

UNIVERSIDADE FEDERAL DO PARANÁ

JEDIAN MARCOS BRAMBILLA

CLASSES LÍMITROFES PARA OS PROBLEMAS DE SATISFAZIBILIDADE

CURITIBA PR

2019

JEDIAN MARCOS BRAMBILLA

CLASSES LIMÍTROFES PARA OS PROBLEMAS DE SATISFAZIBILIDADE

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Ciência da Computação, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: Renato José da Silva Carmo.

CURITIBA PR

2019

RESUMO

Este trabalho de graduação apresenta ao leitor uma série de trabalhos na área da teoria da complexidade computacional, discutindo algumas reduções para o problema da Satisfazibilidade e a separação entre instâncias tratáveis e intratáveis do problema. Uma classe hereditária de grafos que representam fórmulas booleanas e os conceitos de classe limite e limítrofe são apresentados. Como resultado principal estudado, uma classe limítrofe, ou seja, uma sequência infinita de restrições do problema que mantém a NP-completude é identificada.

Palavras-chave: Satisfazibilidade (SAT). Complexidade Computacional. Classes limítrofes.

ABSTRACT

This undergraduate thesis introduces to the reader a series of works on computational complexity theory, discussing some reductions for the Satisfiability problem and the separation between tractable and intractable instances of the problem. A hereditary class of graphs that represent boolean formulas and the concepts of limit and boundary classes are shown. As main studied result, a boundary class, in other words, an infinite sequence of restrictions that keep the NP-completeness is identified.

Keywords: Satisfiability (SAT). Computational Complexity. Boundary Classes.

SUMÁRIO

1	INTRODUÇÃO	5
2	TERMINOLOGIA	6
2.1	TEORIA DOS GRAFOS	6
2.2	LÓGICA MATEMÁTICA	7
2.3	TEORIA DA COMPUTAÇÃO	7
3	SATISFAZIBILIDADE E RESTRIÇÃO.	8
3.1	RESTRIÇÃO	8
3.2	GRAFOS FÓRMULA	8
3.3	CLASSES HEREDITÁRIAS, LIMITES E LÍMITROFES DE GRAFOS	9
4	UMA CLASSE LÍMITROFE PARA SATISFAZIBILIDADE	10
4.1	UMA CLASSE LIMITE PARA SATISFAZIBILIDADE	10
4.2	MINIMALIDADE DE S	11
4.3	UNICIDADE DE S	12
5	CONCLUSÃO	14
	REFERÊNCIAS	15

1 INTRODUÇÃO

O estudo da complexidade de problemas computacionais envolve a transformação de problemas em outros, seja através da generalização, em que se busca uma maneira de descrever o problema de forma a abranger outros parecidos, ou da restrição, que, em contraste, torna o domínio do problema mais particular e exclusivo. Estas técnicas funcionam de forma que, ao resolvermos o novo problema, resolve-se também o primeiro.

Tais transformações permitem a análise e subsequente prova de que certos problemas podem ser transformados um nos outros de maneira em tempo polinomial [Karp, 1972], mesmo que não se conheça solução com tempo de execução polinomial para nenhum deles. Segue daí que, caso uma solução de tempo polinomial para algum deles surja, esta servirá para todos deste grupo. Estes problemas pertencem a uma classe conhecida como problemas NP-completos [Cook, 1971, Levin, 2010].

Em alguns casos, adicionar certas restrições a problemas desta classe gera problemas solúveis em tempo polinomial [Aspvall et al., 1979, Even et al., 1976]. Há, entretanto, casos em que o novo problema permanece sem solução de tempo polinomial conhecida e, além disso, ele continua NP-completo [Cook, 1971, Tovey, 1984, Lichtenstein, 1982]. Estes estudos induzem a reflexão sobre os limites destas restrições e o que separa instâncias de problemas tratáveis e intratáveis na computação teórica.

Este trabalho tem o objetivo de expor detalhadamente alguns resultados sobre a “fronteira” da complexidade do problema da Satisfazibilidade, com ênfase no resultado de [Lozin and Purcell, 2013], cuja abordagem utiliza conceitos da teoria dos grafos para provar uma sequência de restrições que mantêm o problema NP-completo.

O Capítulo 2 revisa as definições e conceitos necessários para a leitura e acompanhamento do trabalho, além de estabelecer a notação utilizada no decorrer do mesmo. O Capítulo 3 sintetiza a bibliografia, demonstrando resultados anteriores e justificando a abordagem e metodologia do trabalho. O Capítulo 4 apresenta os resultados estudados e os discute.

2 TERMINOLOGIA

Este capítulo apresenta ao leitor a terminologia comum utilizada no restante do trabalho. Os termos estão relacionados à Teoria dos Grafos, Teoria da Computação e Lógica Matemática.

