



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE RUSSAS
CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

GABRIEL ENNOS DA SILVA

**PROBLEMA DE MINIMIZAÇÃO DE CUSTO DE PROPAGAÇÃO DE INFLUÊNCIA
EM REDES SOCIAIS: UMA ABORDAGEM META-HEURÍSTICA**

RUSSAS

2025

GABRIEL ENNOS DA SILVA

PROBLEMA DE MINIMIZAÇÃO DE CUSTO DE PROPAGAÇÃO DE INFLUÊNCIA EM
REDES SOCIAIS: UMA ABORDAGEM META-HEURÍSTICA

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Engenharia de Software
do CAMPUS DE RUSSAS da Universidade
Federal do Ceará, como requisito parcial à
obtenção do grau de bacharel em Engenharia de
Software.

Orientadora: Profa. Dra. Tatiane Fernan-
des Figueiredo

RUSSAS

2025

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

S58p Silva, Gabriel Ennos da.
Problema de Minimização de Custo de Propagação de Influência em Redes Sociais : Uma Abordagem Meta-Heurística / Gabriel Ennos da Silva. – 2025.
41 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas, Curso de Engenharia de Software, Russas, 2025.
Orientação: Profa. Dra. Tatiane Fernandes Figueiredo.

1. Meta-Heurística. 2. Redes Sociais. 3. Influenciadores. I. Título.

CDD 005.1

GABRIEL ENNOS DA SILVA

PROBLEMA DE MINIMIZAÇÃO DE CUSTO DE PROPAGAÇÃO DE INFLUÊNCIA EM
REDES SOCIAIS: UMA ABORDAGEM META-HEURÍSTICA

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Engenharia de Software
do CAMPUS DE RUSSAS da Universidade
Federal do Ceará, como requisito parcial à
obtenção do grau de bacharel em Engenharia de
Software.

Aprovada em: 08/08/2025.

BANCA EXAMINADORA

Profa. Dra. Tatiane Fernandes
Figueiredo (Orientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Cenez Araújo de Rezende
Universidade Federal do Ceará (UFC)

Prof. Dr. Eurinaldo Rodrigues Costa
Universidade Federal do Ceará (UFC)

AGRADECIMENTOS

Agradeço primeiramente a Deus, pela graça e apoio concedidos ao longo de todo este percurso acadêmico.

Expresso minha imensa gratidão à Profa. Dra. Tatiane Fernandes. Sua dedicação na orientação, a clareza de seus conselhos e o aprendizado contínuo foram decisivos para a conclusão deste projeto.

À minha família, dedico este trabalho. O apoio constante e o amor que recebi foram o motor essencial para superar todos os desafios.

RESUMO

Com o crescimento do uso das redes sociais, empresas têm buscado cada vez mais vincular seus produtos a influenciadores digitais, sendo uma forma eficaz para divulgação e conseqüentemente aumento das vendas. Porém, a decisão de qual influenciador contratar com o melhor custo benefício para a empresa pode se tornar uma tarefa difícil quando pensamos em escalabilidade das redes. Com o intuito de apresentar soluções para este problema, o presente trabalho teve como objetivo estudar e propor ajustes, para melhorar um dos algoritmos meta-heurístico apresentado na literatura, para resolução do Problema de Minimização de Custo de Propagação de Influência em Redes Sociais. Este problema busca encontrar um número fixo de influenciadores em uma rede social, que possam disseminar propagandas de um produto em seus perfis. Após a realização de testes e análises estatísticas dos resultados obtidos utilizando as mesmas instâncias apresentadas pelo autor do algoritmo meta-heurístico estudado, concluímos que as melhorias propostas apresentam de fato um ganho de desempenho, melhorando o *GAP* médio em 34,47%.

Palavras-chave: meta-heurística;redes sociais;influenciadores.

ABSTRACT

With the growing use of social media, companies have increasingly sought to link their products to digital influencers, as this has proven to be an effective way to promote products and consequently increase sales. However, the decision of which influencer to hire with the best cost-benefit ratio can become a challenging task, especially when considering the scalability of networks. Aiming to present solutions to this problem, this work aimed to study and propose improvements to one of the metaheuristic algorithms found in the literature, for solving the Influence Propagation Cost Minimization Problem in Social Networks. This problem seeks to find a fixed number of influencers in a social network who can disseminate advertisements for a product through their profiles. After conducting tests and statistical analyses using the same instances presented by the author of the studied metaheuristic algorithm, we concluded that the proposed improvements indeed result in performance gains, improving the average *GAP* by 34,47%.

keywords: meta-heuristic;social networks;influencers.

LISTA DE FIGURAS

Figura 1 – Representação de um grafo direcionado.	16
Figura 2 – Representação do modelo para resolução do PMCPIRS.	24
Figura 3 – Visão geral do algoritmo genético proposto por Costa (2019).	25
Figura 4 – Método <i>Nova Solução</i> proposto por Costa (2019).	26
Figura 5 – Operador de <i>crossover</i> utilizado por Costa (2019).	27
Figura 6 – Operador de mutação utilizado por Costa (2019).	28
Figura 7 – Grafo direcionado para geração de população inicial de indivíduos.	32
Figura 8 – Grafo direcionado para geração de população inicial de indivíduos, primeiro indivíduo.	32
Figura 9 – Grafo direcionado para geração de população inicial de indivíduos, segundo indivíduo.	33
Figura 10 – Grafo direcionado para geração de população inicial de indivíduos, terceiro indivíduo.	34
Figura 11 – Representação de solução no processo de mutação do algoritmo.	34
Figura 12 – Mutação de solução realizada.	35
Figura 13 – Mutação de solução realizada, com propagação de influência.	36

LISTA DE TABELAS

Tabela 1 – Comparação de desempenho para instâncias do mundo 1, entre o algoritmo de Costa (2019) e o do autor.	37
Tabela 2 – Comparação de desempenho para instâncias do mundo 2, entre o algoritmo de Costa (2019) e o do autor.	37
Tabela 3 – Comparação de desempenho para instâncias do mundo 3, entre o algoritmo de Costa (2019) e o do autor.	38

SUMÁRIO

1	INTRODUÇÃO	11
2	OBJETIVOS	12
2.1	Objetivo Geral	12
2.2	Objetivos específicos	12
3	FUNDAMENTAÇÃO TEÓRICA	13
3.1	Pesquisa Operacional	13
3.2	Modelos Matemáticos	13
3.3	Programação Linear	14
3.4	Teoria dos grafos	15
3.5	Modelo Limiar Linear	16
3.6	Meta-heurísticas	17
3.7	Algoritmos genéticos	18
4	TRABALHOS RELACIONADOS	19
4.1	Extração de nós influentes para difusão de informação numa rede social	19
4.2	Maximização da disseminação de influência por meio de redes sociais	20
4.3	Uma Abordagem para o Problema de Seleção de Conjuntos-Alvo Ponderados em Redes Sociais	21
4.4	Integração dos efeitos da rede social no problema da partilha de escolha	21
5	PMCPIRS	23
5.1	O algoritmo exato proposto por Fischetti <i>et al.</i> (2018) para resolução do PMCPIRS	23
5.1.1	<i>Price-and-Cut</i>	23
5.1.2	<i>Modelo para resolução do PMCPIRS</i>	23
5.2	O algoritmo genético proposto por Costa (2019) para resolução do PMCPIRS	24
5.2.1	<i>Heurística fundamentada em Algoritmo Genético</i>	25
5.2.2	<i>Método Nova Solução</i>	26
5.2.3	<i>Operador de Crossover</i>	27
5.2.4	<i>Operador de Mutação</i>	28
6	METODOLOGIA	30

6.1	Estudo do domínio	30
6.2	Propostas de melhorias para o algoritmo meta-heurístico da literatura	30
6.2.1	<i>Seleção das instâncias para etapa de testes</i>	36
6.2.2	<i>Avaliação dos resultados obtidos</i>	36
7	CONCLUSÕES E TRABALHOS FUTUROS	39
	REFERÊNCIAS	40

1 INTRODUÇÃO

Desde a década passada, o crescimento do número de pessoas que acompanham outras nas redes sociais tornou-se um hábito comum em todo o mundo. Essas pessoas, seguidas por uma quantidade considerável de usuários, são denominadas influenciadores digitais, que, principalmente após a pandemia, apresentaram um crescimento exponencial nas visualizações de seus perfis (Imédia Comunicação, 2023). Nesse contexto, as empresas perceberam que vincular seus produtos a influenciadores digitais é uma estratégia eficaz de divulgação e, consequentemente, de aumento nas vendas. Assim, um problema a ser solucionado pelas empresas é naturalmente a seleção dos influenciadores mais adequados, com o objetivo de propagar seus produtos para o maior número possível de pessoas, com um investimento financeiro reduzido (Forbes Brasil, 2023).

Procurando apresentar soluções computacionais para este problema, Fischetti *et al.* (2018) apresentaram um modelo limiar linear para o problema denominado Problema de Minimização de Custo de Propagação de Influência em Redes Sociais (PMCPIRS). Neste problema, uma determinada pessoa é influenciada se a soma das influências da sua vizinhança atingir determinado parâmetro chamado de *hurdle* (barreira) de influência. Para resolução do problema, os autores apresentam um algoritmo de *price-and-cut*, assim como uma análise dos resultados obtidos com um grupo de instâncias criadas pelos próprios autores. Costa (2019) também estudou o problema mencionado, foco deste estudo, apresentando um algoritmo meta-heurístico para sua resolução. Buscando colaborar com a obtenção de soluções de alta qualidade para resolução do Problema de Propagação de Influência, este trabalho tem por objetivo apresentar melhorias para o algoritmo meta-heurístico apresentado por Costa (2019). Como resultado principal obtivemos bons *GAPs* em comparação com os resultados exatos obtidos por Fischetti *et al.* (2018) e que superam os *GAPs* obtidos por Costa (2019).

