



GUILHERME OLIVEIRA CHAGAS

**UMA AVALIAÇÃO DE HEURÍSTICAS PARA
REDUÇÃO DE LARGURA DE BANDA DE
MATRIZES**

LAVRAS - MG

2015

GUILHERME OLIVEIRA CHAGAS

**UMA AVALIAÇÃO DE HEURÍSTICAS PARA REDUÇÃO DE
LARGURA DE BANDA DE MATRIZES**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, área de concentração em Inteligência Computacional e Processamento Digital de Imagens, para a obtenção do título de Mestre.

Orientador

Dr. Sanderson Lincoln Gonzaga de Oliveira

LAVRAS - MG

2015

**Ficha catalográfica elaborada pelo Sistema de Geração de Ficha
Catalográfica da Biblioteca Universitária da UFLA, com dados
informados pelo próprio autor.**

Chagas, Guilherme Oliveira.

Uma avaliação de heurísticas para redução de largura de banda
de matrizes/ Guilherme Oliveira Chagas. – Lavras : UFLA, 2015.
159 p. : il.

Dissertação (mestrado acadêmico) – Universidade Federal de
Lavras, 2015.

Orientador: Sanderson L. Gonzaga de Oliveira.

Bibliografia.

1. Redução de largura de bandao. 2. Heurísticas. 3. Método dos
Gradientes Conjugados. I. Universidade Federal de Lavras. II.
Título.

GUILHERME OLIVEIRA CHAGAS

**UMA AVALIAÇÃO DE HEURÍSTICAS PARA REDUÇÃO DE
LARGURA DE BANDA DE MATRIZES**

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, área de concentração em Inteligência Computacional e Processamento Digital de Imagens, para a obtenção do título de Mestre.

APROVADA em 17 de julho de 2015.

Dr. Alexandre Santos Francisco UFF

Dr. Mayron César de Oliveira Moreira UFLA

Dr. Sanderson Lincoln Gonzaga de Oliveira
Orientador

LAVRAS - MG

2015

À memória de meu avô, Paulo Tomé das Chagas.

AGRADECIMENTOS

Acima de tudo, agradeço às pessoas mais importantes da minha vida, os meus pais. Obrigado Zaqueu e Irene, sem vocês, jamais teria chegado até aqui. Agradeço, também, ao meu irmão Gabriel e a todos os meus familiares que, de alguma forma, me ajudaram ao longo desses anos.

Agradeço ao meu orientador e amigo, professor Sanderson, por todos os ensinamentos e conselhos dados. Ainda, agradeço a todos os professores e funcionários do DCC-UFLA, em especial, a Eunice.

Também, agradeço a todos os meus amigos que participaram desta caminhada. Em especial, aos amigos da república Riskafaka, minha segunda família; ao XisDê, pelos conselhos e produtivas conversas diárias; ao Limaum, parceiro de longa data; aos amigos Alexandre, Frederico e Júnior, por toda ajuda dada e a Jéssica, por toda ajuda e conselhos dados.

Ainda, agradeço ao professor Dr. Dragan Urosevic pelos envios dos executáveis da heurística VNS-*band* e ao professor Dr. Fei Xiao pelo envio do código da heurística FNCHC. Finalmente, agradeço os apoios financeiros da Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG).

RESUMO

Pode-se obter redução no custo computacional na resolução de sistemas de equações lineares com a redução da largura de banda das matrizes de coeficientes. O problema da redução de largura de banda de matrizes consiste em realizar permutações de linhas e colunas de uma matriz, deixando-a com uma estrutura compacta e com coeficientes não nulos próximos à diagonal principal. Identificou-se, na literatura, oito heurísticas que apresentaram os melhores benefícios (i.e. redução de largura de banda) por custos computacionais e essas heurísticas foram implementadas. Também, foi implementada a heurística GPS, que é uma das heurísticas mais clássicas nesse problema. Ainda, duas novas heurísticas são propostas neste trabalho. Simulações computacionais foram realizadas com essas 11 heurísticas em 113 instâncias de matrizes da base Harwell-Boeing e em três conjuntos de instâncias de sistemas de equações lineares oriundos de discretizações da equação da condução do calor e da equação de Laplace pelo método dos volumes finitos. Ainda, esses sistemas de equações lineares foram resolvidos pelo método dos gradientes conjugados preconditionado. Com os testes nas instâncias da base Harwell-Boeing, identificou-se que a heurística *VNS-Band* é a melhor heurística para a redução de largura de banda. Porém, essa heurística não é a mais adequada para a redução do custo computacional do método dos gradientes conjugados preconditionado em instâncias muito grandes. Especificamente, melhores resultados na redução do custo computacional da resolução de sistemas de equações lineares foram obtidos por heurísticas que não reduzem muito a largura de banda, mas que têm baixo custo computacional. Então, pode-se considerar que essas heurísticas são mais indicadas para se reduzir o custo computacional do método dos gradientes conjugados preconditionado na resolução de sistemas de equações lineares.

Palavras-chave: Redução de largura de banda. Heurísticas. Método dos Gradientes Conjugados Precondicionado. Matrizes Esparsas.

ABSTRACT

Computational cost of a linear system solver can be reduced by matrix bandwidth reduction. Bandwidth reduction consists of carrying out permutations of lines and columns so that they allow coefficients to remain near the main diagonal. By a systematic review, eight heuristics were identified with the best benefits, i.e., bandwidth reduction per computational cost, and then were implemented. In addition, the GPS heuristic, one of the most known heuristic in this problem, was implemented. Furthermore, two new heuristics are proposed in this work. Computational simulations were performed with these 11 heuristics in 113 instances of the Harwell-Boeing Sparse Matrix Collection and with three sets of instances with linear systems obtained from discretizations of the heat conduction and the Laplace equations by finite volumes. These linear systems were solved using the preconditioned Conjugate Gradient Method. According to the results presented here, the best heuristic in the simulations performed with the Harwell-Boeing Sparse Matrix Collection was the Variable neighborhood search for bandwidth reduction. However, this heuristic is not indicated to reduce the computational cost of preconditioned Conjugate Gradient Method in large-scale sparse linear systems. In particular, the better results in reducing the computational cost of solving linear systems were obtained by low-cost heuristics. Then, low-cost heuristics can be considered the best option to reduce the computational cost of the preconditioned Conjugate Gradient Method.

Keywords: Bandwidth Reduction. Heuristics. Preconditioned Conjugate Gradient Method. Sparse Matrices.

LISTA DE FIGURAS

Figura 1	Representação da matriz por meio de um grafo.	26
Figura 2	Grafo da Figura 1 com a numeração dos vértices alterada..	26
Figura 3	Pseudo-código da sub-rotina <i>Reduz_LargNível</i> da segunda etapa da heurística de Burgess e Lai (1986).	36
Figura 4	Pseudo-código da sub-rotina <i>Renumeração</i> da heurística WBRA.	42
Figura 5	Pseudo-código da sub-rotina <i>AtribuiçãoGargalo</i> da heurística WBRA.....	43
Figura 6	Pseudo-código da heurística WBRA.....	46
Figura 7	Exemplo de busca pela heurística VNS. A busca é iniciada na solução S . A solução S'_1 é encontrada em $N_k(S'')$. A solução S''_1 é melhor que a solução corrente S . Então, uma nova busca é iniciada na vizinhança $N_k(S'_1)$ e a solução S''_2 é encontrada. Com a primeira busca na vizinhança $N_k(S''_2)$, é encontrada a solução S''_3 . Como a solução S''_3 não é melhor que a solução corrente, então, o raio de busca na vizinhança $N_k(S''_2)$ é incrementado. Com isso, a solução S''_4 é encontrada dentro da vizinhança $N_{k+1}(S''_2)$	50
Figura 8	Pseudo-código do método dos gradientes conjugados.	59
Figura 9	Pseudo-código do método dos gradientes conjugados preconditionado à esquerda.	61
Figura 10	Pseudo-código do método dos gradientes conjugados preconditionado por <i>split</i>	62
Figura 11	Pseudo-código da fatoração incompleta de Cholesky com zero <i>fill-in</i>	64
Figura 12	Fluxograma do projeto computacional <i>LinearSystemSolver</i>	77
Figura 13	Gráfico das porcentagens das reduções do custo computacional do MGCPJ de todas as 10 heurísticas testadas.	107
Figura 14	Gráfico das porcentagens das reduções do custo computacional do MGCPJ sem os resultados das heurísticas BL e CSS- <i>Band</i>	108
Figura 15	Gráfico das porcentagens das reduções do custo computacional do MGCPJ das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.	108

Figura 16	Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 de todas as 10 heurísticas testadas.	112
Figura 17	Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 sem os resultados das heurísticas BL e CSS- <i>Band</i>	113
Figura 18	Gráfico das percentuais das reduções do custo computacional do MGC-IC0 das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.	113
Figura 19	Gráfico das ocupações de memória, em <i>mebibytes</i> , das heurísticas WBRA-GL, VNS- <i>Band</i> , FNCHC, GPS, GGPS e CSS- <i>Band</i> nas instâncias oriundas da discretização da equação da condução do calor.	114
Figura 20	Gráfico das ocupações de memória, em <i>mebibytes</i> , das heurísticas WBRA-GL, FNCHC, GPS e GGPS nas instâncias oriundas da discretização da equação da condução do calor.	115
Figura 21	Custos computacionais, em segundos, nas instâncias com até 232052 vértices, do MGCPJ e do MGC-IC0. Mostram-se os resultados dos resolutores MGCPJ e MGC-IC0 com as reduções de largura de banda e <i>profile</i> pela heurísticas CMr-GL.	116
Figura 22	Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 de todas as 10 heurísticas testadas.	124
Figura 23	Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 sem os resultados das heurísticas BL e CSS- <i>Band</i>	124
Figura 24	Gráfico das percentuais das reduções do custo computacional do MGC-IC0 das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.	125
Figura 25	Gráficos das ocupações de memória, em <i>mebibytes</i> , das heurísticas WBRA-GL, VNS- <i>Band</i> , FNCHC, GPS, GGPS e CSS- <i>Band</i> nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.....	126

Figura 26	Gráficos das ocupações de memória, em <i>mebibytes</i> , das heurísticas WBRA-GL, FNCHC, GPS e GGPS nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.	126
Figura 27	Gráfico dos custos computacionais, em cada instância, dos resolutores MGCPJ e MGC-IC0. Utilizaram-se os resultados da melhor combinação para cada resolutor, ou seja, utilizou-se os resultados do MGCPJ sem redução de largura de banda e utilizou-se os resultados do MGC-IC0 com redução de largura de banda pela heurística CMr-GL.	128
Figura 28	Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 de todas as 10 heurísticas testadas.	135
Figura 29	Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 sem os resultados das heurísticas BL, CSS-Band e GGPS.....	136
Figura 30	Gráfico das percentuais das reduções do custo computacional do MGC-IC0 das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.	136
Figura 31	Gráficos das ocupações de memória, em <i>mebibytes</i> , das heurísticas WBRA-GL, VNS-Band, FNCHC, GPS, GGPS e CSS-Band nas instâncias oriundas da discretização da equação de Laplace numeradas pela Curva de Sierpiński modificada.....	137
Figura 32	Gráficos das ocupações de memória, em <i>mebibytes</i> , das heurísticas WBRA-GL, FNCHC, GPS e GGPS nas instâncias oriundas da discretização da equação de Laplace numeradas pela Curva de Sierpiński modificada.	138
Figura 33	Gráfico dos custos computacionais, em cada instância, dos resolutores MGCPJ e MGC-IC0. Utilizaram-se os resultados da melhor combinação para cada resolutor, ou seja, utilizou-se os resultados do MGCPJ sem redução de largura de banda e utilizou-se os resultados do MGC-IC0 com redução de largura de banda pela heurística CMr-GL.	139

LISTA DE TABELAS

Tabela 1	Resultados de cada heurística em relação à largura de banda, ao <i>profile</i> e ao tempo de execução das simulações nas 15 instâncias simétricas muito pequenas da base Harwell-Boeing.....	87
Tabela 2	Resultados de cada heurística em relação à largura de banda e ao <i>profile</i> das simulações nas 35 instâncias simétricas pequenas da base Harwell-Boeing.	89
Tabela 3	Resultados de cada heurística em relação aos tempos médios, em segundos, de execução das simulações nas 35 instâncias simétricas pequenas da base Harwell-Boeing.	90
Tabela 4	Resultados de cada heurística em relação à largura de banda, ao <i>profile</i> e ao custo computacional das simulações nas 18 instâncias assimétricas muito pequenas da base Harwell-Boeing.....	92
Tabela 5	Resultados de cada heurística em relação a largura de banda e ao <i>profile</i> das simulações nas 45 instâncias assimétricas pequenas da base Harwell-Boeing.....	94
Tabela 6	Resultados de cada heurística em relação ao <i>profile</i> das simulações nas 45 instâncias assimétricas pequenas da base Harwell-Boeing.	95
Tabela 7	Resultados de cada heurística em relação aos tempos médios, em segundos, de execução das simulações nas 45 instâncias assimétricas pequenas da base Harwell-Boeing. ..	96
Tabela 8	Posições, de cada heurística, em relação ao número de “vitórias” obtidas nas 113 matrizes da base Harwell-Boeing. São apresentadas as respectivas posições em relação à largura de banda (β), ao <i>profile</i> e ao custo computacional (t).	97
Tabela 9	Heurísticas que obtiveram os menores somatórios de escore padronizado ($\sum \frac{x-\bar{x}}{\sigma}$), em cada conjunto de instância. São apresentados os resultados em relação à largura de banda (β) e <i>profile</i>	98
Tabela 10	Resultados referentes às execuções do MGCPJ e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da condução calor.	105
Tabela 10	Conclusão.....	106

Tabela 11	Resultados referentes às execuções do MGC-IC0 e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da condução calor.	110
Tabela 11	Conclusão.....	111
Tabela 12	Resultados referentes às execuções do MGCPJ e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.....	119
Tabela 12	Conclusão.....	120
Tabela 13	Resultados referentes às execuções do MGC-IC0 e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.	122
Tabela 13	Conclusão.....	123
Tabela 14	Resultados referentes às execuções do MGCPJ e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas pela curva de Sierpiński modificada.	130
Tabela 14	Conclusão.....	131
Tabela 15	Resultados referentes às execuções do MGC-IC0 e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas pela curva de Sierpiński modificada.....	133
Tabela 15	Conclusão.....	134
Tabela 16	Vantagens das heurísticas CMr-GL e BeL-GL.	143
Tabela 17	Desvio padrão (σ) e coeficiente de variação (C_v) dos resultados obtidos pelas heurísticas em relação aos tempos médios de execução nas 15 instâncias simétricas muito pequenas da base Harwell-Boeing.	156
Tabela 18	Desvio padrão (σ) e coeficiente de variação (C_v) dos resultados obtidos pelas heurísticas em relação aos tempos médios de execução nas 35 instâncias simétricas pequenas da base Harwell-Boeing.	157
Tabela 19	Desvio padrão (σ) e coeficiente de variação (C_v) dos resultados obtidos pelas heurísticas em relação aos tempos médios de execução nas 18 instâncias assimétricas muito pequenas da base Harwell-Boeing.....	158

Tabela 20	Desvio padrão (σ) e coeficiente de variação (C_v) dos resultados obtidos pelas heurísticas em relação aos tempos médios de execução nas 45 instâncias assimétricas pequenas da base Harwell-Boeing.....	159
-----------	--	-----

LISTA DE SIGLAS

BeL-GL	Busca em largura iniciando por um vértice pseudo-periférico pelo algoritmo de George e Liu (1979)
BL	<i>Burgess-Lai</i>
CM	<i>Cuthill-McKee</i>
CMr	<i>Cuthill-McKee reverso</i>
CMr-GL	<i>Cuthill-McKee reverso</i> iniciando por um vértice pseudo-periférico pelo algoritmo de George e Liu (1979)
CSS-band	<i>Charged Search System for bandwidth reduction</i>
EDP	Equação diferencial parcial
FNCHC	<i>Fast Node-Centroid with Hill Climbing</i>
GGPS	<i>Generalized Gibbs-Poole-Stockmeyer</i>
GPS	<i>Gibbs-Poole-Stockmeyer</i>
hGPHH	Heurística gerada pela <i>Genetic Programming Hyper-Heuristic</i>
MGC	Método dos Gradientes Conjugados
MGCPJ	Método dos Gradientes Conjugados Precondicionado por Jacobi
MGC-IC0	Método dos Gradientes Conjugados Precondicionado pela fatoração incompleta de Cholesky com zero <i>fill-in</i>
SEL	Sistema de equações lineares
UFF	Universidade Federal Fluminense
UFLA	Universidade Federal de Lavras
VNS-band	<i>Variable Neighbourhood Search for bandwidth reduction</i>
WBRA	<i>Wonder Bandwidth Reduction Algorithm</i>
WBRA-GL	<i>Wonder Bandwidth Reduction Algorithm</i> modificado iniciando por um vértice pseudo-periférico pelo algoritmo de George e Liu (1979)

SUMÁRIO

1	INTRODUÇÃO	18
1.1	Justificativa	21
1.2	Objetivos	23
2	REFERENCIAL TEÓRICO	24
2.1	Conceitos básicos	25
2.2	Heurísticas Cuthill-McKee e Cuthill-McKee reverso .	28
2.3	Heurística GPS	29
2.4	Heurística Burgess-Lai.....	31
2.4.1	Modificação da estrutura de nível enraizada.....	32
2.4.2	Renumeração dos vértices.....	35
2.5	Heurística WBRA.....	39
2.5.1	Inicialização e redução da largura de nível.....	39
2.5.2	Renumeração	40
2.5.3	Pseudocódigo da heurística WBRA.....	45
2.6	Heurística FNCHC.....	46
2.7	Heurística GGPS	47
2.8	Heurística VNS- <i>Band</i>	49
2.9	Heurística hGPHH.....	51
2.10	Heurística CSS- <i>band</i>	54
2.11	Algoritmo de George-Liu.....	55
2.12	Método dos gradientes conjugados	56
2.12.1	Método dos gradientes conjugados preconditionado ..	58
2.13	Métodos preconditionadores	63
2.14	Estruturas CSR-SSS e CSC-SSS	64
3	METODOLOGIA	67
3.1	Revisões sistemáticas	67
3.2	Heurísticas propostas	69
3.2.1	Heurística BeL-GL	69
3.2.2	Heurística WBRA-GL	70
3.3	Instâncias utilizadas	71
3.3.1	Instâncias da Harwell-Boeing	71
3.3.2	Instâncias oriundas da discretização da condução do calor	72
3.3.3	Instâncias oriundas da discretização da equação de laplace	73
3.4	Detalhes de Implementação.....	74
3.4.1	Projeto computacional.....	75

