, 0, 1)$  obtida pelo algoritmo  $A$ . Assim,  $\frac{Opt_{\text{MAXSIM-}j\text{-NOTAS}}(M)}{c(y)} = R_A(M, y) \leq R_A \leq n^{1-\varepsilon}$ . Do Teorema 3.2.1,  $Opt_{\text{MAXSIM-}j\text{-NOTAS}}(M) = Opt_{\text{CONJUNTO INDEPENDENTE}}(G)$  e existe um conjunto independente  $R \subset V$  em  $G$ , tal que  $|R| = c(y)$ .

Assim,  $\frac{Opt_{\text{CONJUNTO INDEPENDENTE}}(G)}{|R|} = \frac{Opt_{\text{MAXSIM-}j\text{-NOTAS}}(M)}{c(y)} \leq n^{1-\varepsilon}$ . Assim, pelo Teorema 3.2.6,  $P=NP$  uma contradição com nossa hipótese. Portanto, não existe algoritmo de aproximação em tempo polinomial para MAXSIM- $j$ -NOTAS com fator de aproximação absoluto menor ou igual a  $n^{1-\varepsilon}$ .  $\square$

### 3.2.2 Um resultado de complexidade parametrizada

Seja  $\mathcal{H}$  uma partitura. Para  $H \in \mathcal{H}$ , definimos  $N_{\mathcal{H}}(H)$  como a coleção de pautas de  $\mathcal{H}$  que tocam em algum tempo com  $H$  e  $N_{\mathcal{H}}[H] = N_{\mathcal{H}}(H) \cup \{H\}$ . Seja  $\Delta(\mathcal{H}) = \max_{H \in \mathcal{H}} |N_{\mathcal{H}}(H)|$ . A seguir exibimos um algoritmo FPT para SIM- $j$ -NOTAS no parâmetro  $\Delta(\mathcal{H})$  e  $k$ .

**Lema 3.2.1.** *Dada  $M = (\mathcal{H}, 0, 1, k)$  uma instância de SIM- $j$ -NOTAS e  $H \in \mathcal{H}$  então, se existe solução para  $M$ , então existe solução  $y$  com  $H \in y$  ou  $N_{\mathcal{H}}(H) \cap y \neq \emptyset$ .*

**Teorema 3.2.8.** *SIM- $j$ -NOTAS é FPT com respeito ao número máximo  $\Delta$  de pautas que tocam em cada tempo e ao tamanho  $k$  da solução, quando  $j = 1$  e  $p = 0$ . Mais especificamente existe um algoritmo para SIM- $j$ -NOTAS que roda em tempo  $O((\Delta + 1)^k \cdot n^2)$ .*

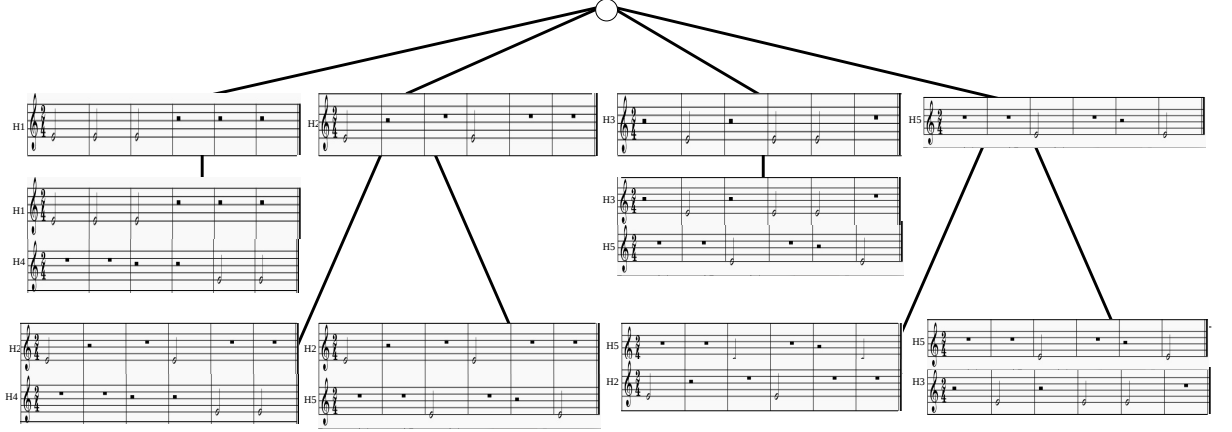
*Demonstração.* Seja  $M = (\mathcal{H}, 0, 1, k)$  uma instância de SIM- $j$ -NOTAS. O algoritmo considera uma árvore  $(\Delta + 1)$ -ária de execução. Cada nó  $v$  da árvore contém uma solução parcial  $S_v$  que contém por sua vez algumas pautas de  $\mathcal{H}$  e um conjunto corrente  $\mathcal{H}_v$ , de pautas que ainda podem fazer parte da solução. O nó raiz  $r$  contém  $S_r = \emptyset$  e  $\mathcal{H}_r = \mathcal{H}$ . Um nó  $v$  é um nó folha se e somente se  $|S_v| = k$  (e concluímos que temos uma instância SIM) ou  $\mathcal{H}_v = \emptyset$  (e a solução corrente não pode ser aumentada).

Seja  $v$  um nó não folha e seja  $H \in \mathcal{H}_v$ . Como, pelo Lema 3.2.1, se existe solução, então existe uma solução com  $H$  ou com um de seus vizinhos, os filhos de  $v$  representarão soluções parciais onde cada um destes elementos,  $H$  ou um de seus vizinhos, é adicionado. Ou seja,  $v$  possui um filho  $w$  com  $S_w = S_v \cup \{H\}$  e  $\mathcal{H}_w = \mathcal{H}_v \setminus N_{\mathcal{H}_v}[H]$  e, para cada  $H' \in N_{\mathcal{H}_v}(H)$ ,  $v$  possui um filho  $x$  com  $S_x = S_v \cup \{H'\}$  e  $\mathcal{H}_x = \mathcal{H}_v \setminus N_{\mathcal{H}_v}[H']$ .

Pela construção, em cada nível adicional da árvore a cardinalidade de  $S$  é incrementada em 1 unidade. Assim, caso exista nó  $v$  com  $|S_v| = k$ , temos uma instância SIM. Como a profundidade de um nó  $v$  corresponde a  $|S_v|$ , temos que a altura máxima

da árvore é  $k$  e portanto temos  $O((\Delta + 1)^k)$  nós na árvore. Assumindo que  $N_{\mathcal{H}}(H)$  foi pré-computado para cada  $H$ , podemos executar as operações de cada nó da árvore em tempo  $n^2$ . Para a conveniência do leitor, nós oferecemos na Figura 38 um exemplo de execução do algoritmo.  $\square$

Figura 38 - Algoritmo FPT em  $\Delta$  e em  $k$  para o problema SIM- $j$ -NOTAS.



Legenda: Exemplo de construção de árvore de execução do algoritmo FPT para a instância  $M = (\mathcal{H}, 0, 1, k)$  de SIM- $j$ -NOTAS da Figura 37a, para algum  $k \geq 2$ . Neste caso, temos  $\Delta = 3$  e, caso  $k = 2$ , obtemos resposta SIM e, se  $k > 2$ , resposta NÃO.

Fonte: O autor, 2024.

### 3.3 Problema das pautas com limitação de velocidade de transição

No problema ARR-CON apresentado neste capítulo, queremos evitar que o arranjo tenha intervalos dissonantes, no segundo problema, SIM- $j$ -NOTAS, garantimos que a quantidade de notas tocadas simultaneamente possa ser executada pela quantidade de instrumentos. Nesta seção queremos garantir que os instrumentos do arranjo, nos quais as notas tocam com uma duração mínima, não comecem a tocar durante a duração de outra nota do arranjo. Apesar de ser bem diferente, é um problema com uma redução e complexidade semelhante à do problema ARR-CON.

O problema de decisão VELOCIDADE DE TRANSIÇÃO LIMITADA foi introduzido por Demaine e Moses [1]. VELOCIDADE DE TRANSIÇÃO LIMITADA consiste em uma instância  $(\mathcal{H}, t, p)$  com um conjunto de pautas  $\mathcal{H} = \{H_1, \dots, H_n\}$ , tal que em cada pauta  $H_i$  cada nota tem uma duração de pelo menos  $t$  tempos e um percentual  $0 \leq p\% \leq 1$ . A pergunta é se existe uma subcoleção de pautas  $\mathcal{H}' \subset \mathcal{H}$ , tal que  $p\%$  das notas de  $\mathcal{H}$  são tocadas e nenhuma nota de  $\mathcal{H}'$  começa a ser tocada durante a duração de outra nota de  $\mathcal{H}'$ .

Assim como os outros problemas trabalhados até agora, formalizamos uma generalização do problema a seguir. O problema original já foi demonstrado NP-completo por

Demaine e Moses [1]. Ainda assim, faremos a redução do problema generalizado que será utilizada na subseção 3.3 para abordar a versão de otimização.

VELOCIDADE DE TRANSIÇÃO LIMITADA GERAL (VTL)

**Instância:**  $M = (\mathcal{H}, t, p, s, k)$  onde  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  é uma coleção de pautas,  $t$  é um inteiro, tal que cada nota toca com duração de pelo menos  $t$  tempos,  $0 \leq p\% \leq 1$  um percentual e um par de inteiros  $s, k \geq 0$ .

**Pergunta:** Existe um subconjunto  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq s$ , tal que nenhuma nota de  $\mathcal{H}'$  começa a tocar durante a duração  $t$  de outra nota de  $\mathcal{H}'$  e em pelo menos  $k$  tempos,  $p\%$  das notas de  $\mathcal{H}$  são tocadas?

Nós observamos que uma instância  $M = (\mathcal{H}, p, t, s, k)$  de VELOCIDADE DE TRANSIÇÃO LIMITADA GERAL particular quando  $k$  é o número de tempos total de  $\mathcal{H}$  com  $s = 0$  é equivalente a uma instância  $(\mathcal{H}, p, t)$  do problema LIMITAÇÃO NA VELOCIDADE DE TRANSIÇÃO de Demaine e Moses [1].