2.1 TEORIA DOS GRAFOS

Grafos são estruturas combinatórias que podem ser utilizadas para representar objetos e relações entre os mesmos. Um *grafo* simples, não-direcionado G é um par $(V(G), E(G))$ onde $V(G)$ é um conjunto finito e $E(G) \subseteq \binom{V(G)}{2}$. Todo grafo mencionado neste trabalho é simples e não-direcionado.

Cada elemento de $V(G)$ é chamado de *vértice* de G e cada elemento de $E(G)$ é chamado de *aresta* de G . Se $a = \{u, v\}$ é uma aresta de G , então u e v são chamados de *pontas* da aresta a . Também se diz que u e v são *vizinhos* em G . A quantidade de vizinhos de um vértice é chamada de *grau* do vértice. Chamamos G de *grafo completo* quando $E(G) = \binom{V(G)}{2}$, ou seja, cada vértice de G é vizinho de todos os outros vértices de G . Denotamos por K_n um grafo completo com n vértices.

Um grafo H é *subgrafo* de um grafo G se $V(H) \subseteq V(G)$ e $E(H) \subseteq E(G)$, ou seja, se H pode ser obtido a partir de G através da remoção de vértices e/ou arestas de G . Um grafo H é *subgrafo induzido* de um grafo G se $V(H) \subseteq V(G)$ e $E(H) := E(G) \cap \binom{V(H)}{2}$, ou seja, se H pode ser obtido a partir de G através da remoção de vértices (e consequentemente das arestas cujas pontas são vértices que foram removidos) de G . Em caso contrário, dizemos que G é *livre* de H .

Um *passeio* P em G é uma sequência de vértices na qual vértices consecutivos são vizinhos em G . O primeiro e o último vértices de P são chamados de *pontas* de P e os demais vértices são chamados de *vértices internos* de P . O *tamanho* do passeio equivale a quantidade de arestas representadas nele e é denotado por $|P|$. Uma *trilha* em um grafo G é um passeio em G cujas arestas são todas distintas. Um *caminho* em um grafo G é um passeio em G cujos vértices são todos distintos. Um *ciclo* em um grafo G é uma trilha em G cujas pontas coincidem e cujos vértices internos são todos distintos.

Um grafo G é dito *cordal* se cada um de seus ciclos com tamanho maior ou igual a quatro possui uma *corda*, isto é, uma aresta que não faz parte do ciclo em questão, mas que liga dois vértices do ciclo.

O grafo G é *conexo* se, para todo $u, v \in V(G)$, existe um passeio em G de u a v , ou seja, existe um passeio em G cujas pontas sejam u e v , e G é *desconexo* em caso contrário. Um *componente* de um grafo G é um subgrafo conexo maximal de G . O grafo G é uma *árvore* se é conexo e não possui ciclos. O grafo G é uma *floresta* se cada um de seus componentes é uma árvore.

Um grafo G é dito *K-vértice-conexo* se permanece conexo quando removidos quaisquer $K - 1$ vértices, ou seja, é necessário remover pelo menos K vértices para que G se torne desconexo.

Dizemos que um grafo G é *bipartido* quando seus vértices podem ser particionados em dois conjuntos X e Y disjuntos de maneira que todas as arestas de G tenham uma ponta em X e outra em Y . Dizemos ainda que G é *bipartido completo* se para todo $u \in X, v \in Y$, u e v são vizinhos em G . Denotamos por $K_{n,m}$ um grafo bipartido completo com n vértices em X e m vértices em Y .

Um *isomorfismo* entre dois grafos G e H é uma bijeção entre $V(G)$ e $V(H)$ tal que, dados $u, v \in V(G)$, u e v são vizinhos em G se e somente se $f(u)$ e $f(v)$ são vizinhos em H . Quando existe um isomorfismo entre dois grafos G e H , dizemos que G é *isomorfo* a H .

Uma *subdivisão* de uma aresta $e = \{u, v\}$ de um grafo G é o ato de se adicionar um novo vértice w ao grafo G , remover a aresta e e adicionar duas novas arestas $f = \{u, w\}$ e $g = \{w, v\}$. O grafo G' resultante de uma ou mais subdivisões de arestas de G é chamado de *subdivisão* de G .

Dizemos que G é *planar* se ele não possuir como subgrafo nenhuma subdivisão do grafo completo K_5 nem do grafo bipartido $K_{3,3}$.

2.2 LÓGICA MATEMÁTICA

Uma *variável* booleana pode ter valor 0 ou 1. Um *literal* é uma variável booleana ou uma variável booleana negada (com valor invertido).

Uma *conjunção* é uma operação lógica entre literais representada por $\vee$. Sejam a, b literais, o resultado de $a \vee b$ é igual a 1 se pelo menos um deles tem valor 1 e é 0 em caso contrário.

Uma *disjunção* é uma operação lógica entre literais representada por $\wedge$. Sejam a, b literais, o resultado de $a \wedge b$ é igual a 1 se ambos tem valor 1 e é 0 em caso contrário.

Uma *cláusula* em Forma Normal Conjuntiva (CNF) é uma disjunção de literais distintos.