3.4.2	Instâncias assimétricas da Harwell-Boeing	78
3.4.3	Inversão de matrizes	79
3.5	Ferramentas utilizadas	82
4	SIMULAÇÕES COM INSTÂNCIAS DA BASE HARWELL-BOEING	84
4.1	Instâncias simétricas muito pequenas	85
4.2	Instâncias simétricas pequenas	87
4.3	Instâncias assimétricas muito pequenas	91
4.4	Instâncias assimétricas pequenas	93
4.5	Análise dos resultados dos testes nas instâncias da Harwell-Boeing	96
5	SIMULAÇÕES COM RESOLUÇÃO DE SISTEMAS DE EQUAÇÕES LINEARES	99
5.1	Estruturas das Tabelas dos Resultados das Simulações com o MGCPJ	100
5.2	Estruturas das Tabelas dos Resultados das Simulações com o MGC-IC0	101
5.3	Simulações Com instâncias oriundas da discretização da condução do calor	103
5.3.1	Simulações com o método dos gradientes conjugados precondicionado pelo método de Jacobi	103
5.3.2	Simulações com o método dos gradientes conjugados precondicionado pelo método incompleto de Cholesky com zero <i>fill-in</i>	108
5.3.3	Ocupações de memória	113
5.3.4	Comparação dos resultados dos resolutores MGCPJ e MGC-IC0	115
5.4	Simulações com instâncias oriundas da discretização equação de Laplace	117
5.4.1	Instâncias numeradas de forma aleatória	117
5.4.1.1	Simulações com o método dos gradientes conjugados precondicionado pelo método de Jacobi	118
5.4.1.2	Simulações com o método dos gradientes conjugados precondicionado pelo método incompleto de Cholesky com zero <i>fill-in</i>	120
5.4.1.3	Ocupações de memória	124
5.4.1.4	Comparação dos resultados dos resolutores MGCPJ e MGC-IC0	127

5.4.2	Instâncias numeradas pela curva de sierpiński modificada	128
5.4.2.1	Simulações com o método dos gradientes conjugados preconditionado pelo método Jacobi.....	129
5.4.2.2	Simulações Com o método dos gradientes conjugados preconditionado pelo método incompleto de Cholesky com zero <i>fill-in</i>	132
5.4.2.3	Ocupações de memória	136
5.4.2.4	Comparação dos resultados dos resolutores MGCPJ e MGC-IC0	138
6	CONCLUSÃO E TRABALHOS FUTUROS.....	140
6.1	Considerações sobre as simulações na base Harwell-Boeing	141
6.2	Considerações sobre os testes com resolução de SELs pelo MGCPJ.....	142
6.3	Considerações sobre os testes com resolução de SELs pelo MGC-IC0.....	144
6.4	Considerações sobre as heurísticas propostas: BeL-GL e WBRA-GL	145
6.5	Trabalhos futuros	146
	REFERÊNCIAS.....	148
	APÊNDICE.....	156

1 INTRODUÇÃO

Nas últimas décadas, o avanço da tecnologia em termos de *hardware* permitiu que simulações computacionais atingissem níveis satisfatórios de precisão. Essas simulações são realizadas por meio da modelagem dos fenômenos físicos por equações diferenciais parciais (EDPs). Alguns exemplos de fenômenos físicos e problemas modelados por EDPs podem ser encontrados na indústria de petróleo, em simulações aeroespaciais e estruturais, em previsões meteorológicas e em estudos de poluição de oceanos.

Para se discretizar fenômenos físicos modelados por EDPs, utilizam-se métodos numéricos, como, por exemplo, o método dos volumes finitos. Sistemas de equações lineares (SEs) são gerados nas aplicações desses métodos. Na maioria dos casos, as matrizes desses SEs são de grande porte e esparsas. “Por serem de grande porte, necessitam de muita memória e de alto custo de processamento para armazenar e resolver esses SEs” (OLIVEIRA; CHAGAS, 2014, p. 1). Por exemplo, segundo Kaveh (2004, p. 221), “de 30% a 50% do custo computacional total exigido nas simulações são relacionados às resoluções de SEs”. Entre alguns exemplos práticos os quais têm-se a necessidade de uma resolução eficiente de SEs, podemos citar: sistemas de redes de energia elétrica, processos de engenharia química, modelos econômicos, sistemas de produção industrial e análise de circuitos digitais (BENZI, 2002).

Métodos diretos ou iterativos são utilizados para a resolução de SEs. Embora os primeiros sejam mais confiáveis, seu custo computacional é, geralmente, maior que o custo computacional exigido por um método iterativo. Por isso, utilizam-se métodos iterativos para a resolução de SEs

de grande porte. Ainda, Benzi (2002) explica que, após diversas ideias e técnicas da área de métodos diretos serem transferidas, na forma de preconditionadores para métodos iterativos, os métodos iterativos passaram a ser cada vez mais confiáveis.

Um dos métodos iterativos mais populares é o método dos gradientes conjugados (MGC) (HESTENES; STIEFEL, 1952; LANCZOS, 1952). Esse método é baseado no subespaço de Krylov (1931). Segundo Saad (2003, p. 187), “o MGC é um dos melhores métodos iterativos para a resolução de SELs em que as matrizes de coeficientes são simétricas, esparsas e positivas-definidas”. Neste trabalho, as matrizes das instâncias de SELs, utilizadas nas simulações, são simétricas, esparsas e positivas-definidas.

Embora o MGC seja um resolutor de SELs rápido, há situações em que a convergência desse método é lenta. Isso pode ser evitado por meio do pré-processamento das matrizes de coeficientes dos SELs, denominado de condicionamento. Alguns métodos diretos e iterativos para a resolução de SELs são utilizados, de forma parcial, como preconditionadores. O capítulo 10 de Saad (2003) é dedicado à descrição de alguns destes métodos.

Para matrizes utilizadas neste trabalho, um método preconditionador indicado é a fatoração incompleta de Cholesky. Ainda, essas matrizes de coeficientes são diagonalmente dominantes. Uma matriz $A = [a_{ij}]$ é diagonalmente dominante se $a_{ii} \geq \sum_{j \neq i} |a_{ij}|$, para $1 \leq i \leq n$. Para esse tipo de matrizes, o método de Jacobi (1845) é um bom preconditionador.

“Pode-se obter baixo custo computacional na resolução do sistema de equações lineares pelo MGC com a *redução de largura de banda* da matriz de coeficientes”(OLIVEIRA; CHAGAS, 2014, p. 94). Seja A uma matriz

simétrica. A largura de banda $\beta(A)$ é a maior distância de um elemento não nulo da matriz triangular inferior até a diagonal principal. O problema da *minimização de largura de banda* de uma matriz simétrica A é aproximar, ao máximo possível, todos os coeficientes não nulos da diagonal principal de A . Esse é um problema NP-difícil, conforme foi demonstrado por Papadimitriou (1976).

Duff e Meurant (1989) compararam várias reordenações de coeficientes de matrizes de SELs para a redução do custo computacional do MGC preconditionado. Esses autores verificaram que as reordenações em que os coeficientes ficam próximos uns dos outros, forneceram os melhores resultados. Esse foi o caso das reordenações geradas pelas heurísticas de Cuthill e McKee (1969) e o Cuthill-McKee reverso (CMr) de George (1971).

A redução do custo computacional do MGC por heurísticas para a redução de largura de banda se deve à arquitetura atual de hierarquia de memória e às políticas de paginação. Isso porque a redução de largura de banda proporciona mais localidade em memória e coerência de cache, ou seja, memórias recentemente referenciadas são agrupadas em vez de serem dispostas aleatoriamente (OLIVEIRA; CHAGAS, 2015a). Isso foi verificado por Temam e Jalby (1992) e Burgess e Giles (1997).

Muitas heurísticas para o problema da redução de largura de banda têm sido propostas desde a década de 1960 (CHAGAS; OLIVEIRA, 2015). Nas revisões sistemáticas realizadas por Oliveira e Chagas (2015a) e Chagas e Oliveira (2015), foram identificadas 73 heurísticas para a redução de largura de banda. Com isso, não é uma tarefa trivial identificar qual é a melhor heurística para a redução de largura de banda de matrizes. Ainda, algumas heurísticas reduzem expressivamente a largura de banda;

porém, exigem um grande custo computacional. Então, possivelmente os ganhos obtidos na resolução do sistema de equações lineares com a redução da largura de banda da matriz de coeficientes, não compensem o custo computacional exigido para se realizar a redução de largura de banda. Por isso, há a necessidade de se identificar a heurística que reduz expressivamente a largura de banda com um custo computacional razoável.

Este trabalho está dividido como a seguir. Nas seções 1.1 e 1.2, são apresentados, respectivamente, a justificativa e o objetivo deste trabalho. No Capítulo 2 é apresentado o referencial teórico. Nesse Capítulo, são descritos os conceitos básicos para o entendimento deste texto, o método dos gradientes conjugados e as heurísticas e métodos implementados. Abordam-se, no Capítulo 3, a metodologia de pesquisa e desenvolvimento adotados neste trabalho. Na metodologia, são apresentados os detalhes de implementação, as ferramentas e os métodos utilizados. Os resultados e as análises das simulações com as instâncias de matrizes da Harwell-Boeing são mostrados no Capítulo 4. As análises e os resultados das simulações com resolução de SELs são apresentadas no Capítulo 5. Finalmente, no Capítulo 6, apresentam-se as conclusões sobre este trabalho e as proposições de trabalhos futuros.

1.1 Justificativa

Como descrito, há um grande custo computacional envolvido nas resoluções de SELs, por métodos diretos ou por métodos iterativos, tornando-se o verdadeiro gargalo em simulações computacionais. Com a redução de largura de banda de matrizes, pode-se obter redução no custo computacional na resolução de SELs. Segundo Tarjan (1975), o custo da

eliminação gaussiana, um resolutor direto, é $O(n \cdot \beta^2)$, em que n é a ordem da matriz de coeficientes e β é a largura de banda dessa matriz. Se $\beta \cong n$, então serão realizadas $O(n^3)$ operações. Claramente, se $\beta \ll n$, o custo do método será bem menor que $O(n^3)$. Ainda, com a redução de largura de banda, pode-se reduzir o custo de armazenamento da matriz, se a esta é armazenada por *Skyline* (JENNINGS, 1966).

“O MGC é um dos melhores métodos para a resolução de SELs em que a matriz de coeficientes é simétrica e definida-positiva”(SAAD, 2003, p. 187). Também, o MGC preconditionado é um dos resolutores mais utilizados. Pesquisas atuais utilizam esse método em comparações com outros resolutores e na resolução de SELs provenientes de vários problemas da ciência. Como exemplos, têm-se os trabalhos de Abbasbandy, Jafarin e Ezzati (2005), Verschoor e Jalba (2012), Oliveira e Chagas (2015b) e Oliveira, Oliveira e Chagas (2015).

Com a redução de largura de banda é possível gerar reordenações em que haja mais coerência de *cache* e, conseqüentemente, menor custo computacional. Porém, existem muitas heurísticas para a redução de largura de banda na literatura e não é uma tarefa trivial encontrar qual a melhor heurística.

Dada a abrangência de estudos que utilizam o MGC preconditionado e as demandas existentes neste íterim, relativas à redução de largura de banda e à busca pela heurística que melhor execute essa tarefa. Justificamos, assim, a relevância deste trabalho.

1.2 Objetivos

O objetivo geral é avaliar as heurísticas para a redução de largura de banda de matrizes para a redução do custo computacional do MGC preconditionado. Para isso, pretende-se identificar a melhor heurística para a redução de largura de banda com custo computacional razoável. Com essa análise, será possível verificar se o custo de se reduzir expressivamente a largura de banda, como é o caso da heurística *VNS-band* de Mladenovic et al. (2010), é compensado pela redução do custo computacional para a resolução do SELs. Os objetivos deste trabalho são:

1. Identificar a melhor heurística para a redução de largura de banda com custo computacional razoável.
2. Identificar a heurística para a redução de largura de banda mais adequada para a redução do custo computacional do MGC preconditionado pelo método de Jacobi.
3. Identificar a heurística para a redução de largura de banda mais adequada para a redução do custo computacional do MGC preconditionado pela fatoração incompleta de Cholesky com zero *fill-in*.
4. Identificar qual o menor custo computacional: o MGC preconditionado pelo método de Jacobi com a heurística identificada em 2 ou o MGC preconditionado pela fatoração incompleta de Cholesky com zero *fill-in* com a heurística identificada em 3.

2 REFERENCIAL TEÓRICO

Neste Capítulo, é apresentado o conteúdo necessário para o entendimento deste trabalho. Primeiramente, na seção 2.1, são apresentadas definições sobre grafos e matrizes de adjacências utilizadas neste texto.

Apresentam-se, na seção 2.2, as heurísticas Cuthill-McKee (CM) (CUTHILL; MCKEE, 1969), Cuthill-McKee reverso (CMr) (GEORGE, 1971) e Cuthill-McKee reverso com vértice pseudo-periférico pelo algoritmo de George e Liu (1979) (CMr-GL). Aborda-se, brevemente, a heurística de Gibbs, Poole e Stockmeyer (1976) (GPS) na seção 2.3. É apresentada na seção 2.4 a heurística de Burgess e Lai (1986) (BL). A heurística *Wonder Bandwidth Reduction Algorithm* (WBRA) (ESPOSITO et al., 1998) é mostrada na seção 2.5. Aborda-se, na seção 2.6, a heurística *Fast Node Centroid with Hill-Climbing* (FNCHC) de Lim, Rodrigues e Xiao (2007).

Na seção 2.7, a heurística *Generalized GPS* (GGPS), de Wang, Guo e Shi (2009), é descrita. Na seção 2.8, descreve-se a heurística *Variable Neighbourhood Search for bandwidth reduction* (VNS-band) de Mladenovic et al. (2010). Apresenta-se, na seção 2.9, a heurística gerada pela hiper-heurística por programação genética (hGPHH) de Koohestani e Poli (2011). A heurística *Charged Search System for bandwidth reduction* (CSS-band) de Kaveh e Sharafi (2012) é mostrada na seção 2.10.

O algoritmo para se encontrar um vértice pseudo-periférico de George e Liu (1979) é descrito na seção 2.11. Mostra-se, na seção 2.12, o método dos gradientes conjugados e o método dos gradientes conjugados condicionado. O condicionador de Jacobi e a fatoração incompleta de Cholesky com zero *fill-in* são detalhadas na seção 2.13. As estruturas de armazenamento de matrizes simétricas esparsas CSC-SSS e CSR-SSS são

apresentadas na seção 2.14.

2.1 Conceitos básicos

Para o entendimento deste texto, considere que os sistemas de equações lineares aqui abordados são da forma $Ax = b$. A matriz de coeficientes $A = [a_{ij}]$ é uma matriz $n \times n$ esparsa, simétrica e positiva-definida, o vetor x é o vetor de incógnitas e o vetor b é o vetor de termos independentes. Ainda, considere o particionamento $A = L + U$, em que L é a matriz triangular inferior de A e U é a matriz triangular superior de A .

Pode-se considerar a matriz de coeficientes A de um sistema de equações lineares como uma matriz de adjacências. Com isso, pode-se obter permutações de linha e colunas dessa matriz pela renumeração dos vértices do grafo correspondente a essa matriz de adjacências. Para isso, considere que $G = (V, E)$, um grafo não orientado e não direcionado, em que V é um conjunto finito de vértices e E é um conjunto finito de arestas.

Como exemplo, na Figura 1, tem-se a representação de um grafo por meio de uma matriz de adjacências. Cada linha da matriz A corresponde a um vértice no grafo da Figura 1, ou seja, para $1 \leq i \leq 6$, a linha i da matriz corresponde ao vértice i do grafo.

Para o grafo da Figura 2, utilizou-se uma ordem diferente da anterior para numerar os vértices do grafo e obteve-se uma representação matricial também diferente. Nota-se que a disposição dos coeficientes não nulos na representação matricial do grafo depende da ordem em que se numeram os vértices do grafo para a montagem da matriz. É da ordem dessa numeração de que se trata este texto. A seguir, têm-se as seguintes definições (OLIVEIRA; CHAGAS, 2014, p. 7-9):

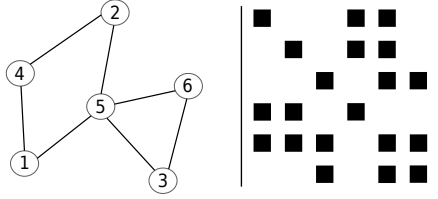


Figura 1 Representação da matriz por meio de um grafo.

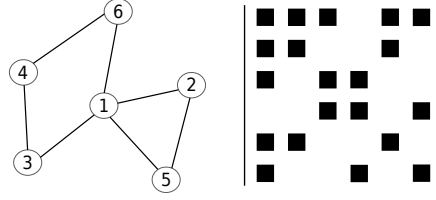


Figura 2 Grafo da Figura 1 com a numeração dos vértices alterada.

Definição 1 (largura de banda). *Seja A uma matriz simétrica $n \times n$, com coeficientes a_{ij} e $1 \leq i, j \leq n$. A largura de banda da i -ésima linha é $\beta_i(A) = i - \min(\{1 \leq j < i \mid a_{ij} \neq 0\})$. A largura de banda $\beta(A)$ é a maior distância de coeficiente não nulo da matriz triangular inferior até a diagonal principal, considerando-se todas as n linhas da matriz, ou seja, $\beta(A) = \max(\{1 \leq i \leq n\} \beta_i(A))$.*

Para uma matriz A assimétrica, a largura de banda da i -ésima linha é $\beta_i(A) = \max(\{|i - j| \mid a_{ij} \neq 0\})$, para $1 \leq i, j \leq n$. A largura de banda de A é $\beta(A) = \max(\beta_i(A))$, $1 \leq i \leq n$.

Definição 2 (profile). *O profile de A , uma matriz simétrica, é definido como $profile = \sum_{i=1}^n \beta_i(A)$.*

Seja A uma matriz assimétrica e L e U sejam as partições de A em matrizes triangulares inferior e superior, respectivamente. O profile de A é definido como $profile = \sum_{i=1}^n (\beta_i(L) + \beta_i(U))$.