A estrutura deste trabalho está dividida da seguinte maneira: Na seção 2, são apresentados os objetivos que nortearam o estudo. Em seguida, a seção 3 dedica-se à fundamentação teórica, estabelecendo as bases conceituais do trabalho. A seção 4 explora os trabalhos relacionados, contextualizando a pesquisa no campo existente. A seção 5 aprofunda-se no problema central, o PMCPIRS, oferecendo uma explicação detalhada. A seção 6 descreve a metodologia utilizada para condução da pesquisa. Por fim, a seção 7 consolida as conclusões e apresenta as considerações finais do estudo e sugestões para trabalhos futuros.

2 OBJETIVOS

2.1 Objetivo Geral

Estudar e propor melhorias para o algoritmo meta-heurístico proposto por Costa (2019) para resolução do Problema de Minimização de Custos de Propagação de Influência em Redes Sociais (PMCPIRS).

2.2 Objetivos específicos

- Estudar e compreender todas as definições acerca do Problema de Minimização de Custos de Propagação de Influência em Redes Sociais (PMCPIRS);
- Estudar e compreender todas as etapas do algoritmo meta-heurístico proposto por Costa (2019);
- Apresentar ideias de melhorias para as principais etapas do algoritmo meta-heurístico proposto por Costa (2019);
- Implementar e testar as melhorias propostas, considerando as instâncias apresentadas na literatura;
- Realizar uma análise estatística, com foco nos *GAPs* obtidos para cada uma das instâncias testadas.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Pesquisa Operacional

A Pesquisa Operacional (PO) é um ramo da matemática aplicada que busca resolver problemas complexos de decisão, sendo amplamente utilizada no meio acadêmico e industrial por meio de modelagem matemática e algoritmos computacionais (Martins, 2019). A PO combina técnicas de estatística, algoritmos e modelos matemáticos com o objetivo de resolver esses problemas de decisão em diversos contextos. As principais técnicas da pesquisa operacional incluem programação linear e não linear, simulação, heurísticas entre outros métodos, geralmente com o intuito de minimizar, ou maximizar algo.

3.2 Modelos Matemáticos

Modelos matemáticos são representações de sistemas reais que descrevem, por meio de equações, fórmulas ou outros elementos matemáticos, o comportamento e as transformações desses sistemas (Sodré, 2007). Esses modelos são usados em diversas áreas como física, engenharia, biologia, economia, entre outras áreas. Em programação linear esses modelos trabalham basicamente com duas técnicas de programação, onde em ambas as técnicas os modelos matemáticos buscam maximizar ou minimizar uma função linear. O modelo matemático padrão contém: variáveis, que são responsáveis por definir os valores alvo para solucionar o problema em questão, a função objetivo, que representa a abstração do problema a ser resolvido matematicamente e as restrições, que são limitadores das variáveis do problema. Como exemplo de modelo matemático linear, considere uma empresa que precisa comprar ao menos 5 máquinas para produção de calçados, e essas 5 máquinas precisam produzir pelo menos 5000 calçados por dia, essa empresa possui n máquinas no catálogo, disponíveis para comprar e deseja adquirir máquinas de tal forma que pague a menor quantidade possível e produza pelo menos 5000 calçados por dia. Então um modelo matemático linear para solucionar esse problema é descrito da seguinte forma:

Variáveis do problema:

- C_i = custo da máquina i ;
- Q_i = quantidade de calçados produzidos por dia pela máquina i ;
- X_i = variável binária que define se a máquina i será comprada(1) ou não(0).

(a) Função objetivo do problema:

$$\min \sum_{i=1}^n c_i x_i$$

(b) Restrições do problema:

$$\sum_{i=1}^n x_i \geq 5$$

$$\sum_{i=1}^n x_i q_i \geq 5000$$

$$X \in \{0, 1\}$$

A expressão linear a (função objetivo), se refere ao que o problema quer solucionar, neste caso, o problema busca minimizar a variável C_i , que representa o custo de cada máquina, utilizando a variável binária X_i , que é uma variável gerenciada, para apontar se uma determinada máquina será escolhida pelo modelo, ou não. Já as desigualdades b (restrições do problema), delimitam o espaço amostral onde o problema trabalha, dando determinadas condições de escolha, que serão utilizadas na função objetivo(a), a primeira restrição delimita que no mínimo 5 máquinas precisarão ser selecionadas pelo modelo, já a segunda restrição diz que pelo menos 5000 calçados deverão ser produzidos por dia, ou seja, o modelo tem de achar uma combinação de máquinas que consigam respeitar essas restrições pelo menor custo possível.

3.3 Programação Linear

A programação Linear (PL) é uma técnica amplamente utilizada na pesquisa operacional com o objetivo de otimizar sistemas que envolvem recursos limitados (Almeida, 2018). Essa técnica busca encontrar a melhor solução, seja ela, por exemplo, a, maximização de lucros ou a minimização de custos por meio da otimização de uma função objetivo, sujeita a um conjunto de restrições, todas representadas por relações lineares.

Os modelos de PL são compostos por três elementos principais: a função objetivo, que representa o critério a ser otimizado, as restrições, que traduzem as limitações dos recursos disponíveis, e as variáveis de decisão, cujos valores devem ser determinados de modo a alcançar a melhor solução possível.

Entre os métodos para resolução de problemas de PL, destaca-se o método Simplex, desenvolvido por George Dantzig em 1947. Esse algoritmo itera sobre soluções básicas viáveis, os vértices da região de factibilidade até encontrar a solução ótima. O método parte de uma solução inicial trivial e realiza melhorias sucessivas com base na análise dos coeficientes da função objetivo, escolhendo os vértices que oferecem os maiores ganhos até que nenhuma melhoria seja possível (Wilhelm, 2020).

Em muitos casos práticos, no entanto, as variáveis de decisão devem assumir apenas valores inteiros. Nesses cenários, aplica-se a programação Linear Inteira (PLI), que se divide em programação Linear Inteira Pura (quando todas as variáveis devem ser inteiras) e programação Linear Inteira Mista (quando apenas parte delas possui essa restrição). Essa abordagem é fundamental em situações que envolvem decisões discretas, como alocação de recursos, planejamento logístico e seleção de projetos (Alves; Delgado, 1997).

Para resolver problemas de PLI, destacam-se métodos como o *Branch-and-Bound*, que consiste em dividir o problema em subproblemas menores e eliminar regiões do espaço de soluções com base em limites inferiores e superiores, e o método dos planos de corte (*cutting planes*), que adiciona restrições adicionais ao modelo relaxado para aproximá-lo da solução ótima inteira (Alves; Delgado, 1997).

Assim, a programação Linear e suas variantes inteiras, constituem ferramentas essenciais para a modelagem e resolução de problemas complexos de otimização nas mais diversas áreas, como engenharia, economia, logística e administração.

3.4 Teoria dos grafos

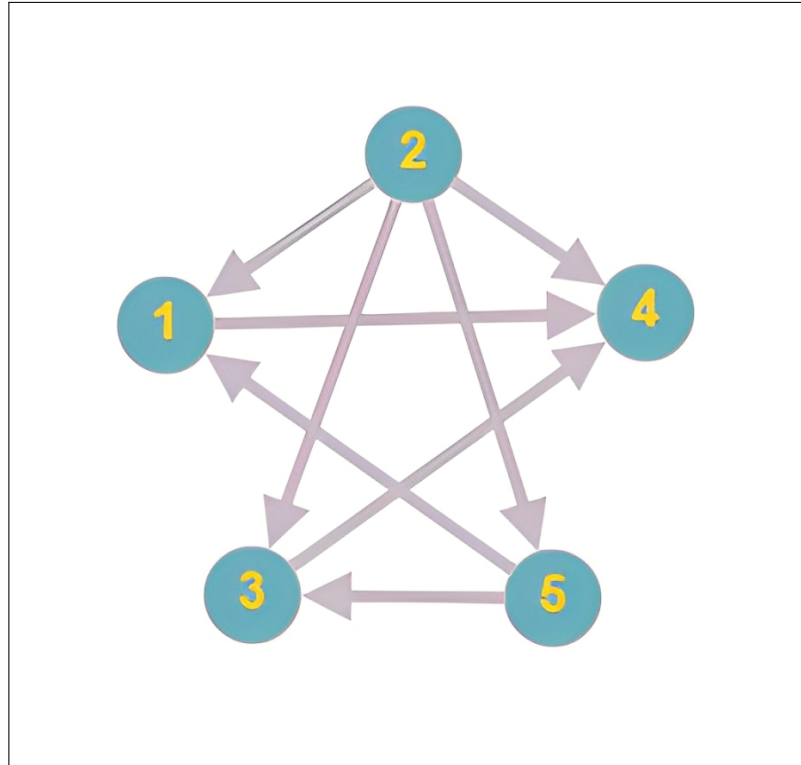
Um grafo é uma estrutura matemática, utilizada para representar elementos de um conjunto conhecido e as relações entre eles (Gomes, 2024). Por exemplo, pode-se representar uma rede social em que os vértices correspondem a pessoas e as arestas representam as conexões entre elas.

Formalmente, um grafo é definido como $G = (V, E)$, onde $V(G)$ é o conjunto de vértices (ou nós) e $E(G)$ é o conjunto de arestas (ou ligações) entre esses vértices. Os grafos podem ser classificados em dois tipos principais: não direcionados e direcionados (dígrafos).

Em grafos não direcionados, as conexões entre os vértices são bidirecionais, ou seja, a relação ocorre mutuamente entre os dois nós conectados. Já nos grafos direcionados, as conexões possuem um sentido definido, partindo de um vértice de origem (vértice de saída) e

chegando a um vértice de destino (vértice de chegada), o que é geralmente representado por uma seta, como ilustrado na Figura 1.

Figura 1 - Representação de um grafo direcionado.



Fonte: Autor(2025).

Cada vértice de um grafo possui um grau, esse grau é calculado pelo número de arestas que incidem nesse vértice. Um conceito muito importante da teoria grafos, que é utilizado no PMCPIRS, é o conceito de vizinhança, que são o conjunto de vértices adjacentes (ligados através de arestas) a um determinado vértice, ou seja, considerando um vértice "a", a sua vizinhança, são os outros vértices do grafo que estão ligados a esse vértice "a".

