

Aprendizado por Reforço em uma Heurística Multi-Centralidades para a Redução de Largura de Banda em Matrizes Esparsas

João Vitor Maués Dias Carneiro

Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro
jvmaues@cos.ufrj.br

Rafael Schneider

Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro
rschneider@cos.ufrj.br

Laura Assis

Programa de Pós-graduação em Ciência da Computação
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca
laura.assis@cefet-rj.br

Sanderson L. Gonzaga de Oliveira

Departamento de Ciência e Tecnologia
Universidade Federal De São Paulo
sgonzaga@unifesp.br

Diego Nunes Brandão

Escola de Informática e Computação
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca
diego.brandao@cefet-rj.br

Claudio Miceli Farias

Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro
cmicelifarias@cos.ufrj.br

Pedro Henrique González

Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro
pegonzalez@cos.ufrj.br

RESUMO

O Problema da Minimização da Largura de Banda para Matrizes Esparsas é um problema NP-Difícil relevante em várias aplicações científicas. Abordagens tradicionais, como o algoritmo de Cuthill-McKee, usam a centralidade de grau para minimizar a largura de banda. Este trabalho apresenta uma abordagem multicentralidade, utilizando Aprendizado por Reforço (AR) com Q-Learning para otimizar a seleção de diferentes medidas de centralidade na rotulagem dos vértices de um grafo. O processo de seleção é modelado como um problema de tomada de decisão, onde um agente aprende a escolher a medida de centralidade mais eficaz com base na classe da instância. O método baseado em AR adapta-se dinamicamente às características específicas de diferentes matrizes esparsas, resultando em uma minimização de largura de banda mais eficiente. Resultados experimentais mostram que essa abordagem adaptativa melhora substancialmente a resolução de problemas complexos de minimização de largura de banda, superando heurísticas tradicionais em eficiência computacional.

PALAVRAS CHAVE. Problema de Redução de Largura de Banda Centralidade de Grafo Método de Cuthill-McKee

Paper topics (IC - Computational Intelligence, OC - Combinatorial Optimization)

ABSTRACT

The Bandwidth Minimization Problem for Sparse Matrices is a well-known NP-Hard problem relevant in various scientific applications. Traditional approaches, such as the Cuthill-McKee algorithm, use degree centrality to minimize bandwidth. This work presents a multi-centrality approach using Reinforcement Learning (RL) with Q-Learning to optimize the selection of different centrality measures in vertex labeling of a graph. The selection process is modeled as a decision-making problem, where an agent learns to choose the most effective centrality measure based on the instance class. The RL-based method dynamically adapts to the specific characteristics of different sparse matrices, resulting in more efficient bandwidth minimization. Experimental results show that this adaptive approach substantially improves the resolution of complex bandwidth minimization problems, outperforming traditional heuristics in computational efficiency.

KEYWORDS. Bandwidth Reduction Problem Graph Centrality Cuthill-McKee method

Paper topics (IC - Computational Intelligence, OC - Combinatorial Optimization)

1. Introdução

Resolver sistemas lineares esparsos de grande escala, representados por $Ax = b$, onde A é uma matriz esparsa $n \times n$, x é o vetor incógnita de dimensão n e b é um vetor conhecido de dimensão n , é crucial em várias áreas científicas e de engenharia. O Problema de Minimização de Largura de Banda (BMP) surgiu nos anos 1950, quando engenheiros estruturais começaram a analisar estruturas de aço usando computadores [Chinn et al., 1982]. Concentrar elementos não nulos em uma faixa estreita ao redor da diagonal principal e usar uma matriz de reordenação aumenta a eficiência em operações como inversão e cálculo de determinantes, economizando tempo [Chinn et al., 1982]. Heurísticas de redução de largura de banda buscam posicionar coeficientes não nulos o mais próximo possível da diagonal principal. A Figura 1 mostra um exemplo de aplicação dessas heurísticas a uma matriz.

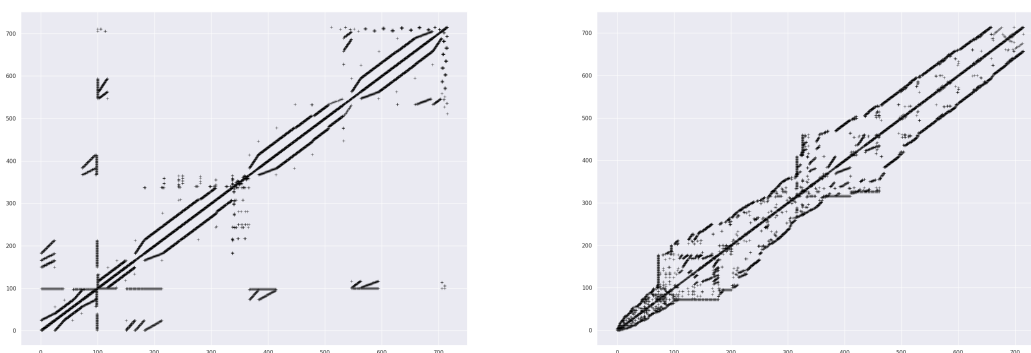


Figura 1: Exemplo de como uma instância de matriz esparsa se comporta antes e depois da redução de largura de banda usando o algoritmo de Cuthill-McKee para tal redução.