Definição 3 (distância). *A distância $d(v, u)$ entre dois vértices v e $u \in V$ é o tamanho do menor caminho entre eles.*

Definição 4 (excentricidade). *A excentricidade $\ell : V \rightarrow \mathbb{N}$ de um vértice $v \in V$ é dada por $\ell(v) = \max(\{\forall u \in V\} d(v, u))$.*

Definição 5 (diâmetro do grafo). O diâmetro $\Phi(G) = \max((\forall v \in V) \ell(v)) = \max((\forall v, u \in V) d(v, u))$ do grafo $G = (V, E)$ é a maior excentricidade encontrada em $G = (V, E)$

Definição 6 (vértice periférico). Um vértice v é considerado periférico se a sua excentricidade é igual ao diâmetro do grafo, ou seja, $\ell(v) = \Phi(G)$.

Para Gibbs, Poole e Stockmeyer (1976), um vértice é considerado *pseudo-periférico* se sua excentricidade for próxima ao diâmetro do grafo.

Definição 7 (estrutura de nível enraizada). Seja $G = (V, E)$ conexo e simples. Dado um vértice $v \in V$, a estrutura de nível com raiz v e profundidade $\ell(v)$ é o particionamento $\mathcal{L}(v) = \{L_0(v), L_1(v), \dots, L_{\ell(v)}(v)\}$, em que $L_0(v) = \{v\}$ e $L_i(v) = Adj(L_{i-1}(v)) - \bigcup_{j=0}^{i-1} L_j(v)$, para $i = 1, 2, 3, \dots, \ell(v)$ e $Adj(\cdot)$ retorna os vértices adjacentes aos vértices do argumento.

Definição 8 (estrutura de nível). Seja $G = (V, E)$ conexo e simples. A estrutura de nível é o particionamento $\mathcal{K}(v, \dots) = \{K_0(v, \dots), K_1(v, \dots), \dots, K_{\ell(v)}(v, \dots)\}$, em que $v \in K_0(v, \dots)$ e $K_i(v, \dots) = Adj(K_{i-1}(v, \dots)) - \bigcup_{j=0}^{i-1} K_j(v, \dots)$, para $i = 1, 2, 3, \dots, \ell(v)$.

Definição 9 (largura de nível). A largura de nível $b(\mathcal{L}(u))$ é o número de vértices do nível com mais vértices, ou seja, $b(\mathcal{L}(u)) = \max((0 \leq i \leq \ell(u)) |L_i(u)|)$.

Segundo Cuthill e McKee (1969), se a renumeração dos vértices de $G = (V, E)$ é realizada nível a nível de $\mathcal{K}(v, \dots)$, então, $\beta(A) \leq 2b(\mathcal{K}(v, \dots)) - 1$. Ainda, se a estrutura de nível

é enraizada, a largura de banda de A não pode ser menor que a largura de nível da estrutura de nível enraizada, ou seja, $b(\mathcal{L}(v)) \leq \beta(A) \leq 2b(\mathcal{L}(v)) - 1$.

2.2 Heurísticas Cuthill-McKee e Cuthill-McKee reverso

As heurísticas Cuthill-McKee (CM) e Cuthill-McKee reverso (CMr) foram propostas por Cuthill e McKee (1969) e George (1971), respectivamente. Essas heurísticas podem ser consideradas como duas das heurísticas mais conhecidas na literatura. Essas heurísticas podem ser encontradas implementadas em *softwares* e bibliotecas, como, por exemplo, MATLAB e *Boost c++ Libraries*.

As heurísticas CM e CMr são semelhantes, com a exceção que, na heurística CMr, a renumeração final dos vértices do grafo é invertida. George (1971) constatou que inverter a renumeração obtida pela heurística CM resulta em um *profile* pelo menos tão bom quanto o *profile* sem inversão da renumeração. Além disso, a largura de banda não é alterada. Essa constatação foi provada posteriormente por Liu e Sherman (1976) e Liu (1976).

As heurísticas CM e CMr originais podem ser divididas em três etapas. Na primeira etapa, são escolhidos os vértices iniciais com base no número de adjacências. Assim, vértices com menor grau tem maior prioridade. Na segunda etapa, são geradas as estruturas de nível enraizadas de todos os vértices selecionados na primeira etapa. Aquela(s) estrutura(s) que apresentar(em) a menor largura de banda é(são) armazenada(s). Finalmente, realiza-se a renumeração dos vértices de cada estrutura de nível enraizada selecionada na segunda etapa. A renumeração é realizada nível a nível da estrutura de nível. Para o primeiro nível, como é uma estrutura

de nível enraizada, há somente um vértice, o vértice inicial. Então, esse vértice é o primeiro a ser renumerado. Posteriormente, a renumeração é realizada como a busca em largura, com a diferença que os vértices “visitados” (renumerados) são em ordem crescente de grau. Após todos os vértices de todas as estruturas de nível enraizadas armazenadas forem renumerados, a renumeração que resultar na menor largura de banda, será a renumeração final dos vértices do grafo.

Gibbs, Poole e Stockmeyer (1976) verificaram que as heurísticas CM e CMr realizavam um esforço computacional desnecessário gerando várias estruturas de nível. Com isso, esses autores propuseram a heurística GPS, descrita na subseção 2.3. Além disso, Gibbs, Poole e Stockmeyer (1976) popularizaram o uso de um vértice pseudo-periférico como vértice inicial de heurísticas, baseadas em estrutura de nível, para a redução de largura de banda. Então, as heurísticas CM e CMr passaram a realizar a renumeração dos vértices de um grafo iniciando-se de um vértice pseudo-periférico. Posteriormente, George e Liu (1981) disponibilizaram o código da heurística CMr com vértice pseudo-periférico pelo algoritmo de George e Liu (1979). Essa versão da heurística, denominada neste trabalho como CMr-GL, é a heurística que se encontra implementada no MATLAB. Para mais detalhes sobre as heurísticas CM e CMr, veja Oliveira e Chagas (2014) a partir da página 14 até a 18.

2.3 Heurística GPS

A heurística de Gibbs, Poole e Stockmeyer (1976) (GPS) foi proposta, inicialmente, como uma melhoria nas heurísticas CM e CMr. Gibbs, Poole e Stockmeyer (1976) verificaram que o custo computacional

das heurísticas CM e CMr era demasiadamente alto. Isso porque, nessas heurísticas, originalmente, selecionavam-se todos os vértice de grau mínimo e geravam-se as estruturas de nível enraizadas de todos esses vértices. Em seguida, era realizada a renumeração dos vértices da estrutura de nível enraizada que apresentasse a menor largura de nível. Se houvesse mais de uma estrutura de nível com a menor largura de nível, então, renumeravam-se os vértices de todas essas estruturas de nível enraizadas e a renumeração que resultasse na menor largura de banda era a renumeração final.

Divide-se a heurística GPS em três etapas. Nessa heurística, em vez de se construírem várias estruturas de nível enraizadas, como nas heurísticas CM e CMr, constroem-se apenas duas estruturas de nível enraizadas de dois vértices pseudo-periféricos. Gibbs, Poole e Stockmeyer (1976) foram os primeiros autores a utilizarem um vértice pseudo-periférico como vértice inicial para a renumeração. Um vértice pseudo-periférico é um bom vértice inicial para a renumeração. Isso porque na estrutura de nível enraizada desse vértice haverá mais níveis e, conseqüentemente, menor largura de nível do que um vértice que não esteja na periferia ou próximo da periferia do grafo. Para definição de vértice pseudo-periférico, veja a definição 6, na página 27. Então, na primeira etapa da heurística GPS, são selecionados dois vértices pseudo-periféricos, v e $u \in V$, e as suas estruturas de nível enraizadas, $\mathcal{L}(v)$ e $\mathcal{L}(u)$, são geradas.

Na segunda etapa da heurística GPS, cria-se uma nova estrutura de nível, $\mathcal{H}(v, \dots)$, a partir das estruturas de nível enraizadas, $\mathcal{L}(v)$ e $\mathcal{L}(u)$, geradas na primeira etapa. As estruturas de nível enraizadas são combinadas com o objetivo de se reduzir a largura de nível. Com esta etapa, garante-se que a largura de nível da estrutura de nível $\mathcal{H}(v, \dots)$ seja, no máximo, a

menor largura de nível das estruturas de nível enraizadas $\mathcal{L}(v)$ e $\mathcal{L}(u)$.

Finalmente, na última etapa, realiza-se a renumeração dos vértices percorrendo-se a estrutura de nível $\mathcal{K}(v, \dots)$, gerada na segunda etapa da heurística GPS. Os níveis são percorridos na ordem inversa, ou seja, iniciando-se pelo nível $K_{\ell(v)}(v, \dots)$ até o nível $K_0(v, \dots)$. Essa renumeração é realizada nível a nível da estrutura de nível, em que a renumeração dos vértices é realizada em ordem crescente de grau. Para mais detalhes da heurística GPS, veja Oliveira e Chagas (2014), a partir da página 18 até a 27.

2.4 Heurística Burgess-Lai

A heurística para a redução de largura de banda de matrizes proposta por Burgess e Lai (1986) (BL) é inspirada nas heurísticas CM e GPS. Como a heurística GPS, a heurística BL é, também, dividida em três etapas. Na primeira etapa, um vértice pseudo-periférico é encontrado. Burgess e Lai (1986) utilizaram o mesmo algoritmo para encontrar um vértice pseudo-periférico de Gibbs, Poole e Stockmeyer (1976). A segunda etapa consiste na modificação da estrutura de nível enraizada gerada do vértice encontrado na primeira etapa. Na terceira etapa, é realizada a renumeração dos vértices do grafo. Nesta seção, apresenta-se uma descrição da heurística BL. Essa descrição é baseada no artigo de Burgess e Lai (1986).

Burgess e Lai (1986) utilizaram uma definição diferente das definições 7 e 8, apresentadas na subseção 2.1, para a estrutura de nível enraizada e a estrutura de nível. Na descrição da heurística BL, esses autores consideram que, na estrutura de nível (enraizada ou não), também são armazenadas as conectividades entre os vértices. Essa estrutura é

semelhante à estrutura *Shortest Route Tree* (SRT) (KAVEH, 1986), que é detalhada em Oliveira e Chagas (2014), a partir da página 32. Porém, na SRT são armazenadas somente as conectividades de vértices entre níveis diferentes. Então, por simplicidade, para a descrição da heurística BL, neste trabalho, considera-se que são armazenadas, nas estruturas de nível enraizadas e estruturas de nível, as conectividades entres os vértices.

A seguir, na subseção 2.4.1, é descrita a segunda etapa da heurística BL. Descreve-se, na subseção 2.4.2, a etapa de renumeração da heurística BL.

2.4.1 Modificação da estrutura de nível enraizada

Na etapa de modificação da estrutura de nível enraizada, a estrutura de nível enraizada do vértice pseudo-periférico encontrado na primeira etapa é alterada para se reduzir a largura de nível. Isso é realizado para que a renumeração realizada na terceira etapa obtenha uma redução maior na largura de banda.

Para o entendimento desta etapa, considere as seguintes definições utilizadas por Burgess e Lai (1986). Seja $v \in V$ um vértice pseudo-periférico selecionado na primeira etapa da heurística BL, $\mathcal{L}(v)$ sua estrutura de nível enraizada e u e $w \in V$. Então,

- uma aresta $\{u, w\} \in E$ é considerada uma *aresta de entrada* do vértice w se $u \in L_{i-1}(v)$ e $w \in L_i(v)$, para $0 < i \leq \ell(v) + 1$;
- uma aresta $\{u, w\} \in E$ é considerada uma *aresta de saída* do vértice u se $u \in L_i(v)$ e $w \in L_{i+1}(v)$, para $0 \leq i < \ell(v) + 1$;
- se uma aresta $\{u, w\} \in E$, com $u \in L_i(v)$ e $w \in L_{i+1}(v)$, é uma aresta de

entrada do vértice w e é a única aresta de saída do vértice u , então, $\{u, w\} \in E$ é *aresta móvel de entrada* do vértice w ;

- um vértice u é considerado um *vértice móvel* se $Adj(G, u) \in L_{i-1}(v) \cup L_i(v)$ e $u \in L_i(v)$, para $0 < i \leq \ell(v) + 1$;
- um nível de uma estrutura de nível é *reduzível* se todos os vértices pertencentes a esse nível são vértice móveis;
- uma aresta $\{u, w\} \in E$ é *removível* se u e $w \in L_i(v)$ para $0 \leq i \leq \ell(v) + 1$.

Nesta etapa, objetiva-se reduzir a largura de nível de cada nível reduzível de $\mathcal{L}(v)$. Para isso, tenta-se trocar o nível de todos os vértices móveis de cada nível reduzível deslocando-os para um nível anterior. Mais especificamente, seja $L_i(v) \in \mathcal{L}(v)$, para $0 < i \leq \ell(v) + 1$, um nível reduzível e $u \in L_i(v)$ um vértice móvel. O vértice u será movido do nível $L_i(v)$ para o nível $L_{i-1}(v)$. Porém, pode ser que nem sempre seja uma boa ideia trocar um vértice de nível. Burgess e Lai (1986) utilizaram um valor limiar para se verificar se o número de vértices em um determinado nível é satisfatório ou não. Esses autores denominaram esse valor de *número residual*, dado por $\lceil |V|/\ell(v) + 1 \rceil$, ou seja, a função teto da razão do número de vértices pelo número de níveis da estrutura de nível em questão. Com isso, não são alterados os níveis reduzíveis com números de vértices menores ou iguais ao número residual.

Burgess e Lai (1986) constataram, empiricamente, que utilizar apenas um valor residual para se alterar a estrutura de nível nem sempre era uma boa alternativa. Então, Burgess e Lai (1986) utilizaram, também, os valores $\lceil |V|/\ell(v) + 1 \rceil - 1$ e $\lceil |V|/\ell(v) + 1 \rceil + 1$. Com isso, são geradas três estruturas de nível enraizadas no vértice pseudo-periférico encontrado na

primeira etapa. Cada estrutura de nível enraizada é alterada com um valor residual ($\lceil |V|/\ell(v) + 1 \rceil - 1$, $\lceil |V|/\ell(v) + 1 \rceil$ e $\lceil |V|/\ell(v) + 1 \rceil + 1$). A estrutura de nível que será utilizada na etapa seguinte da heurística BL é aquela que apresentar a menor largura de nível. Em caso de empate na largura de nível, seleciona-se aquela estrutura de nível que possuir o menor número de níveis com a largura de nível máxima.

Apresenta-se um pseudo-código da etapa de redução da largura de nível da heurística BL na Figura 3. O algoritmo recebe um grafo $G = (V, E)$ e o vértice pseudo-periférico v . Na estrutura de repetição *para* das linhas 2 a 19, geram-se as três estruturas de nível enraizadas para que possam ser alteradas cada uma com um número residual. A estrutura de repetição *enquanto*, das linhas 6 a 16, é realizada para cada nível reduzível da estrutura de nível, em que o número de vértices é maior que o número residual. Nessa estrutura, é selecionado um vértice móvel que será colocado no nível anterior da estrutura de nível. A sub-rotina *EMaxNumArestaMEntrada*, da linha 7, retorna, em um vetor, o(s) vértice(s) que possuir(em) o maior número de arestas móveis de entrada. Se houver mais de um vértice, então, a estrutura condicional *se* das linhas 8 a 13 tratará essa exceção. Nessa estrutura condicional, será(ão) selecionado(s) o(s) vértice(s) que possuir(em) o menor número de arestas removíveis e esse(s) vértice(s) permanecerá(ão) no vetor *verticesMoveis*. Caso ocorra de o vetor *verticesMoveis* possuir mais de um vértice, então, será selecionado o vértice com o maior número de adjacências. Se ainda assim houver mais de um vértice, o algoritmo escolherá o primeiro vértice do vetor *verticesMoveis*, como mostrado na linha 14. Na linha 15 o vértice selecionado é movido para o nível anterior da estrutura de nível. Posteriormente, armazena-se, na sub-rotina da linha 18, a estrutura de

nível corrente que possuir a menor largura de nível. Caso haja mais de uma, armazena-se a estrutura de nível que possuir menos níveis com a maior largura de nível. O algoritmo retorna a melhor estrutura de nível encontrada.

2.4.2 Renumeração dos vértices

A renumeração dos vértices pela heurística BL é realizada a cada nível da estrutura de nível. Para isso, utiliza-se a estrutura $\mathcal{K}(v, \dots)$ proveniente da segunda etapa da heurística BL, apresentada na subseção 2.4.1. Note que, a estrutura de nível alterada pela etapa descrita na subseção 2.4.1 não é necessariamente enraizada.

Para o entendimento desta etapa, outras definições, além das definições apresentadas na subseção 2.4.1, são necessárias. Considere, então, que:

- uma *aresta de saída falsa* $\{u, w\}$, em que $u \in K_i(v, \dots)$ e $w \in K_{i+1}(v, \dots)$, para $0 \leq i \leq \ell(v) + 1$, é uma aresta de saída em que o vértice w é adjacente a um ou mais vértices de $K_i(v, \dots)$, que já foram renumerados;
- o *número máximo de entrada* de um vértice $u \in K_i(v, \dots)$, para $0 < i \leq \ell(v) + 1$, é a maior numeração de um vértice adjacente ao vértice u e pertencente ao nível $K_{i-1}(v, \dots)$;
- o *número mínimo de entrada* de um vértice $u \in K_i(v, \dots)$, para $0 < i \leq \ell(v) + 1$, é a menor numeração de um vértice adjacente ao vértice u e pertencente ao nível $K_{i-1}(v, \dots)$.

Algoritmo: Reduz_LargNível.

```

1 início
2   para (  $i \leftarrow -1; i < 2; i \leftarrow i + 1$  ) faça
3      $\mathcal{L}(v) \leftarrow \text{BuscaEmLargura}(v)$ ;
4      $\text{numeroResidual} \leftarrow \lceil |V|/\ell(v) + i \rceil$ ;
5     para cada ( nível reduzível  $L_j(v) \in \mathcal{L}(v)$  ) faça
6       enquanto (  $|L_j(v)| > \text{numeroResidual}$  ) faça
7         // encontra-se vértice(s) com maior
8         // número de arestas móveis de entrada
9          $\text{verticesMoveis} \leftarrow \text{EMaxNumArestaMEntrada}(v)$ ;
10        se (  $|\text{verticesMoveis}| > 1$  ) então
11          // encontra-se vértice(s) com menor
12          // número de arestas removíveis
13           $\text{EMinNumArestaRem}(\text{verticesMoveis})$ ;
14          se (  $|\text{verticesMoveis}| > 1$  ) então
15            // vértice(s) com maior grau
16             $\text{EMaxGrau}(\text{verticesMoveis})$ ;
17          fim-se;
18        fim-se;
19         $u \leftarrow \text{verticesMoveis}[1]$ ;
20        // move o vértice para o nível anterior
21         $\text{moveVerticeNivel}(u)$ ;
22      fim-enquanto;
23    fim-para-cada;
24    // seleciona e armazena a melhor  $\mathcal{L}$  de acordo
25    // com a descrição da heurística BL
26     $\text{melhorEstrutura} \leftarrow \text{verificaMelhorEstrutura}(\mathcal{L}(v))$ ;
27  fim-para;
28  retorna  $\text{melhorEstrutura}$ ;
29 fim.
```

Figura 3 Pseudo-código da sub-rotina *Reduz_LargNível* da segunda etapa da heurística de Burgess e Lai (1986).