**Teorema 3.3.1.** VTL é NP-completo.

*Demonstração.* VTL está em NP, pois um certificado  $\mathcal{H}'$  pode ser verificado em tempo polinomial checando se  $|\mathcal{H}'| \geq s$ , se em pelo menos  $k$  tempos são tocadas pelo menos  $p\%$  das notas e se em cada tempo.

Vamos usar o problema NP-completo MAX3SAT<sub>3</sub> [35] para provar que VTL é NP-completo, através de uma redução polinomial MAX3SAT<sub>3</sub>  $\propto$  VTL.

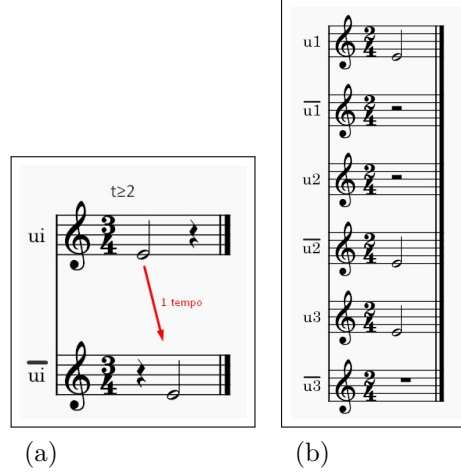
Dada uma instância  $I = ((U, C), k)$  de MAX3SAT<sub>3</sub> nós construímos uma instância  $M = (\mathcal{H}, p, t, s, k)$  de VTL com  $p = 30\%$  e  $t = 2$  usando a técnica “projeto de componente” de Karp [15].

Diferente do problema ARRANJO CONSONANTE GERAL, no problema VTL precisamos nos preocupar mais com tempos e compassos. Nessa transformação, estamos considerando  $t = 2$ , o que significa que não podemos continuar usando os compassos  $\frac{2}{4}$  nos primeiros  $n$  compassos, precisamos de mais espaço no compasso para podermos tocar a segunda nota um tempo depois da primeira e também para podermos tocar a primeira nota depois da segunda nota do compasso anterior. Então nos  $n$  primeiros compassos usaremos a fórmula de compasso  $\frac{3}{4}$ , nos  $m$  compassos seguintes podemos continuar usando a fórmula  $\frac{2}{4}$ .

Para cada um dos literais  $u_1, \bar{u}_1, u_2, \bar{u}_2, \dots, u_n, \bar{u}_n$  nós produzimos, respectivamente as pautas  $H_1, \dots, H_{2n}$ , tal que para cada  $i \in \{1, \dots, n\}$ :

1. no primeiro tempo do compasso  $i$  de  $H_{2i-1}$  seja tocada a nota  $E$  (Mi).
2. no segundo tempo do compasso  $i$  de  $H_{2i}$  seja tocada a nota  $E$  (Mi).

Figura 39 - Construção de definidor de verdade  $T_i$  (a) e testador de satisfação  $S_j$  (b).



Legenda: Exemplo de construção de variável gadget  $T_i$  (a), onde a literal associada  $u_i$  toca a nota E (Mi) no primeiro tempo do compasso e a literal  $\overline{u_i}$  toca a nota E (Mi) no segundo tempo do compasso, criando um conflito com a restrição de que nenhuma nota pode tocar na duração  $t$  de outra e impedindo que ambos os literais sejam verdadeiros ao mesmo tempo. E exemplo de cláusula gadget  $S_j$  (b), onde as pautas  $H_1$ ,  $H_4$  e  $H_5$  correspondentes às variáveis  $u_1$ ,  $\overline{u_2}$  e  $u_3$  tocam a nota Mi simultaneamente em  $\mathcal{H}$ , permitindo que de zero a três pautas possam ser tocadas simultaneamente em  $\mathcal{H}'$ .

Fonte: O autor, 2024.

Para cada  $j \in \{1, \dots, m\}$  com a cláusula  $c_j = (x \vee y \vee w) \in \{c_1, \dots, c_m\} = C$  as pautas  $H_x$ ,  $H_y$  e  $H_z$ , respectivamente, correspondentes aos literais  $x, y, z$ , possuem a nota  $E$  no primeiro tempo do compasso  $n + j$ .

Nós lembramos que  $t \geq 2$  e a pauta  $H_{2i-1}$  toca sempre no primeiro tempo do compasso, enquanto  $H_{2i}$  toca no segundo tempo, o que significa que há uma transição menor que  $t$ . Assim, as pautas  $H_{2i-1}$  e  $H_{2i}$ , em um certificado positivo  $\mathcal{H}'$ , não podem ser tocadas simultaneamente e para qualquer  $p > 0$  pelo menos uma delas precisa ser tocada porque  $p = 30\%$ . Nós adiantamos para o leitor que a resposta sim de  $(\mathcal{H}, p, s, k)$  implica a definição de uma atribuição de verdade satisfável onde  $u_i = V$  se e somente se  $H_{2i-1}$  pertence ao arranjo.

A construção é concluída pela instância  $(\mathcal{H}, p, t, s, k') = (\{H_1, \dots, H_{2n}\}, 3/10, 2, n, n + k)$ .

Observe que, na partitura  $\mathcal{H} = \{H_1, \dots, H_{2n}\}$ , temos um número total de compassos igual a  $n + m$ .

Suponha que  $I = ((U, C), k)$  é uma instância SIM de  $\text{MAX3SAT}_3$ . Então existe uma atribuição de verdade  $f: U \rightarrow \{V, F\}$ , tal que  $k$  ou mais cláusulas de  $C$  têm pelo menos um literal verdadeiro. Mostraremos que  $M = (\{H_1, \dots, H_{2n}\}, 3/10, 2, n, n + k)$  é também uma instância SIM escolhendo a pauta  $H_{2i-1}$  para adicionar para a subcoleção  $\mathcal{H}'$  se e somente se  $f(u_i) = V$ . Veja primeiro que  $|\mathcal{H}'| = n$ , portanto  $|\mathcal{H}'| \geq n$ . Observe que nos primeiros  $n$  compassos são tocadas 50% das notas de cada compasso, enquanto nos

Figura 40 - Exemplo de partitura  $\mathcal{H}$  para uma Instância

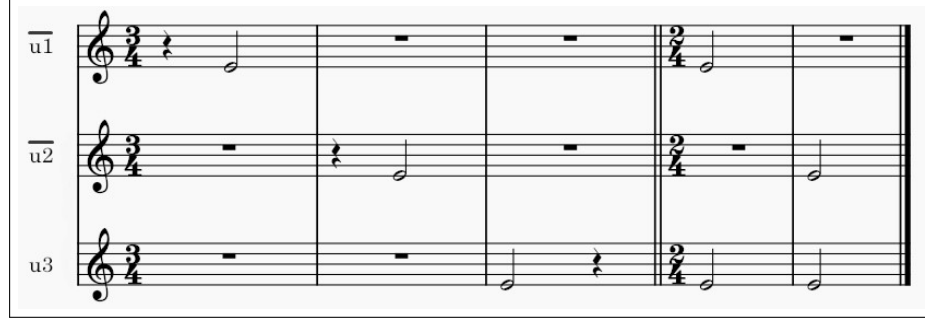
$M = (\mathcal{H}, p, t, s, k) = (\mathcal{H}, {}^3/_{10}, 2, n, n + k)$  de VTL, construída a partir de  $I = ((U, C), k)$  de  $\text{MAX3SAT}_3$  com  $|U| = n$  e  $|C| = m$ .

The image shows a musical score for six staves, labeled  $u_1, \overline{u_1}, u_2, \overline{u_2}, u_3, \overline{u_3}$  from top to bottom. The score is divided into two systems. The first system consists of three measures, and the second system consists of two measures. In each measure, different staves have notes, while others have rests. For example, in the first measure of the first system,  $u_1$  and  $\overline{u_1}$  have notes, while  $u_2, \overline{u_2}, u_3, \overline{u_3}$  have rests. This pattern continues across the measures, illustrating the construction of the partition  $\mathcal{H}$ .

Legenda: Exemplo de partitura  $\mathcal{H} = \{H_1, H_2, H_3, H_4, H_5, H_6\}$  de ARRANJO CONSONANTE, construída a partir de  $I = (U, C)$ , com  $U = \{u_1, u_2, u_3\}$  e  $C = \{(\overline{u_1}, u_2, u_3), (u_1, \overline{u_2}, u_3)\}$ , de  $\text{MAX3SAT}_3$ . No compasso  $i \in \{1, \dots, n\}$ , a pauta  $H_{2i-1}$  toca uma nota no primeiro tempo e  $H_{2i}$  toca uma nota no segundo tempo. Se  $\mathcal{H}'$  é um certificado para a resposta SIM de VTL para a instância  $M$  e  $t = 2$ , então  $H_{2i-1}$  e  $H_{2i}$  não podem ambos pertencer a  $\mathcal{H}'$ . Como  $|\mathcal{H}'| \geq n$ , e existem  $2n$  pautas, então para cada  $i$ , exatamente um de  $H_{2i-1}$  e  $H_{2i}$  pertence a  $\mathcal{H}'$ . Para cada  $j \in \{1, \dots, m\}$  com a cláusula  $c_j = (x \vee y \vee w) \in \{c_1, \dots, c_m\} = C$ , as pautas  $H_x, H_y$  e  $H_z$ , respectivamente, correspondentes aos literais  $x, y, z$ , possuem a nota  $E$  no compasso  $n + j$ .

Fonte: ‘O autor, 2024’.

Figura 41 - Exemplo de certificados para a resposta SIM de VTL e de  $\text{MAX3SAT}_3$  para a resposta SIM.



Legenda: Exemplo de certificado  $\mathcal{H}'$  para a resposta SIM de VTL para a instância  $M = (\mathcal{H}, 3/10, 2, n, n + k)$  associado à atribuição de verdade satisfatível  $\bar{u}_1 = \bar{u}_2 = u_3 = V$  para a instância  $I = (U, C)$  de  $\text{MAX3SAT}_3$ .

Fonte: 'O autor, 2024'.