3.5 Modelo Limiar Linear

O modelo limiar linear é um modelo de propagação de influência que fundamentou o processo de disseminação de ideias ou produtos, proposto por Granovetter (1978), neste modelo os autores consideram que um determinado indivíduo, posteriormente representado no modelo limiar linear como nó ou vértice, pode adquirir determinada ideia ou produto, baseado na influência que sua vizinhança exerce sobre esse indivíduo. Essa influência é medida pelos pesos de influência que cada vértice tem de seus vizinhos, cada vértice possui um limiar de ativação, e após a soma das influências dos vértices vizinhos à determinado vértice atingirem o limiar

de ativação dele, esse vértice se torna ativo. Após esse limiar ser alcançado pelo somatório da influência de sua vizinhança, tal vértice (indivíduo) adquire uma ideia ou compra um produto. Para calcular o limiar linear do problema tem-se como entrada um grafo direcionado $G = (V, E)$, onde V representa os vértices desse grafo e E as arestas deste grafo. Para ativar um determinado vértice $i \in V$, cada vértice i possui um valor de influência de seus vizinhos j , representados por W_{ij} , e um limiar θ_i que é gerado aleatoriamente considerando intervalo $[0, 1]$. A seguir é apresentada a expressão que representa, o que precisa para um determinado vértice ser ativado.

$$\sum_{i=1}^V W_{ij} \geq \theta_i$$

Em suma, a expressão acima, afirma que o somatório de W_{ij} , ou seja os valores de influência dos vértices j que estão ligados ao vértice i , precisam, quando somados, atingir o θ_i (limiar de ativação do vértice i), para que i seja ativado.

3.6 Meta-heurísticas

As meta-heurísticas são estratégias utilizadas para resolver problemas de otimização, em espaços de busca com alta complexidade (Luzia; Rodrigues, 2009). Geralmente, meta-heurísticas são utilizadas em problemas onde a checagem de todas as soluções possíveis é inviável, pela demasiada quantidade de combinações necessárias para checar todas as soluções existentes. Em geral, as meta-heurísticas sacrificam a busca da solução ótima, ou seja, não percorrem todo o espaço de soluções do problema, buscando por uma solução com um certo grau de qualidade em detrimento de um tempo reduzido de execução.

Um exemplo de meta-heurística é a Estratégia Evolutiva (EE), que é uma meta-heurística de otimização que modela a evolução natural para encontrar soluções eficazes. Utiliza-se uma população de soluções candidatas, aplicando operador de mutação para explorar o espaço de busca. As soluções são avaliadas por uma função de aptidão, e o processo evolutivo envolve a seleção de pais com base na aptidão, recombinação e mutação para gerar a próxima geração. A EE é aplicada em problemas complexos, adaptando-se a diferentes estratégias de mutação para melhorar a busca global por soluções ótimas (Nogueira; Saavedra, 1999).

3.7 Algoritmos genéticos

Algoritmos genéticos são algoritmos inspirados no princípio da evolução das espécies (Alam et al., 2020). Tais algoritmos disponibilizam métodos de busca baseados na aptidão populacional. Os algoritmos genéticos se baseiam basicamente nos seguintes conceitos vindos da genética:

- **População:** Grupo de indivíduos que são possíveis soluções.
- **Cromossomo:** Conjunto de genes, que possuem informações que representam uma possível solução.
- **Gene:** Parte específica do cromossomo que codifica uma característica da solução.
- **Alelos:** Diferentes variações de um gene, como 0 ou 1 em problemas binários.
- **Cruzamento (Crossover):** Processo que combina informações genéticas de dois pais para criar filhos.
- **Mutação:** Alteração aleatória de um gene para introduzir diversidade genética.
- **Função de Aptidão:** Avaliação de quão bem um cromossomo se encaixa no problema.

De forma geral, os algoritmos genéticos iniciam gerando uma população inicial de indivíduos. Em seguida, avaliam a aptidão de cada indivíduo, selecionando aqueles mais aptos para reprodução. Esse processo se repete até que um critério de parada seja atingido. Durante as iterações, pode ocorrer cruzamento entre os cromossomos selecionados e mutação em alguns genes. Após essas operações genéticas, uma nova avaliação é realizada e a população antiga é substituída pela nova. Esse ciclo persiste até que os critérios de parada sejam alcançados, resultando em uma população evoluída que representa soluções mais adequadas ao problema em questão (Alam et al., 2020).

4 TRABALHOS RELACIONADOS

Essa seção tem por objetivo apresentar os trabalhos relacionados sobre o problema abordado nesta monografia.

4.1 Extração de nós influentes para difusão de informação numa rede social

A difusão de ideias ou produtos em redes sociais, impulsionada pela influência entre indivíduos conectados, é um fenômeno amplamente observado em contextos como marketing viral, política e propagação de inovações. Entender como selecionar estrategicamente os nós mais influentes para iniciar essa difusão é fundamental para otimizar campanhas de disseminação. Esse é o cerne do Problema de Maximização de Influência, estudado em diferentes modelos de propagação, como os modelos de cascata independente (IC) e limiar linear (LT) (Kempe et al., 2003; Fischetti et al., 2018).

Kimura *et al.* (2007) propuseram um método eficiente para a extração de nós influentes em redes sociais de grande escala, considerando os modelos de propagação de cascata independente (IC) e limiar linear (LT). Esses modelos utilizam a estratégia gananciosa (*greedy*) para resolver o problema, oferecendo garantias teóricas de aproximação de pelo menos $(1 - 1/e)$ do valor ótimo. No entanto, a forma tradicional de aplicar essa estratégia exige realizar diversas simulações do processo de difusão para cada vértice da rede. Isso acontece porque, para decidir se um vértice deve ser incluído no conjunto inicial (chamado de conjunto de sementes), é necessário estimar o seu ganho marginal, ou seja, o quanto ele contribui para aumentar o número de pessoas influenciadas quando adicionado ao grupo.

Como essa estimativa depende de um processo probabilístico (baseado em sorteios e interações entre os nós), é preciso repetir as simulações várias vezes para obter uma média confiável. Esse procedimento se torna muito demorado e pesado computacionalmente, principalmente em redes sociais grandes, com milhares de vértices e conexões. Por isso, o espaço de busca cresce rapidamente, e o tempo de execução aumenta consideravelmente.

Para superar essa limitação, os autores introduzem uma abordagem baseada em percolação de ligações e teoria dos grafos, que permite estimar os ganhos marginais de forma mais eficiente. Essa abordagem reconstrói o processo de difusão como um modelo determinístico em grafos derivados de amostras probabilísticas (modelos de percolação), reduzindo drasticamente o número de simulações necessárias.

A solução é implementada por meio de um algoritmo de subida de encosta (*hill-climbing*), que utiliza essa estimativa eficiente para selecionar os k vértices com maior impacto esperado, ou seja, aqueles com maior capacidade de influenciar outros vértices e propagar ativações pela rede. Os experimentos realizados em redes reais, como a rede de blogs sobre o acidente ferroviário de Fukuchiyama, demonstraram que o método proposto não apenas mantém a qualidade das soluções, como também reduz o tempo de processamento em ordens de grandeza quando comparado aos métodos convencionais baseados em simulação

Em suma, o estudo mostra que a modelagem da propagação por meio de percolação de ligações é uma alternativa eficaz e escalável para o problema de extração de vértices influentes, com resultados promissores em aplicações práticas.

4.2 Maximização da disseminação de influência por meio de redes sociais

O Problema de Maximização da Disseminação de Influência em redes sociais, introduzido por Kempe *et al.* (2003), consiste em identificar um subconjunto de indivíduos em uma rede que, ao serem inicialmente influenciados ou convencidos a adotar um produto, maximizem a disseminação do mesmo por meio da rede. A ideia central é selecionar os chamados indivíduos sementes, que possuem alto potencial de influência sobre os demais, de modo a iniciar um efeito cascata de ativações.

Esses indivíduos, uma vez ativados, influenciam outros com os quais se conectam, ampliando progressivamente a difusão do produto ou ideia na rede. Dado um orçamento limitado, representado por k , o objetivo é escolher k indivíduos iniciais que maximizem o número esperado de indivíduos influenciados ao final do processo.

Kempe *et al.* (2003) formalizaram esse problema utilizando modelos clássicos de propagação (o Modelo de Cascata Independente e o Modelo Limiar Linear) e demonstraram que o problema é NP-difícil em ambos os casos. Apesar disso, os autores apresentaram um algoritmo de aproximação baseado na propriedade de submodularidade da função de influência, o qual garante uma solução com fator de aproximação de pelo menos $(1 - 1/e)$ do valor ótimo.

4.3 Uma Abordagem para o Problema de Seleção de Conjuntos-Alvo Ponderados em Redes Sociais

O Problema de Seleção de Conjuntos-Alvo Ponderados (Weighted Target Set Selection — WTSS) estende o Problema de Seleção de Conjuntos-Alvo (TSS) ao considerar diferentes níveis de esforço ou custo para ativar os nós de uma rede. Cada vértice da rede possui um limiar de ativação, e a ativação se propaga de acordo com o número de vizinhos previamente ativados.

Raghavan e Zhang (2019) propuseram uma abordagem para resolver o WTSS baseada em programação inteira mista. A proposta consiste em um algoritmo de *branch-and-cut*, fundamentado em uma formulação estendida compacta e robusta do problema, inicialmente válida para árvores e posteriormente generalizada para grafos arbitrários por meio da adição de restrições que forçam a aciclicidade do grafo de propagação.

A formulação apresentada é mais forte do que modelos anteriores, como os de Shakarian *et al.* (2013) e Ackerman *et al.* (2010), cujas relaxações lineares resultam em soluções fracionárias e fracas, mesmo em estruturas simples como árvores. Para contornar essas limitações, os autores desenvolveram uma nova representação do grafo de influência, inserindo nós fictícios e direcionando as arestas de propagação para modelar com precisão o processo de ativação.

Nos experimentos computacionais, a abordagem de *branch-and-cut* mostrou-se altamente eficaz, resolvendo instâncias reais de grande escala com até 155 mil vértices e 327 mil arestas. Em média, as soluções encontradas ficaram a apenas 0,9% da otimalidade, superando significativamente heurísticas conhecidas, como a adaptação do método de Shakarian *et al.* (2013), cujas soluções apresentaram qualidade até 5 vezes inferior.