O método Cuthill-McKee [Cuthill e McKee, 1969] é uma das heurísticas mais clássicas para redução de largura de banda. Outro algoritmo altamente eficaz e considerado por muitos pesquisadores como o estado da arte para o Problema de Minimização de Largura de Banda (PMB) é o DRSA [Torres-Jimenez et al., 2015].

Diversas heurísticas e metaheurísticas têm sido desenvolvidas buscando reduzir a largura de banda [Chagas e Gonzaga de Oliveira, 2015]. No entanto, as metaheurísticas atuais enfrentam

um alto custo computacional durante a execução. Sendo assim, o método Cuthill-McKee segue amplamente utilizado devido a sua fácil implementação e baixo custo computacional. O algoritmo utilizado nesse método é guiado por uma escolha de centralidade de grau na ordem da rotulagem dos vértices.

Motivados por essa alternativa interessante, utilizamos uma abordagem que emprega diversas centralidades, inserindo de forma aleatória as escolhas das centralidades em um processo multi-start. Com o intuito de contornar a aleatoriedade dessa escolha, este artigo propõe o uso de aprendizado por reforço na seleção da centralidade. A aplicação desta técnica tem se revelado uma estratégia eficiente na resolução problemas de otimização combinatória complexos [Mazyavkina et al., 2021; Kobayashi et al., 2024; Souto et al., 2024; Alves et al., 2024]. Dessa forma, no contexto da redução de largura de banda, o algoritmo de aprendizado por reforço buscará identificar a melhor centralidade para um conjunto de classes de instâncias para minimizar a largura de banda da matriz esparsa. Essa abordagem visa combinar a eficácia das heurísticas tradicionais com a capacidade adaptativa proporcionada pelo aprendizado por reforço.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta a descrição do problema e a formulação matemática. A Seção 3 revisa a literatura relacionada ao problema. A Seção 4 detalha a metodologia utilizada. A Seção 5 descreve os experimentos computacionais realizados. Por fim, a Seção 6 apresenta os resultados e trabalhos futuros.

2. Descrição do Problema e formulação Matemática

Utilizamos uma formulação matemática baseada na encontrada em [Garey e Johnson, 1979]. Considere um grafo $G = (V, E)$, onde V é o conjunto de vértices e E é o conjunto de arestas. Seja $n = |V|$. Uma rotulação f de G atribui os inteiros $\{1, 2, \dots, n\}$ aos vértices de G . A largura de banda de um grafo G com a rotulação $f : V \rightarrow \mathbb{N}$ é definida como:

$$B_w(f) = \max_{\{u,v\} \in E} |f(u) - f(v)|$$

O objetivo do problema de redução de largura de banda é encontrar uma rotulação f que minimize $B_w(f)$. Uma abordagem tradicional deste problema é considerar que $A = [a_{ij}]$ representa a matriz de incidência de um grafo V . O problema de redução de largura de banda busca identificar uma permutação das linhas e colunas de forma que todos os elementos não nulos de A fiquem o mais próximo possível da diagonal principal. Esta abordagem é relevante em aplicações práticas para resolver sistemas de equações algébricas lineares não singulares na forma $Ax = b$ [Campos, 2001].

3. Trabalhos relacionados

O problema abordado neste artigo tem dois objetivos principais: maximizar a redução da largura de banda e manter um tempo de execução eficiente. Com base nesses critérios, a Tabela 1 busca destacar os estudos mais relevantes sobre o BMP, identificando os principais métodos propostos para resolver esse desafio.

Um estudo focado em heurísticas de baixo custo computacional, conduzido por [Gonzaga de Oliveira et al., 2018b], concluiu que a meta-heurística de Busca por Vizinhança Variável descrita em [Mladenovic et al., 2010] mostrou-se mais eficaz na redução da largura de banda para instâncias da coleção de matrizes SuiteSparse [Davis e Hu, 2011], superando outras heurísticas de baixo custo. O algoritmo denotado por FNCHC, que consiste em uma versão acelerada do algoritmo Node Centroid with Hill Climbing (NCHC), destacou-se como a segunda melhor neste conjunto, mas alcançou os melhores resultados em 12 instâncias de um conjunto menor. Embora o VNS-band tenha obtido resultados superiores, ele apresentou um custo computacional consideravelmente mais alto.

Tabela 1: Métodos heurísticos mais promissores para redução de largura de banda.