últimos  $m$  compassos pelo menos  $3/10$  das notas são tocadas em pelo menos  $k$  compassos. Nenhuma nota é tocada nos tempos das cláusulas não satisfeitas. Assim, existe um subcoleção  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq n$ , onde pelo menos 30% das notas de  $\mathcal{H}$  são tocadas em pelo menos  $n + k$  compassos de  $\mathcal{H}'$  e nenhuma nota de  $\mathcal{H}'$  começa a ser tocada durante a duração de outra nota de  $\mathcal{H}'$ . E assim,  $M = (\{H_1, \dots, H_{2n}\}, 3/10, 2, n, n + k)$  é uma instância SIM de VTL.

Suponha que  $M = (\{H_1, \dots, H_{2n}\}, 3/10, 2, n, n + k)$  é uma instância SIM de VTL. Seja a subcoleção  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq n$ , onde pelo menos 30% das notas de  $\mathcal{H}$  são tocadas em  $n + k$  tempos tal que nenhuma nota de  $\mathcal{H}'$  começa durante a duração de outra nota de  $\mathcal{H}'$ . Sabemos que para todo  $i \in \{1, \dots, n\}$  nunca aparecem ao mesmo tempo  $H_{2i-1}$  e  $H_{2i}$  em  $\mathcal{H}'$ , e como  $|\mathcal{H}'| \geq n$ , temos que para cada  $i \in \{1, \dots, n\}$  aparece exatamente um entre  $H_{2i-1}$  ou  $H_{2i}$  em  $\mathcal{H}'$ . Nesse caso, sabemos mais,  $|\mathcal{H}'| = n$ . Dessa forma, damos a seguinte receita para produzir uma atribuição de verdade satisfatível  $f: U \rightarrow \{V, F\}$  para  $U$ , fazemos  $f(u_i) = V$  se e somente se  $H_{2i-1}$  aparece em  $\mathcal{H}'$ . Vamos provar que  $f$  satisfaz pelo menos  $k$  cláusulas de  $C$ . Como pelo menos  $\frac{3}{10}$  das notas em cada tempo de  $\mathcal{H}$  são tocadas em pelo menos  $n + k$  compassos, temos que os  $n$  primeiros compassos, onde são tocadas 50% das notas e mais  $k$  compassos adicionais que correspondem as cláusulas com pelo menos um literal verdadeiro. E assim, que  $f$  satisfaz pelo menos  $k$  cláusulas de  $C$ .

Dessa forma,  $I = (U, C)$  é uma instância SIM de  $\text{MAX3SAT}_3$  se e somente se  $M = (\{H_1, \dots, H_{2n}\}, 3/10, 2, n, n + k)$  é uma instância SIM de VTL. Assim,  $\text{MAX3SAT}_3 \propto \text{VTL}$  e dessa forma, VTL é NP-completo.  $\square$

### 3.3.1 Uma versão de maximização para VTL

Nós provaremos que o problema de otimização associado – MÁXIMA VELOCIDADE DE TRANSIÇÃO LIMITADA (MAXVTL) não possui um esquema de aproximação polinomial (a menos que  $P=NP$ ).

MÁXIMA VELOCIDADE DE TRANSIÇÃO LIMITADA (MAXVTL)

**Instância:** Coleção de pautas  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  e inteiro positivo  $t$ , onde cada nota de  $\mathcal{H}$  tem duração de pelo menos  $t$  tempos, percentual  $0 \leq p\% \leq 1$  e um inteiro  $s$ .

**Pergunta:** Determine  $k$  máximo para o qual existe um subconjunto  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \geq s$ , tal que pelo menos  $p\%$  das notas de  $\mathcal{H}$  são tocadas em pelo menos  $k$  tempos de  $\mathcal{H}'$ ?

**Teorema 3.3.2.** MAX VELOCIDADE DE TRANSIÇÃO LIMITADA (MAXVTL) é MAXSNP-difícil.

*Demonstração.* Para provar que MAXVTL é MAXSNP-difícil, provamos que MAX3SAT $_{\frac{3}{3}}$  L-reduz para MAXVTL.

Dado  $I = ((U, C), k)$  uma instância de MAX3SAT $_{\frac{3}{3}}$ . A partir do Teorema 3.1.1 temos que  $I = ((U, C), k)$  é uma instância SIM de MAX3SAT $_{\frac{3}{3}}$  se e somente se  $M = (\{H_1, \dots, H_{2n}\}, \frac{3}{10}, 2, n, n+k)$  é uma instância SIM de VTL. Em outras palavras, significa que quando  $k$  cláusulas são satisfeitas em  $I$ ,  $\mathcal{H}'$  possui  $n+k$  compassos onde cada nota e acorde dura pelo menos  $t$  tempos e são tocadas pelo menos  $\frac{3}{10}$  das notas em cada um desses  $n+k$  compassos. Assim,  $OPT_{\text{MAXVTL}}(M) = n + OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I)$ .

Vamos provar agora a primeira parte da L-redução, isto é, que existe  $\alpha > 0$ , tal que  $OPT_{\text{MAXVTL}}(M) \leq \alpha OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I)$ . A seguir, vamos usar o Teorema 3.3.1 e o Lema 3.1.1 para provar a sequência de desigualdades necessárias. Assim, temos que  $OPT_{\text{MAXVTL}}(M) = n + OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) \leq \frac{3}{2}m + OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) \leq \frac{3}{2} \left( \frac{4}{3} OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) \right) + OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) = 3 OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I)$ . O que mostra que  $\alpha = 3$  resolve.

Vamos provar a segunda parte da L-redução, isto é, que dada uma solução  $s_M$  de custo  $|s_M|$  para  $M$ , existe um algoritmo tempo polinomial  $g$  que obtém uma solução  $g(s_M) = s_I$  de custo  $|s_I|$  para  $I$ , tal que existe  $\beta > 0$  onde  $|OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) - |s_I|| \leq \beta |OPT_{\text{MAXVTL}}(M) - |s_M||$ .

Dada  $s_M$  uma solução viável de custo  $|s_M|$  para  $M$ . Porque  $s_M$  é viável, não existem notas que comecem durante a duração de outras notas de  $s_M$ . Se para algum  $i \in \{1, \dots, n\}$  nem a pauta  $H_{2i-1}$  nem a pauta  $H_{2i}$  pertencem a  $s_M$ , então nós modificamos  $s_M$ , ampliando  $s_M$ , inserindo  $H_{2i-1}$  em  $s_M$ , aumentando o número de compassos de  $s_M$ , onde nenhuma nota começa durante a duração de outra nota. Definimos em tempo polinomial, uma solução viável  $s_I$  para  $I$ , para cada  $i \in \{1, \dots, n\}$ , fazendo  $u_i = V$  se e somente se  $H_{2i-1}$  pertence a  $s_M$ .

Dessa forma, se  $k$  é o número de cláusulas satisfeitas por  $s_I$ , então  $|s_M| = n + k$ . Então  $|OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) - |s_I|| = |OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) - k| = |n + OPT_{\text{MAX3SAT}_{\frac{3}{3}}}(I) - (n + k)| = |OPT_{\text{MAXVTL}}(M) - |s_M||$ . O que mostra que  $\beta = 1$  resolve.  $\square$

### 3.3.2 Complexidade parametrizada de VTL

Nesta subseção, iremos demonstrar que VTL é  $W[1]$ -difícil no tamanho  $k$  da solução mostrando que CONJUNTO INDEPENDENTE reduz para VTL.

CONJUNTO INDEPENDENTE (CI)

**Instância:** Grafo  $G = (V, E)$  e inteiro positivo  $k$ .

**Pergunta:** O grafo  $G$  contém um conjunto independente  $S \subseteq V$  com  $|S| \geq k$ , ou seja, um conjunto  $S \subseteq V$ , tal que para cada  $uv \in E$  ou  $u \notin S$  ou  $v \notin S$ ?

**Teorema 3.3.3** (Downey e Fellows [33]). CONJUNTO INDEPENDENTE é  $W[1]$ -completo no tamanho  $k$  da solução.

**Teorema 3.3.4.** VTL é  $W[1]$ -difícil no tamanho  $k$  da solução.

*Demonstração.* Para provar que VTL é  $W[1]$ -difícil no tamanho  $k$  da solução, fornecemos uma redução parametrizada de CI parametrizado pelo tamanho do conjunto independente parametrizado.

Figura 42 - Instância de CONJUNTO INDEPENDENTE e VTL.



Legenda: Exemplo para a transformação parametrizada de CONJUNTO INDEPENDENTE para VTL, onde a instância  $M = (\mathcal{H}, p, t, k) = (\mathcal{H}, 0, 2, 2)$  de VTL (43b) é obtida a partir da instância  $(G = (V, E), k) = (G = (V, E), 2)$  de CONJUNTO INDEPENDENTE((a)).

Fonte: O autor, 2024.

Dada uma instância  $(G = (V, E), k)$  com  $|V| = n$  e  $|E| = m$  de CONJUNTO INDEPENDENTE nós construímos uma instância  $(\mathcal{H}, p, t, s, k) = (\mathcal{H}, 0, 2, 0, k)$  tal que  $G$  tem um conjunto independente  $S$  com  $|S| \geq k$  se e somente se  $(\mathcal{H}, 0, 2, 0, k)$  é uma instância SIM de VTL.

No problema VTL uma nota pode começar simultaneamente, mas não pode começar a tocar durante a duração  $t$  de outra nota. Portanto, consideramos que os compassos nessa construção tem uma duração de 3 tempos e  $t = 2$ .

Para cada vértice  $v_1, v_2, \dots, v_n \in V$  nós produzimos, respectivamente as pautas  $H_1, H_2, \dots, H_n \in \mathcal{H}$  e para cada aresta  $e_j = v_i v_\ell$ ,  $H_i$  toca a nota E (mi) no primeiro tempo do compasso  $o$  e  $H_\ell$  toca a nota E (mi) no segundo tempo do compasso  $o$ ,  $o \in \{1, \dots, m\}$ . Como as notas não são simultâneas e são tocadas a uma distância em tempos menor que  $t$ , isso cria um conflito. Assim, para cada aresta  $v_i v_\ell \in E$ , como em cada compasso  $H_i$  e  $H_\ell$  tocam notas não simultâneas e tocadas a uma distância em tempos menor que  $t$ , apenas uma das pautas,  $H_i$  ou  $H_\ell$  pode pertencer a solução viável  $\mathcal{H}$ . A construção é concluída com instância  $(\mathcal{H}, p, t, s, k') = (\{H_1, \dots, H_n\}, 0, 2, 0, k)$ .