Esses resultados demonstram que a formulação proposta não apenas melhora a precisão das soluções obtidas, como também oferece escalabilidade para aplicação em redes sociais reais. Além disso, o trabalho reforça a importância de integrar restrições estruturais (como a aciclicidade) na modelagem da propagação de influência, a fim de refletir com fidelidade os aspectos dinâmicos do problema.

4.4 Integração dos efeitos da rede social no problema da partilha de escolha

Gunnec e Raghavan (2017) estudam a integração dos efeitos de rede social no Problema da Partilha de Escolha (Share-of-Choice Problem — SOC), com base no modelo de limiar linear proposto por Granovetter (1978). Neste contexto, os autores consideram que um

indivíduo (representado por um vértice em um grafo) adota um produto se a utilidade percebida superar um valor de limiar (chamado de *hurdle*), que pode ser reduzido à medida que vizinhos também adotam o produto e conseqüentemente exercem influência sobre esse indivíduo.

A rede social é representada por um grafo $G = (V, E)$, em que cada vértice $i \in V$ corresponde a um indivíduo e cada aresta indica uma relação de influência entre dois indivíduos. Cada vértice possui um limiar de adoção, inicialmente definido como h_i^H , que representa a resistência do indivíduo a adotar um determinado produto. Esse limiar, no entanto, pode ser reduzido até um valor mínimo h_i^L à medida que os vizinhos do indivíduo na rede também adotam o produto, refletindo o efeito da influência social. Esse processo segue o modelo de limiar linear, no qual a influência dos vizinhos é proporcional ao número de conexões do vértice na rede.

O modelo proposto, denominado SOCSNE (Share-of-Choice with Social Network Effects), é formulado como um problema de otimização combinatória. Nele, a utilidade percebida por cada indivíduo é ajustada com base nas influências recebidas de seus vizinhos, o que é representado formalmente por meio da matriz de adjacência da rede. A principal inovação do estudo é considerar esses efeitos de rede já durante a fase de concepção do produto — ou seja, antes de seu lançamento no mercado. Com isso, o projeto do produto leva em conta como ele será percebido e adotado dentro da rede social, permitindo que sua configuração seja otimizada para aproveitar ao máximo os efeitos de propagação por influência.

Além disso, os autores propõem um problema complementar chamado LCIP (Least Cost Influence Problem), cujo objetivo é minimizar o custo total de incentivos necessários para alcançar a disseminação desejada do produto na rede, conforme definida pelo modelo SOCSNE. Como ambos os problemas são computacionalmente complexos (NP-difíceis), os autores desenvolvem um algoritmo genético eficiente para encontrar boas soluções em tempo razoável.

Em resumo, o estudo mostra que incorporar a estrutura e considerar os efeitos da rede social já na concepção do produto pode aumentar significativamente seu alcance no mercado, além de permitir uma alocação mais eficiente dos recursos.

5 PMCPIRS

Esta seção tem por objetivo descrever o problema de minimização de custos de propagação de influência em redes sociais (PMCPIRS), proposto por Fischetti *et al.* (2018) e foco central deste trabalho.

5.1 O algoritmo exato proposto por Fischetti *et al.* (2018) para resolução do PMCPIRS

5.1.1 Price-and-Cut

O *Price-and-Cut* proposto por Fischetti *et al.* (2018), é uma heurística responsável por capturar os vértices viáveis que serão utilizados como entrada no modelo matemático, fazendo com que o mesmo não precise verificar todos os vértices do grafo, e sim somente os que não ferem nenhuma restrição imposta pelo problema. Inicialmente, o *Price-and-Cut* usa o procedimento de geração de colunas e cortes de geração de colunas para obter limites primários e duais apertados. Depois o algoritmo continua resolvendo o problema utilizando três critérios gulosos, que são: a escolha do vértice com menor incentivo para ser ativado, a escolha do vértice com menor incentivo sobre seus vizinhos, e o vértice com menor incentivo dividido pelo número de vizinhos que serão ativados na iteração seguinte. Esse processo é repetido até que encontramos um conjunto de vizinhos influentes onde não resta nenhum arco na lista, ou até obtermos um conjunto de vizinhos influentes (nesse caso chamamos de *influencers*) sem a necessidade de incentivos extras.

5.1.2 Modelo para resolução do PMCPIRS

Considere as seguintes entradas e variáveis:

Seja w_{ip} o custo para ativar o vértice i , sendo p o valor de incentivo ao vértice i e d_{ji} o valor de influência repassado do vértice j para i , considere as seguintes variáveis: x_i representa se o vértice i foi ativado (valor 1) ou não (valor 0), x_{ip} possui o valor 1 se o vértice i foi selecionado para ser um influencer e valor 0, caso contrário e z_{ji} representa se aresta $ij \in E$ deve ser computada (valor 1) ou não (valor 0). Por fim, h_i é o *threshold* a ser atendido para ativação do vértice i .

O modelo matemático para resolução do PMCPIRS é dado por:

A função objetivo (4a) minimiza a soma dos custos dos incentivos selecionados.

Figura 2 - Representação do modelo para resolução do PMCPIRS.

(ARC)	min	$\sum_{i \in V} \sum_{p \in P_i} w_{ip} x_{ip}$	(4a)
	s.t.	$\sum_{p \in P_i} p x_{ip} + \sum_{(j,i) \in A} d_{ji} z_{ji} \geq h_i x_i \quad \forall i \in V$	(4b)
		$\sum_{p \in P_i} x_{ip} = x_i \quad \forall i \in V$	(4c)
		$\sum_{(i,j) \in C} z_{ij} \leq \sum_{i \in V(C) \setminus \{k\}} x_i \quad \forall k \in V(C), \forall \text{ cycles } C \subseteq A$	(4d)
		$z_{ij} \leq x_i \quad \forall (i,j) \in A \text{ s.t. } (j,i) \notin A$	(4e)
		$\sum_{i \in V} x_i \geq \lceil \alpha V \rceil$	(4f)
		$x_{ip} \in \{0, 1\} \quad \forall i \in V, \forall p \in P_i$	(4g)
		$x_i \in \{0, 1\} \quad \forall i \in V$	(4h)
		$z_{ij} \in \{0, 1\} \quad \forall (i,j) \in A$	(4i)

Fonte: Fischetti *et al.* (2018).

As restrições de propagação do tipo (4b) comparam a soma dos incentivos selecionados e a influência dos vizinhos com o *threshold* estabelecido (valor necessário para ativar um vértice). A restrição (4c) garante que apenas um incentivo a cada vértice ativado. Restrições generalizadas de eliminação de ciclos são definidas em (4d) enquanto as restrições (4e), (4f) verificam se a quantidade de vértices ativados satisfaz a demanda necessária. Por fim, as restrições 4(g), 4(h) e 4(i) garantem que todas as variáveis sejam binárias.

5.2 O algoritmo genético proposto por Costa (2019) para resolução do PMCPIRS

Costa (2019) propôs uma abordagem meta-heurística baseada em Algoritmos Genéticos (AG) para a resolução do PMCPIRS (problema de Minimização de Custos de Propagação de Influência em Redes Sociais). Essa proposta se diferencia do modelo exato de Fischetti *et al.* (2018) por priorizar soluções aproximadas, visando eficiência computacional, especialmente em instâncias maiores.

A estrutura do algoritmo está dividida em três etapas principais: geração da população inicial de indivíduos, aplicação de operadores genéticos (*crossover* e *mutação*), e seleção dos melhores indivíduos para compor as novas gerações. A seção 5.2.1 mostra o funcionamento do algoritmo, de forma geral.

5.2.1 Heurística fundamentada em Algoritmo Genético

O algoritmo genético desenvolvido por Costa (2019) recebe os seguintes parâmetros de entrada:

- β : Tamanho da população (número de indivíduos gerados), com valor fixado em 300;
- G : Grafo direcionado representado por uma matriz de adjacência;
- α : Percentual mínimo de vértices que devem estar ativos ao final do processo;
- n : Número de vértices no grafo G ;
- γ : Quantidade máxima de épocas (iterações), com valor definido em 200;
- P : Matriz contendo os possíveis contratos de incentivos para cada vértice;
- μ : Taxa de mutação, com valor definido em 0,02;
- θ : Tolerância à repetição de melhores indivíduos entre gerações, com valor 50.

Figura 3 - Visão geral do algoritmo genético proposto por Costa (2019).

<p>Algoritmo 2: Algoritmo Genético</p> <hr/> <p>Input: ($\beta, G, \alpha, n, \gamma, P, \mu, \theta$)</p> <p>Output: Melhor solução encontrada para o $PMCPIRS$</p> <pre> 1 populacao ← gerar_populacao(β); 2 while criterio de parada nao for atendido do 3 populacao ← populacao \cup reproducao(crossover, populacao); 4 populacao ← selecionar_nova_populacao(populacao, β); 5 return Melhor solução encontrada </pre> <hr/>

Fonte: Costa (2019).

Inicialmente, o algoritmo chama a função *gerar_populacao*, responsável por construir uma população inicial com β indivíduos, utilizando o método *Nova Solução*, apresentado na Figura 4.

Em seguida, o método *reproducao* é utilizado dentro de um laço de repetição que se estende até o critério de parada ser satisfeito (quantidade de épocas γ). Esse método aplica operadores genéticos como *crossover* e mutação, que podem ser vistos nas figuras 5 e 6, sobre os indivíduos da população, com o objetivo de gerar novas soluções e melhorar a aptidão da solução.

Após cada iteração, o método *selecionar_nova_populacao* é acionado para selecionar 80% dos melhores indivíduos e 20% dos piores, baseado em seus respectivos *fitness*, garantindo

a diversidade genética da próxima população.

5.2.2 Método Nova Solução

Figura 4 - Método *Nova Solução* proposto por Costa (2019).