Uma *fórmula* booleana em Forma Normal Conjuntiva (CNF) F é um par (V, C) em que C é um conjunto de cláusulas CNF e V o conjunto das variáveis que aparecem nas cláusulas de C . Uma *valoração* de V é uma função $g : V \rightarrow \{0, 1\}$. Dizemos que uma cláusula D é *satisfeita* por uma valoração g se o resultado da disjunção de seus literais é 1, ou seja, se pelo menos um de seus literais tem valor 1 em g . Dizemos que uma fórmula F é satisfazível se existe valoração que satisfaça todas as cláusulas de F .

Toda fórmula ou cláusula mencionada neste trabalho está na Forma Normal Conjuntiva.

2.3 TEORIA DA COMPUTAÇÃO

Os conceitos da Teoria da Computação e Complexidade utilizados no trabalho, como por exemplo definições de problemas de decisão, classes de problemas P e NP, tempo de execução de algoritmos e algoritmos verificadores têm o sentido usual da literatura (como por exemplo em [Garey and Johnson, 2009]).

Chamamos uma instância de um problema de decisão de *tratável* quando se conhece um algoritmo com tempo de execução polinomial que computa sua resposta. Caso contrário, a instância é chamada de *intratável*.

3 SATISFAZIBILIDADE E RESTRIÇÃO

O problema SAT foi um dos primeiros a ser provado NP-completo, se tornando assim um problema de destaque na área da Teoria da Computação. Além disso, ele continua NP-completo mesmo com diversos tipos de restrição. A seguir, algumas restrições relevantes serão citadas.

3.1 RESTRIÇÃO

Sabe-se que o problema restrito a instâncias em que cada cláusula possui exatamente três variáveis (versão restringida conhecida como 3-SAT) continua NP-completo[Cook, 1971]. Pode-se dizer que essa restrição é forte, visto que quando restringido a instâncias em que cada cláusula possui 2 variáveis (2-SAT), o problema possui solução com tempo de execução polinomial[Aspvall et al., 1979, Even et al., 1976]. Mas não é a restrição mais forte possível, visto que 3-SAT permanece NP-completo mesmo para instâncias em que cada variável aparece em no máximo cinco cláusulas. Esta restrição permitiu a prova de que o problema do Caixeiro Viajante Euclidiano é NP-completo[Papadimitriou, 1977].

Além disto, Tovey[Tovey, 1984] mostrou que 3-SAT permanece NP-completo mesmo restringido a instâncias em que cada variável aparece em no máximo três cláusulas. Ele provou ainda que esta restrição é a mais forte possível no sentido de que 3-SAT restringido a instâncias em que cada variável aparece em no máximo duas cláusulas possui solução em tempo polinomial.

É perceptível a existência de uma espécie de fronteira entre as versões restringidas que as separa em solúveis em tempo de execução polinomial ou não. Esclarecer esta fronteira é importante porque facilita o trabalho de provar a NP-completude de novos problemas, permitindo transformações e reduções mais fáceis e diretas. Também ajuda a distinguir instâncias tratáveis e intratáveis do problema.

3.2 GRAFOS FÓRMULA

Também é possível utilizar conceitos da Teoria dos Grafos para permitir novos tipos de restrição. Dada uma instância F do problema SAT, o *grafo fórmula* G_F é um grafo bipartido em que uma das partes dos vértices corresponde às variáveis e a outra parte às cláusulas. As arestas de G_F ligam suas variáveis às cláusulas que as contém. Lichtenstein provou que 3-SAT permanece NP-completo mesmo se restringido a instâncias cujo grafo fórmula é planar e usou isto para provar a NP-completude do problema do conjunto dominante conexo geométrico [Lichtenstein, 1982].

Motivado pelo estudo e prova da NP-completude de outros problemas, como o problema de reconhecimento de grafos 2-DIR, Kratochvíl[Kratochvíl, 1994] acrescentou ainda mais restrições ao problema 3-SAT Planar, exigindo que o grafo fórmula fosse 3-vértice-conexo e que cada variável aparecesse em no máximo quatro cláusulas.

Percebe-se também, ao analisarmos o trabalho de Tovey[Tovey, 1984], que a restrição acrescentada por ele pode ser limitada ao caso planar e, portanto, 3-SAT Planar permanece NP-completo para instâncias onde cada variável aparece em no máximo três cláusulas.

Cada uma das restrições citadas anteriormente também pode ser expressa em termos dos grafos fórmula. Limitar o número de variáveis por cláusulas em uma fórmula equivale a limitar o grau dos vértices de cláusulas no grafo fórmula e limitar a quantidade de aparições de cada variável equivale a limitar o grau do vértice de cada variável.

Todas as classes de grafo já mencionadas (grafos planares, grafos bipartidos e grafos com grau de vértice limitado) são classes *hereditárias*. Classes hereditárias permitem a descrição de seus elementos através de subgrafos induzidos proibidos. Elas serão usadas neste trabalho para a identificação de uma sequência infinita de versões restringidas do problema da satisfazibilidade que convergem para uma versão particular do problema.