Figura 43 - Instâncias de CONJUNTO INDEPENDENTE e de VTL.



Legenda: Arranjo  $\mathcal{H}'$  de VTL com duas pautas  $\{H_1, H_4\}$  associado ao conjunto independente

$S = \{v_1, v_4\}$  de  $G$  de CONJUNTO INDEPENDENTE

Fonte: O autor, 2024.

Suponha que  $(G = (V, E), k)$  é uma instância SIM de CONJUNTO INDEPENDENTE. Então existe um conjunto independente  $S$  em  $G$  de tamanho  $|S| \geq k$ . Nós damos uma receita para mostrar que  $(\{H_1, \dots, H_n\}, 0, 2, 0, k)$  é também uma instância SIM de VTL escolhendo a pauta  $H_i$  para adicionar para a subcoleção  $\mathcal{H}'$  se e somente se  $v_i \in S$ . Para a conveniência do leitor, oferecemos na Figura 43 o certificado associado de uma instância de VTL construída a partir de uma instância de CONJUNTO INDEPENDENTE. Observe que no  $x$ -ésimo compasso, a aresta  $e_x = v_i v_\ell$ , pelo menos uma das pautas  $H_i$  ou  $H_\ell$ , não pode estar em  $\mathcal{H}'$ , porque tocam notas não simultâneas e tocadas a uma distância em tempos menor que  $t$ . Assim como  $v_i$  e  $v_\ell$  extremidades da aresta  $v_i v_\ell$  não podem estar ambos em  $S$ . Assim, existe uma subcoleção  $\mathcal{H}' \subset \mathcal{H}$  onde nenhuma nota toca durante a duração  $t$  de outra e  $|\mathcal{H}'| \geq k$ .

Suponha que  $(\{H_1, \dots, H_n\}, 0, 2, 0, k)$  é uma instância SIM de VTL. Então existe uma subcoleção  $\mathcal{H}' \subset \mathcal{H}$ , tal que nenhuma nota toca durante a duração  $t$  de outra e  $|\mathcal{H}'| \geq k$ . Sabemos que para cada aresta  $v_i v_\ell$ , as pautas  $H_i$  e  $H_\ell$  não podem estar no mesmo arranjo pois tem notas tocadas a uma distância em tempos menor que  $t$  e portanto não podem estar ambos em  $\mathcal{H}'$ . Dessa forma, fornecemos a seguinte receita para produzir um conjunto independente  $S \subseteq V$  com tamanho  $|S| \geq k$ . Fazemos  $v_i \in S$  se e somente se  $H_i$  aparece em  $\mathcal{H}'$ . Vamos provar que  $|S| \geq k$  e que nenhum par de vértices em  $S$  são adjacentes. Assim, uma aresta  $v_i v_\ell \in E$  implica em ou  $H_i \notin \mathcal{H}'$  ou  $H_\ell \notin \mathcal{H}'$ . Portanto, se  $v_i v_\ell \in E$  e  $v_i \in S$ , então  $v_\ell \notin S$ . Assim,  $S$  é independente. Como  $|\mathcal{H}'| \geq k$ , então  $|S| \geq k$ .

Ou seja,  $(G = (V, E), k)$  é uma instância SIM de CONJUNTO INDEPENDENTE se e somente se  $(\{H_1, \dots, H_n\}, 0, 2, 0, k)$  é uma instância SIM de VTL. Assim, CONJUNTO INDEPENDENTE reduz para VTL. Então de acordo com a Definição 1.5.9, VTL é W[1]-difícil no tamanho  $k$  da solução.  $\square$

## 4 DOIS NOVOS PROBLEMAS

Os estudos sobre os três problemas originais de Demaine e Moses nos levaram naturalmente à formulação, nessa dissertação, de dois novos problemas musicais. Nas próximas duas seções apresentamos esses problemas, os quais contribuem para o estado-da-arte de arranjo musical no sentido de serem problemas que não se encontram abordados na literatura. O primeiro é o problema do ARRANJO PARA  $k$  INSTRUMENTOS (ou  $k$ -ARRANJO) e o segundo é o problema de PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS (PRE-INST).

### 4.1 Problema de ARRANJO PARA $k$ INSTRUMENTOS ( $k$ -ARRANJO)

Imaginamos um arranjador que deseja fazer uma versão popular de uma música clássica. Normalmente, partituras de orquestras possuem um grande número de instrumentos. Uma possível estratégia para o arranjador é agrupar diferentes instrumentos, que toquem um mesmo conjunto de notas, em um único instrumento de um conjunto com  $k$  instrumentos disponíveis sem deixar de tocar nenhuma nota.

O problema do  $k$ -ARRANJO envolve a tarefa de combinar diferentes pautas musicais para serem tocadas por um número limitado de instrumentos, preservando todas as notas da música original. Dado uma partitura  $\mathcal{H}$  com  $n$  pautas e um inteiro  $k$ , o objetivo é particionar essas pautas em  $k$  coleções, de forma que as pautas atribuídas à mesma coleção não toquem notas diferentes simultaneamente. Em outras palavras, se duas pautas ocupam o mesmo tempo em uma coleção, suas notas precisam ser idênticas, ou uma deve estar completamente contida na outra. Cada coleção representa uma pauta no arranjo, permitindo que um instrumento no arranjo toque as notas de vários instrumentos da música original.

Esse problema é relevante na criação de arranjos musicais quando há uma limitação no número de instrumentos disponíveis. O método descrito assegura que todas as notas da música original sejam tocadas, oferecendo uma solução ainda mais eficaz preservar a essência da obra do que a restrição do percentual  $p\%$  das notas tocadas dos três problemas originais. De fato, um arranjo feito com as restrições do  $k$ -ARRANJO pretende manter uma sonoridade compatível à original, sendo o objetivo principal reduzir, apropriadamente, o número de instrumentos com o mínimo de alteração na música.

Para ver uma aplicação no mundo real de  $k$ -arranjo, mostramos na Figura 44 um trecho da 5a. sinfonia de Beethoven. O trecho mostrado conta com uma partitura formada por um conjunto de instrumentos  $\mathcal{H}$  que possui  $|\mathcal{H}| = 12$  instrumentos. Uma partição de instrumentos  $\mathcal{H}'$  obtida a partir de  $\mathcal{H}$  com  $|\mathcal{H}'| = 4$  é mostrada na Figura 45 com a propriedade que a pauta de cada instrumento está contida em uma das 4 pautas de  $\mathcal{H}'$ .

Figura 44 - Trecho da sinfonia número 5 de Beethoven.

**Allegro con brio.**

Public Domain

Legenda: Partitura  $\mathcal{H}$  de um trecho da 5a. sinfonia de Beethoven com

$|\mathcal{H}| = 12$  instrumentos.

Fonte: Site Cantorion, acessado em 2024, disponível em: <<https://pt.cantorion.org/music/94/Sinfonia-n.%C2%BA-5-%28Symphony-No.-5%29-1.-Allegro-con-brio-%28typeset%29>>.

E em cada pauta, quando dois instrumentos tocam no mesmo tempo um deles contém o conjunto de notas do outro.

Figura 45 - Arranjo obtido do trecho da sinfonia número 5 de Beethoven.

Legenda: Arranjo  $\mathcal{H}'$ , que obtido a partir de  $\mathcal{H}$  com  $|\mathcal{H}'| = 4$  satisfazendo que a pauta de cada um dos  $|\mathcal{H}| = 12$  instrumentos de  $\mathcal{H}$  está contida em alguma pauta. E em uma mesma pauta, quando dois instrumentos tocam no mesmo tempo, nesse tempo um deles contém o conjunto de notas do outro.

Fonte: O autor, 2024.

Formalizamos este problema a seguir.

ARRANJO PARA  $k$  INSTRUMENTOS ( $k$ -ARRANJO)

**Entrada:** Coleção de pautas  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  e inteiro  $k \geq 1$ .

**Pergunta:** Existe uma partição  $\{A_1, \dots, A_k\}$  de  $\mathcal{H}$  em  $k$  coleções de pautas, tal que duas pautas  $H_i, H_j$  pertencem ao mesmo conjunto  $A_\ell$  satisfazem que se  $H_i$  e  $H_j$  tocam no mesmo tempo ou nesse tempo  $H_i$  contém as notas de  $H_j$  ou  $H_j$  contém as notas de  $H_i$ ?

**Teorema 4.1.1.**  $k$ -ARRANJO é NP-completo para qualquer  $k > 2$ .

*Demonstração.*  $k$ -ARRANJO está em NP, pois um certificado  $\{A_1, \dots, A_k\}$  pode ser verificado para a resposta SIM em tempo polinomial checando que sempre que  $H_i$  e  $H_j$  pertencem ao mesmo instrumento  $A_\ell$  e tocam ao mesmo tempo, então eles estão tocando as mesmas notas. Vamos usar o problema NP-completo  $k$ -COLORAÇÃO [15],  $k \geq 3$  para provar que  $k$ -COLORAÇÃO  $\propto k$ -ARRANJO.

**Problema 4.1.1.**  $k$ -COLORAÇÃO

**Instância:** Grafo  $G = (V, E)$ .

**Pergunta:**  $G$  é  $k$ -colorível? Ou seja, existe uma função  $f : V \rightarrow \{1, 2, \dots, k\}$ , tal que se  $uv \in E$ , então  $f(u) \neq f(v)$ ?