Na heurística BL, a renumeração dos vértices do primeiro nível de $\mathcal{H}(v, \dots)$ é diferente da renumeração dos vértices pertencentes aos outros níveis. No primeiro nível, os vértices são renumerados em ordem crescente

do número de arestas que não são arestas de saída falsas. Para os outros níveis, a renumeração é diferente. Para isso, calcula-se o *número falso*. O número falso é calculado depois que todos os vértices do nível atual foram renumerados. Esse valor é calculado para cada vértice desse nível. Considere que todos os vértices pertencem a $K_i(v, \dots)$, para $0 \leq i < \ell(v) + 1$, tenham sido renumerados e considere que $u \in K_i(v, \dots)$. O número falso de u é dado por $nF(u) = nl + b(\mathcal{K}(v, \dots)) - nUv + 1$ se $nUv > 0$ e $nF = nl + b(\mathcal{K}(v, \dots)) - nUv$ caso contrário, em que $nF(u)$ é o número falso de u , nl é a nova numeração atribuída a u e nUv é o número de vértices ainda não renumerados no próximo nível da estrutura de nível adjacentes a u .

Para a renumeração dos vértices dos níveis $K_1(v, \dots), K_2(v, \dots), \dots, K_{\ell(v)}(v)$ considere os passos a seguir. Considere, também, que os vértices do nível $K_i(v, \dots)$, para $0 < i < \ell(v) + 1$, ainda não foram renumerados.

1. Após os vértices do nível $K_{i-1}(v, \dots)$ serem renumerados, os números falsos para cada vértice pertencente a $K_{i-1}(v, \dots)$ são calculados. Para a renumeração do nível $K_i(v, \dots)$, verificam-se se todos os números falsos dos vértices do nível $K_{i-1}(v, \dots)$ são maiores que o próximo número utilizado para a renumeração. Por exemplo, suponha que há quatro vértices no nível $K_{i-1}(v, \dots)$ e esses vértices já foram renumerados. Necessariamente, esses quatro vértices foram renumerados de $vn + 1$ a $vn + 4$, em que $vn = \sum_{j=0}^{i-1} |K_j(v, \dots)|$. Com isso, a próxima numeração, a ser atribuída a um vértice do nível $K_i(v, \dots)$, será o número $vn + 1$. Então, é verificado se os números falsos, dos vértices do nível $K_{i-1}(v, \dots)$, são maiores que o número $vn + 1$. Caso todos os números falsos forem maiores que o próximo número utilizado

na renumeração, então, é realizado o passo 2. Em caso contrário, seleciona-se um vértice ainda não renumerado com o menor número de arestas de saída. Então, realiza-se o passo 6;

2. Neste passo, seleciona-se um vértice do nível $K_i(v, \dots)$ para ser renumerado. Se houver apenas um vértice que ainda não foi renumerado, então é realizado o passo 6. Caso haja mais de um vértice para ser renumerado, então seleciona-se aquele vértice que possuir o menor número de arestas que não sejam arestas de saída falsas. Se houver mais de um vértice com o menor número de arestas que não sejam arestas de saída falsas, então, realiza-se o passo 3. Se houver apenas um, vá para o passo 6.
3. Seleciona-se, para a renumeração, o vértice que possuir o menor número mínimo de entrada. Caso haja mais de um vértice, vá para o passo 4. Em caso contrário, realiza-se o passo 6.
4. Seleciona-se, para a renumeração, o vértice que possuir o menor número máximo de entrada. Caso haja mais de um vértice, vá para o passo 5. Em caso contrário, realiza-se o passo 6.
5. Seleciona-se o vértice que possuir o menor número de arestas de saída e vá para o passo 6.
6. Realiza-se a renumeração do vértice selecionado. Se ainda existirem vértices para serem renumerados no nível $K_i(v, \dots)$, recalculam-se os números falsos dos vértices $K_{i-1}(v, \dots)$ e retorna para o passo 1. Caso contrário, calculam-se os números falsos para os vértices do nível $K_i(v, \dots)$, vá para o passo 1 e inicie a renumeração dos vértices do nível

$K_{i+1}(v, \dots)$. O algoritmo para quando os vértices do nível $K_{\ell(v)}(v, \dots)$ forem reenumerados.

2.5 Heurística WBRA

A heurística *Wonder Bandwidth Reduction Algorithm* (WBRA) (ESPOSITO et al., 1998) é baseada na heurística CM. Assim como as heurísticas CM e CMr, a heurística WBRA realiza a reenumeração dos vértices nível a nível da estrutura de nível. Nesta seção, apresenta-se uma descrição da heurística WBRA. Essa descrição é baseada no artigo de Esposito et al. (1998).

A heurística WBRA pode ser dividida em duas etapas. Na primeira etapa, tem-se a construção das estruturas de nível enraizadas dos vértices do grafo e, em seguida, tem-se a tentativa de se reduzir as larguras de nível dessas estruturas de nível enraizadas. Essa etapa é descrita na subseção 2.5.1. Na segunda etapa, realiza-se a reenumeração dos vértices do grafo. Essa reenumeração é realizada nível a nível da estrutura de nível, considerando-se o problema de associação linear com gargalo. Descreve-se essa etapa na subseção 2.5.2. Apresenta-se, na subseção 2.5.3, o pseudocódigo da heurística WBRA.

2.5.1 Inicialização e redução da largura de nível

Nesta etapa, constroem-se as estruturas de nível enraizadas de todos os vértices pertencentes a V . Em seguida, reduz-se a largura de nível de cada estrutura de nível gerada. Para se realizar a redução da largura de nível, Esposito et al. (1998) utilizaram uma heurística denominada *PushUp*. A redução da largura de nível é realizada ao se mover alguns vértices do nível

da estrutura de nível com maior largura para outros níveis da estrutura de nível, considerando-se a limitação de que, para cada aresta $(i, j) \in E$, os vértices i e j pertencem ao mesmo nível da estrutura de nível ou pertencem a níveis adjacentes. Isso pode fazer com que as estruturas de nível enraizadas tornem-se estruturas de nível (*não enraizadas*). Segundo Esposito et al. (1998), a complexidade do *PushUp* é $O(|V|^2)$.

Após a redução de largura de nível de cada estrutura de nível, são armazenadas, em uma fila de prioridades, as estruturas de nível que tiverem largura de nível menor que a largura de banda inicial. Isso significa que, ao se considerar cada vértice $\mathbf{v} \in V$, se $b(\mathcal{K}(\mathbf{v}, \dots)) < \beta(A)$, armazena-se $\mathcal{K}(\mathbf{v}, \dots)$ na fila de prioridades. Essa fila de prioridades é organizada pela ordem crescente de largura de nível das estruturas de nível.

2.5.2 Renumeração

Nesta etapa, renumeram-se os vértices de cada estrutura de nível armazenada na fila de prioridades do passo anterior. Primeiramente, remove-se a estrutura de nível $\mathcal{K}(\mathbf{v}, \dots)$ da fila de prioridades. Os vértices de cada nível da estrutura de nível são organizados em ordem crescente de grau. A renumeração é realizada nível a nível de $\mathcal{K}(\mathbf{v}, \dots)$. Nessa renumeração, minimizam-se as larguras de bandas das submatrizes geradas pela numeração dos vértices de cada nível da estrutura de nível.

Como a renumeração dos vértices de $\mathcal{K}(\mathbf{v}, \dots)$ é realizada nível a nível, considere que $K_h(\mathbf{v}, \dots)$, com $h = 0, 1, \dots, \ell(\mathbf{v})$, é o nível de $\mathcal{K}(\mathbf{v}, \dots)$ corrente na renumeração dos vértices. Também, considere que os vértices dos níveis $K_0(\mathbf{v}, \dots), K_1(\mathbf{v}, \dots), \dots, K_{h-1}(\mathbf{v}, \dots)$ foram renumerados. A largura de banda correspondente ao nível $K_h(\mathbf{v}, \dots)$, representada pela

submatriz $M_h \times M_h$, só será influenciada pela largura de banda da submatriz $M_h \times M_{h-1}$.

Esposito et al. (1998) interpretaram o problema da redução de largura de banda de matrizes como um *problema de atribuição linear com gargalo*. Seja t_{ij} o tempo para a máquina j realizar a tarefa i com $i, j = 1, 2, \dots, n$, em que n é o número de máquinas e tarefas. Considere que todas as máquinas são executadas paralelamente. Com isso, o problema de atribuição linear com gargalo é $\min(\max_{i,j}(t_{ij}x_{ij}))$, em que x_{ij} é o coeficiente da matriz de permutações P . Considere as seguintes restrições: $\sum_{i=1}^n x_{ij} = 1$, para todo $j = 1, \dots, n$; $\sum_{j=1}^n x_{ij} = 1$, para todo $i = 1, \dots, n$ e $x_{ij} \in \{0, 1\}$, para todo $i, j = 1, \dots, n$ (BURKARD; CELA, 1999). Considerando-se a estrutura de nível $\mathcal{K}(\mathbf{v}, \dots)$ e o coeficiente a_{ij} da matriz A , o problema da minimização da largura de banda pode ser aproximado por $\min(o)$, sujeito a $o \geq f_{ik}x_{ik}$; $\sum_{i=1}^{|\mathcal{K}_h(\mathbf{v}, \dots)|} x_{ik} = 1$, $k = 1, \dots, |\mathcal{K}_h(\mathbf{v}, \dots)|$; $\sum_{k=1}^{|\mathcal{K}_h(\mathbf{v}, \dots)|} x_{ik} = 1$, $i = 1, \dots, |\mathcal{K}_h(\mathbf{v}, \dots)|$; $x_{ik} \in \{0, 1\}$, em que $f_{ik} = \max\{m_{ik}, g_{ik}\}$, para todo $i, k = 1, \dots, |\mathcal{K}_h(\mathbf{v}, \dots)|$ com $m_{ik} = (|\mathcal{K}_{h-1}(\mathbf{v}, \dots)| - \min\{l \in \mathcal{K}_h(\mathbf{v}, \dots) \mid a_{kl} = 1\}) + i$ e $g_{ik} = |\mathcal{K}_h(\mathbf{v}, \dots)| + \sum_l(a_{kl}) - i$, $l \in \mathcal{K}_{h+1}(\mathbf{v}, \dots)$.

Com os coeficientes f_{ik} , pode-se obter boas renumerações para os vértices do nível $\mathcal{K}_h(\mathbf{v}, \dots)$. Para isso, consideram-se as renumerações dos vértices dos níveis $\mathcal{K}_{h-1}(\mathbf{v}, \dots)$ e $\mathcal{K}_{h+1}(\mathbf{v}, \dots)$.

A seguir, é apresentado o pseudocódigo da sub-rotina *Renumeração* na Figura 4. A sub-rotina *Renumeração* recebe a estrutura de nível $\mathcal{K}(\mathbf{v}, \dots)$ e a numeração S dos vértices. Uma solução (ou renumeração) S é o conjunto das renumerações dos vértices, ou seja, $S = \{s(1), s(2), \dots, s(|V|)\}$, em que $s(i)$ é a numeração do vértice i da solução S . Considerando-se uma estrutura de nível $\mathcal{K}(\mathbf{v}, \dots)$, a renumeração $S_{\mathcal{K}_h}$ é o conjunto das

renumerações dos vértices do nível $K_h(\mathbf{v}, \dots)$. A matriz A_S é obtida pela renumeração S do grafo $G = (V, E)$.

Na estrutura de repetição das linhas 2 a 5, calculam-se, iterativamente, os coeficientes f_{ik} , que são representados pela matriz F . Esse passo é realizado, na linha 3, pela sub-rotina *CalculoGargalo*, que recebe a numeração S .

Na linha 4, para cada nível da estrutura de nível $\mathcal{K}(\mathbf{v}, \dots)$, altera-se a numeração correspondente ao nível corrente. Essa numeração é representada por S_{K_h} e é realizada pela sub-rotina *AtribuiçãoGargalo*. Essa sub-rotina recebe a matriz F , o nível $K_h(\mathbf{v}, \dots)$ corrente da iteração e a numeração dos vértices S_{K_h} correspondente ao nível $K_h(\mathbf{v}, \dots)$. Com a renumeração de cada S_{K_h} , o algoritmo da Figura 4, na linha 6, retorna S renumerado.

Algoritmo: Numeração.

```

1 início
2   para cada ( nível  $K_h(\mathbf{v}, \dots)$  de  $\mathcal{K}(\mathbf{v}, \dots)$  ) faça
3     // a matriz  $F$  é calculada iterativamente
4      $F \leftarrow \text{CalculaGargalo}(S_{K_h});$ 
5     // com a numeração de cada nível  $S_{K_h}$ 
6     // tem-se ao final das iterações
7     // a numeração  $S$ 
8      $S_{K_h} \leftarrow \text{AtribuicaoGargalo}(F, K_h(\mathbf{v}, \dots), S_{K_h});$ 
9   fim-para-cada;
10  retorna  $S$ ;
11 fim.
```

Figura 4 Pseudo-código da sub-rotina *Renumeração* da heurística WBRA.

A sub-rotina *AtribuiçãoGargalo*, apresentada na Figura 5, recebe a matriz F , o nível $K_h(\mathbf{v}, \dots)$ de $\mathcal{K}(\mathbf{v}, \dots)$ e a renumeração dos vértices de $K_h(\mathbf{v}, \dots)$ representada por S_{K_h} . Na linha 2, calcula-se o valor mínimo do

gargalo e armazena-se esse valor limite na variável *limite*.

Algoritmo: AtribuiçãoGargalo.

```

1 início
  // menor valor do gargalo com  $i, k = 1, \dots, |K_h(v, \dots)|$ 
2  $limite \leftarrow \max(\max_i(\min_k(f_{ik})), \max_k(\min_i(f_{ik})))$ ;
3  $S_{K_h} \leftarrow MaxCardMatching(K_h(v, \dots), F, limite, S_{K_h})$ ;
4 se (  $|S_{K_h}| = |K_h(v, \dots)|$  ) então
  // retorna a numeração  $S_{K_h}$  obtida pela
  // sub-rotina MaxCardMatching
5   retorna  $S_{K_h}$ ;
6 fim-se;
7  $ini \leftarrow limite$ ;
8  $fim \leftarrow limite + |K_h(v, \dots)|$ ;
9 enquanto (  $ini \neq (fim - 1)$  ) faça
10    $meio \leftarrow \lfloor \frac{ini + fim}{2} \rfloor$ ;
11    $S_{K_h} \leftarrow MaxCardMatching(K_h(v, \dots), F, meio, S_{K_h})$ ;
12   se (  $|S_{K_h}| = |K_h(v, \dots)|$  ) então  $fim \leftarrow meio$ ;
13   senão  $ini \leftarrow meio$ ;
14 fim-enquanto;
  // retorna a numeração obtida pela última
  // chamada da
  // sub-rotina MaxCardMatching
15 retorna  $S_{K_h}$ ;
16 fim.
```

Figura 5 Pseudo-código da sub-rotina *AtribuiçãoGargalo* da heurística WBRA.

Seja E' um subconjunto de E do grafo $G = (V, E)$. E' é um conjunto de arestas independentes se as arestas de E' são incidentes a todos os vértices de V sem possuir vértices em comum. Seja $B = (U, W, Q)$ um grafo bipartido, em que U e W são conjuntos de vértices com $|U| = |W|$ e Q é um conjunto de arestas. Os vértices de U são considerados de saída e os vértices de W são os vértices de chegada. Os vértices de chegada são os vértices pertencentes ao nível $K_h(v, \dots)$ e os vértices de saída são os vértices que representam as

numerações que devem ser associadas a cada vértice pertencente ao nível $K_h(\mathbf{v}, \dots)$.

Na linha 3 da Figura 5, utiliza-se a sub-rotina *MaxCardMatching*. Essa sub-rotina encontra um conjunto máximo de arestas independentes de $B = (U, W, Q)$. Para isso, Esposito et al. (1998) utilizaram o algoritmo de Steiner e Yeomans (1996). Steiner e Yeomans (1996) propuseram um algoritmo em tempo linear para encontrar um conjunto máximo de arestas independentes em um grafo bipartido.

A sub-rotina *MaxCardMatching* recebe o nível $K_h(\mathbf{v}, \dots)$, a matriz F , um valor limite para se encontrar o conjunto de arestas independentes e a numeração S_{K_h} referente ao nível $K_h(\mathbf{v}, \dots)$. Seja $u \in U$ e $w \in W$. Há uma aresta $(u, w) \in Q$ se e, somente se, f_{uw} for menor ou igual ao parâmetro limite passado para a sub-rotina *MaxCardMatching*. Na linha 3 da Figura 5, esse valor limite é passado por meio da variável *limite*. A sub-rotina *MaxCardMatching* retorna nS_{K_h} que é a numeração dos vértices do nível $K_h(\mathbf{v}, \dots)$ de acordo com o conjunto de arestas independentes de B encontrado pela sub-rotina.

Seja S_{K_h} a renumeração dos vértices de $K_h(\mathbf{v}, \dots)$ realizada pela sub-rotina *MaxCardMatching*, na linha 3. Se $|S_{K_h}|$ é igual a $|K_h(\mathbf{v}, \dots)|$, então, significa que todos os vértices de $K_h(\mathbf{v}, \dots)$ foram renumerados. Com isso, retorna-se a renumeração dos vértices obtida pelo *MaxCardMatching*. Caso contrário, inicializam-se as variáveis *ini* e *fim* com os valores de *limite* e de $(\textit{limite} + |K_h(\mathbf{v}, \dots)|)$, respectivamente. Essas variáveis são inicializadas com esses valores para se limitar o intervalo de busca do valor do gargalo. Na estrutura de repetição das linhas 9 a 14, procura-se a melhor atribuição da numeração dos vértices do nível $K_h(\mathbf{v}, \dots)$. O algoritmo retorna a

renumeração obtida pela última chamada da sub-rotina *MaxCardMatching* na última iteração da estrutura de repetição das linhas 9 a 14.