3.3 CLASSES HEREDITÁRIAS, LIMITES E LIMÍTROFES DE GRAFOS

Uma *classe* de grafos é um conjunto de grafos fechado por isomorfismo. Uma classe é chamada *hereditária* se é fechada em relação a remoção de vértices. Ou seja, se a remoção de vértices de um grafo da classe resulta em um grafo da mesma classe. Seja M um conjunto de grafos, denotamos por $\text{Livre}(M)$ a classe de todos os grafos que não contém subgrafos induzidos isomorfos a grafos no conjunto M . Uma classe Y é hereditária se e somente se $Y = \text{Livre}(M)$ para algum conjunto M (Teorema 16 de [Lozin, 2019]). Além disso, para toda classe hereditária, o conjunto minimal de subgrafos induzidos proibidos é único (Teorema 18 de [Lozin, 2019]).

Como exemplo de classe hereditária, pode-se citar a classe dos grafos cordais, que pode ser expressa por $\text{Livre}(\{C_j \mid j \geq 4\})$, ou seja, é a classe dos grafos que não possuem subgrafos induzidos isomorfos a ciclos de tamanho maior que 4 (sem cordas, ou seja, arestas que ligam vértices internos do ciclo).

Uma classe de grafos é *monotônica* se é fechada em relação a remoção de vértices e a remoção de arestas. De maneira similar às classes hereditárias, podemos descrever classes monotônicas a partir de um conjunto de grafos proibidos. Denotamos por $Y = \text{Livre}_m(M)$ a classe monotônica Y que não possui grafos com subgrafos (agora não mais necessariamente induzidos) isomorfos a grafos em M .

Como exemplo de classe monotônica, pode-se citar a classe dos grafos bipartidos, que por sua vez pode ser expressa por $\text{Livre}_M(\{C_{2j+1} \mid j \geq 1\})$, ou seja, é a classe dos grafos que não possuem subgrafos induzidos isomorfos a ciclos de tamanho ímpar.

No decorrer do trabalho, serão chamadas de *tratáveis* as classes hereditárias de grafos fórmula cujo problema de satisfazibilidade é tratável. De maneira análoga, chamaremos de *intratáveis* classes hereditárias de grafos fórmula cujo problema de satisfazibilidade é intratável.

Uma classe hereditária Z de grafos fórmula será chamada *classe limite* se existe uma sequência $Y_1 \supseteq Y_2 \supseteq \dots$ de classes intratáveis tais que $Z = \bigcap_{i \geq 1} Y_i$. Dizemos que a sequência $\{Y_i\}_{i \geq 1}$ *converge* para Z . Uma classe limite minimal será chamada *classe limítrofe*. O critério para minimalidade utilizado é dado no seguinte lema.

Lema 1. *Uma classe limite $Y = \text{Livre}(M)$ é minimal (limítrofe) se e somente se, para todo grafo G de Y , existe um conjunto finito $T \subseteq M$ tal que $\text{Livre}(\{G\} \cup T)$ é tratável. [Lozin and Purcell, 2013]*

Prova. Suponha que Y é limítrofe e, por contradição, exista um grafo G para todo conjunto finito $T \subseteq M$ tal que $\text{Livre}(\{G\} \cup T)$ é intratável. Como $M := \{m_1, m_2, \dots\}$, podemos definir $Z_i = \text{Livre}\{G, m_1, m_2, \dots, m_i\}$. Então, segundo a primeira suposição, Z_i é intratável para todo i . Então $Z := \bigcap_i Z_i$ é uma classe limite. Ela não contém nenhum grafo de M e não contém G . Portanto, é subconjunto próprio de Y , contradizendo a minimalidade de Y .

Agora, suponha que, para todo grafo $G \in Y$, existe um conjunto finito $T \subseteq M$ tal que $\text{Free}(\{G\} \cup T)$ é tratável, e que, por contradição, existe uma classe limite Z contida propriamente em Y . Então existe uma sequência $Z_1 \supseteq Z_2 \supseteq \dots$ convergindo em Z . Escolha um grafo $G \in Y \setminus Z$ e um conjunto finito $T \subseteq M$ tal que $\text{Livre}(\{G\} \cup T)$ é tratável. Então, como $Z \subset Y$, Z também é livre de M e portanto existe n tal que Z_n é livre de $(\{G\} \cup T)$. Neste caso, Z_n deve ser tratável, contradizendo a suposição de que Z é uma classe limite. Essa contradição termina a prova. $\square$

4 UMA CLASSE LIMÍTROFE PARA SATISFAZIBILIDADE

Neste capítulo, uma classe de grafos fórmula S será apresentada. Além disso, ela será provada limite e limítrofe para o problema da satisfazibilidade.