Dada uma instância  $G = (V, E)$  de  $k$ -COLORAÇÃO com  $n = |V|$  e  $m = |E|$ . Uma instância  $\mathcal{H} = \{H_1, \dots, H_n\}$  de  $k$ -ARRANJO é construída a partir de  $G$ , tal que

$G$  é  $k$ -colorível se e somente se  $\mathcal{H}$  é uma instância SIM de  $k$ -ARRANJO. Nossa partitura será formada por  $n$  pautas, cada uma com  $m$  tempos. A transformação polinomial  $T: k\text{-COLORAÇÃO} \rightarrow k\text{-ARRANJO}$  tempo  $O(nm)$  é assim definida:

$T(G = (V, E), k);$

1.  $r \leftarrow 1;$

2.  $E' \leftarrow E;$

3. Para  $i \leftarrow 1, \dots, n$  faça

Para cada aresta  $e_j$  de  $E'$  faça

Se  $v_i$  ocorre em  $e_j = v_i v_s$ , então

(a)  $H_i$  no tempo  $r$  tem a nota  $E$  (Mi);

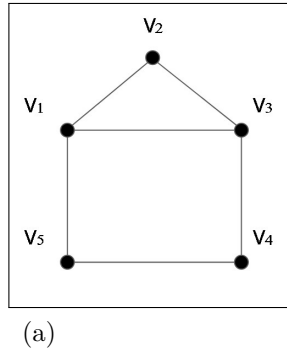
(b)  $H_s$  no tempo  $r$  tem a nota  $F$  (Fá);

(c)  $r \leftarrow r + 1;$

(d)  $E' \leftarrow E' \setminus \{e_j\};$

4. Retorne  $(\{H_1, \dots, H_n\}, k);$

Figura 46 - Instâncias de  $k$ -COLORAÇÃO e  $k$ -ARRANJO.



Legenda: Exemplo para a transformação  $k\text{-COLORAÇÃO} \propto \text{ARRANJO PARA } k$   
 INSTRUMENTOS ( $k$ -ARRANJO), onde a instância  $\mathcal{H}$  de 3-arranjo (46b) é obtida a  
 partir da instância  $(G = (V, E), 3)$  de  $k$ -COLORAÇÃO (46a).

Fonte: O autor, 2024.

Para a conveniência do leitor nós oferecemos na Figura 46 um exemplo da transformação. Na Figura 47 um arranjo com os 3 partes  $(\{H_1, H_4\}, \{H_2\}, \{H_3, H_5\})$  está

Figura 47 -  $k$ -COLORAÇÃO  $\propto$  ( $k$ -ARRANJO)

Legenda: 3-arranjo  $(A_1, A_2, A_3) = (\{H_1, H_4\}, \{H_2, H_5\}, \{H_3\})$  em (47b) associado a 3-coloração da instância  $(G = (V, E), 3)$  (47a).

Fonte: O autor, 2024.

associado à 3-coloração de  $G$  na qual aos vértices  $v_1, v_4$  foi atribuída a cor 1, ao vértice  $v_2$  foi atribuída a cor 2 e aos vértices  $v_3, v_5$  foi atribuída a cor 3.

Uma propriedade particular e crucial da construção é que:  $H_i$  e  $H_j$  tocam em um mesmo tempo se e somente se  $H_i$  e  $H_j$  tocam notas diferentes se e somente se  $v_i v_j \in E$ .

Suponha que  $G$  é uma instância SIM de  $k$ -COLORAÇÃO. Então  $G$  é  $k$ -colorível, ou seja, existe uma função  $f: V \rightarrow \{1, 2, \dots, k\}$ , tal que se  $uv \in E$ , então  $f(u) \neq f(v)$ . Mostramos que  $\mathcal{H}$  é uma instância SIM de  $k$ -ARRANJO. Para isso escolhemos a pauta  $H_i$  para adicionar a parte  $A_\ell$  se e somente se o vértice  $v_i$  tiver sido colorido com a cor  $\ell$ ,  $\ell \in \{1, 2, \dots, k\}$ . Suponha duas pautas  $H_i$  e  $H_j$  pertencem ao mesmo  $A_\ell$ . Vamos provar que quando  $H_i$  e  $H_j$  tocam ao mesmo tempo (Eles não tocam), eles tocam o mesmo conjunto de notas. Como  $H_i$  e  $H_j$  pertencem a parte  $A_\ell$  a cor  $\ell$  foi atribuída para  $v_i$  e  $v_j$  e assim  $v_i v_j \notin E$ . Logo,  $H_i$  e  $H_j$  não tocam ao mesmo tempo. E portanto, por vacuidade, quando  $H_i$  e  $H_j$  tocam ao mesmo tempo eles tocam o mesmo conjunto de notas.

Suponha que  $\mathcal{H}$  é uma instância SIM de  $k$ -ARRANJO. Seja  $\{A_1, \dots, A_k\}$  um  $k$ -arranjo para  $\mathcal{H}$ . Vamos definir uma  $k$ -coloração  $f: V \rightarrow \{1, \dots, k\}$  para  $G$  atribuindo a cor  $f(v_i) = \ell \in \{1, \dots, k\}$  ao vértice  $v_i \in V$  se e somente se  $H_i \in A_\ell$ . Vamos provar que vértices adjacentes receberam cores diferentes. Suponha que  $u_i v_j \in E$ . Portanto, existe um tempo em que  $H_i$  toca ao mesmo tempo que  $H_j$  e tocam notas diferentes: E (Mi) e F (Fá). Logo, existem  $r, s \in \{1, \dots, k\}$  com  $r < s$ , tal que  $H_i \in A_r$  e  $H_j \in A_s$ . E portanto,  $v_i$  recebeu a cor  $r$  e  $v_j$  recebeu a cor  $s$  com  $r \neq s$ . E assim, a atribuição dada é uma  $k$ -coloração para  $G$ . Assim,  $\mathcal{H}$  é uma instância SIM de  $k$ -ARRANJO se e somente se  $G$  é  $k$ -colorível. Portanto,  $k$ -ARRANJO é NP-completo.  $\square$

**Corolário 4.1.1.**  $k$ -ARRANJO é paraNP-difícil com respeito ao parâmetro tamanho  $k$  da solução.

Exploraremos novamente a conexão entre  $k$ -ARRANJO e  $k$ -COLORAÇÃO, para obtenção de um resultado de inaproximabilidade. Utilizaremos o resultado a seguir, que

considera o problema COLORAÇÃO, que é a versão de minimização do número de cores do problema de coloração em grafos.

**Teorema 4.1.2** (Zuckerman 2006 [34]). *Se  $P \neq NP$  e  $A$  é um algoritmo de aproximação em tempo polinomial para COLORAÇÃO, então  $R_A > n^{1-\varepsilon}$  para todo  $\varepsilon > 0$ .*

Seja ARRANJO o problema de minimização associado ao  $k$ -ARRANJO, isto é, dada uma partitura  $\mathcal{H}$ , desejamos determinar o menor  $k$  para o qual  $\mathcal{H}$  é uma instância SIM para  $k$ -ARRANJO. Note que, se  $\mathcal{H}$  possui  $n$  pautas, então  $k \leq n$ .

**Teorema 4.1.3.** *Se  $P \neq NP$  e  $A$  é um algoritmo de aproximação em tempo polinomial para ARRANJO, então  $R_A > n^{1-\varepsilon}$  para todo  $\varepsilon > 0$ .*

*Demonstração.* A demonstração desse teorema se utiliza do resultado de Zuckerman [34], enunciado no Teorema 4.1.2 e do fato que  $OPT_{\text{COLORAÇÃO}}(G) = OPT_{\text{ARRANJO}}(M)$ .  $\square$

**Teorema 4.1.4.**  *$k$ -ARRANJO é polinomial se  $k \leq 2$ .*

*Demonstração.* Vamos provar agora que 2-ARRANJO pode ser resolvido em tempo polinomial. Para isso, basta fazer a redução polinomial 2-ARRANJO  $\propto$  2-COLORAÇÃO, pois 2-COLORAÇÃO é um problema polinomial.

Dada uma instância  $\mathcal{H} = \{H_1, \dots, H_n\}$  de 2-ARRANJO, uma instância  $G = (V, E)$  de 2-COLORAÇÃO é construída a partir de  $\mathcal{H}$ , tal que  $\mathcal{H}$  é uma instância SIM se e somente se a instância  $G$  é 2-colorível. A construção é feita da seguinte forma:

- Para cada pauta  $H_i$ , para  $i$  de 1 a  $n$  temos um vértice  $v_i$ .
- Para cada conflito entre as pautas  $H_i$  e  $H_j$ , cuja existência pode ser checada em tempo  $O(T)$ , temos uma aresta  $v_i v_j \in E$ .

Suponha que  $\mathcal{H} = \{H_1, \dots, H_n\}$  é uma instância SIM de 2-ARRANJO. Seja  $(A_1, A_2)$  uma partição para  $\mathcal{H}$ , tal que se  $H_i, H_j \in A_\ell, \ell \in \{1, 2\}$ , então em cada tempo as notas de  $H_i$  contém as notas de  $H_j$  ou as notas de  $H_j$  contém as notas de  $H_i$ . Mostramos que existe  $f: V \rightarrow \{1, 2\}$  uma 2-coloração para  $G$  colorindo o vértice  $v_i$  com a cor  $f(v_i) = \ell$  se e somente se a pauta  $H_i$  pertencer a parte  $A_\ell, \ell \in \{1, 2\}$ . Suponha dois vértices  $v_i$  e  $v_j$  com a mesma cor  $f(v_i) = f(v_j) = \ell$ . Vamos provar que  $v_i v_j \notin E$ . Como  $v_i$  e  $v_j$  têm a mesma cor atribuída  $\ell$ ,  $H_i$  e  $H_j$  pertencem à parte  $A_\ell$  e assim  $H_i$  e  $H_j$  em cada tempo não tocam notas diferentes simultaneamente. Assim,  $v_i v_j \notin E$ . Logo,  $f: V \rightarrow \{1, 2\}$  é uma 2-coloração para  $G$ . E portanto,  $G$  é 2-colorível.