2.5.3 Pseudocódigo da heurística WBRA

Mostra-se o pseudocódigo da heurística WBRA na Figura 6. O algoritmo recebe um grafo $G = (V, E)$ e a numeração S dos vértices de $G = (V, E)$. Considere que a matriz A_S é a matriz obtida pela renumeração S .

Inicializa-se a fila de prioridades na linha 2. Na estrutura de repetição das linhas 3 a 9, constroem-se as estruturas de nível enraizadas de todos os vértices de V . Essas estruturas de nível têm a largura de nível reduzida pela heurística *PushUp*. Armazenam-se, na fila de prioridades, as estruturas de nível que tiveram a largura de nível menor que a largura de banda correspondente à numeração original S dos vértices de $G = (V, E)$. Essas estruturas de nível são referenciadas na fila de prioridades em ordem crescente de largura de nível. Em seguida, remove-se a primeira estrutura de nível $\mathcal{H}(u, \dots)$, da fila de prioridades. $\mathcal{H}(u, \dots)$ é passada como parâmetro para a sub-rotina *Numeração*, apresentada na Figura 4. Se a largura de banda obtida pela numeração S' de $\mathcal{H}(u, \dots)$ for menor que a largura de banda S corrente, S' é atribuída a S . Em seguida, remove-se a próxima estrutura de nível da fila de prioridades e repetem-se esses passos. A estrutura de repetição das linhas 11 a 15 termina quando a largura de nível da estrutura de nível corrente for maior ou igual à largura de banda da matriz de adjacências correspondente à numeração S corrente. O algoritmo retorna a numeração S . Segundo Esposito, Malucelli e Tarricone (1998), a complexidade da heurística WBRA é $O(|V| \cdot |E| + |V|^2 \lg |V|)$.

Algoritmo: WBRA.

```

1 início
2   para cada ( vértice  $v \in V$  ) faça
3     BuscaEmLargura( $v$ ); // constrói-se estrutura de
        nível  $\mathcal{L}(v)$ 
        // reduz-se a largura de nível de  $\mathcal{L}(v)$ 
4     PushUp( $\mathcal{L}(v)$ );
5     se (  $b(\mathcal{K}(v, \dots)) < \beta(A)$  ) então
        // a fila de prioridades é organizada na
        ordem
        // crescente das larguras de nível
        // insere a estrutura de nível  $\mathcal{K}(v, \dots)$  na
        fila
6     filaPrioridade.Insere( $\mathcal{K}(v, \dots)$ );
7   fim-se;
8 fim-para-cada;
9  $\mathcal{K}(u, \dots) \leftarrow$  filaPrioridade.Remove();
10 enquanto (  $b(\mathcal{K}(u, \dots)) < \beta(A_S)$  ) faça
    // realiza-se a numeração dos vértices de
     $G = (V, E)$ 
11    $S' \leftarrow$  Numeracao( $\mathcal{K}(u, \dots), S$ );
12   se (  $\beta(A_{S'}) < \beta(A_S)$  ) então  $S \leftarrow S'$ ;
13    $\mathcal{K}(u, \dots) \leftarrow$  filaPrioridade.Remove();
14 fim-enquanto;
15 retorna  $S$ ;
16 fim.
```

Figura 6 Pseudo-código da heurística WBRA.

2.6 Heurística FNCHC

Proposta por Lim, Rodrigues e Xiao (2007), a heurística *Node Centroid Hill Climbing* (NCHC) baseia-se no ajuste de numeração para a busca global e no *Hill Climbing* para a busca local. Foi proposta com o intuito de ser um método rápido, simples e que gerasse bons resultados.

De forma geral, a heurística NCHC inicia por uma solução gerada

pela busca em largura. Altera-se essa numeração inicial pelo ajuste de numeração em relação a um determinado vértice e seus vértices adjacentes, denominado de *node centroid* (NC). Isso é realizado atribuindo-se um valor a cada vértice do grafo de acordo com os vértices que são considerados *node centroid*. Com isso, os vértices são renumerados em ordem crescente desse valor. Finalmente, aplica-se a busca local pelo *Hill Climbing* para obter um mínimo local. Essa busca local foi proposta por Lim, Rodrigues e Xiao (2006) e outros autores também a utilizaram como, por exemplo, Rodriguez-Tello, Kao e Torres-Jimenez (2008) e Mladenovic et al. (2010). Como a etapa da busca local tem grande custo computacional, o *Hill Climbing* é aplicado em apenas metade das vezes.

A heurística *Fast Node Centroid Hill Climbing* (FNCHC) é uma variação da heurística NCHC de Lim, Rodrigues e Xiao (2007). Na heurística FNCHC, esses autores utilizaram alguns parâmetros pré-definidos, como por exemplo, número de soluções mínimas geradas e número de vértices aos quais o *Hill Climbing* é aplicado. Com isso, uma redução do custo computacional pode ser alcançada em relação à heurística NCHC. Para mais detalhes sobre a heurística NCHC e FNCHC, veja Oliveira e Chagas (2014), a partir da página 43 até a 48.

2.7 Heurística GGPS

A heurística *Generalized Gibbs-Poole-Stockmeyer* (GGPS) foi proposta por Wang, Guo e Shi (2009) para a redução de largura de banda. Essa heurística, assim como a heurística de Gibbs, Poole e Stockmeyer (1976) (GPS), é dividida em três etapas: escolha de vértices pseudo-periféricos, redução da largura de nível e renumeração.

A heurística GGPS tem duas diferenças principais em relação à heurística GPS. A primeira diferença é que, na heurística GPS, apenas dois vértices pseudo-periféricos são encontrados na etapa de inicialização. Na heurística GGPS, dois conjuntos de vértices pseudo-periféricos são encontrados na etapa de inicialização. Primeiramente, selecionam-se dois vértices pseudo-periféricos v e u , em que u é o vértice com menor largura de nível do último nível da estrutura de nível enraizada de v . Essa escolha ocorre da mesma forma que o passo de inicialização da heurística GPS.

No primeiro conjunto de vértices pseudo-periféricos, selecionam-se todos os vértices do último nível da estrutura de nível de v . No segundo conjunto, selecionam-se todos os vértices do grafo que possuem o mesmo grau e a *excentricidade* de v e que não tenham sido escolhidos no primeiro conjunto.

Com mais vértices iniciais, mais estruturas de nível enraizadas são geradas. Com isso, pode ser que se obtenham larguras de nível menores que as larguras de nível das estruturas de nível obtidas pelo GPS. Isso porque, como na heurística GPS, as estruturas de nível enraizadas dos vértices pseudo-periféricos são combinadas para se gerar uma nova estrutura de nível.

A segunda diferença entre o GPS e o GGPS é que, na etapa de renumeração, em vez de se renumerar os vértices nível a nível da estrutura de nível em ordem crescente de grau, renumeram-se os vértices em ordem crescente do somatório dos rótulos dos seus vértices adjacentes já renumerados. Assim como no GPS, a renumeração final é invertida no GGPS. Para mais detalhes sobre a heurística GGPS, veja Oliveira e Chagas (2014), a partir da página 36 até a 41.

2.8 Heurística VNS-*Band*

A meta-heurística *variable neighbourhood search* (VNS) foi proposta por Mladenović e Hansen (1997). Divide-se o VNS em três fases: perturbação, busca local e troca de vizinhança. Inicialmente, o raio k da vizinhança da solução inicial S é atualizado como k_{min} . Geram-se perturbações para tentar escapar dos vales que os mínimos locais pertencem. Na etapa de perturbação, uma solução S' é encontrada dentro da vizinhança $N_{k_{min}}(S)$. Realiza-se uma exploração dentro do raio da vizinhança corrente. Uma busca local é aplicada na solução S' para encontrar uma solução S'' . Por meio dessa busca, procura-se encontrar um ótimo local. Na terceira etapa, verifica-se se a solução encontrada S'' é melhor que a solução corrente S . Caso seja, S'' passará a ser a solução corrente e k será atualizado novamente para k_{min} . Caso contrário, o raio da vizinhança k é incrementado de um k_{step} , que é ajustado de acordo com o problema. Repetem-se esses passos até que um critério de parada seja satisfeito. Um número máximo de iterações ou um tempo máximo de processamento são exemplos de critério de parada.

Um exemplo da meta-heurística VNS é mostrada na Figura 7. A busca começa na solução inicial S . Em seguida, é gerada, dentro da vizinhança $N_k(S)$, a solução S''_1 . Essa solução é gerada pelo passo de perturbação e pela busca local. Como a solução gerada S''_1 é melhor que a solução S corrente, então, S''_1 passa a ser a solução corrente e a busca continua em $N_k(S''_1)$. Dentro da vizinhança da solução S''_1 é encontrada a solução S''_2 . Mais uma vez, a nova solução é melhor que a solução corrente. Com isso, a solução S''_2 passa a ser a solução corrente. Uma nova busca é iniciada na vizinhança $N_k(S''_2)$. Com a busca em $N_k(S''_2)$, é gerada a solução S''_3 . Como a solução S''_3 não é melhor que a solução corrente S''_2 , o raio de

busca na vizinhança é aumentado, então, uma nova busca é realizada na vizinhança $N_{k+1}(S_2'')$. Em seguida, a solução S_4'' é gerada. Esses passos são repetidos até que o critério de parada estabelecido seja alcançado.

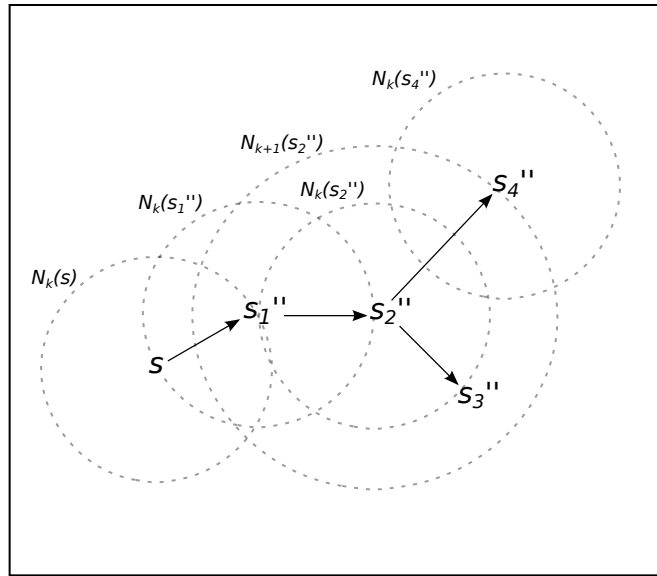


Figura 7 Exemplo de busca pela heurística VNS. A busca é iniciada na solução S . A solução S_1'' é encontrada em $N_k(S)$. A solução S_1'' é melhor que a solução corrente S . Então, uma nova busca é iniciada na vizinhança $N_k(S_1'')$ e a solução S_2'' é encontrada. Com a primeira busca na vizinhança $N_k(S_2'')$, é encontrada a solução S_3'' . Como a solução S_3'' não é melhor que a solução corrente, então, o raio de busca na vizinhança $N_k(S_2'')$ é incrementado. Com isso, a solução S_4'' é encontrada dentro da vizinhança $N_{k+1}(S_2'')$.

Na heurística *VNS-band*, uma solução S é o conjunto das numerações dos vértices, ou seja, $S = \{s(1), s(2), \dots, s(|V|)\}$, em que $s(i)$ é a numeração do vértice i da solução S . No *VNS-band*, uma solução inicial das numerações dos vértices é construída por uma busca em largura aleatória, iniciando-se por um vértice aleatório. Na busca em largura aleatória, os vértices adjacentes de cada vértice são visitados aleatoriamente. Os passos do VNS (perturbação, busca local e troca de vizinhança) são seguidos até que o

critério de parada seja satisfeito. Segundo Mladenovic et al. (2010), o VNS-*band* é, ou pelo menos era, até a data de publicação do artigo, o estado da arte em métodos para a redução de largura de banda. Para mais detalhes sobre a heurística VNS-*band*, veja Oliveira e Chagas (2014), a partir da página 51 até a 60.

2.9 Heurística hGPHH

Koohestani e Poli (2011) propuseram uma *híper-heurística* baseada em *programação genética* para o problema da redução de largura de banda de matrizes. O termo híper-heurística foi introduzido por Cowling, Kendall e Soubeiga (2001) e, em termos gerais, é uma heurística de alto nível especializada na geração de heurísticas para um determinado problema.

Proposta por Koza (1992), a programação genética é um método evolutivo inspirado na evolução biológica. Conforme Baeck, Fogel e Michalewicz (1997) explicam, na programação genética, indivíduos, que são algoritmos, são modificados pelas regras da programação genética até que os resultados desejados sejam alcançados, ou seja, um algoritmo adequado para o problema em questão. Essas regras são similares ao que observamos na natureza em relação à teoria da evolução, como, por exemplo, mutação, cruzamento de indivíduos (*crossover*), herança de características e outros. Segundo Koohestani e Poli (2011), pode-se considerar a programação genética como uma especialização da meta-heurística algoritmo genético. A principal diferença entre essas abordagens é que no algoritmo genético os indivíduos são *strings* de tamanho e alfabeto limitado. Já na programação genética, os indivíduos são algoritmos. Outro exemplo de diferença entre esses métodos é a função de avaliação (*fitness*). Na meta-heurística

algoritmo genético, essa função é diretamente atrelada à solução. Já na programação genética, a função de avaliação é ligada à execução dos algoritmos e nas soluções por eles geradas.

Koohestani e Poli (2011) utilizaram a híper-heurística *Genetic Programming Hyper-Heuristic for Bandwidth Reduction* (GPHH-*band*) para a geração de heurísticas para a redução de largura de banda. Os autores basearam-se na heurística CMr para a geração das heurísticas pela GPHH-*band*. A heurística CMr foi utilizada como base por ser uma heurística rápida, bem difundida na literatura e por gerar bons resultados. Todas as heurísticas geradas pela híper-heurística GPHH-*band* realizam a renumeração dos vértices de forma similar à heurística CMr, ou seja, nível a nível da estrutura de nível enraizada em um vértice pseudo-periférico. Os autores não informaram qual algoritmo para se encontrar um vértice pseudo-periférico foi utilizado nas heurísticas geradas pela GPHH-*band*. Koohestani e Poli (2011) apenas explicam que essas heurísticas iniciam-se por um vértice pseudo-periférico.

A única diferença entre as heurísticas geradas pela GPHH-*band* e a heurística CMr está na forma em que os vértices de cada nível da estrutura de nível enraizada são renumerados. Como descrito na subseção 2.2, a heurística CMr é similar à busca em largura, porém os vértices são visitados (e renumerados) em ordem crescente de grau. Já as heurísticas geradas pela GPHH-*band*, os vértices são visitados (e renumerados) em ordem crescente do resultado de uma equação, encontrada pela própria híper-heurística GPHH-*Band* (KOOHESTANI; POLI, 2011).

Para a geração das equações de cada heurística, são utilizados alguns operadores primitivos e as expressões são representadas, e armazenadas,

em forma de árvores. Embora a heurística CMr realize a renumeração dos vértices em ordem crescente de grau, Koohestani e Poli (2011) não utilizaram um operador primitivo que retornasse o grau do vértice. Isso porque os autores observaram que utilizar esse operador primitivo gerava heurísticas ruins. Em vez disso, os autores utilizaram a soma dos graus dos vértices adjacentes ao vértice em questão. Esse operador primitivo é denominado de SDV. Os outros operadores primitivos utilizados na geração das heurísticas pela GPHH-*Band* são: adição, subtração, multiplicação, número de vértices do grafo e constantes uniformemente distribuídos no intervalo real $[-1, 1]$.

A hiper-heurística GPHH-*Band* inicia com uma população de indivíduos, ou heurísticas, geradas aleatoriamente. Posteriormente, cada heurística dessa população é executada em cada instância do conjunto de treinamento. Koohestani e Poli (2011) utilizaram, para o conjunto de treinamento, 25 instâncias da base *Harwell-Boeing*. Após todas as heurísticas terem renumerado todas as instâncias do conjunto de treinamento, é calculada a largura de banda para cada instância. A função de avaliação, ou *fitness*, de cada heurística é a soma das larguras de banda de cada instância do conjunto de treinamento. Após o cálculo da função de avaliação de cada heurística, são realizadas as operações genéticas. Aplica-se a seleção de heurísticas baseada na função de avaliação. Em sequência, a nova geração de heurísticas é gerada por meio do cruzamento, mutação e elitismo aplicados às heurísticas selecionadas. Esses passos são repetidos até que o critério de parada estabelecido seja atingido, ou seja, o número de gerações. A melhor heurística encontrada é retornada.

Koohestani e Poli (2011) realizaram testes na redução de largura de

banda com a melhor heurística gerada pela híper-heurística *GPHH-Band*. Essa heurística, denominada neste trabalho de *hGPHH*, foi comparada com a heurística *CMr-GL* do MATLAB e obteve resultados competitivos na redução de largura de banda e no custo computacional. Por isso, na revisão sistemática realizada em Oliveira e Chagas (2015a), essa heurística foi selecionada como sendo uma possível melhor heurística na relação benefício (i.e. redução de largura de banda) por custo computacional.

A heurística *hGPHH* é similar à heurística *CMr*. Inicia a renumeração por um vértice pseudo-periférico, neste trabalho utilizou-se o algoritmo de George e Liu (1979), e realiza a renumeração dos vértices em ordem crescente do valor da equação $0,179492928171 \times SDV^3 + 0,292849834929 \times SDV^2 - 0,208926175433 \times |V| - 0,736485142138 \times |V| \times SDV - 1,77524579882 \times SDV - 1,75681383404$. Essa equação foi encontrada pela híper-heurística *GPHH-Band* nos testes realizados por Koohestani e Poli (2011).

2.10 Heurística *CSS-band*

A meta-heurística *busca em sistema carregado*, *Charged System Search* (*CSS*), é uma meta-heurística desenvolvida por Kaveh e Talatahari (2010) para a resolução de problemas relacionados à análise de estruturas. Os autores utilizaram as leis de Coulomb e Gauss, da eletrostática, e das leis da mecânica de Newton para desenvolver essa meta-heurística.