4.1 UMA CLASSE LIMITE PARA SATISFAZIBILIDADE

Seja F uma fórmula, x uma variável e C uma cláusula que a contém. Denotaremos o literal de x contido em C por x^α , onde $x^\alpha = x$ se $\alpha = 1$, e $x^\alpha = \bar{x}$ se $\alpha = 0$. Denotaremos por $F(x, C)$ a fórmula obtida de F da seguinte forma: adicione uma nova variável y , adicione uma nova cláusula $D = (x^\alpha \vee \bar{y})$, substitua x^α em C por y .

Lema 2. *Seja F uma fórmula. Para toda cláusula $C \in F$ e toda variável $x \in C$, vale a afirmação que F é satisfazível se e somente se $F(x, C)$ é satisfazível. [Lozin and Purcell, 2013]*

Prova. Assuma que F é satisfazível e seja γ uma valoração que a satisfaça. Se C é satisfeita por x^α , então x^α satisfaz a nova cláusula em $F(x, C)$. Portanto, a valoração γ juntamente com $y \rightarrow 1$ satisfaz $F(x, C)$. Se C é satisfeita por um literal diferente de x^α , então $F(x, C)$ pode ser satisfeita pela valoração γ juntamente com $y \rightarrow 0$.

Agora, suponha que $F(x, C)$ é satisfazível e seja γ uma valoração que a satisfaça. Se a cláusula $D = (x^\alpha \vee \bar{y})$ é satisfeita por x^α , então γ também satisfaz F . Se D não é satisfeita por x^α , então é satisfeita por $\bar{y}$ e, neste caso, $\gamma(y) = 0$ e portanto, C é satisfeita em $F(x, C)$ por uma variável diferente de y . Portanto, γ satisfaz F novamente. $\square$

Esta transformação, em termos do grafo fórmula, equivale a uma subdivisão dupla de uma aresta no grafo fórmula G_F , como pode ser visto no exemplo da Figura 1. Perceber isto permite a seguinte conclusão, onde H_n é um grafo que é uma subdivisão do K_2 com n arestas acrescido de dois vértices ligados a cada ponta (representado na Figura 2).

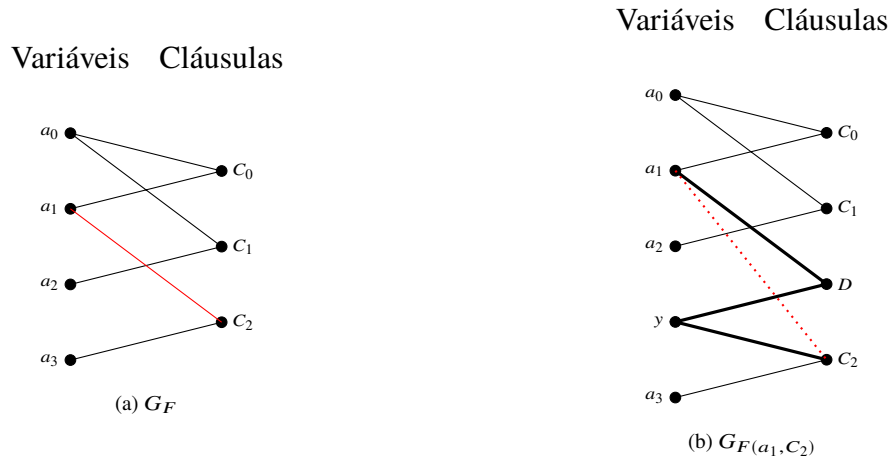


Figura 1: Transformação $F(x, C)$ visualizada no grafo fórmula.

Lema 3. *Para todo k fixo, o problema da satisfazibilidade restrito a instâncias cujos grafos fórmula pertencem à classe $S_k = \text{Livre}(K_{1,4}, C_3, C_4, \dots, C_k, H_1, H_2, \dots, H_k)$ é NP-completo. [Lozin and Purcell, 2013]*

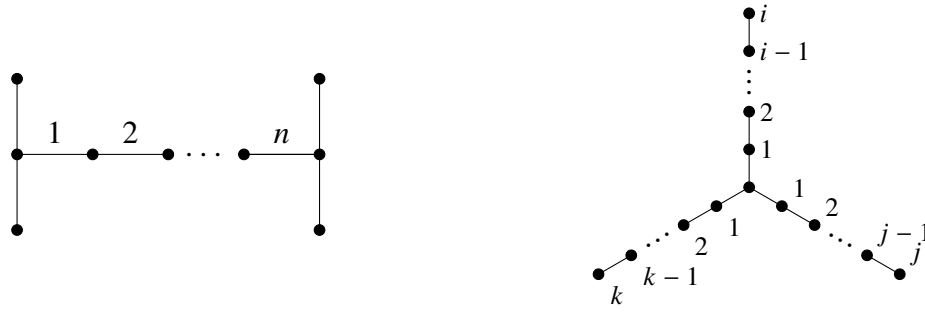


Figura 2: Grafo H_n (a esquerda) e $S_{i,j,k}$ (a direita).