Heurística	Ano	Redução de largura de banda	Custo
Cuthill-McKee [Cuthill e McKee, 1969]	1969	baixa	baixo
RCM [George, 1971]	1971	baixa	baixo
FNCHC [Lim et al., 2007]	2007	razoável	alto
VNS-band [Mladenovic et al., 2010]	2010	alta	alto
KP-band [Koohestani e Poli, 2011]	2011	baixa	baixo
DRSA [Torres-Jimenez et al., 2015]	2015	alta	alto
RBFS-GL [Gonzaga de Oliveira et al., 2018a]	2018	baixa	baixo
ACHH [Gonzaga de Oliveira e Silva, 2020a,b, 2021]	2020	alta	baixo
FNCHC+ [Gonzaga de Oliveira e Carvalho, 2022; Gonzaga de Oliveira, 2023]	2022	alta	razoável

O trabalho relevante mais recente para o BMP foi realizado por [Gonzaga de Oliveira e Silva, 2020b], que introduziu uma abordagem hiperheurística de colônia de formigas como um novo e promissor método para a otimização da largura de banda em matrizes de grande escala. Este estudo comparou sua eficácia com heurísticas de baixo custo, e os resultados baseados em uma ampla gama de instâncias de matrizes esparsas indicam que a abordagem proposta pode ser mais eficaz e eficiente do que as heurísticas de baixo custo existentes na otimização da largura de banda para matrizes de grande escala.

Outro ponto relevante destacado na Tabela 1 é o método de estado da arte, o DRSA [Torres-Jimenez et al., 2015], cujos tempos de execução são maiores do que os do VNS-band [Gonzaga de Oliveira e Silva, 2020b; Gonzaga de Oliveira e Carvalho, 2022; Gonzaga de Oliveira, 2023]. Embora o DRSA tenha uma alta taxa de redução da largura de banda, na prática, não é viável utilizá-lo, pois a resolução de sistemas lineares sem esses métodos geralmente é mais rápida.

4. Metodologia

Esta seção apresenta o método heurístico de multi-centralidades com a integração de escolhas de centralidades via aprendizado por reforço (Q-MCH), que combina uma heurística construtiva de múltiplos reinícios com uma abordagem de aprendizado por reforço na escolha de multi-centralidades para a ordenação da rotulação dos vértices para o problema de redução de largura de banda. A Seção 4.1 introduz o Algoritmo Construtivo de multi-centralidade, que opera com uma lista restrita de candidatos para a escolha dos vértices a serem rotulados. A segunda subseção explica como o algoritmo de aprendizado por reforço foi implementado junto a heurística MCH com lista de candidatos restrita, onde a escolha das centralidades é realizada via aprendizado de máquina.

4.1. Algoritmo Construtivo de Multi-Centralidade

A Heurística Construtiva proposta busca gerar soluções de qualidade de forma eficiente por meio de reinicializações iterativas, explorando diversas opções na busca por uma solução de boa qualidade. No Algoritmo 1, construímos uma sequência de vértices q usando busca em largura para obter uma solução construtiva de boa qualidade, rotulando-os como $1, 2, \dots, n$. A rotulação segue a ordem crescente da centralidade no nível de vizinhança atual. A ideia é que vértices adjacentes tenham rótulos próximos em uma boa solução (com largura de banda máxima pequena), conforme explicado em [Mladenovic et al., 2010]. Além disso, é construída uma lista de candidatos restrita a partir da seguinte construção: escolhemos c_i como o critério guloso referente à inserção de um elemento i na solução corrente, o que corresponde ao valor da centralidade daquele vértice.

$$h_{\min} = \min\{c_i\}, h_{\max} = \max\{c_i\}, \alpha \in [0, 1]$$

$$LCR = \{i \mid h_{\min} \leq c_i \leq h_{\max} + \alpha(h_{\min} - h_{\max})\}$$

A seleção do próximo vértice a ser adicionado à solução é conduzida por meio de uma Lista de Candidatos Restrita (LCR), que é elaborada com base em uma função adaptativa gulosa

que avalia o benefício da inclusão de cada elemento possível na solução. A LCR contém apenas os elementos que podem ser adicionados à solução e é atualizada a cada iteração do algoritmo de construção. A escolha final do elemento a ser incorporado à solução é realizada aleatoriamente a partir dos elementos presentes na LCR, possibilitando a geração de soluções distintas a cada execução do algoritmo multi-start.