Suponha que  $G = (V, E)$  seja 2-colorível. Seja  $f: V \rightarrow \{1, 2\}$  uma 2-coloração. Vamos definir um 2-arranjo  $(A_1, A_2)$  para  $\mathcal{H}$  onde  $H_i \in A_\ell, \ell \in \{1, 2\}$  se e somente se o vértice  $v_i$  tem a cor  $f(v_i) = \ell$  atribuída. Vamos provar que pautas com conflitos não pertencem a mesma parte  $A_\ell, \ell \in \{1, 2\}$ . Suponha, por absurdo, que  $H_i, H_j \in A_\ell$  e que

existe um tempo em que  $H_i$  e  $H_j$  tocam notas diferentes. Assim, pela definição de  $G$ ,  $v_i v_j \in E$ . Portanto, do fato que  $f$  é uma 2-coloração,  $v_i$  recebeu uma cor diferente de  $v_j$ , uma contradição porque  $H_i$  não poderia estar na mesma parte de  $H_j$ . Assim,  $(A_1, A_2)$  é um 2-arranjo.  $\square$

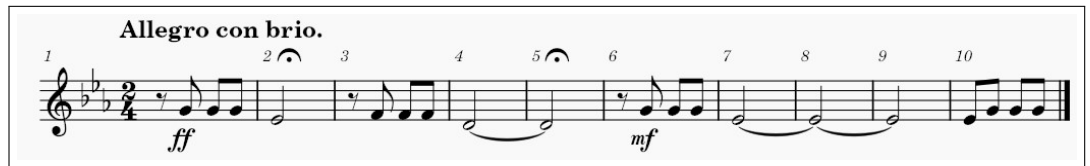
## 4.2 Problema do PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS

Neste problema, considera-se um arranjador que deseja reduzir o número de instrumentos, garantindo que, em cada tempo em que alguma pauta da música original é tocada, pelo menos uma das pautas do arranjo final também seja executada, evitando assim a criação de novos silêncios que não estavam presentes na música original.

A relevância do problema PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS surge quando um arranjador precisa determinar o número mínimo de instrumentos necessários para garantir que em cada tempo da música original que algum instrumento toca existe um instrumento representante do arranjo que toca também. Como em todos os tempos da música original, alguma pauta do arranjo também deve ser tocada, isso assegura que a melodia principal, ou pelo menos uma variação adequada, será reproduzida em cada tempo. Dessa forma, há a garantia de que os instrumentos disponíveis são suficientes para manter a melodia principal ou permitir que adaptações mantenham sua presença ao longo da execução.

Para uma aplicação desse método no mundo real, usamos de novo o trecho da 5a. sinfonia de Beethoven da Figura 44, mostrando na Figura 48 que existe uma coleção de pautas  $\mathcal{H}'$  com  $|\mathcal{H}'| = 1$  instrumento com a propriedade de que algum instrumento de  $\mathcal{H}'$  toca em cada tempo não silencioso de  $\mathcal{H}$ .

Figura 48 - Arranjo obtido do trecho da sinfonia número 5 de Beethoven.



Legenda: Arranjo  $\mathcal{H}'$ , obtido a partir de  $\mathcal{H}$  da Figura 44 com  $|\mathcal{H}'| = 1$  satisfazendo que algum instrumento de  $\mathcal{H}'$  toca em cada tempo não silencioso de  $\mathcal{H}$ .

Fonte: O autor, 2024.

Definimos, agora, o problema de decisão associado:

PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS - (PRE-INST)

**Entrada:** Coleção de pautas  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$  e inteiro positivo  $k > 0$ .

**Pergunta:** Existe uma coleção  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \leq k$ , tal que para cada tempo que alguma pauta de  $\mathcal{H}$  toca, existe  $H \in \mathcal{H}'$  que toca nesse tempo?

**Teorema 4.2.1.** PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS é NP-completo.

*Demonstração.* PRE-INST está em NP, pois um certificado  $\mathcal{H}'$  pode ser verificado em tempo polinomial checando se  $|\mathcal{H}'| \leq k$  e em cada tempo se existe  $H \in \mathcal{H}'$  que toca nesse tempo. Vamos usar o problema NP-completo COBERTURA DE VÉRTICES [15] para provar que COBERTURA POR VÉRTICES  $\propto$  PRE-INST.

**Problema 4.2.1.** COBERTURA POR VÉRTICES

**Instância:** Grafo  $G = (V, E)$  e inteiro positivo  $k$ .

**Pergunta:** Existe uma cobertura por vértices  $S$  de tamanho  $k$  ou menos para  $G$ , ou seja, um subconjunto  $S \subseteq V$  com  $|S| \leq k$  tal que, para cada aresta  $uv \in E$ , ou  $u \in S$  ou  $v \in S$ ?

Dado um grafo  $G = (V, E)$  com  $n = |V|$  e  $m = |E|$  e um inteiro positivo  $k$  uma instância de COBERTURA POR VÉRTICES. Constrói-se uma instância  $M = (\mathcal{H} = \{H_1, \dots, H_n\}, k)$  de PRE-INST a partir de  $(G, k)$ , tal que existe uma cobertura de vértices  $S$  de  $G$  se e somente se  $(\mathcal{H}, k)$  é uma instância SIM de PRE-INST.

Nossa partitura será formada por  $n$  pautas, cada uma com  $m$  tempos. Nossa transformação polinomial  $T: \text{COBERTURA POR VÉRTICES} \rightarrow \text{PRE-INST}$  é semelhante a da transformação polinomial  $O(mn)$  usada para provar que  $k$ -ARRANJO é NP-completo e é definida pelo seguinte algoritmo:

$T((G = (V, E), k));$

1.  $r \leftarrow 1;$

2.  $E' \leftarrow E;$

3. Para  $i \leftarrow 1, \dots, n$  faça

Para cada aresta  $e_j$  de  $E'$  faça

Se  $v_i$  ocorre em  $e_j = v_i v_s$ , então

(a)  $H_i$  no tempo  $r$  tem a nota  $E$  (Mi);

(b)  $H_s$  no tempo  $r$  tem a nota  $E$  (Mi);

(c)  $r \leftarrow r + 1;$

(d)  $E' \leftarrow E' \setminus \{e_j\};$

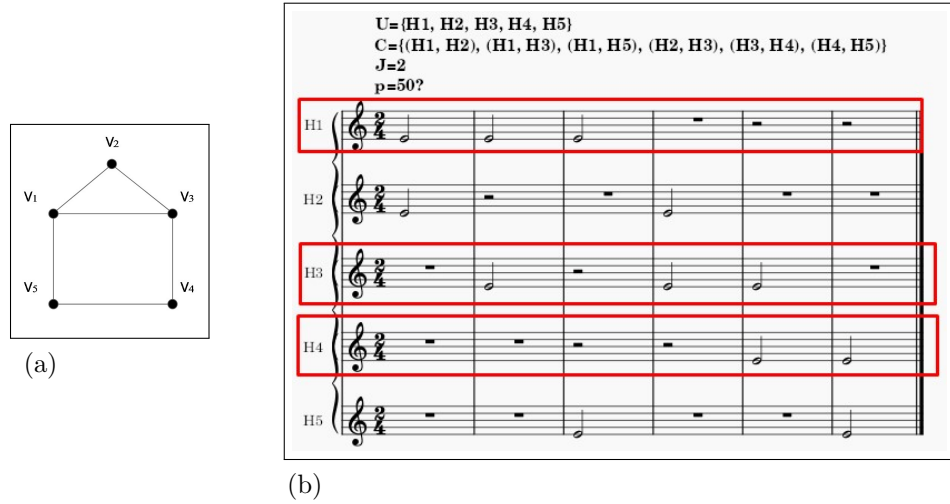
4. Retorne  $\{H_1, \dots, H_n\};$

Para a conveniência do leitor nós apresentamos na Figura 49 um exemplo da transformação.

Suponha COBERTURA DE VÉRTICES tem resposta SIM para  $(G, k)$ . Então existe uma cobertura de vértices em  $G$  de tamanho  $k$  ou menor, ou seja, um subconjunto  $V' \subseteq V$ ,

tal que  $|V'| \leq k$ , onde para cada aresta  $uv \in E$ , pelo menos um dos vértices  $u$  ou  $v$  pertence a  $V'$ . Nós damos uma receita para mostrar que  $M = (\{H_1, \dots, H_n\}, k)$  é também uma instância SIM, escolhendo a pauta  $H_i$  para adicionar para a subcoleção  $\mathcal{H}'$  se e somente se  $v_i \in V'$ . Como existem  $m$  compassos associados a cada aresta de  $E$ , temos que em cada tempo é tocado pelo menos uma nota por algum  $H \in \mathcal{H}'$ .

Figura 49 - COBERTURA POR VÉRTICES  $\propto$  PRE-INST



Legenda: Instância de PRE-INST em (49b) obtida em tempo polinomial a partir da instância  $(G = (V, E), 3)$  de COBERTURA POR VÉRTICES em (49a). A cobertura por vértices  $S = \{v_1, v_3, v_4\}$  associada ao conjunto de pautas  $\{H_1, H_3, H_4\}$  de tamanho 3 que toca em cada tempo da partitura  $\mathcal{H}$ .

Fonte: O autor, 2024.

Suponha que  $M = (\{H_1, \dots, H_n\}, k)$  é uma instância SIM de PRE-INST. Então existe  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \leq k$ , tal que para cada tempo existe  $H \in \mathcal{H}'$  que toca nesse tempo. Nós observamos que em cada tempo da música exatamente 2 pautas tem notas. Assim, em cada tempo uma das duas pautas tem que estar presente em  $\mathcal{H}'$ . Produz-se uma cobertura por vértices  $V'$  colocando  $v_i$  em  $V'$  se e somente se  $H_i$  toca no tempo  $j \in \{1, \dots, m\}$ . Dessa forma para cada aresta  $e \in E$  uma das suas extremidades pertencem a  $V'$ . E assim temos que  $V'$  é uma cobertura por vértices e  $|V'| \leq k$ . Dessa forma, COBERTURA POR VÉRTICES tem resposta sim para  $(G, k)$  se e somente se PRE-INST tem resposta sim para  $(\mathcal{H}, k)$ . Assim, PRE-INST é NP-completo.  $\square$

Uma redução análoga pode ser feita a partir do problema de COBERTURA POR CONJUNTOS, onde cada conjunto corresponde a uma pauta e, para o  $i$ -ésimo elemento do conjunto universo, os conjuntos que o contêm tocam uma nota no  $i$ -ésimo tempo. Tal redução, permite, por exemplo, concluir que, a menos que  $P=NP$ , PRE-INST não pode ser aproximado por uma razão de aproximação melhor que logarítmica em  $T$ , onde  $T$  é

o número de tempos não silenciosos da entrada [36]. Utilizando uma redução análoga na direção oposta, é possível obter um algoritmo com razão de aproximação  $O(\log T)$ , compondo a redução com o clássico algoritmo guloso para COBERTURA POR CONJUNTOS [37]. Obtemos, então, o seguinte resultado.