Prova. Sabe-se que o problema da satisfazibilidade é NP-completo quando restrito a grafos fórmula com grau de vértice no máximo três [Tovey, 1984]. Dado um grafo G_F nesta classe, subdivimos cada aresta de G_F duas vezes e denotamos o grafo resultante por $G_{F'}$. Observe que, quando fazemos isso, aumentamos o tamanho de todas as cópias induzidas de ciclos e grafos da forma H_i . De acordo com o Lema 2, F é satisfazível se e somente se F' é. Se aplicarmos a transformação acima um número suficiente de vezes, podemos transformar G_F em um grafo $G_{F''}$ que pertence à classe $\text{Livre}(C_3, \dots, C_k, H_1, \dots, H_k)$. Como antes, F é satisfazível se e somente se F'' também é, e para um k fixo, essa transformação pode ser feita em tempo polinomial. Finalmente, como G_F tem grau no máximo 3, $G_{F''}$ também tem. Portanto, $G_{F''}$ é livre de $K_{1,4}$. $\square$

É fácil perceber que $S_3 \supset S_4 \supset S_5 \supset \dots$ e, portanto, pelo Lema 3, a intersecção $S = \bigcap_k S_k$ é uma classe limite. Essa intersecção não contém ciclos, portanto é uma classe de florestas. Como a intersecção não contém $K_{1,4}$, é uma classe de florestas cujos vértices tem grau máximo 3. Finalmente, como a intersecção também não tem grafos da forma H_n , toda componente de qualquer grafo nesta classe é da forma $S_{i,j,k}$ (para alguns valores de i, j, k), ou seja, uma subdivisão do $K_{1,3}$ onde i, j e k indicam quantas subdivisões foram feitas em cada uma das arestas originais do $K_{1,3}$ (representado na Figura 2).

Da discussão acima, temos a seguinte conclusão:

Teorema 4. *S é uma classe limite para o problema da satisfazibilidade. [Lozin and Purcell, 2013]*

4.2 MINIMALIDADE DE S

A prova para a minimalidade da classe S é baseada em alguns resultados auxiliares envolvendo a noção de *largura arbórea* [Gottlob and Szeider, 2007].

Teorema 5. *O problema da satisfazibilidade restrito a qualquer classe de grafos fórmula com largura arbórea limitada é resolvível em tempo polinomial.*

O próximo resultado lida com classes monotônicas e pode ser encontrado em [Kaminski and Lozin, 2007].

Lema 6. *Para qualquer $k > 2$ fixo, a classe $\text{Livre}_m(\{C_j | j \geq k\})$ tem largura arbórea limitada.*

A partir destes resultados, enunciamos o seguinte teorema.

Teorema 7. *Seja Y uma classe monotônica de grafos. Se pelo menos um grafo de S não pertence a Y , então a largura arbórea dos grafos de Y é limitada. [Lozin and Purcell, 2013]*

Prova. Vamos denotar por $F_{t,k}$ o grafo em S com exatamente t componentes, cada uma da forma $S_{k,k,k}$. Claramente, todo grafo em S é um subgrafo induzido de $F_{t,k}$ para alguns valores fixos de t e k . Portanto, é suficiente provar o teorema para classes monotônicas que não possuem $F_{t,k}$ para alguns t e k . Provaremos por indução em t .

Primeiro, assumindo $t = 1$. Mostraremos que grafos em $Livre_m(S_{k,k,k})$ tem largura arbórea limitada. Seja G um grafo conexo nesta classe. Considere um caminho P de tamanho $2k - 2$, e um ciclo C de tamanho no mínimo $2k + 1$ em G . Se G não contém tais P ou C , então a largura arbórea de G é limitada pelo Lema 6. Assuma que P e C são disjuntos, isto é, não possuem vértices em comum. Como G é conexo, então deve haver um caminho P' cujas pontas pertencem a C e P , e cujos vértices restantes estão fora de C e P . Então a união de C , P e P' contém um subgrafo isomórfico a $S_{k,k,k}$. Esta contradição mostra que P intersecta todo ciclo de tamanho pelo menos $2k + 1$. Portanto, ao remover de G os vértices de P obtemos um grafo sem ciclos de tamanho pelo menos $2k + 1$. De acordo com o Lema 6, o grafo resultante é um grafo de largura arbórea limitada. A remoção de P diminui a largura arbórea de G em no máximo $2k - 2$, portanto a largura arbórea de G também é limitada.

Agora, seja $t > 1$ e G um grafo em $Livre_m(F_{t,k})$. Se G não contém $S_{k,k,k}$ como subgrafo (não necessariamente induzido), então a largura arbórea de G é limitada pelo parágrafo anterior (a base da indução). Se G contém uma cópia de $S_{k,k,k}$, então a remoção desta cópia resulta em um grafo em $Livre_m(F_{t-1,k})$. Pela hipótese da indução, este grafo tem largura arbórea limitada. Portanto, a largura arbórea de G é limitada também, já que removemos apenas uma quantidade constante de vértices ($3k + 1$). $\square$

Com a ajuda dos resultados acima, agora provaremos que S é uma classe limítrofe, ou seja, uma classe limite minimal.