Algoritmo 3: Nova Solução	
Input: (G, α, n)	
Output: $S(Z, V, \varphi)$	
1	$Z \leftarrow$ matriz de $n \times n$ posições;
2	$V \leftarrow$ vetor de n posições;
3	$T \leftarrow$ vetor com $[0, 1, 2 \dots n]$;
4	$\varphi \leftarrow 0$;
5	while tamanho de $V < \lceil \alpha n \rceil$ do
6	$novo_individuo \leftarrow$ elemento aleatorio de T ;
7	$T.remove(novo_individuo)$;
8	$V.adiciona(novo_individuo)$;
9	for $i : V[0 : -1]$ do
10	if $G[novo_elemento][i] > 0$ then
11	$Z[novo_elemento][i] \leftarrow 1$;
12	$\varphi \leftarrow calcular_aptidao(V, Z)$;
13	return $S(Z, V, \varphi)$;

Fonte: Costa (2019).

O método *Nova Solução* é responsável pela criação de um indivíduo da população. As linhas 2 a 5 do algoritmo inicializam as variáveis utilizadas, sendo T o conjunto de vértices candidatos à solução.

O laço entre as linhas 5 e 11 executa a seleção de vértices até atingir a quantidade exigida, proporcional a α . Nas linhas 7 e 8, um vértice é escolhido aleatoriamente e removido do conjunto T . Esse vértice é adicionado ao vetor de solução V na linha 11.

As linhas 9 a 11 tratam da atualização da matriz Z , responsável por armazenar as influências entre os vértices selecionados. O valor de *fitness* da solução é calculado na linha 12,

e a solução completa é retornada na linha 13.

5.2.3 Operador de Crossover

Figura 5 - Operador de *crossover* utilizado por Costa (2019).

Algoritmo 4: Crossover	
Input:	$(S_i = (V^i, Z^i, \varphi^i), S_k = (V^k, Z^k, \varphi^k))$
Output:	$S_1^n(Z_1^n, V_1^n, \varphi_1^n), S_2^n(Z_2^n, V_2^n, \varphi_2^n)$
1	$Z_1^n \leftarrow$ matriz de $n \times n$ posicoes;
2	$Z_2^n \leftarrow$ matriz de $n \times n$ posicoes;
3	$V_1^n \leftarrow$ vetor de n posicoes;
4	$V_2^n \leftarrow$ vetor de n posicoes;
5	$\varphi_1^n \leftarrow 0$;
6	$\varphi_2^n \leftarrow 0$;
7	$S_1^n \leftarrow Falso$;
8	$S_2^n \leftarrow Falso$
9	$corte_1 \leftarrow$ Valor randomico entre 1 e $n/2$;
10	$corte_2 \leftarrow$ Valor randomico entre $n/2$ e $n - 1$
11	if nao existe qualquer intersecao entre $V^i[0 : corte_1]$, $V^k[corte_1 : corte_2]$ e $V^i[corte_2 :]$ then
12	$V_1^n \leftarrow V^i[0 : corte_1] \cup V^k[corte_1 : corte_2] \cup V^i[corte_2 :]$;
13	$V_1^n \leftarrow realizar_mutacao(V_1^n)$;
14	$Z_1^n \leftarrow preencher_matriz(V_1^n)$
15	$\varphi_1^n \leftarrow calcular_aptidao(V_1^n, Z_1^n)$;
16	if nao existe qualquer intersecao entre $V^k[0 : corte_1]$, $V^i[corte_1 : corte_2]$ e $V^k[corte_2 :]$ then
17	$V_2^n \leftarrow V^k[0 : corte_1] \cup V^i[corte_1 : corte_2] \cup V^k[corte_2 :]$;
18	$V_2^n \leftarrow realizar_mutacao(V_2^n)$;
19	$Z_2^n \leftarrow preencher_matriz(V_2^n)$
20	$\varphi_2^n \leftarrow calcular_aptidao(V_2^n, Z_2^n)$;
21	return $S_1^n(Z_1^n, V_1^n, \varphi_1^n), S_2^n(Z_2^n, V_2^n, \varphi_2^n)$;

Fonte: Costa (2019).

O operador de *crossover* é responsável por combinar pares de indivíduos da população atual, gerando novas soluções a partir do corte dos vetores V_i e V_k nas posições indicadas nas linhas 9 e 10.

A linha 11 verifica a ausência de duplicação de vértices no novo vetor V_n . O preenchimento ocorre na linha 12, podendo ser seguido de mutação (linha 13). A matriz Z_n é atualizada conforme as restrições de propagação (linha 14), e o *fitness* da nova solução S_n^1 é calculado na linha 15. O processo é repetido entre as linhas 16 e 20 para gerar um segundo descendente.

5.2.4 Operador de Mutação

Figura 6 - Operador de mutação utilizado por Costa (2019).

Algoritmo 5: Mutação	
	Input: (V^n, μ, n)
	Output: V^n
1	begin
2	$contador \leftarrow 0;$
3	while $contador < \text{valor aleatorio entre } 0 \text{ e } n\mu$ do
4	$posicao \leftarrow \text{valor_randomico}(0, \text{tamanho}(V^n));$
5	$f_1 \leftarrow V^n[posicao];$
6	$f_2 \leftarrow \text{valor_randomico}(0, n - 1);$
7	while $V^n \text{ conter } f_2 \text{ ou } f_1 \text{ igual a } f_2$ do
8	$f_2 \leftarrow \text{valor_randomico}(0, n - 1);$
9	$V^n[posicao] \leftarrow f_2;$
10	$contador \leftarrow contador + 1;$
11	return $V^n;$
12	

Fonte: Costa (2019).

A mutação é aplicada ao vetor V_n , que representa os vértices ativos de uma solução $S_n = (V_n, Z_n)$. Inicialmente, é sorteado uma posição aleatória entre 0 e n , indicando o vértice que será alterado.

O vértice selecionado é substituído por outro que não pertence à solução atual, promovendo diversidade genética. Esse processo se repete até atingir $n \cdot \mu$ interações, conforme a taxa de mutação definida.

Esse operador busca evitar o estagnamento da população em ótimos locais, incentivando a exploração de novas soluções viáveis ao longo das épocas.

6 METODOLOGIA

Nesta seção será apresentado todas as etapas necessárias para se atingir os objetivos propostos nesta monografia.

6.1 Estudo do domínio

Primeiramente foi estudado o artigo de Fischetti *et al.* (2018), para entender como o autor aborda e resolve o problema PMCPIRS. Posteriormente, também foi estudado o algoritmo meta-heurístico fundamentado em algoritmos genéticos proposto por Costa (2019).

6.2 Propostas de melhorias para o algoritmo meta-heurístico da literatura

Foram propostas duas melhorias no algoritmo meta-heurístico adotado. A primeira proposta refere-se à geração da população inicial de indivíduos. No algoritmo original da literatura, essa população é gerada de forma totalmente aleatória. Na proposta do trabalho vigente, a geração inicial é feita com base nas melhores soluções encontradas, a partir dos vértices que apresentam menor custo para ativação. Assim, cada indivíduo da população inicial é construído de forma ordenada, conforme sua aptidão (*fitness*), em ordem crescente de custo.

Ou seja, o primeiro indivíduo da população inicial representa a melhor solução possível (com o menor custo para ativar o percentual desejado de vértices); o segundo indivíduo representa a segunda melhor solução, e assim sucessivamente. As figuras 7, 8 e 9 ilustram essa dinâmica, mostrando como a nova geração da população inicial é estruturada. O exemplo utiliza um grafo direcionado, com pesos de influência entre os vértices, uma lista de incentivos I , na qual cada vértice possui uma tupla do tipo (custo, incentivo), e uma lista de *thresholds* H , onde cada valor representa o nível necessário para ativação de um vértice, considerando os incentivos, somado a influência recebida de vértices vizinhos ativos.

A segunda proposta de melhoria envolve o operador de mutação do algoritmo, onde acontece a troca de alguns vértices, de determinados indivíduos, com o intuito de melhorar o *fitness* desses indivíduos. Nesse caso, os indivíduos utilizados, são previamente gerados na população inicial, com o objetivo de aprimorá-los. Um percentual de vértices num indivíduo, previamente configurado no algoritmo, é selecionado para alteração. Diferentemente do algoritmo original, os novos vértices que compõem a solução não são escolhidos totalmente aleatoriamente, mas sim, aleatoriamente a partir de uma lista ordenada de vértices, com forte influência sobre os

demais vértices já presentes no indivíduo e que obviamente não estejam presentes na solução.

Dessa forma, ao incluir vértices com alta capacidade de influência, os custos necessários para a ativação de determinados nós são reduzidos, podendo, em alguns casos, chegar a custo zero. A dinâmica dessa segunda proposta é ilustrada nas Figuras 8, 9 e 10. A seguir, são apresentadas as dinâmicas das duas propostas por meio de figuras, considerando um valor de α (percentual de ativação de vértices) igual a 25% e uma taxa de mutação de 15%.

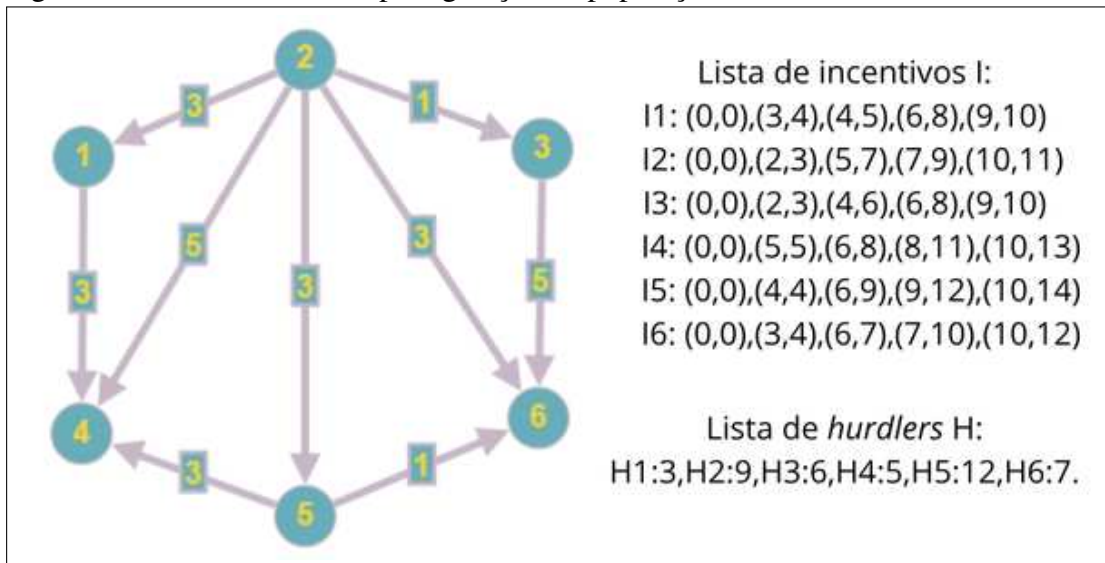
Nas figuras, os vértices são representados pelas seguintes cores: azul (não ativados), verde (ativados com incentivo), vermelho (pertencentes a outro indivíduo, portanto, indisponíveis para ativação) e amarelo (ativados por influência).