Algorithm 1: Algoritmo Construtivo de Multi-Centralidades LCR

```

1 Data: Grafo  $A$ ,  $ArrayCentrality$ 
2  $centrality \leftarrow randomChoice(ArrayCentrality)$ ;
3 for  $i \leftarrow 1$  to  $|V|$  do
4    $mark[i] \leftarrow false$ ;
5 end
6  $k \leftarrow RandomInt(1, |V|)$ ;
7  $mark[k] \leftarrow true$ ;
8  $q[1] \leftarrow k$ ;  $ql \leftarrow 1$ ;  $f[k] \leftarrow 1$ ;  $l \leftarrow 1$ ;
9 while  $l < |V|$  do
10    $r \leftarrow 0$ ;
11   for  $il \leftarrow 1$  to  $ql$  do
12      $i \leftarrow q[il]$ ;
13     for  $jl \leftarrow 1$  to  $|A[i]|$  do
14        $j \leftarrow a[i, jl]$ ;
15       if not  $mark[j]$  then
16          $r \leftarrow r + 1$ ;  $s[r] \leftarrow j$ ;
17       end
18     end
19   end
20    $q \leftarrow getLCR(q, lambda(x) : centrality[x], alpha)$ ;
21    $j^* \leftarrow 0$ ;
22   for  $jl \leftarrow 1$  to  $ql$  do
23      $j \leftarrow q[j^*]$ ;  $l \leftarrow l + 1$ ;  $f[j] \leftarrow l$ ;  $j^* \leftarrow j^* + 1$ ;
24   end
25    $q \leftarrow s$ ;  $ql \leftarrow r$ ;
26 end
27 return  $f$ 

```

Os vértices para a busca em largura são colocados em níveis da seguinte forma: o primeiro nível contém apenas o vértice k (selecionado aleatoriamente), e o rótulo 1 é atribuído a ele ($f(k) \leftarrow 1$); enquanto o segundo nível é composto pelos vértices adjacentes a k , ordenados de maneira decrescente, com os rótulos 2, 3, ... O terceiro nível contém vértices não incluídos em níveis anteriores, mas conectados a vértices no segundo nível, com a ordem de rotulação sempre seguindo os valores de centralidade de cada vértice na lista de candidatos restrita (LCR).

Suponha que haja n_i vértices no nível i . O método escolhe aleatoriamente uma centralidade para ordenar os vértices q . O processo de rotulação ocorre então pela sequência de vértices ordenados. Além disso, o Algoritmo 2 foi implementado com um recurso de controle de múltiplos inícios para executar o Algoritmo 1. Ele recebe como entrada um grafo, o número máximo de iterações max_{iter} , e uma lista de dicionários de centralidade, onde cada dicionário é construído com um vértice vinculado a um valor de centralidade no vértice em questão. Em seguida, exe-

cuta o algoritmo de múltiplos inícios max_{iter} vezes, utilizando a lista calculada de centralidade como parâmetro. O objetivo do algoritmo é realizar múltiplas inicializações (multi-start) do algoritmo principal, possivelmente variando a medida de centralidade usada, para explorar diferentes soluções iniciais e aumentar as chances de encontrar uma solução de alta qualidade.

Algorithm 2: Heurística de Multi-Centralidade com Múltiplos Inícios

```

1 Data: Grafo  $A$ , int  $max_{iter}$ , Lista  $ArrayCentrality$ 
2  $f_{sol} \leftarrow inf$ ;
3 for  $k \leftarrow 1$  to  $max_{iter}$  do
4    $f \leftarrow ConstructiveCentralities(A, ArrayCentrality)$ 
5   if  $f < f_{sol}$  then
6      $f_{sol} \leftarrow f$ 
7   end
8 end
9 return  $f_{sol}$ 

```

Este algoritmo combina a flexibilidade de múltiplas chamadas aleatórias para a escolha inicial do primeiro vértice a ser rotulado pelo algoritmo com a diversificação proporcionada pela seleção de diferentes medidas de centralidade. A ideia é que centralidades diversificadas possam guiar a construção da solução de várias maneiras, explorando vários aspectos da estrutura do grafo.

4.2. Algoritmo Construtivo Multi-Centralidade com Q-Learning

Integrar o Algoritmo Construtivo de multi-centralidade com Q-Learning com o método descrito envolve a adaptação de técnicas de aprendizado por reforço para otimizar a seleção de centralidades durante os reinícios do algoritmo. Embora a introdução de uma nova camada possa afetar o desempenho temporal, ela também oferece a oportunidade de alcançar recompensas significativas.

Nossa abordagem busca aprimorar os resultados do algoritmo de linha de base, o MCH com lista de candidatos restrita, identificando a centralidade mais adequada para cada conjunto de classes de instâncias analisadas. Inicialmente, definimos parâmetros iniciais, como a configuração de α e max_{iter} . Em seguida, o algoritmo Q-MCH é executado em uma fase inicial de treinamento, agrupando as instâncias em classes. Após o treinamento, a rede neural torna-se responsável por determinar a medida de centralidade que será utilizada para alcançar a maior redução de largura de banda em uma instância previamente escolhida, o que permite explorar os espaços de solução de maneira mais eficiente. O processo é repetido até que um critério de parada seja alcançado, como o número de iterações. Outro parâmetro do Q-MCH é o limite dual. Esse limite é utilizado para substituir o valor ótimo no cálculo do gap, que representa a diferença entre a solução atual e a solução ideal, em cada iteração do processo. Conforme apontado em [Lam et al., 1997], usamos o limite inferior definido pela seguinte fórmula:

$$B_f(G) \geq n \pm \frac{1 + \sqrt{(2n \pm 1)^2 - 8m}}{2}, \text{ onde } n = |V_G| \text{ e } m = |E_G|,$$

Essa integração visa otimizar a seleção de centralidades, aproveitando a flexibilidade e a diversificação inerentes ao método multi-centralidade e ao algoritmo Q-Learning.