**Teorema 4.2.2.** *Existe  $O(\log T)$ -aproximação polinomial para PRE-INST, onde  $T$  é o número de tempos não silenciosos de uma entrada de PRE-INST e existe  $c > 0$  tal que não existe algoritmo com razão de aproximação melhor que  $c \log T$ , a menos que  $P=NP$ .*

#### 4.2.1 Um resultado parametrizado para PREENCHIMENTO POR INSTRUMENTOS

Utilizando hipergrafos, é possível determinar a complexidade parametrizada do problema PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS  $\tau$  LIMITADO, que é uma versão do problema restrita pelos parâmetros  $k$  e  $\tau$ .

PREENCHIMENTO DE TEMPOS POR INSTRUMENTOS  $\tau$  LIMITADO (PRE-INST $_{\tau}$ )

**Entrada:** Coleção de pautas  $\mathcal{H} = \{H_1, H_2, \dots, H_n\}$ , tal que  $H(M)$  tem tamanho máximo de aresta  $\tau$  e inteiro positivo  $k > 0$ .

**Pergunta:** Existe uma coleção  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \leq k$ , tal que para cada tempo que alguma pauta de  $\mathcal{H}$  toca, existe  $H \in \mathcal{H}'$  que toca nesse tempo?

**Corolário 4.2.1.** PRE-INST $_2$  é NP-completo.

*Demonstração.* Vem diretamente da construção do Teorema 4.2.1 □

**Lema 4.2.1.** *Seja  $M = (\mathcal{H}, k)$  uma instância de PRE-INST $_{\tau}$  e seja  $H(M) = (V, E)$  o hipergrafo de restrição de  $M$ . Suponha que  $M$  é um instância SIM e seja  $\mathcal{H}' \subset \mathcal{H}$  com  $|\mathcal{H}'| \leq k$ , tal que para cada tempo que alguma pauta de  $\mathcal{H}$  toca, existe  $H \in \mathcal{H}'$  que toca nesse tempo. Então, para todo  $e_t \in E$ ,  $e_t \cap \mathcal{H}' \neq \emptyset$ .*

**Teorema 4.2.3.** PRE-INST $_{\tau}$  é FPT com respeito ao tamanho  $k$  da solução. Mais especificamente, PRE-INST $_{\tau}$  pode ser resolvido em tempo  $O(\tau^k \cdot n^2)$ .

*Demonstração.* Seja  $M = (\mathcal{H}, k)$  uma instância de PRE-INST $_{\tau}$ . O algoritmo considera uma árvore  $\tau$ -ária. Cada nó  $v$  da árvore contém uma solução parcial  $S_v$  que contém por sua vez algumas pautas de  $\mathcal{H}$ . O nó raiz contém  $S_v = \emptyset$ . Um nó  $v$  será um nó folha se e somente se para toda aresta  $e_t \in E$  vale  $S_v \cap e_t \neq \emptyset$  (e nossa instância é SIM) ou  $|S_v| = k$  (e  $S_v$  já é grande demais e não pode ser estendido).

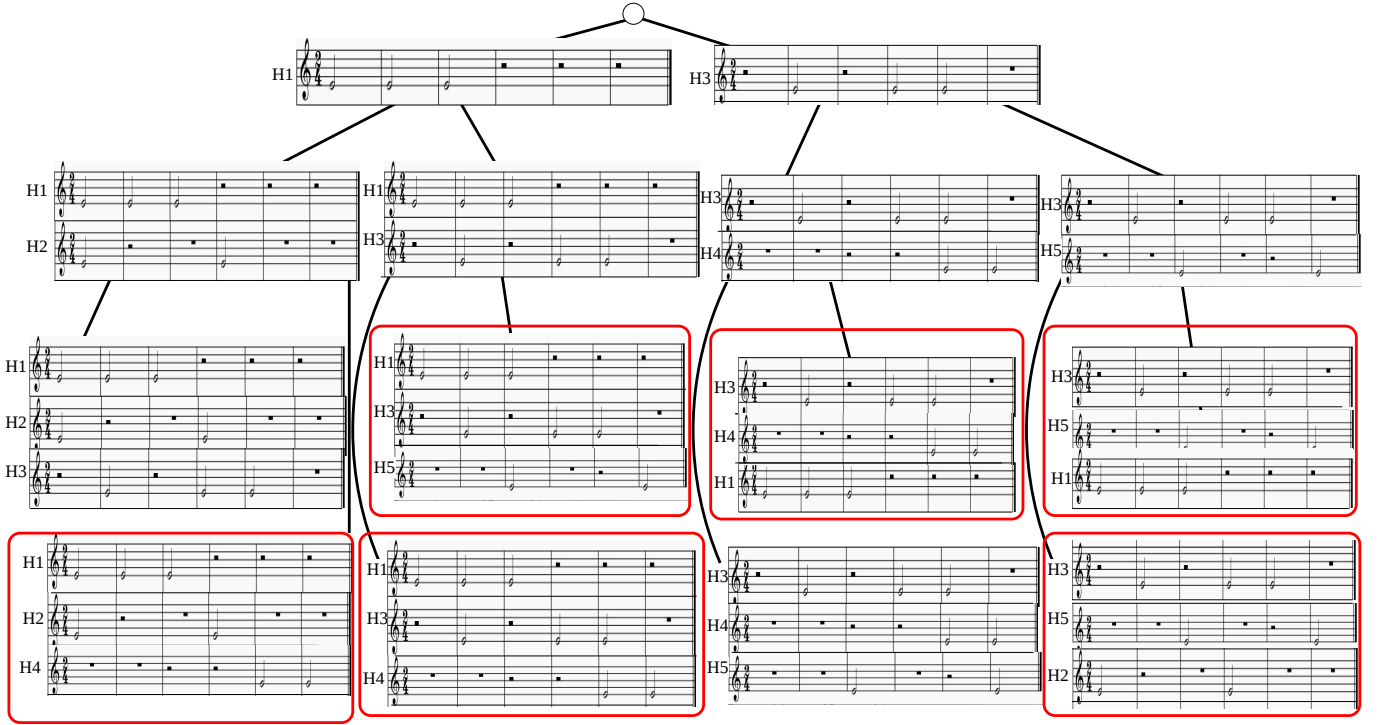
Seja  $v$  um nó não folha, seja  $t$  um tempo da partitura  $\mathcal{H}$  onde nenhuma pauta de  $S_v$  toca, ou seja  $e_t \cap S_v = \emptyset$ . Pelo Lema 4.2.1, sabemos que qualquer solução contém pelo menos um elemento de  $e_t$ . Então, cada filho de  $v$  corresponderá à extensão da solução

parcial  $S_v$  com um elemento de  $e_t$ , ou seja, para cada  $H \in e_t$ , teremos um filho  $w$  de  $v$  com  $S_w = S_v \cup \{H\}$ .

Pela construção, em cada nível adicional da árvore o valor de  $S_v$  é incrementado em 1 unidade e, pela definição das folhas, a árvore tem altura máxima  $k$ . Logo, temos no máximo  $O(\tau^k)$  nós e, caso a instância seja SIM, uma das folhas terá o certificado correspondente.  $\square$

É mostrada na Figura 50 um exemplo de execução do algoritmo.

Figura 50 - Algoritmo FPT no tamanho da solução  $k$  para o problema  $\text{PRE-INST}_2$ .



Legenda: Exemplo de construção de árvore de execução do algoritmo FPT no tamanho da solução para a instância  $M = (\mathcal{H}, k)$  de  $\text{PRE-INST}_2$  obtida a partir do grafo  $G = (V, E)$  da Figura 49.

Fonte: O autor, 2024.

## CONCLUSÃO

Até 2017, a literatura de produção de arranjos foi formada por heurísticas sem garantias de qualidade ou tempo de execução. Tais heurísticas podiam ser empiricamente comparadas e selecionadas entre si pelos músicos. O trabalho de Demaine e Moses [1] foi um marco que mostrou que os algoritmos arranjadores formulados por eles eram computacionalmente ineficientes.

Nesta dissertação, apresentamos versões mais gerais dos 3 problemas originais de Demaine e Moses [1], os quais denominamos ARRANJO-CONSONANTE GERAL,  $j$ -NOTAS SIMULTÂNEAS GERAL (MAXSIM- $j$ -NOTAS) e VELOCIDADE DE TRANSIÇÃO LIMITADA GERAL. Nós concebemos essas novas versões pensando que podíamos reduzi-las a partir de problemas NP-difíceis mais “ricos” em propriedades que os usados por Demaine e Moses no artigo original. As versões de otimização desses problemas tem propriedades importantes de MAXSNP-dureza, aproximabilidade, não aproximabilidade e parametrização. Isso nos permitiu mapear o comportamento dos problemas de otimização de arranjo musical.

Nós provamos a NP-completude das versões gerais relacionando os valores de ótimo dos problemas envolvidos. Usando essas relações, provamos que as versões de otimização ARRANJO-CONSONANTE GERAL e VELOCIDADE DE TRANSIÇÃO LIMITADA são MAXSNP-difícil, implicando que não têm esquema de aproximação polinomial, a menos que  $P=NP$ . Provamos também que os problemas ARRANJO CONSONANTE GERAL e VELOCIDADE DE TRANSIÇÃO LIMITADA GERAL são  $W[1]$ -difíceis no tamanho  $k$  da solução.

Provamos que não existe algoritmo de aproximação em tempo polinomial para MAXSIM- $j$ -NOTAS com fator de aproximação absoluto menor ou igual a  $n^{1-\varepsilon}$  para todo  $\varepsilon > 0$ . Nós consideramos uma versão particular de otimização de MAXSIM- $j$ -NOTAS com  $p = 0$  e  $j = 1$ . Provamos que MAXSIM- $j$ -NOTAS $\Delta 4$  – uma versão particular do problema MAXSIM- $j$ -NOTAS, onde cada pauta toca simultaneamente com no máximo 4 outras pautas – é um problema MAXSNP-difícil. Por outro lado, em relação a complexidade parametrizada, o problema MAXSIM- $j$ -NOTAS foi mostrado ser FPT com respeito ao número máximo  $\Delta$  – o número máximo de pautas que toca com uma pauta – e ao tamanho  $k$  da solução.