A figura 7 apresenta um grafo direcionado que representa uma rede social entre indivíduos. Conforme já discutido neste trabalho, o problema abordado considera que os vértices representam pessoas, enquanto as arestas indicam o peso de influência que cada indivíduo exerce sobre os demais.

Além do grafo, também é fornecida uma lista de incentivos para cada vértice, composta por tuplas no formato $(custo, incentivo)$. Por exemplo, um vértice pode ter um custo x associado à compra de um incentivo y . Caso a soma desse incentivo com a influência proveniente dos vizinhos já ativados atinja o *hurdle* (limiar) correspondente ao vértice, ele é ativado. Uma vez ativado, o vértice passa a influenciar outros vértices conectados a ele, dando continuidade ao processo de propagação.

Por fim, também é fornecida uma lista contendo o valor do *hurdle* (limiar de ativação) para cada vértice da rede.

Figura 7 - Grafo direcionado para geração de população inicial de indivíduos.

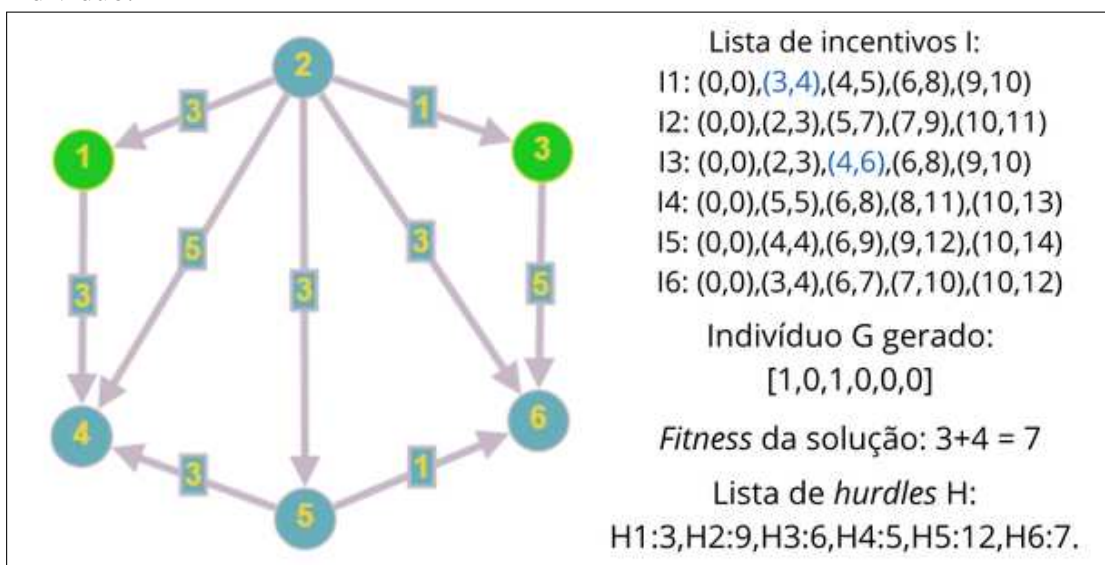


Fonte: Autor(2025).

A figura 8 apresenta a seleção dos dois vértices com menor custo de ativação, com base na lista de incentivos fornecida.

Assim, foi construído o indivíduo $[1, 0, 1, 0, 0, 0]$, onde os vértices 1 e 3 encontram-se ativados. Esses dois vértices foram selecionados por apresentarem os menores custos de ativação entre todos os vértices disponíveis, respeitando a lista de incentivos, o que resulta na melhor solução possível dentro dessa abordagem *gulosa*.

Figura 8 - Grafo direcionado para geração de população inicial de indivíduos, primeiro indivíduo.



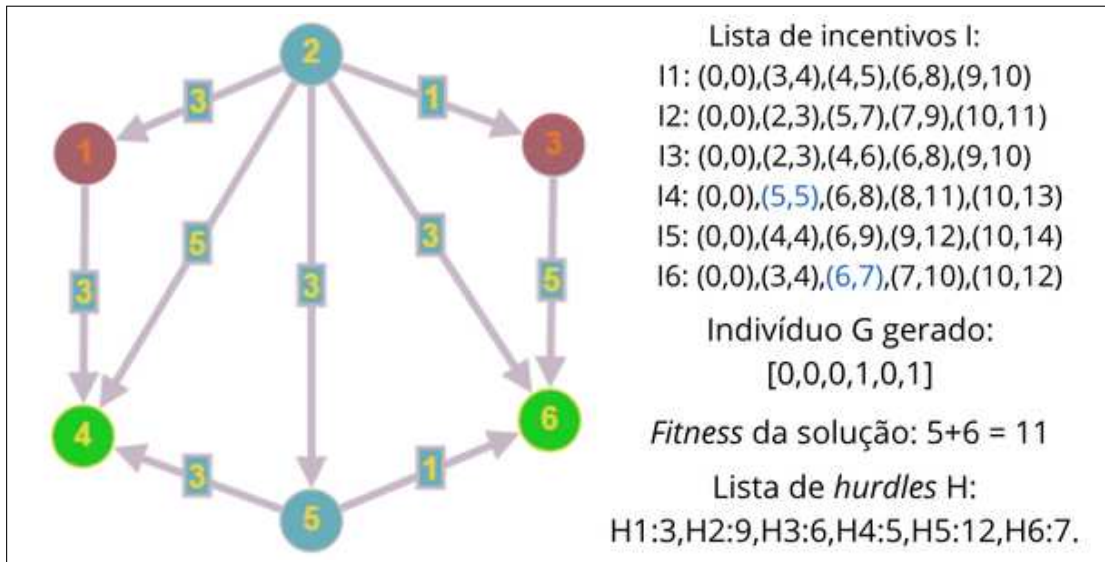
Fonte: Autor(2025).

A figura 9 apresenta uma nova seleção de vértices, onde os vértices 1 e 3 estão

indisponíveis para escolha. Diante dessa restrição, os vértices selecionados são os vértices 4 e 6, que apresentam os menores custos de ativação dentre os restantes.

Essa escolha gera o indivíduo $[0, 0, 0, 1, 0, 1]$, representando a segunda melhor solução dentro do critério *guloso* adotado, considerando a ordem crescente dos custos de ativação.

Figura 9 - Grafo direcionado para geração de população inicial de indivíduos, segundo indivíduo.

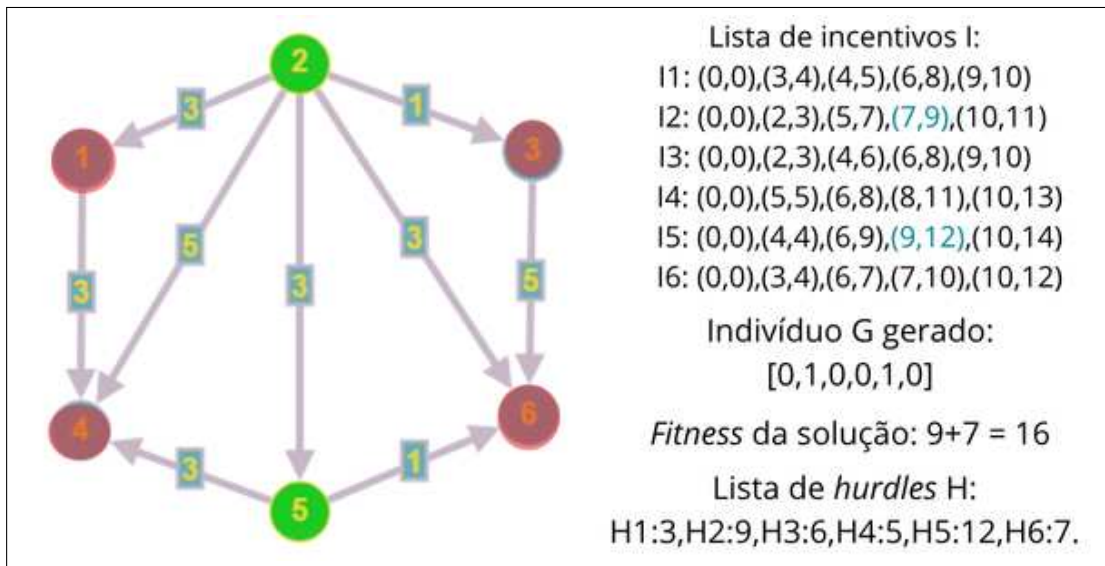


Fonte: Autor(2025).

Por fim, a figura 10 apresenta o cenário em que os vértices 1, 3, 4 e 6 estão indisponíveis para seleção. Diante disso, restam apenas os vértices 2 e 5 como opções viáveis.

Essa combinação gera o indivíduo $[0, 1, 0, 0, 1, 0]$, que representa a terceira melhor solução segundo o critério *guloso*, baseado na seleção dos vértices com menor custo de ativação disponíveis.

Figura 10 - Grafo direcionado para geração de população inicial de indivíduos, terceiro indivíduo.