4.3. Aprendizado por Reforço

O aprendizado por reforço (AR) é um paradigma de aprendizado de máquina em que um agente aprende a tomar decisões interagindo com um ambiente. O agente recebe recompensas por

suas ações, podendo ser tanto positivas quanto negativas, sendo estas últimas chamadas de penalidades. A Figura 2 representa as etapas fundamentais do processo de Aprendizado por Reforço. Em um determinado instante t , o agente é apresentado com o estado atual S_t do ambiente. Utilizando sua política e o estado atual, o agente seleciona uma ação A_t para ser executada no ambiente. Após a ação para o próximo estado S_{t+1} no instante $t + 1$, o ambiente concede uma recompensa R_{t+1} . Essa informação de recompensa é então utilizada pelo agente para ajustar e aprimorar sua política de decisão [Arulkumaran et al., 2017].

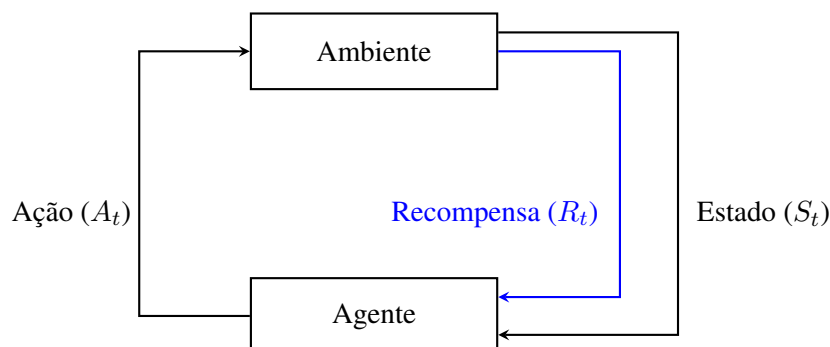


Figura 2: Fluxo de trabalho do Aprendizado por Reforço.

Através da exploração, os agentes são capazes de aprender com suas próprias ações e experiências, refinando a estratégia de tomada de decisão para maximizar as recompensas acumuladas ao longo do tempo, o que torna o uso de algoritmos com essa abordagem especialmente úteis em cenários complexos onde as soluções não são facilmente programáveis. Adicionalmente, uma estratégia para enfrentar espaços de estado complexos e de alta dimensão é o Deep Q-Learning, que consiste na integração do AR com Redes Neurais Profundas (DNN) [Mnih et al., 2013].

5. Experimentos Computacionais

Implementamos o método proposto em Python 3.9.12 e o executamos para cada instância em um Mac Pro Max M2 com 32 GB de RAM. As matrizes utilizadas neste trabalho para comparação foram as mesmas usadas por Mladenovic et al. [Mladenovic et al., 2010]. Utilizamos um conjunto de 26 instâncias, oriundas de 3 classes da SuiteSparse Matrix Collection [Davis e Hu, 2011] para avaliar a eficácia de cada algoritmo implementado na redução de largura de banda.

Dividimos esta seção em três subseções distintas. A primeira parte descreve os detalhes dos parâmetros de implementação e ajuste. As duas partes seguintes analisam de forma abrangente as principais considerações do algoritmo proposto neste trabalho, estabelecendo comparações criteriosas com a literatura existente.

5.1. MCH $\times$ Q-MCH

Neste estudo, uma abordagem combinada foi adotada integrando o algoritmo construtivo MCH com aprendizagem por reforço, utilizando especificamente a Deep Q-Network (DQN). Para melhorar o desempenho do DQN, ele foi treinado usando uma amostra de instâncias do problema, o DQN utilizado é uma rede neural feedforward com três camadas totalmente conectadas.

A configuração de parâmetros para o Q-MCH adaptada a um conjunto de instâncias e o desempenho do DQN, foi treinado usando uma amostra de instâncias do problema, o DQN usado é uma rede neural feedforward com três camadas totalmente conectadas, em que os parâmetros utilizados: γ , ϵ , ϵ_{min} e ϵ_{dec} , estão listados na Tabela 2.

Assim, foi essencial estabelecer um parâmetro de iteração máxima. Este parâmetro não poderia ser muito baixo, pois testaria poucas soluções de reinício. Além disso, não poderia ser

muito alto, pois aumentaria o custo computacional do método. Após avaliar algumas possibilidades, estabelecemos $max_{iter} = 100$.