Apresentamos dois novos problemas de decisão  $k$ -ARRANJO e PREENCHIMENTO POR INSTRUMENTOS. Acreditamos que esses novos problemas possuem algum sentido e aplicação musical.

O problema  $k$ -ARRANJO que visa criar um arranjo com um número reduzido de pautas, sem que nenhuma nota seja omitida. Isso é alcançado ao combinar pautas que obedecem a certas restrições. O problema é NP-completo para qualquer  $k \geq 3$ . Provamos que a versão de minimização de  $k$ -ARRANJO não é aproximável com  $R_A \leq n^{1-\epsilon}$ , a menos que  $P = NP$ . O problema é paraNP-difícil em relação ao tamanho  $k$  da solução, para

qualquer  $k \geq 3$ .

O problema PREENCHIMENTO POR INSTRUMENTOS tem o objetivo de criar um arranjo com poucos instrumentos, sem gerar novos silêncios na música. Mostramos que o problema de decisão é NP-completo. Mostramos que a menos que  $P=NP$ , não há uma aproximação polinomial menor que  $O(\log T)$  para PREENCHIMENTO POR INSTRUMENTOS, onde  $T$  representa o número de tempos não silenciosos na entrada. Mostramos que o problema é FPT em relação ao parâmetro  $\tau$  – o número máximo de instrumentos que toca em conjunto com outro instrumento e ao tamanho  $k$  da solução.

Dentre os trabalhos futuros que pretendemos fazer está a determinação da complexidade de algoritmos exatos para todos os problemas tratados.

Inspirados nos trabalhos de Demaine e Moses, essa dissertação avançou ainda mais o estudo sobre a complexidade de tempo dos problemas de arranjos musicais. Além de estender e mapear os resultados de dificuldade para os problemas de otimização associados, imaginamos novos problemas e desenvolvemos os primeiros algoritmos polinomiais com garantia para enfrentar a complexidade.

## REFERÊNCIAS

- 1 DEMAINE, E. D.; MOSES, W. S. Computational complexity of arranging music. *The Mathematics of Various Entertaining Subjects: Research in Games, Graphs, Counting, and Complexity, Volume 2*, Princeton University Press, v. 2, p. 364, 2017. Citado 20 vezes nas páginas 4, 5, 10, 17, 18, 19, 20, 37, 39, 40, 42, 43, 44, 47, 48, 49, 58, 64, 65 e 84.
- 2 BIBBY, N. Tuning and temperament: Closing the spiral. *Music and mathematics: From Pythagoras to fractals*, Oxford University Press Oxford, p. 13–27, 2003. Citado 2 vezes nas páginas 10 e 26.
- 3 FERREIRA, I. M.; MACHADO, J. T. Algorithmic music composition: a survey. In: *2nd Conference on Nonlinear Science and Complexity*. [S.l.: s.n.], 2008. p. 1–18. Citado 3 vezes nas páginas 11, 13 e 37.
- 4 MILLER, S. D. Guido d'arezzo: Medieval musician and educator. *Journal of Research in Music Education*, SAGE Publications Sage CA: Los Angeles, CA, v. 21, n. 3, p. 239–245, 1973. Citado 2 vezes nas páginas 11 e 13.
- 5 MERSENNE, M. 1636–1637. harmonie universelle, contenant la théorie et la pratique de la musique, vol. 2. *Paris: S. Cramoisy*. Citado na página 12.
- 6 NOLAN, C. et al. On musical space and combinatorics: historical and conceptual perspectives in music theory. *Bridges*, 2000. Citado na página 12.
- 7 KOETSIER, T.; BERGMANS, L. *Mathematics and the divine: a historical study*. [S.l.]: Elsevier, 2004. Citado na página 12.
- 8 RUTTKAY, Z. Composing mozart variations with dice. *Teaching Statistics*, Wiley Online Library, v. 19, n. 1, p. 18–19, 1997. Citado na página 12.
- 9 BOWLES, E. Musicke's handmaiden: Or technology in the service of the arts. *The computer and music*, Cornell University Press Ithaca, p. 3–20, 1970. Citado na página 12.
- 10 ROADS, C. *The computer music tutorial*. [S.l.]: MIT press, 1996. Citado na página 12.
- 11 ROADS, C. Research in music and artificial intelligence. *ACM Computing Surveys (CSUR)*, ACM New York, NY, USA, v. 17, n. 2, p. 163–190, 1985. Citado na página 13.
- 12 PANDA, R. E. S. *Emotion-based analysis and classification of audio music*. Tese (Doutorado) — 00500:: Universidade de Coimbra, 2019. Citado na página 13.
- 13 ATHAYDE, E. de M. et al. Teoria dos grafos aplicada à análise musical. Universidade Federal de Minas Gerais, 2020. Citado na página 13.
- 14 GEEM, Z. W.; KIM, J. H.; LOGANATHAN, G. V. A new heuristic optimization algorithm: harmony search. *simulation*, Sage Publications Sage CA: Thousand Oaks, CA, v. 76, n. 2, p. 60–68, 2001. Citado na página 14.

- 15 KARP, R. Reducibility among combinatorial problems. In: MILLER, R.; THATCHER, J. (Ed.). *Complexity of Computer Computations*. [S.l.]: Plenum Press, 1972. p. 85–103. Citado 5 vezes nas páginas 14, 49, 65, 75 e 80.
- 16 GAREY, M. R.; JOHNSON, D. S. Computers and intractability. *A Guide to the*, 1979. Citado 3 vezes nas páginas 14, 58 e 62.
- 17 TOUSSAINT, G. T. Algorithmic, geometric, and combinatorial problems in computational music theory. *Proceedings of X Encuentros de Geometria Computacional*, Citeseer, p. 101–107, 2003. Citado 2 vezes nas páginas 17 e 37.
- 18 READ, R. C. Combinatorial problems in the theory of music. *Discrete Mathematics*, Elsevier, v. 167, p. 543–551, 1997. Citado na página 17.
- 19 DEMAINE, E. D. et al. The distance geometry of music. *Computational geometry*, Elsevier, v. 42, n. 5, p. 429–454, 2009. Citado 2 vezes nas páginas 17 e 37.
- 20 DOE, P. Tallis's 'spem in alium' and the elizabethan respond-motet. *Music & Letters*, JSTOR, v. 51, n. 1, p. 1–14, 1970. Disponível em: <<https://www.youtube.com/watch?v=iT-ZAAi4UQQ>>. Citado na página 17.
- 21 FARIA, L.; FIGUEIREDO, N.; SANTOS, V. F. dos. Problemas musicais-estrutura, propostas e complexidade. In: SBC. *Encontro de Teoria da Computação (ETC)*. [S.l.], 2024. p. 86–89. Citado na página 19.
- 22 FERREIRA, A. B. H. *Aurélio - Edição Especial - o Dicionário da Língua Portuguesa*. [S.l.]: Editora Positivo, 2004. Citado 4 vezes nas páginas 21, 23, 26 e 27.
- 23 GROVE music online. *Oxford University Press*. Retrieved September. Citado na página 23.
- 24 MÚSICA, H. G. da; BENNETT, R. Uma breve história da música. *Cadernos de Música da Universidade de Cambridge*. RJ: Jorge Zahar Ed, 1986. Citado na página 24.
- 25 TSUJI, K.; MÜLLER, S. C. Intervals and scales. In: *Physics and Music: Essential Connections and Illuminating Excursions*. [S.l.]: Springer, 2021. p. 47–70. Citado na página 26.
- 26 GILBERT, P. Chapter 16 - musical scales. In: GILBERT, P. (Ed.). *Physics in the Arts (Third Edition)*. Third edition. Academic Press, 2022. p. 271–284. ISBN 978-0-12-824347-3. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780128243473000169>>. Citado na página 26.
- 27 HELMHOLTZ, H. L. *On the Sensations of Tone as a Physiological Basis for the Theory of Music*. [S.l.]: Library of the University of California, 1895. Citado na página 26.
- 28 SZWARCFITER, J. L. Teoria computacional de grafos, 1ª Edição. *Editora Elsevier - Série Sociedade Brasileira de Computação*, 2018. Citado 2 vezes nas páginas 29 e 31.
- 29 SIPSER, M. Introduction to the theory of computation. *ACM Sigact News*, ACM New York, NY, USA, v. 27, n. 1, p. 27–29, 1996. Citado na página 31.

- 30 PAPADIMITRIOU, C. H. Computational complexity. In: *Encyclopedia of computer science*. [S.l.: s.n.], 2003. p. 260–265. Citado 4 vezes nas páginas 32, 49, 54 e 61.
- 31 FAUVEL, J.; FLOOD, R.; WILSON, R. J. *Music and mathematics: From Pythagoras to fractals*. [S.l.]: Oxford University Press, USA, 2006. Citado na página 37.
- 32 MITZENMACHER, M.; UPFAL, E. *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. [S.l.]: Cambridge university press, 2017. Citado na página 53.
- 33 DOWNEY, R. G.; FELLOWS, M. R. *Fundamentals of parameterized complexity*. [S.l.]: Springer, 2013. v. 4. Citado 2 vezes nas páginas 55 e 70.
- 34 ZUCKERMAN, D. Linear degree extractors and the inapproximability of max clique and chromatic number. In: *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*. [S.l.: s.n.], 2006. p. 681–690. Citado 2 vezes nas páginas 62 e 78.
- 35 PAPADIMITRIOU, C.; YANNAKAKIS, M. Optimization, approximation, and complexity classes. In: *Proceedings of the twentieth annual ACM symposium on Theory of computing*. [S.l.: s.n.], 1988. p. 229–234. Citado na página 65.
- 36 LUND, C.; YANNAKAKIS, M. On the hardness of approximating minimization problems. *Journal of the ACM (JACM)*, ACM New York, NY, USA, v. 41, n. 5, p. 960–981, 1994. Citado na página 82.
- 37 CHVATAL, V. A greedy heuristic for the set-covering problem. *Mathematics of operations research*, INFORMS, v. 4, n. 3, p. 233–235, 1979. Citado na página 82.