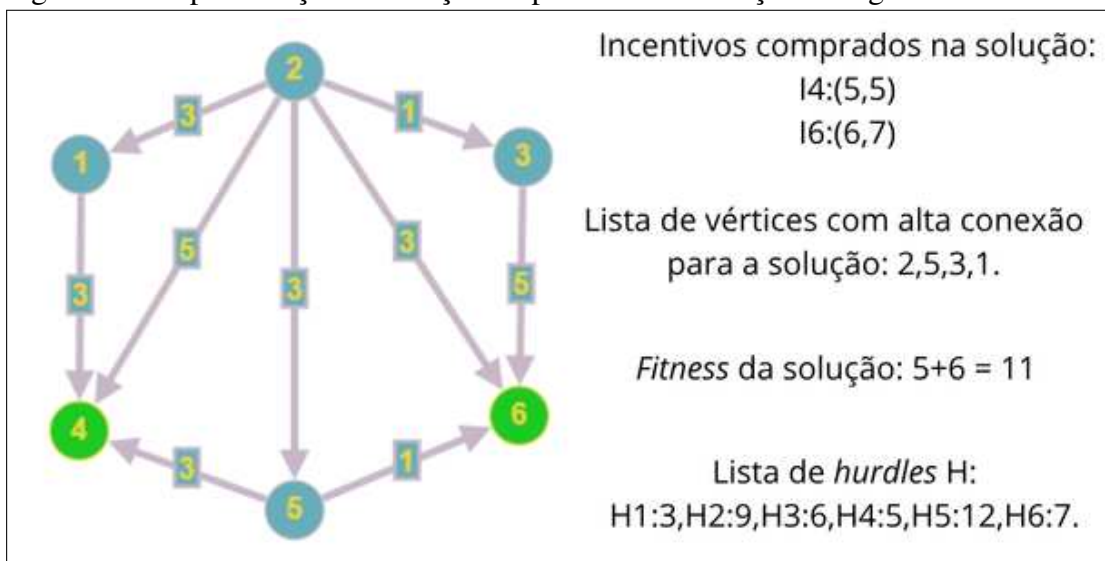


Fonte: Autor(2025).

A figura 11 apresenta uma solução gerada durante o processo de criação inicial da população de indivíduos. Nesta solução, foram selecionados os vértices 4 e 6, com custos 5 e 6, respectivamente, totalizando um valor de *fitness* igual a 11.

Além disso, a partir dessa solução, é construída uma lista de vértices com alta conectividade com os vértices selecionados, essa lista, possui vértices altamente candidatos para substituírem alguns vértices, presente no indivíduo, durante o processo de mutação. No exemplo, os vértices 2, 5, 3 e 1 exercem algum tipo de influência sobre os vértices 4 e 6, sendo, portanto, considerados bons candidatos para o processo de mutação.

Figura 11 - Representação de solução no processo de mutação do algoritmo.



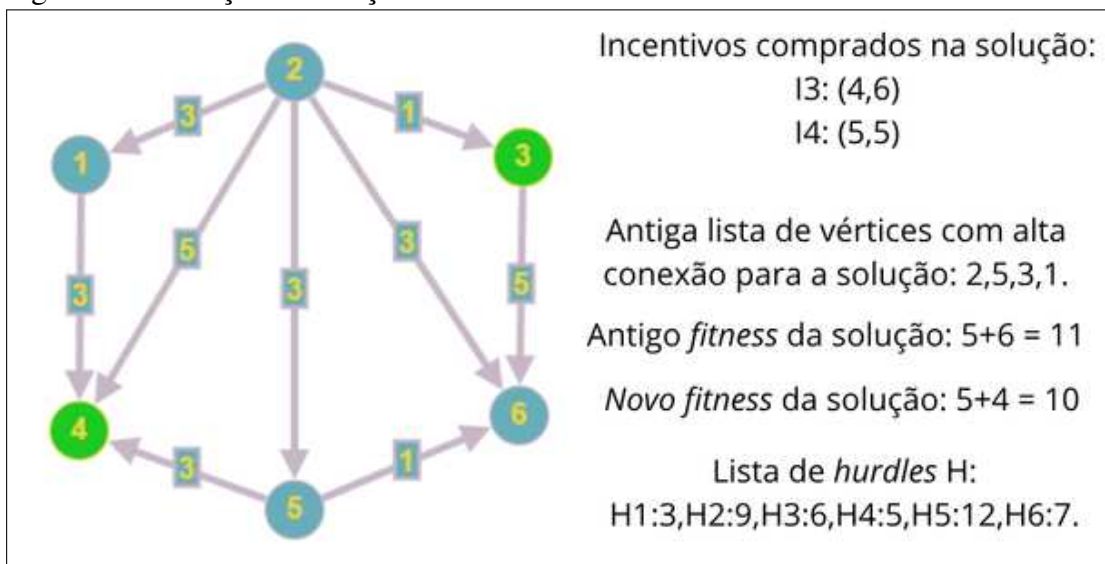
Fonte: Autor(2025).

Na figura 12, o algoritmo de mutação substituiu o vértice 6 pelo vértice 3, reduzindo o custo de ativação de 6 para 4. Como resultado, o novo valor de *fitness* da solução passa a ser 10, em vez de 11.

O vértice 3 foi selecionado aleatoriamente a partir da lista de vértices com alto grau de influência sobre os vértices da solução original.

Esse processo de mutação é aplicado a todos os indivíduos gerados na população inicial, buscando sempre melhorar seus respectivos valores de *fitness*. Ao final da execução, o algoritmo retorna o indivíduo com o menor valor de *fitness* encontrado na população.

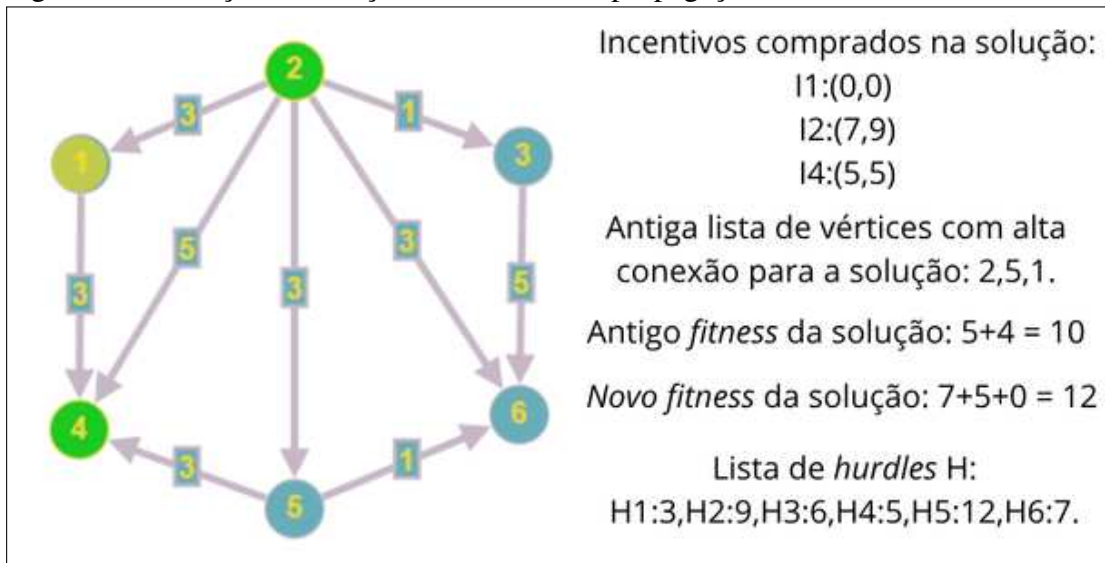
Figura 12 - Mutação de solução realizada.



Fonte: Autor(2025).

Na Figura 13, o algoritmo de mutação substituiu o vértice 3 pelo vértice 2. Nesse caso, o valor de *fitness* da solução piora, passando de 10 na solução anterior (mostrada na Figura 12) para 12. Contudo, o vértice 1 tem seu limiar de ativação atingido por conta da influência recebida do vértice 2, sendo ativado sem a necessidade de incentivo adicional, ou seja, sem custo.

Figura 13 - Mutaç o de soluç o realizada, com propagaç o de influ ncia.



Fonte: Autor(2025).

6.2.1 Seleç o das inst ncias para etapa de testes

Para testar as melhorias propostas para o algoritmo meta-heur stico foi utilizado o mesmo grupo de inst ncias disponibilizadas no trabalho de Costa (2019).

6.2.2 Avaliaç o dos resultados obtidos

Para avaliar a qualidade do algoritmo e das soluç es encontradas, foi realizado o c lculo do *GAP* para tr s conjuntos de inst ncias, denominados mundos 1, 2 e 3. As Tabelas 1,2 e 3 apresentam os resultados obtidos para cada um desses mundos.

Na tabela, o par metro $|V|$ representa a quantidade de v rtices da inst ncia, e α , o percentual de v rtices da rede que devem ser ativados. O *GAP* indica a diferenç a percentual entre a soluç o encontrada pelo algoritmo testado e a soluç o  tima fornecida pelo modelo exato. A coluna Tempo refere-se ao tempo de execuç o de cada algoritmo, e a coluna MELHOR indica qual dos algoritmos obteve o menor *GAP*, ou seja, o melhor desempenho.

  poss vel observar na Tabela 1 que as melhorias propostas resultaram em uma reduç o significativa do *GAP* para todas as inst ncias analisadas. Para esse conjunto de testes, a m dia do *GAP* do algoritmo com as melhorias foi de 38%, enquanto, sem as melhorias, o valor m dio foi de 69%. Essa diferenç a evidencia que as modificaç es implementadas contribuíram para aprimorar os resultados obtidos. Em outras palavras, sem as melhorias, a meta-heur stica apresentou desempenho, em m dia, 69% inferior ao do algoritmo exato; com as melhorias,

essa diferença foi reduzida para 38%, o que representa um ganho considerável em termos de qualidade das soluções.

Tabela 1 – Comparação de desempenho para instâncias do mundo 1, entre o algoritmo de Costa (2019) e o do autor.

IVI	α	GAP COSTA	TEMPO COSTA	GAP AUTOR	TEMPO AUTOR	MELHOR
50	0,1	0,45	0:00:01.62	0,00	0:00:02.70	Autor
	0,5	0,95	0:00:01.74	0,55	0:00:04.10	Autor
	1,0	0,61	0:00:00.49	0,39	0:00:02.00	Autor
75	0,1	0,17	0:00:04.57	0,02	0:00:09.00	Autor
	0,5	1,23	0:00:02.86	0,73	0:00:06.10	Autor
	1,0	0,70	0:00:00.45	0,45	0:00:02.40	Autor
100	0,1	0,68	0:00:07.45	0,28	0:00:14.00	Autor
	0,5	0,96	0:00:04.65	0,74	0:00:13.00	Autor
	1,0	0,46	0:00:00.72	0,34	0:00:02.10	Autor

Fonte: Autor (2025).