Em nossa proposta, o agente é responsável perante um determinado estado por definir a melhor escolha sobre a centralidade a ser aplicada utilizando o HMC. A representação do estado é composta pelo número de iterações, pela lacuna da melhor solução conhecida e pela lacuna da solução anterior. A recompensa é positiva se a nova solução for melhor que a anterior e negativa caso contrário. Como estamos utilizando duas centralidades, o agente tem duas ações possíveis para escolher vezes o número de vértices na lista de candidatos restrita da instância para iniciar o processo de rotulagem.

Tabela 2: Configurações de parâmetros do agente

Parâmetro & Valor	
learning rate	0.001
γ	0.9
ϵ	1
ϵ_{min}	0.1
ϵ_{dec}	0.995

A Tabela 2 mostra os valores utilizados no processo, a taxa de aprendizagem determina o tamanho do passo no qual o agente atualiza seus valores Q durante o treinamento, γ é o fator de desconto que determina a importância das recompensas futuras em comparação com as recompensas imediatas, ϵ determina o trade-off no processo de tomada de decisão mínima do agente, ϵ_{min} define o limite inferior para ϵ durante o processo de exploração e, finalmente, ϵ_{dec} controla a rapidez com que o ϵ diminui ao longo do tempo.

5.2. Comparações com a Literatura

Procuramos avaliar algoritmos descritos na literatura que apresentavam um tempo de execução inferior a 0,5 segundos, a fim de comparar os resultados obtidos pela abordagem proposta com heurísticas de baixo custo. Além dos algoritmos inicialmente desenvolvidos, também implementamos o componente construtivo da meta-heurística VNS-band proposto em [Mladenovic et al., 2010], para fins de comparação.

Apresentamos na Tabela 3 os resultados para cada instância testada. A coluna “Original” exibe a largura de banda original do grafo. Nas demais colunas, são mostradas a melhor solução encontrada (Best Sol) e o tempo de execução em segundos (Tempo exec) o algoritmo correspondente.

Os resultados apontam que o Q-MCH alcançou reduções significativas na largura de banda em comparação com o algoritmo de Cuthill-McKee. Especialmente para as classes de instância optimization e structural, a foi na ordem de 40, 5% e 38, 5%, respectivamente.

Para validar o melhor desempenho do Q-MCH, avaliamos a diferença estatística significativa entre os dois métodos através do teste de hipótese, considerando os valores encontrados como a melhor solução. Como os dados não seguiam uma distribuição normal e as medidas dos experimentos eram emparelhadas, foi empregado o teste não paramétrico Wilcoxon Signed Rank Test [Hettmansperger e McKean, 2010; Pratt, 1959] a um nível de significância de $\theta = 0,05$. As hipóteses testadas foram:

- Hipótese Nula (H_0): A diferença das melhores soluções é zero.
- Hipótese Alternativa (H_1): A diferença das melhores soluções não é zero.

Tabela 3: Comparativo Q-MCH x Cuthill por instância

Instancia	Classe	Original	Cuthill		Q-MCH	
		Best Sol	Best Sol	Tempo exec	Best Sol	Tempo exec
bp_0.mtx	optimization	1549	1031	0,021288	585	0,356708
bp_1000.mtx	optimization	1643	1071	0,025355	640	0,452862
bp_1200.mtx	optimization	1643	1099	0,019303	647	0,446598
bp_1400.mtx	optimization	1643	1095	0,031912	642	0,457749
bp_1600.mtx	optimization	1643	1079	0,033441	650	0,465606
bp_200.mtx	optimization	1633	1019	0,021413	596	0,390006
bp_400.mtx	optimization	1627	999	0,025189	621	0,407727
bp_600.mtx	optimization	1627	1075	0,022407	631	0,408325
bp_800.mtx	optimization	1643	1085	0,09957	639	0,450543
shl_0.mtx	optimization	1285	849	0,018475	531	0,215352
bcsstk01.mtx	structural	71	55	0,00112	28	0,026956
bcsstk02.mtx	structural	131	131	0,003741	65	0,146682
bcsstk04.mtx	structural	109	109	0,004125	58	0,12903
bcsstk05.mtx	structural	55	49	0,004122	31	0,095921
bcsstk06.mtx	structural	147	101	0,010224	78	0,289728
bcsstk07.mtx	structural	147	101	0,010306	76	0,287401
lshp_265.mtx	thermal	511	37	0,004225	25	0,089789
lshp_406.mtx	thermal	783	45	0,006336	32	0,138994
lshp1009.mtx	thermal	1953	69	0,028204	54	0,343484
lshp1270.mtx	thermal	2457	77	0,024355	61	0,422652
lshp1561.mtx	thermal	3019	85	0,027688	66	0,516754
lshp1882.mtx	thermal	3641	93	0,037781	73	0,62875
lshp2233.mtx	thermal	4333	101	0,056463	82	0,743319
lshp2614.mtx	thermal	5085	109	0,050563	87	0,86284
lshp3025.mtx	thermal	5897	117	0,067248	94	1,006613
lshp3466.mtx	thermal	6769	125	0,085357	104	1,182524

A hipótese nula postula que não há diferenças entre os valores das melhores soluções de cada versão implementada. Entretanto, a hipótese nula foi rejeitada, pois o valor p encontrado foi de $2,98 \times 10^{-8}$, indicando que o número de diferenças entre os dois algoritmos é estatisticamente significativa. Em particular, isso confirma a conclusão de que o Q-MCH supera o Cuthill.