Teorema 8. *S é uma classe limítrofe para o problema da satisfazibilidade. [Lozin and Purcell, 2013]*

Prova. Seja G um grafo de S . Sem perda de generalidade, podemos assumir que toda componente de G tem a forma $S_{k,k,k}$ para algum valor finito k , já que todo grafo em S é um subgrafo induzido de um grafo desta forma.

Considere a classe $Z := Livre(G, K_{1,4}, C_3, \dots, C_{2k+1}, H_1, \dots, H_{2k+1})$. Por definição, nenhum grafo em Z contém uma cópia induzida de G . Mais que isso: podemos mostrar que nenhum grafo em Z contém G como subgrafo, não necessariamente induzido. Assuma que, pelo contrário, um grafo em Z contém uma cópia de G como subgrafo. Sabemos que esta cópia não é induzida, portanto deve conter uma aresta que não pertence a G . Se esta aresta conecta dois vértices da mesma componente de G , então um ciclo de tamanho no máximo $2k + 1$ surge, o que é impossível, pois estes ciclos estão, por definição, proibidos. De maneira similar, um ciclo menor pode ser encontrado se duas componentes diferentes de G estão conectadas por pelo menos duas arestas. Finalmente, se duas componentes diferentes estão conectadas por uma única aresta, então uma cópia induzida de um grafo H_i com $i \leq 2k + 1$ aparece, o que também é, por definição, impossível. Portanto, nenhum grafo em Z contém G como subgrafo. Em outras palavras, $Z \subseteq Livre_m(G)$. De acordo com o Teorema 7, isto implica que a largura arbórea dos grafos em Z é limitada. Como resultado, de acordo com o Teorema 5, Z é uma classe tratável e, portanto, de acordo com o Lema 1, S é uma classe limite minimal, ou seja, limítrofe. $\square$

4.3 UNICIDADE DE S

A classe S é a única classe limítrofe para o problema da satisfazibilidade se e somente se, para todo grafo $G \in S$ o problema é resolvível em tempo polinomial para instâncias cujo grafo

fórmula é livre de G . Para alguns grafos pequenos, como por exemplo, se $G = S_{1,1,1}$, qualquer grafo bipartido livre de G tem grau máximo 2 e neste caso, estamos lidando com uma instância de 2-SAT, que é resolvível em tempo polinomial. Outro exemplo, no caso em que $G = S_{0,1,2}$, então o problema também é resolvível em tempo polinomial, pois todo grafo bipartido livre de $S_{0,1,2}$ é bipartido cordal, caso tratado pelo resultado em [Ordyniak et al., 2013].

Teorema 9. *O problema da satisfazibilidade restringido a instâncias cujos grafos fórmula são livres de $S_{1,1,2}$ é resolvível em tempo polinomial. [Lozin and Purcell, 2013]*

Prova. Observe que todo grafo bipartido conexo livre de $S_{1,1,2}$ é ou um grafo de grau máximo de vértice 2 ou um grafo bipartido quase completo [Alekseev, 2004], ou seja, um grafo bipartido em que cada vértice tem no máximo um vértice que não é vizinho na parte oposta. Para grafos de grau máximo de vértice 2, o problema é simples. Para resolver o problema para grafos bipartidos quase completos, utilizamos a noção da resolução de Davis-Putnam definida a seguir.

Para uma fórmula F contendo uma variável x , sejam C, D duas cláusulas tais que uma delas contém x positivamente e a outra negativamente e, além disso, x é a única variável em que C e D "discordam". Então a cláusula $C \cup D \setminus \{x, \bar{x}\}$ é chamada *resolvente* de x de C e D . Seja $DP_x(F)$ a fórmula obtida de F ao adicionar todos os possíveis resolventes de x e removendo todas as cláusulas contendo x . Essa operação é chamada de resolução de Davis-Putnam. É sabido [Davis and Putnam, 1983] que F é satisfazível se e somente se $DP_x(F)$ é satisfazível.

Afirmção 10. *Se F é uma instância cujo grafo fórmula é bipartido quase completo e seja x uma variável arbitrária, então o grafo fórmula de $DP_x(F)$ é bipartido completo. [Lozin and Purcell, 2013]*

Prova. Seja G o grafo bipartido quase completo representando F , e seja G' o grafo representando $DP_x(F)$. O grafo G' é obtido de G ao remover todas as cláusulas contendo x , introduzindo todos os resolventes de x e removendo o vértice que representa x .

Seja C um vértice representando uma cláusula em G' . Já que x tem no máximo um vértice que não é seu vizinho na parte oposta em G , C é ou um resolvente de x ou o único vértice que não é vizinho de x em G . No último caso, C deve ser adjacente a todos os outros vértices na parte oposta de G' , já que x foi removido.