Observa-se na Tabela 2 que, em apenas uma das instâncias, o algoritmo sem as melhorias apresentou um *GAP* inferior ao do algoritmo com melhorias. No entanto, mesmo nesse cenário, a média geral do *GAP* foi consideravelmente melhor com as melhorias propostas: 44% contra 71% do algoritmo original. Esse resultado reforça a eficácia das alterações implementadas, demonstrando que, de forma geral, o desempenho do algoritmo foi significativamente aprimorado.

Tabela 2 – Comparação de desempenho para instâncias do mundo 2, entre o algoritmo de Costa (2019) e o do autor.

IVI	α	GAP COSTA	TEMPO COSTA	GAP AUTOR	TEMPO AUTOR	MELHOR
50	0,1	0,25	0:00:01.90	0,33	0:00:01.90	Costa
	0,5	0,36	0:03:22.84	0,31	0:00:05.50	Autor
	1,0	0,59	0:17:05.43	0,40	0:00:01.40	Autor
75	0,1	1,21	0:00:04.57	0,12	0:00:07.00	Autor
	0,5	0,95	0:00:02.86	0,78	0:00:06.20	Autor
	1,0	0,97	0:00:00.45	0,41	0:00:01.20	Autor
100	0,1	0,40	0:00:06.82	0,24	0:00:12.70	Autor
	0,5	1,21	0:00:04.58	0,95	0:00:12.80	Autor
	1,0	0,49	0:00:00.77	0,44	0:00:02.40	Autor

Fonte: Autor (2025).

Observa-se na Tabela 3 o mesmo fenômeno identificado na Tabela 2, apenas uma das instâncias apresentou um *GAP* inferior na versão do algoritmo sem as melhorias. Ainda

assim, a média do *GAP* foi significativamente melhor com as melhorias propostas, com 72%, em comparação com os 95% observados na versão original do algoritmo. Esse resultado reforça o impacto positivo das modificações realizadas, indicando uma melhoria consistente no desempenho da meta-heurística.

Tabela 3 – Comparação de desempenho para instâncias do mundo 3, entre o algoritmo de Costa (2019) e o do autor.

IVI	α	GAP COSTA	TEMPO COSTA	GAP AUTOR	TEMPO AUTOR	MELHOR
50	0,1	2,34	0:00:01.70	2,29	0:00:03.00	Autor
	0,5	0,66	0:00:01.41	0,35	0:00:04.20	Autor
	1,0	0,68	0:00:00.23	0,44	0:00:01.30	Autor
75	0,1	0,49	0:00:03.95	0,09	0:00:08.30	Autor
	0,5	0,48	0:00:03.13	0,43	0:00:07.70	Autor
	1,0	0,48	0:00:00.42	0,53	0:00:03.70	Costa
100	0,1	1,57	0:00:06.26	1,31	0:00:11.50	Autor
	0,5	1,24	0:00:04.57	0,76	0:00:10.80	Autor
	1,0	0,68	0:00:00.70	0,38	0:00:02.70	Autor

Fonte: Autor (2025).

7 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho abordou o problema da disseminação de influência em redes sociais, mais especificamente o Problema da Minimização do Custo de Propagação de Influência em Redes Sociais (PMCPIRS). A motivação central foi explorar maneiras eficientes de selecionar um subconjunto de nós capazes de propagar uma ideia, produto ou comportamento com o menor custo possível, aproveitando as relações de influência presentes na estrutura da rede.

Para isso, foi implementada uma versão baseada no Algoritmo Genético proposto por Costa (2019), e, em seguida, melhorias foram introduzidas no processo de geração da população inicial e no operador de mutação, utilizando características estruturais do grafo, como centralidade e distribuição de graus. Essas modificações tiveram como objetivo acelerar a convergência e melhorar a qualidade das soluções encontradas.

Os resultados experimentais demonstraram que as melhorias propostas foram eficazes, com reduções significativas no *GAP* médio entre as soluções obtidas, através das melhorias propostas pelo autor, em relação ao algoritmo sem as melhorias, proposto por Costa (2019). As comparações mostraram que a nova abordagem superou o algoritmo original em diversas instâncias, comprovando a utilidade de considerar informações estruturais do grafo durante a execução da heurística.

Como trabalhos futuros, esperam-se os seguintes acréscimos:

- Adicionar uma nova restrição ao problema, para limitar os custos finais;
- Explorar diferentes operadores genéticos adaptados à estrutura de grafos;
- Aplicar o modelo proposto a redes reais de grande escala, como redes de colaboração científica ou redes sociais online;
- Estender o modelo para diferentes variantes do problema, como restrições de tempo, limiares dinâmicos ou múltiplas fontes de influência.

Em síntese, este trabalho contribui para o avanço na resolução eficiente de problemas de propagação em redes sociais e abre caminho para novas investigações na área de otimização em grafos com aplicação em contextos reais.

REFERÊNCIAS

- ACKERMAN, E.; BEN-ZWI, O.; WOLFOVITZ, G. Combinatorial model and bounds for target set selection. **Theoretical Computer Science**, v. 411, n. 44, p. 4017–4022, 2010.
- ALAM, T.; QAMAR, S.; DIXIT, A.; BENAIDA, M. Genetic algorithm: Reviews, implementations, and applications. **International Journal of Engineering Pedagogy**, v. 10, n. 6, p. 60–63, 2020. Disponível em <https://online-journals.org/index.php/ijep/article/view/14567/8299>. Acesso em: 10 jul 2024.
- ALMEIDA, M. R. **Programação linear**: Uma aplicação ao problema de compras de um supermercado da cidade de macaúbas-ba. 2018. Trabalho de Conclusão de curso (Licenciatura em Matemática) - Universidade Estadual do Sudoeste da Bahia, Vitória da Conquista, BA, 2018.
- ALVES, R.; DELGADO, C. **Programação linear inteira**. 1997. Disponível em <https://repositorio-aberto.up.pt/bitstream/10216/74369/2/40539.pdf>. Acesso em: 11 dez. 2023.
- COSTA, R. D. A. **Problema de minimização de custo de propagação de influência em redes sociais**: uma abordagem heurística. 2019. Trabalho de Conclusão de Curso (Graduação em Engenharia de Software) – Universidade Federal do Ceará, Russas, CE, 2019.
- FISCHETTI, M.; KAHR, M.; LEITNER, M.; MONACI, M.; RUTHMAIR, M. Least cost influence propagation in (social) networks. **Mathematical Programming**, Springer, v. 170, n. 1, p. 293–325, 2018.
- FORBES BRASIL. **Orcamentos milionários ganham destaque no marketing de influência brasileiro**. 2023. Disponível em: <https://forbes.com.br/forbes-tech/2023/04/orcamentos-milionarios-ganham-destaque-no-marketing-de-influencia-brasileiro/>. Acesso em: 11 dez. 2023.
- GOMES, P. F. Uma introdução à ciência de redes e teoria de grafos. **Revista Brasileira de Ensino de Física**, SciELO Brasil, v. 46, p. e20240190, 2024.
- GRANOVETTER, M. Threshold models of collective behavior. **American Journal of Sociology**, University of Chicago Press, v. 83, n. 6, p. 1420–1443, 1978.
- GUNNEC, D.; RAGHAVAN, S. Integrating social network effects in the share-of-choice problem. **Decision Sciences**, Wiley Online Library, v. 48, n. 6, p. 1098–1131, 2017.
- IMEDIA COMUNICAÇÃO. **O crescimento do marketing de influência durante o coronavírus**. 2023. Disponível em: <https://imediacomunicacao.com.br/o-crescimento-do-marketing-de-influencia-durante-o-coronavirus/>. Acesso em: 11 dez. 2023.
- KEMPE, D.; KLEINBERG, J.; TARDOS, Maximizing the spread of influence through a social network. In: ACM. **Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [S.l.], 2003. p. 137–146.
- KIMURA, M.; SAITO, K.; NAKANO, R. **Extracting influential nodes for information diffusion on a social network**. 2007. 1371–1376 p.
- LUZIA, L.; RODRIGUES, M. **Introdução ao escalonamento e aplicações: estudo sobre as metaheurísticas**. 2009. Instituto de Matemática e Estatística – IME, Universidade de São Paulo – USP. 38 p. Disponível em: <https://www.ime.usp.br/gold/cursos/2009/mac5758/LeandroMauricioHeuristica.pdf>. Acesso em: 11 dez. 2023.

MARTINS, E. R. Pesquisa operacional e sua atuação multidisciplinar. **Ponta Grossa, PR: Atena Editora**, Atena Editora, 2019.

NOGUEIRA, M.; SAAVEDRA, O. Estratégias evolutivas aplicadas à resolução de otimização multimodal. In: **Simpósio Brasileiro de Automação Inteligente**. [S.l.: s.n.], 1999.

RAGHAVAN, S.; ZHANG, R. A branch-and-cut approach for the weighted target set selection problem on social networks. **INFORMS Journal on Optimization**, INFORMS, v. 1, n. 4, p. 304–322, 2019.

SHAKARIAN, P.; EYRE, S.; PAULO, D. A scalable heuristic for viral marketing under the tipping model. **Social Network Analysis and Mining**, Springer, v. 3, n. 4, p. 1225–1248, 2013.

SODRÉ, U. Modelos matemáticos. **Londrina: UEL**, 2007. Disponível em: <https://www.uel.br/projetos/matessencial/superior/pdfs/modelos.pdf>. Acesso em: 5 dez. 2023.

WILHELM, V. A. **Programação linear: método simplex algébrico**. 2020. Disponível em: <https://docs.ufpr.br/volmir/POI/DOCs%20simplex%20algebrico.pdf> Acesso em: 11 dez. 2023. Acesso em: 11 dez. 2023.