Em complemento, a Tabela 4 consolida a comparação entre o Q-MCH e outros métodos em termos de melhoria na qualidade da solução. Os resultados estão agrupados da seguinte forma: cada célula da tabela representa o número de instâncias da classe em que o Q-MCH melhorou a solução em comparação com o algoritmo destacado na respectiva coluna.

Tabela 4: Melhora na qualidade de solução pelo Q-MCH x Demais Algoritmos

Classe	Original	Cuthill	MCH
Optimization	10	10	5
Structural	10	10	7
Thermal	6	6	5
Total (%)	100%	100%	65%

Nota-se que o desempenho do Q-MCH é quantitativamente superior aos demais, melhorando o valor original da largura de banda e as soluções do algoritmo Cuthill-McKee em todas as

instâncias, além de alcançar uma melhoria de 65% em relação ao MCH. Mesmo na classe Optimization, onde o desempenho foi mais modesto, 50% das instâncias apresentaram soluções melhoradas.

Tabela 5: Comparação de tempos entre heurísticas em segundos.

Heurística	Média de Tempo(s)
Cuthill-McKee	0.03
VNS Constructive	0.16
Q-MCH	0.42
GPS	31.80
FNCHC	146.20
VNS-band	157.33

Por fim, apresentamos na Tabela 5, uma comparação entre o tempo de médio de execução do Q-MCH em relação aos algoritmos tradicionais da literatura. O Q-MCH apresentou tempo de execução inferior, com média de 0.42 segundos, frente a algoritmos como GPS, FNCHC e VNS-band, que demandam mais tempo. Isso destaca sua eficiência como heurística competitiva para reduzir a largura de banda em grafos.

6. Conclusão e Trabalhos Futuros

A pesquisa desenvolvida neste estudo foca no desenvolvimento e avaliação de uma nova abordagem através do algoritmo Q-MCH, que integra a técnica de Aprendizado por Reforço, especificamente o método de Q-Learning, em uma heurística multi-centralidades, visando potencializar o uso de medidas de centralidade no algoritmo construtivo para gerar a maior eficácia das técnicas de redução de largura de banda para sistemas lineares extensos. Este estudo alcançou os objetivos propostos ao mostrar que o Q-MCH não apenas maximiza a redução da largura de banda, mas também mantém um tempo de execução eficiente. Os resultados evidenciam sua vantagem significativa em relação a outros métodos tradicionais, consolidando-o como uma escolha promissora para aplicações que exigem otimização de grafos com eficiência computacional.

Com base nos resultados obtidos, uma direção futura de pesquisa é explorar ainda mais as capacidades do Q-Learning em combinação com outras heurísticas e métodos híbridos de otimização, a fim de desenvolver abordagens ainda mais eficientes e poderosas para a redução de largura de banda em matrizes esparsas.

Agradecimentos

Os autores agradecem a CAPES, ao CNPq e à FAPERJ pelo financiamento parcial desta pesquisa.