Assuma agora que C é um resolvente de x de duas cláusulas $D_1, D_2 \in F$. Por definição, todo vértice y representando uma variável em G é adjacente a pelo menos uma destas cláusulas, já que, em caso contrário, G não é bipartido quase completo. Portanto y deve ser adjacente a C em G' , já que, por definição $C = D_1 \cup D_2 \setminus \{x, \bar{x}\}$. Como resultado, C é adjacente a todos os vértices na parte oposta de G' e, portanto, G' é bipartido completo. $\square$

Como mencionado anteriormente, grafos bipartidos completos são bipartidos cordais. Portanto, a afirmação acima juntamente com o resultado em [Ordyniak et al., 2013] que soluciona o problema para instâncias representáveis por grafos bipartidos cordais completa a prova do teorema. $\square$

5 CONCLUSÃO

Devido a NP-completude do problema da satisfazibilidade, estudar e identificar propriedades de instâncias deste problema pode, além de render conhecimento e domínio para a solução deste problema, trazer avanços no que conhecemos sobre diversos outros problemas. Por isso, a pesquisa e o estudo detalhado destes problemas é muito importante.

Os resultados expostos por este trabalho identificam, de maneira construtiva, a classe de grafos fórmula da forma $S_{i,j,k}$ como uma classe limítrofe para o problema da Satisfazibilidade. Isto significa que o problema SAT restrito a classes de grafos fórmula que contém S é NP-completo.

Como a unicidade de S como classe limítrofe para SAT ainda é desconhecida, saber que um certo conjunto de instâncias cujas classes de grafos não contém grafos de S não garante que estas instâncias sejam tratáveis. Mas provamos, por exemplo, que classes livres de $S_{1,1,2}$ são tratáveis e, por isso, problemas cujas instâncias tem grafos fórmula nesta classe podem ser resolvidos eficientemente. A prova para classes livres de outros grafos maiores em S é um possível e desafiador trabalho futuro.

REFERÊNCIAS

- V.e. Alekseev. Polynomial algorithm for finding the largest independent sets in graphs without forks. *Discrete Applied Mathematics*, 135(1-3):3–16, 2004. doi: 10.1016/s0166-218x(02)00290-1.
- Bengt F. Aspvall, Michael Endre Plass, and Robert undefined Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 8(3): 121–123, 1979. doi: 10.1016/0020-0190(79)90002-4.
- Stephen A. Cook. The complexity of theorem-proving procedures. *Proceedings of the third annual ACM symposium on Theory of computing - STOC 71*, 1971. doi: 10.1145/800157.805047.
- M. Davis and H. Putnam. A computing procedure for quantification theory. *Automation of Reasoning*, page 125–139, 1983. doi: 10.1007/978-3-642-81952-0_9.
- S. Even, A. Itai, and A. Shamir. On the complexity of timetable and multicommodity flow problems. *SIAM Journal on Computing*, 5(4):691–703, 1976. doi: 10.1137/0205048.
- Michael R. Garey and David S. Johnson. *Computers and intractability: a guide to the theory of NP-completeness*. Freeman, 2009.
- G. Gottlob and S. Szeider. Fixed-parameter algorithms for artificial intelligence, constraint satisfaction and database problems. *The Computer Journal*, 51(3):303–325, 2007. doi: 10.1093/comjnl/bxm056.
- M. Kaminski and V. V. Lozin. Coloring edges and vertices of graphs without short or long cycles. *Contributions to Discrete Mathematics*, 2:61–66, 2007.
- Richard M. Karp. Reducibility among combinatorial problems. *Complexity of Computer Computations*, page 85–103, 1972. doi: 10.1007/978-1-4684-2001-2_9.
- Jan Kratochvíl. A special planar satisfiability problem and a consequence of its np-completeness. *Discrete Applied Mathematics*, 52(3):233–252, 1994. doi: 10.1016/0166-218x(94)90143-0.
- Leonid A. Levin. Some theorems on the algorithmic approach to probability theory and information theory. *Annals of Pure and Applied Logic*, 162(3):224–235, 2010. doi: 10.1016/j.apal.2010.09.007.
- David Lichtenstein. Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343, 1982. doi: 10.1137/0211025.
- Vadim Lozin. Graph theory notes. <https://homepages.warwick.ac.uk/~masgax/Graph-Theory-notes.pdf>, 2019. Acessado em 16/11/2019.
- Vadim Lozin and Christopher Purcell. Boundary properties of the satisfiability problems. *Information Processing Letters*, 113(9):313–317, 2013. doi: 10.1016/j.ipl.2013.01.022.
- Sebastian Ordyniak, Daniel Paulusma, and Stefan Szeider. Satisfiability of acyclic and almost acyclic cnf formulas. *Theoretical Computer Science*, 481:85–99, 2013. doi: 10.1016/j.tcs.2012.12.039.

- Christos H. Papadimitriou. The euclidean travelling salesman problem is np-complete. *Theoretical Computer Science*, 4(3):237–244, 1977. doi: 10.1016/0304-3975(77)90012-3.
- Craig A. Tovey. A simplified np-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85–89, 1984. doi: 10.1016/0166-218x(84)90081-7.