A *CSS* é composta por *partículas carregadas* (*PC*), em que uma *PC* é uma solução válida para o problema em questão. Cada *PC* é considerada uma esfera eletricamente carregada que exerce uma força elétrica sobre as outras partículas, de acordo com as leis de Coulomb e de Gauss. A

força resultante e as leis da mecânica determinam as novas posições e as velocidades das PCs. A utilização de tais leis pelo CSS provê um modelo que pode ser utilizado em problemas de otimização discretos e contínuos. Kaveh e Sharafi (2012) utilizaram um modelo baseado nessas leis para criar uma proposta para a renumeração dos vértices de um grafo. Com isso, utilizaram essa meta-heurística para a redução de banda de matrizes esparsas, denominada, neste trabalho, como *CSS-band*.

A heurística *CSS-band* possui nove regras básicas e todos os passos da heurística devem respeitar essas regras. Inicialmente, soluções aleatórias são geradas. Cada PC inicia em uma posição aleatória que é definida justamente de acordo com a solução, ou seja, a renumeração dos vértices do grafo definem a posição da PC. Além disso, as PCs iniciam com velocidade 0. A cada iteração, as PCs influenciam umas às outras de acordo com as leis de Coulomb e Gauss. Com isso, novas soluções (posições) são geradas. Esse processo é realizado até que o critério de parada estabelecido seja alcançado. Kaveh e Sharafi (2012) utilizaram o número de iterações como critério de parada. Para mais detalhes sobre a heurística *CSS-band*, veja Oliveira e Chagas (2014), a partir da página 60 até a 66.

2.11 Algoritmo de George-Liu

George e Liu (1979) propuseram um algoritmo para se encontrar um vértice pseudo-periférico. Esse algoritmo foi proposto como melhorias em relação ao algoritmo da primeira etapa da heurística GPS. George e Liu (1979) propuseram quatro mudanças nesse algoritmo das quais duas apresentaram os melhores resultados.

Primeiramente, é escolhido um vértice $v \in V$ qualquer. Em seguida,

gera-se a estrutura de nível enraizada $\mathcal{L}(v)$. Um vértice $u \in L_{\ell(v)}(v)$ de grau mínimo é selecionado. $\mathcal{L}(u)$ é gerada e verifica-se se $\ell(u) > \ell(v)$. Se isso ocorrer, o vértice u passa a ser o corrente e todo o processo é repetido. Caso contrário, o algoritmo termina e o vértice v é selecionado como um vértice pseudo-periférico. Para mais detalhes sobre o algoritmo de George e Liu (1979), veja Oliveira e Chagas (2014), a partir da página 70 até a página 71.

2.12 Método dos gradientes conjugados

Apresenta-se, nesta seção, uma breve descrição sobre o método dos gradientes conjugados (MGC). Considere, a seguir, a descrição do MGC (CHAGAS, 2013):

O método dos gradientes conjugados foi proposto por Hestenes e Stiefel (1952) e Lanczos (1952). “Verificou-se que o método dos gradientes conjugados é um eficiente método iterativo para solução de sistemas de equações lineares, simétricos, positivos-definidos e esparsos” (BURDEN; FAIRES, 2003, p. 404). Todos os sistemas de equações lineares abordados neste trabalho são simétricos e positivos-definidos.

Seja um sistema de equações lineares na forma $Ax = b$, em que $A \in \mathbb{R}^{n \times n}$, x e $b \in \mathbb{R}^n$. A solução desse sistema de equações lineares pode ser encontrada por meio da minimização de $f(x)$, com $x \in \mathbb{R}^n$. Seja f uma função quadrática,

$$f(x) = \frac{x^T A x}{2} - b^T x, \quad (1)$$

o vetor x^* é uma solução do sistema de equações lineares se,

e somente se, x^* minimiza f . Ao derivar f da equação 1 em relação ao vetor x e igualar a zero, obtém-se

$$\nabla f(x^*) = Ax^* - b = 0, \quad (2)$$

ou seja, o método dos gradientes conjugados soluciona o sistema de equações lineares quando encontra o mínimo de f , $f'(x) = 0$.

Inicia-se o método dos gradientes conjugados escolhendo-se uma solução inicial x_0 , uma solução aproximada de $Ax^* = b$. Considere que, inicialmente, $r_0 = b - Ax_0$ e $d_1 = r_0$, em que r_k é o vetor resíduo na iteração k e d_k é o vetor de direção de busca na iteração k , com $k = 0, 1, \dots, n-1$. O método dos gradientes conjugados constrói os vetores r e d , a cada iteração, por

$$r_k = r_{k-1} - \alpha_k A d_k \quad (3)$$

e

$$d_{k+1} = r_k + \beta_k d_k. \quad (4)$$

Os escalares α da equação 3 e β (não confunda com largura de banda) da equação 4 são dados por

$$\alpha_k = \frac{r_{k-1}^T r_{k-1}}{d_k^T A d_k} \quad (5)$$

e

$$\beta_k = \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}. \quad (6)$$

A solução em cada iteração é dada por

$$x_k = x_{k-1} + \alpha_k d_k. \quad (7)$$

Os vetores r e d auxiliam o método dos gradientes conjugados na busca do mínimo de f , em que cada iteração a solução x_k aproxima-se mais de x^* . O vetor d satisfaz a condição de *A-ortogonalidade* (ou conjugada), $d_i^T A d_j = 0$, com $i \neq j$. Em Luenberger (2008, p. 263), tem-se a definição formal de *A-ortogonalidade*. “O uso dos vetores de direção d *A-ortogonais* produz o método de direção conjugada” (BURDEN; FAIRES, 2003, p. 409). Resíduos são ortogonais entre si, portanto, $r_i^T r_j = 0$, para $i \neq j$. Ainda, os vetores r_i e d_j , com $i \neq j$, são mutuamente ortogonais, ou seja, $r_i^T d_j = 0$.

Com as equações 3, 4, 5, 6 e 7, forma-se o algoritmo iterativo do método dos gradientes conjugados. Esse algoritmo é mostrado na Figura 8. Inicialmente, define-se a precisão numérica ε desejada. Nas linhas 3 a 5, inicializam-se os vetores x, r e d . As variáveis k e *erro* são inicializadas na linha 6, em que k é o contador para a estrutura de repetição *enquanto* e o *erro* é a variável para o critério de parada. A cada iteração atualiza-se $erro \leftarrow \frac{\|x_k - x_{k-1}\|_\infty}{\|x_k\|_\infty}$, em que $\|x\|_\infty = \max\{|x_1|, |x_2|, \dots, |x_n|\}$. Se $erro \geq \varepsilon$, o algoritmo para.

O método dos gradientes conjugados converge para a solução em no máximo n iterações. Para mais detalhes sobre o método dos gradientes conjugados, veja Shewchuk (1994) e Burden e Faires (2003), da página 404 a 415, por exemplo.

2.12.1 Método dos gradientes conjugados preconditionado

Embora o método dos gradientes conjugados seja um resolutor rápido em relação aos métodos diretos, há casos em que a convergência desse método é lenta. Segundo Saad (2003, p. 261), “isso ocorre com frequência em sistemas de equações lineares (SEL) oriundos de problemas

Algoritmo: método dos gradientes conjugados.

```

1 início
  // define a precisão numérica  $\varepsilon$ , ex.  $10^{-16}$ 
2  $\varepsilon \leftarrow 10^{-16}$ ;
3  $x_0 \leftarrow 0$ ; // estimativa inicial
4  $r_0 \leftarrow b - Ax_0$ ;
5  $d_0 \leftarrow r_0$ ;
6  $k \leftarrow 1$ ;  $erro \leftarrow \varepsilon + 1$ ;
  //  $n$  é a dimensão de  $x$ 
7 enquanto (  $(k < n) \wedge (erro > \varepsilon)$  ) faça
8    $\alpha_k \leftarrow \frac{r_{k-1}^T r_{k-1}}{d_k^T A d_k}$ ; // equação 5
9    $x_k \leftarrow x_{k-1} + \alpha_k d_k$ ; // equação 7
10   $r_k \leftarrow r_{k-1} - \alpha_k A d_k$ ; // equação 3
11   $erro \leftarrow \frac{\|x_k - x_{k-1}\|_\infty}{\|x_k\|_\infty}$ ;
12   $k \leftarrow k + 1$ ;
13   $\beta_k \leftarrow \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}$ ; // equação 6
14   $d_{k+1} \leftarrow r_k + \beta_k d_k$ ; // equação 4
15 fim-enquanto;
16 fim.
```

Figura 8 Pseudo-código do método dos gradientes conjugados.

em dinâmica dos fluídos e dispositivos eletrônicos”. Com isso, métodos de condicionamento de SEL são utilizadas para acelerar a convergência do método dos gradientes conjugados.

O condicionamento da matriz de coeficientes de um SEL é realizado de forma que seja gerado um SEL equivalente ao SEL original mas que possa ser resolvido mais rapidamente e que apresente a mesma solução. Embora não seja garantido que a resolução de um SEL condicionado convirja com menos iterações, isso geralmente ocorre (SAAD, 2003, p. 261).

Considere um SEL $Ax = b$. Em termos simples, os métodos de condicionamento encontram uma matriz M não singular, ou seja, uma

matriz que possua inversa, em que o resultado da operação $P^{-1}A$ tenha um número de condição menor que A . Além disso, M deverá ser de tal forma que, a resolução de $Mx = b$ apresente o menor custo computacional possível. Isso porque, em cada iteração, esse SEL deverá ser resolvido. Ainda, no caso do método dos gradientes conjugados preconditionado, a matriz M também deve ser simétrica e positiva-definida.

Como Saad (2003, p. 262) explica, “há três formas de preconditionamento: à esquerda, à direita e o preconditionamento conhecido como *split*”. Especificamente, neste trabalho, utilizou-se os preconditionamentos à esquerda e *split*.

A matriz M pode ser aplicada à esquerda do SEL $Ax = b$. Nesse caso, gera-se o SEL preconditionado

$$M^{-1}Ax = M^{-1}b. \quad (8)$$

No preconditionamento *split* utiliza-se a matriz M em sua forma fatorada $M = M_L M_R$, em que as matrizes M_L e M_R são matrizes triangulares. Com isso, o preconditionador pode ser separado em

$$M_L^{-1} A M_R^{-1} u = M_L^{-1} b, \quad x \equiv M_R^{-1} u. \quad (9)$$

Como descrito neste trabalho, utilizou-se os preconditionamento à esquerda e o preconditionamento *split*. O preconditionamento à esquerda, apresentado na equação 8, foi utilizado com o método preconditionador de Jacobi. O preconditionamento *split*, apresentado na equação 9, foi utilizado com o método incompleto de Cholesky. Esses métodos preconditionadores são detalhados na subseção 2.13.

A seguir, são apresentados os pseudo-códigos do método dos gradientes conjugados preconditionado à esquerda e pelo *split*. Esses métodos são apresentados nas Figuras 9 e 10, respectivamente. Note que, no algoritmo da Figura 9, utiliza-se a matriz preconditionadora M e a variável z_k para alterar o algoritmo do método dos gradientes conjugados mostrado na Figura 8.

Algoritmo: método dos gradientes conjugados preconditionado à esquerda.

```

1 início
    // define a precisão numérica  $\varepsilon$ , ex.  $10^{-16}$ 
2  $\varepsilon \leftarrow 10^{-16}$ ;
3  $x_0 \leftarrow 0$ ; // estimativa inicial
4  $r_0 \leftarrow b - Ax_0$ ;
5  $z_0 \leftarrow M^{-1}r_0$ ;
6  $d_0 \leftarrow r_0$ ;
7  $k \leftarrow 1$ ;  $erro \leftarrow \varepsilon + 1$ ;
    //  $n$  é a dimensão de  $x$ 
8 enquanto (  $(k < n) \wedge (erro > \varepsilon)$  ) faça
9      $\alpha_k \leftarrow \frac{r_{k-1}^T z_{k-1}}{d_k^T A d_k}$ ;
10     $x_k \leftarrow x_{k-1} + \alpha_k d_k$ ;
11     $r_k \leftarrow r_{k-1} - \alpha_k A d_k$ ;
12     $z_k \leftarrow M^{-1}r_k$ ;
13     $erro \leftarrow \frac{\|x_k - x_{k-1}\|_\infty}{\|x_k\|_\infty}$ ;
14     $k \leftarrow k + 1$ ;
15     $\beta_k \leftarrow \frac{r_k^T r_k}{z_{k-1}^T z_{k-1}}$ ;
16     $d_{k+1} \leftarrow z_k + \beta_k d_k$ ;
17 fim-enquanto;
18 fim.
```

Figura 9 Pseudo-código do método dos gradientes conjugados preconditionado à esquerda.

No algoritmo da Figura 10, utiliza-se a matriz M na forma de uma

fatoração incompleta de Cholesky, ou seja, $M = LL^T$. Isso é possível pois pode-se utilizar o preconditionador do tipo *split*, mostrado na equação 9, para que a simetria seja mantida. Com isso, pode-se reescrever a equação 9 na forma $L^{-1}AL^{-T}u = L^{-1}b$, em que $x = L^{-T}u$. Além disso, a variável z_k é também utilizada. Para mais detalhes sobre o método dos gradientes conjugados preconditionado, veja Saad (2003), a partir da página 262 até a 264.

Algoritmo: método dos gradientes conjugados preconditionado por *split*.

```

1 início
  // define a precisão numérica  $\varepsilon$ , ex.  $10^{-16}$ 
2  $\varepsilon \leftarrow 10^{-16}$ ;
3  $x_0 \leftarrow 0$ ; // estimativa inicial
4  $r_0 \leftarrow b - Ax_0$ ;
5  $z_0 \leftarrow L^{-1}r_0$ ;
6  $d_0 \leftarrow L^{-T}z_0$ ;
7  $k \leftarrow 1$ ;  $erro \leftarrow \varepsilon + 1$ ;
  //  $n$  é a dimensão de  $x$ 
8 enquanto (  $(k < n) \wedge (erro > \varepsilon)$  ) faça
9    $\alpha_k \leftarrow \frac{z_{k-1}^T z_{k-1}}{d_k^T A d_k}$ ;
10   $x_k \leftarrow x_{k-1} + \alpha_k d_k$ ;
11   $z_k \leftarrow z_{k-1} - \alpha_k L^{-1} A d_k$ ;
12   $erro \leftarrow \frac{\|x_k - x_{k-1}\|_\infty}{\|x_k\|_\infty}$ ;
13   $k \leftarrow k + 1$ ;
14   $\beta_k \leftarrow \frac{z_k^T z_k}{z_{k-1}^T z_{k-1}}$ ;
15   $d_{k+1} \leftarrow L^{-T} z_k + \beta_k d_k$ ;
16 fim-enquanto;
17 fim.
```

Figura 10 Pseudo-código do método dos gradientes conjugados preconditionado por *split*.

2.13 Métodos preconditionadores

Alguns métodos diretos e iterativos para a resolução de SEL são utilizados como preconditionadores. Tem-se o método de Cholesky, como exemplo de método direto utilizado como preconditionador, e o método de Jacobi (1845) como método iterativo. Contudo, esses métodos são aplicados de forma parcial.

O método de Jacobi (1845) é um resolutor clássico de SEL e é um dos métodos de preconditionamento mais simples. Seja A uma matriz $n \times n$, D a matriz composta pela diagonal principal de A , $(L - D)$ a matriz estritamente triangular inferior de A e $(U - D)$ a matriz estritamente triangular superior de A , em que $A = D + (L - D) + (U - D)$. A matriz M preconditionadora de Jacobi é definida por $M = D$.

Na fatoração de Cholesy, a matriz A é fatorada em L e L^T em que $A = LL^T$. Para o preconditionamento, utiliza-se a fatoração incompleta de Cholesky. Com isso, calcula-se $M = LL^T$. Segundo Benzi (2002), os autores Meijerink e van der Vorst (1977) foram os primeiros autores a utilizarem a fatoração incompleta de Cholesky como preconditionador do MGC.

Especificamente, neste trabalho, utilizou-se a fatoração incompleta de Cholesky com zero *fill-in* (IC0) para realizar o preconditionamento das matrizes de coeficientes dos SEL. O IC0 mantém, na matriz M , a estrutura de matriz A sem realizar *fill-in*. Em outras palavras, se $a_{ij} \neq 0$, então, $m_{ij} \neq 0$ e se $a_{ij} = 0$, então, $m_{ij} = 0$, para $1 \leq i, j \leq n$, $A = [a_{ij}]$ e $M = [m_{ij}]$.

Mostra-se, na Figura 11, o algoritmo da fatoração incompleta de Cholesky com zero *fill-in* (GOLUB; VAN LOAN, 1996, p. 535). O algoritmo recebe uma matriz A simétrica e retorna a matriz M preconditionadora.

Para mais detalhes sobre o método preconditionador de Jacobi, veja

Algoritmo: IC0.

Entrada: matriz $A = [a_{ij}]$ simétrica $n \times n$;
Saída: matriz $M = [m_{ij}]$ condicionadora;

```

1 início
2    $M \leftarrow A$ ; // copia-se os elementos de A para M
3   para (  $k \leftarrow 1$ ;  $k \leq n$ ;  $k \leftarrow k + 1$  ) faça
4      $m_{kk} \leftarrow \sqrt{m_{kk}}$ ;
5     para (  $i \leftarrow k + 1$ ;  $i \leq n$ ;  $i \leftarrow i + 1$  ) faça
6       se (  $m_{ik} \neq 0$  ) então  $m_{ik} \leftarrow \frac{m_{ik}}{m_{kk}}$ ;
7     fim-para;
8     para (  $j \leftarrow k + 1$ ;  $j \leq n$ ;  $j \leftarrow j + 1$  ) faça
9       para (  $i \leftarrow j$ ;  $i \leq n$ ;  $i \leftarrow i + 1$  ) faça
10        se (  $m_{ij} \neq 0$  ) então  $m_{ij} \leftarrow m_{ij} - m_{ik} * m_{jk}$ ;
11      fim-para;
12    fim-para;
13  fim-para;
14  retorna M;
15 fim.
```

Figura 11 Pseudo-código da fatoração incompleta de Cholesky com zero *fill-in*.

Saad (2003, p. 284). Para mais detalhes sobre a fatoração incompleta de Cholesky com zero *fill-in*, veja Golub e Van Loan (1996, p. 535).

2.14 Estruturas CSR-SSS e CSC-SSS

Pode-se economizar espaço em memória apenas armazenando-se as informações essenciais das matrizes esparsas, por exemplo, somente os elementos não nulos e os índices. Para isso, há vários esquemas de armazenamento de matrizes esparsas e Saad (2003, p. 89-92) apresenta alguns.