Referências

- Alves, C. A. S., Mendonca, I., Guimaraes, V. A., e González, P. H. (2024). Aco with reinforcement learning applied to rescues operations on urban forests. *Congresso IEEE sobre Computação Evolucionária (CEC)*, 2024:1–8.
- Arulkumaran, K., Deisenroth, M. P., Brundage, M., e Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38.
- Campos, R. M. . M. L. . F. G. . V. (2001). Reducing the bandwidth of a sparse matrix with tabu search. *European Journal of Operational Research*, 135(2):450–459. ISSN 0377-2217. URL <https://www.sciencedirect.com/science/article/pii/S0377221700003258>. Financial Modelling.
- Chagas, G. O. e Gonzaga de Oliveira, S. L. (2015). Metaheuristic-based heuristics for symmetric-matrix bandwidth reduction: A systematic review. *Procedia Computer Science*, 51:211–220. ISSN 1877-0509. URL <https://www.sciencedirect.com/science/article/pii/S1877050915010376>. International Conference On Computational Science, ICCS 2015.
- Chinn, P. Z., Chvátalová, J., Dewdney, A. K., e Gibbs, N. E. (1982). The bandwidth problem for graphs and matrices—a survey. *Journal of Graph Theory*, 6(3):223–254. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/jgt.3190060302>.
- Cuthill, E. e McKee, J. (1969). Reducing the bandwidth of sparse symmetric matrices. *Proceedings of the 1969 24th national conference*, p. 157–172.
- Davis, T. A. e Hu, Y. (2011). The University of Florida sparse matrix collection. *ACM Transactions on Mathematical Software*, 38(1):1–25.
- Garey, M. R. e Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)*. W. H. Freeman, first edition edition. ISBN 0716710455. URL <http://www.amazon.com/Computers-Intractability-NP-Completeness-Mathematical-Sciences/dp/0716710455>.
- George, J. A. (1971). *Computer implementation of the finite element method*. PhD thesis, Computer Science Department, Stanford University, CA, USA.
- Gonzaga de Oliveira, S. L. (2023). An evaluation of heuristic methods for the bandwidth reduction of large-scale graphs. *Revista Pesquisa Operacional*, 43(e268255):1–22.
- Gonzaga de Oliveira, S. L., Bernardes, J. A. B., e Chagas, G. O. (2018a). An evaluation of reordering algorithms to reduce the computational cost of the incomplete Cholesky-conjugate gradient method. *Computational & Applied Mathematics*, 37:2965–3004.
- Gonzaga de Oliveira, S. L. e Carvalho, C. (2022). Metaheuristic algorithms for the bandwidth reduction of large-scale matrices. *Journal of Combinatorial Optimization*, 43:727–784.
- Gonzaga de Oliveira, S. L. e Silva, L. M. (2020a). Evolving reordering algorithms using an ant colony hyperheuristic approach for accelerating the convergence of the ICCG method. *Engineering with Computers*, 36:1857–1873.

- Gonzaga de Oliveira, S. L. e Silva, L. M. (2021). Low-cost heuristics for matrix bandwidth reduction combined with a Hill-Climbing strategy. *Rairo - Operations Research*, 55(4):2247–2264.
- Gonzaga de Oliveira, S. L., Bernardes, J. A. B., e Chagas, G. O. (2018b). An evaluation of low-cost heuristics for matrix bandwidth and profile reductions. *Computational and Applied Mathematics*, 37(2):1412–1471. ISSN 1807-0302. URL <https://doi.org/10.1007/s40314-016-0394-9>.
- Gonzaga de Oliveira, S. e Silva, L. (2020b). An ant colony hyperheuristic approach for matrix bandwidth reduction. *Applied Soft Computing*, 94:106434. ISSN 1568-4946. URL <https://www.sciencedirect.com/science/article/pii/S1568494620303744>.
- Hettmansperger, T. P. e McKean, J. W. (2010). *Robust nonparametric statistical methods*. CRC Press.
- Kobayashi, K., Aritsugi, M., González, P. H., e Mendonca, I. (2024). Efficient hyperparameter optimization using deep q-network and brkga. *Congresso IEEE sobre Computação Evolucionária (CEC)*, 2024:1–8.
- Koohestani, B. e Poli, R. (2011). A hyper-heuristic approach to evolving algorithms for bandwidth reduction based on genetic programming. In Bramer, M., Petridis, M., e Nolle, L., editors, *Research and Development in Intelligent Systems XXVIII*, p. 93–106, London. Springer London. ISBN 978-1-4471-2318-7.
- Lam, P., Shiu, W., e Lin, Y. (1997). Duality in bandwidth problems. *Congressus Numerantium*, 124:117–127.
- Lim, A., Rodrigues, B., e Xiao, F. (2007). International journal on artificial intelligence tools 2007 lim rodrigues xiao. *International Journal of Artificial Intelligence Tools*, 16:537–544.
- Mazyavkina, N., Sviridov, S., Ivanov, S., e Burnaev, E. (2021). Reinforcement learning for combinatorial optimization: A survey. *Computers Operations Research*, 134:105400.
- Mladenovic, N., Urošević, D., Pérez-Brito, D., e García-González, C. (2010). Variable neighbourhood search for bandwidth reduction. *European Journal of Operational Research*, 200: 14–27.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., e Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.
- Pratt, J. W. (1959). Remarks on zeros and ties in the Wilcoxon signed rank procedures. *Journal of the American Statistical Association*, 54:655–667. URL <https://api.semanticscholar.org/CorpusID:120225998>.
- Souto, G., Aritsugi, M., Mendonca, I., Simonetti, L., e González, P. H. (2024). A q-learning hybrid brkga applied to the knapsack problem with forfeits. *Congresso IEEE sobre Computação Evolucionária (CEC)*, 2024:1–8.
- Torres-Jimenez, J., Izquierdo-Marquez, I., Garcia-Robledo, A., Gonzalez-Gomez, A., Bernal, J., e Kacker, R. N. (2015). A dual representation simulated annealing algorithm for the bandwidth minimization problem on graphs. *Information Sciences*, 303:33–49.