Os esquemas de armazenamento de matrizes esparsas *Compressed Sparse Row/Column Sparse Symmetric Skyline* (CSR-SSS/CSC-SSS) são

a combinação de outros dois esquemas conhecidos: *Compressed Sparse Row/Column* (CSR/CSC) e *Skyline*. Para detalhes sobre os esquemas de armazenamento de matrizes esparsas CSR/CSC e *Skyline*, veja Saad (2003, p. 90) e Jennings (1966) respectivamente.

As estruturas CSR-SSS/CSC-SSS são para o armazenamento de matrizes esparsas simétricas. Essas estruturas utilizam quatro vetores para armazenar uma matriz esparsa A : um vetor para armazenar os coeficientes da diagonal principal; um vetor em que se armazenam os coeficientes não nulos da matriz estritamente triangular inferior (ou superior); um vetor que armazena os índices das colunas (CSR-SSS) ou os índices das linhas (CSC-SSS); um vetor armazena as posições em que cada linha (CSR-SSS) ou coluna (CSC-SSS) são iniciadas nos vetores dos coeficientes e índices.

A seguir, um exemplo das estruturas CSR-SSS e CSC-SSS é mostrado. Para isso, considere a seguinte matriz:

$$A = \begin{bmatrix} \mathbf{1} & -2 & 0 & -3 & 5 \\ -2 & \mathbf{-2} & 0 & 4 & 0 \\ 0 & 0 & \mathbf{-1} & 0 & -1 \\ -3 & 4 & 0 & \mathbf{3} & -6 \\ 5 & 0 & -1 & -6 & \mathbf{4} \end{bmatrix}$$

A matriz A pode ser armazenada no formato CSR-SSS da seguinte forma:

$$\begin{aligned} dCof &= [\ 1 \ -2 \ -1 \ 3 \ 4 \], \\ linhaPtr &= [\ 0 \ 0 \ 1 \ 1 \ 3 \ 6 \], \\ colunaInd &= [\ 1 \ 1 \ 2 \ 1 \ 3 \ 4 \], \\ coef &= [\ -2 \ -3 \ 4 \ 5 \ -1 \ -6 \], \end{aligned}$$

em que $dCoef$ é o vetor dos coeficientes da diagonal principal, $linhaPtr$ é o vetor de onde se inicia cada linha nos vetores $colInd$ e $coef$, $colunaInd$ é o vetor dos índices das colunas e $coef$ são os coeficientes da matriz estritamente triangular inferior.

De forma similar ao formato CSR-SSS, a matriz A pode ser armazenada no formato CSC-SSS da seguinte forma:

$$\begin{aligned} dCoef &= [1 \quad -2 \quad -1 \quad 3 \quad 4 \quad], \\ colunaPtr &= [0 \quad 3 \quad 4 \quad 5 \quad 6 \quad 6 \quad], \\ linhaInd &= [1 \quad 1 \quad 1 \quad 2 \quad 3 \quad 4 \quad], \\ coef &= [-2 \quad -3 \quad 5 \quad 4 \quad -1 \quad -6 \quad], \end{aligned}$$

em que o $colunaPtr$ é o vetor de onde se inicia cada coluna nos vetores $colInd$ e $coef$, $linhaInd$ é o vetor dos índices das linhas.

3 METODOLOGIA

Neste Capítulo, apresentam-se os procedimentos metodológicos adotados neste trabalho. Explica-se, na subseção 3.1, as revisões sistemáticas realizadas. As heurísticas propostas neste trabalho são mostradas na subseção 3.2. As instâncias utilizadas nas simulações computacionais são detalhadas na subseção 3.3. Apresenta-se, na subseção 3.4, os detalhes das implementações realizadas no desenvolvimento deste trabalho. Na subseção 3.5, as especificações das máquinas e as ferramentas utilizadas nas simulações computacionais são mostradas.

3.1 Revisões sistemáticas

Como descrito, heurísticas para a redução de largura têm sido propostas desde a década de 1960 e é difícil dizer qual heurística é a melhor. Isso porque há outros fatores que devem ser levados em consideração além da redução de largura de banda. Por exemplo, a heurística deve ser rápida, pois tem-se o objetivo de se reduzir o custo computacional na resolução de SELs. Além disso, é importante que essa heurística também possua um baixo consumo de memória. Com isso, realizou-se uma revisão sistemática para identificar heurísticas para a redução de largura de banda com bons benefícios (i. e. redução de largura de banda) por custos computacionais.

A revisão sistemática foi realizada na base de artigos Scopus (<http://www.scopus.com/>). Na busca, foram utilizados os termos “TITLE-ABS-KEY((matrix bandwidth reduction) or (matrix bandwidth minimization))”. Com essa busca, foram obtidos 502 artigos. Foram identificados, entre todos os 502 artigos, aqueles que apresentavam propostas de heurísticas para a redução de largura de banda ou que apresentavam

comparações entre heurísticas para a redução de largura de banda. Além disso, as referências de todos os artigos sobre redução de largura de banda foram analisados na tentativa de se encontrar outros artigos de heurísticas para a redução de largura de banda. Com isso, 73 heurísticas para a redução de largura de banda foram identificadas.

Foram analisados os resultados, em cada artigo, em relação à redução de largura de banda e ao custo computacional de todas as 73 heurísticas. Com isso, foi possível identificar nove heurísticas como sendo as heurísticas com os melhores benefícios (i. e. redução de largura de banda) por custos computacionais. As heurísticas identificadas foram: CMr iniciando-se por um vértice pseudo-periférico pelo algoritmo de George-Liu (CMr-GL) (GEORGE; LIU, 1981), BL (BURGESS; LAI, 1986), WBRA (ESPOSITO et al., 1998), heurística de Boutora et al. (2007), FNCHC (LIM; RODRIGUES; XIAO, 2007), GGPS (WANG; GUO; SHI, 2009), VNS-*band* (MLADENOVIC et al., 2010), hGPHH (KOOHESTANI; POLI, 2011) e CSS-*band* (KAVEH; SHARAFI, 2012). Com a exceção da heurística de Boutora et al. (2007), todas as outras oito heurísticas foram avaliadas neste trabalho.

A heurística de Boutora et al. (2007) não foi implementada neste trabalho, pois é uma heurística específica para malhas (grafos) triangulares. Portanto, a heurística de Boutora et al. (2007) não funcionaria na maioria das instâncias da base Harwell-Boeing. Porém, com a ideia da heurística Boutora et al. (2007), foi possível propor a nova heurística BeL-GL, que é apresentada na subseção 3.2.1. Para mais detalhes sobre a revisão sistemática, veja os trabalhos de Chagas e Oliveira (2015) e Oliveira e Chagas (2015a).

3.2 Heurísticas propostas

Nesta seção, são apresentadas as heurísticas propostas neste trabalho. Na subseção 3.2.1, apresenta-se a heurística por busca em largura iniciada por um vértice pseudo-periférico pelo algoritmo de George e Liu (1979) (BeL-GL). A proposta de modificação da heurística WBRA é mostrada na subseção 3.2.2.

3.2.1 Heurística BeL-GL

A heurística por busca em largura iniciando-se por um vértice pseudo-periférico pelo algoritmo de George e Liu (1979), denominada de BeL-GL, foi proposta neste trabalho.

Na revisão sistemática realizada em Oliveira e Chagas (2015a), a heurística de Boutora et al. (2007) foi selecionada como uma das melhores heurísticas na relação benefício (redução da largura de banda) por custo computacional. Porém, a heurística de Boutora et al. (2007) é específica para malhas triangulares. Com isso, essa heurística não funcionaria, por exemplo, na maioria das instâncias da base *Harwell-Boeing*. A heurística de Boutora et al. (2007) pode ser considerada como uma busca em largura modificada para malhas triangulares. Então, decidiu-se utilizar a busca em largura nos testes. Com a busca em largura, é possível realizar a renumeração em qualquer instância. Além disso, utilizou-se o algoritmo de George e Liu (1979) para se encontrar um vértice pseudo-periférico para a busca em largura. Com isso, a heurística BeL-GL é uma heurística com baixo custo computacional em relação as outras heurísticas testadas. Além disso, a heurística BeL-GL gera bons resultados em relação às reduções de largura de banda e de *profile*.

3.2.2 Heurística WBRA-GL

A heurística WBRA-GL, proposta neste trabalho, é uma modificação da heurística WBRA. Como descrito na subseção 2.5.1, na página 39, a heurística WBRA gera as estruturas de nível enraizadas de todos os vértices de V . Em seguida, são reduzidas as larguras de nível dessas estruturas pelo método *PushUp*. Com isso, as estruturas de nível que possuem a largura de nível menor que a largura de banda inicial são armazenadas. Dependendo da instância, pode ser que sejam armazenadas as estruturas de nível de todos os vértices do grafo. Claramente, isso pode resultar em muita ocupação de memória. Também, há grande custo computacional para a geração das estruturas de nível enraizadas de todos os vértices do grafo.

Esposito et al. (1998) realizaram testes com instâncias de até 2.000 vértices. Como, neste trabalho, foram realizados testes com instâncias de até 1.015.004 vértices, tornou-se impraticável o uso da heurística WBRA. Com isso, foi proposta uma modificação na heurística WBRA para torná-la mais rápida e diminuir seu consumo de memória.

A diferença entre as heurísticas WBRA e WBRA-GL está na primeira etapa dessas heurísticas. Na heurística WBRA-GL, em vez de serem geradas as estruturas de nível de todos os vértices do grafo, é gerada apenas a estrutura de nível enraizada de um vértice pseudo-periférico. Especificamente, utilizou-se o algoritmo de George e Liu (1979) para encontrar um vértice pseudo-periférico. Em seguida, utiliza-se, também, o método *PushUp* para se reduzir a largura de nível da estrutura de nível enraizada de um vértice pseudo-periférico. Essa estrutura de nível com largura de nível reduzida é armazenada. Posteriormente, os mesmos passos da heurística WBRA são executados, com a diferença que a renumeração

será realizada somente em uma estrutura de nível.

Com essas modificações, a heurística WBRA-GL realiza as renumerações dos vértices com custo computacional bem menor que a heurística WBRA e com reduções de largura de banda e de *profile* equiparáveis às reduções obtidas pela heurística WBRA. Além disso, o consumo de memória da heurística WBRA-GL é expressivamente menor que o consumo de memória da heurística WBRA.

3.3 Instâncias utilizadas

Nesta seção, são apresentadas as instâncias de matrizes e de SELs utilizadas neste trabalho. As instâncias de matrizes utilizadas nos testes para avaliar as reduções de largura de banda e de *profile* pelas heurísticas, são detalhadas na subseção 3.3.1. Para os testes com a resolução de SELs, utilizou-se três conjuntos de instâncias. O primeiro conjunto, é composto de SELs oriundos da discretização da condução do calor e é detalhado na subseção 3.3.2. No segundo e terceiro conjuntos, foram utilizados SELs oriundos da discretização da equação de Laplace e esses conjuntos são apresentados na subseção 3.3.3.

3.3.1 Instâncias da Harwell-Boeing

Para a avaliação das reduções de largura de banda e de *profile* pelas heurísticas, foram utilizadas 113 matrizes da base Harwell-Boeing (DUFF; GRIMES; LEWIS, 1992). Esse conjunto de matrizes tem sido utilizado por muitos autores que testam heurísticas para a redução de largura de banda de matrizes. Por exemplo, Lim, Rodrigues e Xiao (2007), Rodriguez-Tello, Kao e Torres-Jimenez (2008) e Mladenovic et al. (2010), realizaram testes

com essas 113 matrizes

Como pode ser visto no Capítulo 4, dividiu-se as 113 matrizes em quatro subconjuntos. As 33 matrizes muito pequenas possuem, desconsiderando linhas sem coeficientes, até 200 linhas. As 80 matrizes pequenas possuem de 201 a 992 linhas, desconsiderando linhas sem coeficientes. Como é explicado na subseção 3.4.2, nem todas essas matrizes são simétricas. Com isso, decidiu-se dividir o conjunto de matrizes, também, em matrizes simétricas e assimétricas. Então, têm-se os quatro subconjuntos de matrizes: 15 matrizes muito pequenas simétricas, 18 matrizes muito pequenas assimétricas, 35 matrizes pequenas simétricas e 45 matrizes pequenas assimétricas.

3.3.2 Instâncias oriundas da discretização da condução do calor

Os SELs oriundos da discretização da condução do calor foram gerados pelo projeto computacional de Oliveira (2014a). Nesse projeto computacional, utilizou-se as condições de contorno de Dirichlet e o domínio discretizado foi um quadrado unitário, com os valores das condições de contorno nos lados superior, esquerdo e direito iguais a dez e, no lado inferior, igual a zero. Ainda, Oliveira (2014a) utilizou o método dos volumes finitos para a discretização da equação da condução do calor, com malhas geradas pelo refinamento de *Delaunay* e diagramas de *Voronoi*. A equação da condução do calor é uma equação parcial parabólica e é definida por

$$\frac{\partial}{\partial t}(\phi\rho) = \nabla \cdot \left(\frac{k\nabla}{c_p} \phi \right),$$

em que t é o tempo, ρ é a massa volumétrica de um corpo sólido, ϕ é a temperatura, k é a condutibilidade térmica e c_p é o calor específico.

Foram utilizadas oito SELs gerados pelo projeto computacional de Oliveira (2014a). Essas instâncias de SELs possuem de 4.816 a 965.545 incógnitas.

3.3.3 Instâncias oriundas da discretização da equação de Laplace

Os SELs oriundos da discretização da equação de Laplace foram gerados pelo projeto computacional de Oliveira, Kischinhevsky e Tavares (2013). Assim como os SELs oriundos da discretização da condução do calor, a discretização da equação de Laplace também foi realizada pelo método dos volumes finitos. A equação de Laplace é uma equação diferencial parcial elíptica e é definida por

$$\nabla^2 \phi = 0 \text{ em } \Omega \in \mathbb{R}^2, \phi = f \text{ em } \partial\Omega,$$

em que ϕ é a variável dependente da equação diferencial parcial, Ω é um domínio limitado em \mathbb{R}^2 e f é uma função suave definida no contorno $\delta\Omega$. Oliveira, Kischinhevsky e Tavares (2013) utilizaram o quadrado unitário como domínio. Foram utilizadas as condições de contorno definidas nos lados superior, inferior e esquerdo com um valor definido único f e no lado direito com um valor diferente.

As instâncias de SELs oriundas da discretização da equação de Laplace, podem ser divididas em dois conjuntos. No primeiro conjunto, têm-se os SELs gerados na ordem curva de Sierpiński modificada (OLIVEIRA; KISCHINHEVSKY, 2008). Esse conjunto possui oito SELs com 16.922 a 1.015.004 incógnitas. No segundo conjunto, têm-se os SELs gerados de forma aleatória. Esse conjunto possui sete SELs com 7.322 a 277.118 incógnitas.

3.4 Detalhes de Implementação

Nesta seção, são apresentados os detalhes das implementações realizadas. Para o desenvolvimento deste trabalho, implementou-se um projeto computacional denominado *LinearSystemSolver* que é abordado, em detalhes, na subseção 3.4.1.

Embora os códigos de todas as heurísticas tenham sido pedidos para os respectivos autores, apenas em duas heurísticas obteve-se resposta. São os casos das heurísticas FNCHC de Lim, Rodrigues e Xiao (2007) e VNS-*band* de Mladenovic et al. (2010). Especificamente, o professor Dr. Fei Xiao enviou o código da heurística FNCHC. Com isso, algumas adaptações tiveram que ser realizadas no código da heurística FNCHC para incorporá-la ao projeto computacional *LinearSystemSolver*.

No caso da heurística VNS-*band*, o professor Dr. Dragan Urosevic enviou o executável dessa heurística. Adaptações foram realizadas para incorporar o executável da heurística VNS-*band* no projeto computacional. Esse executável foi compilado em um sistema operacional Linux de 64 *bits*. Ainda, esse executável aceita, como entrada, instâncias de matrizes $n \times n$, com $n \leq 500.000$, no formato das matrizes da Harwell-Boeing.

As heurísticas CMr-GL (GEORGE; LIU, 1981), GPS de Gibbs, Poole e Stockmeyer (1976), BL de Burgess e Lai (1986), WBRA de Esposito et al. (1998), GGPS de Wang, Guo e Shi (2009), hGPHH de Koohestani e Poli (2011) e CSS-*band* de Kaveh e Sharafi (2012) foram implementadas. Vale ressaltar que, em alguns casos, as descrições das heurísticas, nos respectivos artigos, são incompletas e/ou ambíguas e tentou-se ser o mais fiel possível às descrições.

Implementou-se, também, as heurísticas BeL-GL e WBRA-GL,

propostas neste trabalho. Essas heurísticas são descritas nas subseções 3.2.1 e 3.2.2 respectivamente. Ainda, implementou-se o algoritmo de George e Liu (1979) e a fatoração incompleta de Cholesky com zero *fill-in* (IC0).

Utilizou-se o código do MGC preconditionado pelo método de Jacobi de Oliveira (2014b). Nesse código, foi utilizada a precisão numérica do MGC preconditionado de 10^{-16} . Esse código foi adaptado para o MGC *split* nas estruturas CSR-SSS/CSC-SSS, conforme é explicado na subseção 3.4.1. Além disso, foi implementada a inversão de matrizes triangulares inferiores por substituição avançada múltipla. Porém, por causa do grande custo computacional envolvido na inversão de matrizes, adotou-se a abordagem detalhada na subseção 3.4.3.

Esta seção está dividida como a seguir. Na subseção 3.4.1, detalhes do projeto computacional desenvolvido neste trabalho são explicados. Descreve-se, na subseção 3.4.2, a abordagem utilizada para transformar as matrizes assimétricas da base Harwell-Boeing em simétricas. Apresenta-se, na subseção 3.4.3, como se evitou a inversão de matrizes no MGC preconditionado à esquerda e no MGC *split*.

3.4.1 Projeto computacional

Todas as heurísticas e métodos implementados compõem um único projeto computacional. Ao todo, foram implementadas 28 classes totalizando cerca de 26.000 linhas de código. O fluxograma simplificado desse projeto computacional, desenvolvido neste trabalho, é mostrado na Figura 12.

O projeto computacional lê o arquivo de configuração de entrada e é identificado qual tipo de teste deverá ser realizado e com qual

instância. Caso a instância seja da base Harwell-Boeing gera-se o grafo na estrutura utilizada pelas heurísticas. A estrutura de dados utilizadas pelas heurísticas é baseada em lista de adjacências utilizando-se a classe *vector* (<http://www.cplusplus.com/reference/vector/vector/>) do c++. Em seguida, executa-se a heurística para a redução de largura de banda e os resultados, como largura de banda, *profile* e tempo de execução, são gravados em arquivo.

Se a instância passada como entrada é de SEL, então, gera-se um grafo baseado em listas de adjacências. Caso a simulação definida no arquivo de entrada não seja com redução de largura de banda, então, o SEL é resolvido pelo MGC preconditionado pelo método escolhido, isto é, IC0 ou Jacobi. Se o preconditionador escolhido é o IC0, então, é necessário gerar o SEL na estrutura CSR-SSS/CSC-SSS. Isso porque, a implementação do IC0 em uma estrutura de dados baseada em listas de adjacências é ineficiente (ABREU, 2014). Posteriormente, passa-se essa estrutura para o MGC-IC0 (MGC *split*) que resolve o SEL e grava os resultados em arquivo. Em ocorrendo de o preconditionador de Jacobi ser escolhido, nenhuma estrutura extra é gerada, pois a estrutura de dados do MGC preconditionado por Jacobi (MGCPJ) é baseada em lista de adjacências. Posteriormente, os resultados são salvos em arquivo.

Caso seja escolhida a opção de resolução de SEL com a redução de largura de banda, então, um grafo na estrutura das heurísticas deve ser gerado. Com isso, a redução de largura de banda é realizada pela heurística determinada no arquivo de configuração. Em seguida, a renumeração obtida pela heurística é passada do grafo baseado em lista de adjacências por *vector* para o grafo baseado em lista de adjacências. Posteriormente, o SEL,

3.4.2 Instâncias assimétricas da Harwell-Boeing

Nos testes realizados, utilizou-se 113 matrizes da base Harwell-Boeing (DUFF; GRIMES; LEWIS, 1992) para avaliar as reduções de largura de banda e de *profile* das heurísticas. Essas matrizes são oriundas das mais diversas aplicações científicas e, conseqüentemente, nem todas as matrizes são simétricas. Infelizmente, nem todas as heurísticas podem ser utilizadas com matrizes assimétricas. Esse é o caso das heurísticas BL, GPS, GGPS, WBRA e WBRA-GL, que são específicas para matrizes simétricas. As heurísticas CSS-*band* FNCHC e VNS-*band*, foram projetadas para funcionar em matrizes simétricas e assimétricas. Ainda, as heurísticas Bel-GL, CMr-GL e hGPHH foram facilmente adaptadas para funcionarem em matrizes assimétricas sem ser necessário mexer na estrutura dessas heurísticas. Então, uma abordagem precisou ser adotada para tornar possível a utilização das heurísticas BL, GPS, GGPS, WBRA e WBRA-GL em matrizes assimétricas.

Reid e Scott (2005) apresentaram três técnicas para adaptar matrizes assimétricas para heurísticas para a redução de largura de banda. A primeira técnica é utilizar um grafo bipartido, em que há $2 \cdot |V|$ vértices. Isso pode não ser uma boa alternativa, pois dobra o custo de armazenamento. A segunda técnica é utilizar um grafo denominado de grafo linha. Seja B , uma matriz assimétrica, esse grafo consiste em ser uma representação da matriz BB^T . Claramente, essa não é uma boa opção, pois o custo para computar uma multiplicação de matrizes é alto. A última técnica consiste em transformar a matriz assimétrica em uma matriz simétrica por somar a matriz original com sua transposta, ou seja, $B + B^T$. Essa última técnica foi adotada neste trabalho.

O custo computacional para realizar a soma de duas matrizes é, relativamente, baixo. Como as matrizes são representadas por grafos, para fazer a soma $B + B^T$, considerando $v, u \in V$, basta criar arestas $\{u, v\}$ para todos os vértices $v, u \in V$, em que $\{v, u\} \in E$ e $\{u, v\} \notin E$.

Depois de ter transformado o grafo assimétrico em simétrico, a renumeração é realizada pelas heurísticas nesse grafo simétrico. Posteriormente, o grafo é transformado em sua forma original, isto é, assimétrico. Isso pode ser feito, com facilidade, ao se remover as arestas inseridas. Finalmente, a largura de banda e *profile* são calculados.

3.4.3 Inversão de matrizes

Como pode ser notado nos pseudo-códigos das Figuras 9 e 10, nas páginas 61 e 62, respectivamente, são utilizadas as inversas das matrizes condicionadoras M , L e L^T . Porém, exige-se alto custo computacional para se calcular a inversa de uma matriz. Por exemplo, Cormen et al. (2009, p. 827-828) apresentam um algoritmo $\Theta(n^3)$ para inversão de matrizes baseado na decomposição LUP.

Em Cormen et al. (2009), nas páginas 828 a 831, é mostrado que a inversão de matrizes é equivalente a multiplicação de matrizes. Com isso, algoritmos atuais para inversão de matrizes são, originalmente, propostos para multiplicação de matrizes quadradas. A maioria desses algoritmos são variações do algoritmo de Coppersmith e Winograd (1990), um algoritmo com custo $O(n^{2,376})$. Como exemplo, tem-se o algoritmo de Williams (2012) com complexidade $O(n^{2,373})$. Infelizmente, algoritmos com esses custos são impeditivos. Isso porque pode ser que o custo computacional da inversão da matriz condicionadora seja maior que o custo computacional na resolução

do SEL pelo MGC. Então, deve-se evitar a inversão de matrizes.

A matriz preconditionadora gerada pelo método de Jacobi, apresentado na subseção 2.13, na página 63, é uma matriz diagonal. O cálculo da inversa de uma matriz diagonal é trivial. Para isso, calcula-se o inverso da diagonal principal de A . Então, implementou-se o MGC preconditionado à esquerda ao custo de uma multiplicação vetor-vetor por iteração.

Em relação ao MGC preconditionado por *split*, mostrado na Figura 10, na página 62, não se realizou a inversa das matrizes L e L^T de forma explícita. Note que, no algoritmo da Figura 10, as matrizes L^{-1} e L^{-T} são utilizadas nas linhas 5, 6, 11 e 15. Em vez disso, utilizou-se algebrismo para evitar o cálculo das inversas das matrizes.

Para a linha 5, tem-se $z_0 \leftarrow L^{-1}r_0$, então considere que

$$y = L^{-1}r_0 \quad (10)$$

multiplicando-se a equação 10, à esquerda, por L dos dois lados da igualdade, tem-se

$$Ly = \cancel{L}L^{-1}r_0 = r_0,$$

logo,

$$Ly = r_0. \quad (11)$$

Com isso, em vez de se calcular a inversa da matriz triangular inferior L^{-1} , resolve-se o SEL mostrado na equação 11. Como a matriz L é uma matriz triangular inferior, o SEL mostrado em 11 pode ser resolvido diretamente por substituição avançada. É fácil verificar que, no pior caso, ou seja, quando a matriz L é densa, o algoritmo de substituição avançada

ocorre em $O(n^2)$. Como as matrizes utilizadas neste trabalho são esparsas, espera-se um comportamento linear do algoritmo de substituição avançada. Claramente, o custo computacional de se resolver o SEL mostrado na equação 11 é menor que o custo computacional para se inverter a matriz L . Então, substitui-se a linha 5, do algoritmo da Figura 10, por $z_0 \leftarrow y$, em que y é a solução do SEL $Ly = r_0$.

A mesma ideia pode ser aplicada às linhas 6, 11 e 15 do algoritmo da Figura 10. Porém, nas linhas 6 e 15, é utilizada a matriz L^T , que é triangular superior. Com isso, utiliza-se a substituição regressiva.

A linha 6, $d_0 \leftarrow L^{-T}z_0$, pode ser substituída por

$$d_0 = w, L^T w = z_0.$$

Tem-se, na linha 11, a seguinte atribuição

$$z_k \leftarrow z_{k-1} - \alpha_k L^{-1} A d_k.$$

Essa atribuição pode ser substituída por

$$z_k \leftarrow z_{k-1} - \alpha_k c,$$

em que,

$$Lc = t \text{ e } t = Ad_k.$$

Já a atribuição da linha 15, representada por

$$d_{k+1} \leftarrow L^{-T}z_k + \beta_k d_k,$$

pode ser substituída por

$$d_{k+1} \leftarrow u + \beta_k d_k, L^T u = z_k.$$

Com essas modificações, evita-se o cálculo da inversa das matrizes L e L^T .

3.5 Ferramentas utilizadas

Utilizou-se a IDE *Netbeans* 8.0.2 para auxiliar na escrita dos códigos em linguagem *C++*. O compilador utilizado foi o *G++* 4.8.2. Além disso, utilizou-se a biblioteca *GNU Multiple Precision Floating-point computations with correct-Rounding* (MPFR) para se obter maior precisão nos cálculos na resolução dos SELs.

Para os testes computacionais realizados nas instâncias da *Harwell-Boeing*, utilizou-se um computador com processador Intel® Core™ i3-2120 CPU 3,30GHz \times 4, com 3 MB de memória *cache* e com 8 GB de memória RAM DDR3 1,333GHz. Esse computador também foi utilizado nas simulações envolvendo a resolução dos SELs provenientes da discretização por MVF da equação de Laplace numerados de forma aleatória. Para essas simulações, utilizou-se o sistema operacional Ubuntu 14.04 LTS 64 *bits*, *kernel* 3.13.0-39-*generic* e precisão de 256 *bits* na biblioteca *GNU MPFR*.

Para os testes da heurística *VNS-Band* com a instância SHERMAN4, da base *Harwell-Boeing*, utilizou-se uma máquina com sistema operacional de 32 *bits*. Isso porque o executável da heurística *VNS-Band* utilizado em todas as simulações deste trabalho é 64 *bits* e não funciona corretamente nessa instância. Com isso, utilizou-se uma versão da heurística *VNS-Band* 32 *bits*, disponibilizada por Mladenovic et al. (2010), que funciona corretamente na instância SHERMAN4. Por isso, utilizou-se

um computador com processador Intel® Core™ i5-3337U CPU 1,80GHz \times 4, com 3 MB de memória *cache*, com 6 GB de memória RAM DDR3 1,333GHz e com sistema operacional Ubuntu 12.04 LTS 32 *bits* e *kernel* 3.2.0-83-*generic*.

Para os testes computacionais realizados na resolução dos SELs provenientes da discretização por MVF da equação de Laplace numerados pela curva de Sierpiński modificada (OLIVEIRA; KISCHINHEVSKY, 2008), utilizou-se um computador servidor IBM® com processador Intel® Xeon™ E5620 2,40GHz \times 8 com 12 MB de memória *cache* e 24 GB de memória RAM DDR3 1,333GHz. Para essas simulações, utilizou-se o sistema operacional Ubuntu Server 14.04 LTS 64 *bits*, *kernel* 3.13.0-44-*generic* e precisão de 256 *bits* na biblioteca *GNU MPFR*.

Para os testes computacionais realizados na resolução dos SELs provenientes da discretização por MVF da equação da condução do calor, utilizou-se um computador com processador Intel® Core™ i3 CPU 550 3,20GHz \times 4, com 4 MB de memória *cache* e 16 GB de memória RAM DDR3 1,333GHz. Para essas simulações, utilizou-se o sistema operacional Ubuntu 14.04 LTS 64 *bits*, *kernel* 3.13.0-39-*generic* e precisão de 512 *bits* na biblioteca *GNU MPFR*.

4 SIMULAÇÕES COM INSTÂNCIAS DA BASE *HARWELL-BOEING*

Neste Capítulo, apresentam-se os resultados das heurísticas para a redução de largura de banda nas 113 matrizes da base Harwell-Boeing (DUFF; GRIMES; LEWIS, 1992). Essas 113 matrizes foram divididas em quatro conjuntos. Todas as 11 heurísticas para a redução de largura de banda foram testadas nas 113 matrizes. Resultados em relação à largura de banda, ao *profile* e ao custo computacional, são apresentados. As tabelas de desvio padrão e coeficiente de variação dos custos computacionais das heurísticas, em cada conjunto, são apresentadas no apêndice A.

A heurística *VNS-band* é, provavelmente, a heurística para a redução de largura de banda no estado da arte (MLADENOVIC et al., 2010). Mladenovic et al. (2010) utilizaram, como critério de parada, 500s em todas as instâncias utilizadas nos testes. Porém, como o intuito deste trabalho é identificar a melhor heurística para a redução de largura de banda com custo computacional razoável, utilizou-se tempos diferentes para cada conjunto de matrizes. O tempo de 500s foi utilizado como tempo máximo de execução, ou seja, a heurística *VNS-band* não foi testada com mais de 500s. O objetivo foi encontrar o menor tempo em que a heurística *VNS-band* obtivesse os melhores resultados. Esses tempos foram definidos de forma empírica.

Nas tabelas presentes neste Capítulo, a dimensão da matriz é representada por n , a largura de banda original da matriz por β_0 e o *profile* original da matriz por $profile_0$. No final de cada tabela mostra-se o número de vezes em que cada heurística obteve o melhor resultado. Ainda, no final das tabelas dos resultados referentes à largura de banda e ao *profile*, há o somatório do escore padronizado (*Z-score*), representado por $\sum \frac{x-\bar{x}}{\sigma}$, em que

x é o valor do dado, \bar{x} é a média da população e σ é o desvio padrão. O escore padronizado representa o quanto uma medida se afasta da média em termos de desvios padrões. No caso das reduções de largura de banda e de *profile*, quanto menor esse valor é, melhores são os resultados da heurística. Quanto maior os resultados são, piores são os resultados.

Todas as heurísticas foram executadas 10 vezes. Os resultados, médios, em relação às reduções de largura de banda e de *profile* e os resultados em relação aos custos computacionais são apresentados nas respectivas tabelas das subseções seguintes. Apresentam-se, os coeficientes de variação e os desvios padrões, referentes às 10 execuções no apêndice.

Para a heurística *CSS-Band*, utilizaram-se 500 iterações como critério de parada e uma partícula carregada a cada 100 vértices em todas as 113 instâncias. Esses valores foram os mesmos utilizados por Kaveh e Sharafi (2012).

Este Capítulo é dividido em quatro seções. Na seção 4.1, apresentam-se os resultados das 11 heurísticas nas 15 instâncias simétricas muito pequenas da base Harwell-Boeing. Os resultados das simulações com as 11 heurísticas nas 35 instâncias pequenas da Harwell-Boeing são apresentados na seção 4.2. Mostram-se, na seção 4.3, os resultados referentes às 11 heurísticas nas 18 instâncias assimétricas muito pequenas. Na seção 4.4, são apresentados os resultados das 11 heurísticas nas 45 instâncias assimétricas pequenas. A análise dos resultados é feita na seção 4.5.

4.1 Instâncias simétricas muito pequenas

Nesta subseção apresentam-se os resultados obtidos pelas heurísticas nos testes computacionais nas 15 instâncias simétricas muito pequenas da

base Harwell-Boeing. Essas instâncias possuem de 39 a 237 vértices.

A Tabela 1 é dividida em três partes. Na primeira parte, têm-se os resultados das heurísticas em relação à largura de banda. Na segunda parte, mostram-se os resultados das heurísticas em relação ao *profile*. Finalmente, na terceira parte, têm-se os resultados das heurísticas em relação ao custo computacional.

Para a heurística *VNS-Band*, foram utilizados 500s de tempo máximo de execução em todas as 15 instâncias nos testes para as reduções de largura de banda e de *profile*. Utilizou-se 500s para mostrar que a heurística *VNS-band*, nem com o tempo máximo estabelecido neste trabalho, obteve resultados melhores, nas reduções de largura de banda e de *profile*, que a heurística FNCHC.

Observa-se, na primeira parte da Tabela 1, que a heurística FNCHC obteve os melhores resultados na redução de largura de banda. A heurística FNCHC obteve, em 12 das 15 instâncias, as melhores largura de banda. Também, o menor somatório do escore padronizado foi obtido pela heurística FNCHC. Embora a heurística *VNS-band* tenha ficado executando por 500s, os resultados dessa heurística não foram superiores aos resultados da heurística FNCHC. Em relação ao *profile*, as heurísticas que mais vezes obtiveram os melhores resultados foram as heurísticas CMr-GL e BeL-GL. A heurística FNCHC foi a heurística que apresentou o menor somatório de escore padronizado. Com isso, pode-se afirmar que a melhor heurística na redução de *profile* foi a heurística FNCHC, pois foi a heurística com resultados mais regulares na redução de *profile*. A heurística com menor custo computacional foi a heurística BeL-GL. Claramente, a heurística mais lenta, foi a heurística *VNS-band*. Observa-se, ainda, que os piores resultados

Tabela 1 Resultados de cada heurística em relação à largura de banda, ao *profile* e ao tempo de execução das simulações nas 15 instâncias simétricas muito pequenas da base Harwell-Boeing.

Instância	n	β											
		CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band	
ASH85	85	39	10	11	10	15	14	9	10	10	15	10	76
BCSPWR01	39	38	8	8	8	8	7	5	9	8	9	6	31
BCSPWR02	49	34	13	13	13	12	10	7	10	14	12	11	43
BCSPWR03	118	115	17	19	19	21	10	11	15	19	23	19	104
BCSSTK01	48	35	27	27	27	35	24	16	26	26	33	24	42
BCSSTK04	132	47	54	54	54	47	37	37	46	54	57	47	127
BCSSTK05	153	28	26	32	24	28	20	20	23	28	32	28	146
BCSSTK22	138	111	15	13	14	13	10	10	14	14	16	12	125
CAN_144	144	142	18	19	15	20	13	13	15	14	22	19	134
CAN_161	161	79	30	32	30	46	18	19	18	18	30	24	152
DWT_234	234	48	19	26	17	21	12	11	17	24	23	17	212
LUND_A	147	23	23	23	23	23	23	23	23	23	28	23	140
LUND_B	147	23	23	23	23	23	23	23	23	23	31	23	140
NOS1	237	4	4	4	4	4	3	4	3	4	4	4	220
NOS4	100	13	12	16	13	13	10	10	13	12	17	11	91
Melhores resultados		2	2	2	2	2	10	12	4	3	0	2	0
$\bar{y} \frac{s-\bar{x}}{\bar{y}}$		-4,04	-3,41	-4,13	-2,21	-6,80	-8,86	-5,33	-4,33	-1,43	-5,48	46,02	

Instância	n	profile ₀	<i>profile</i>										
			CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
ASH85	85	1153	589	596	608	715	686	593	589	638	603	619	2434
BCSPWR01	39	292	123	122	171	182	159	126	123	123	121	138	367
BCSPWR02	49	377	214	200	344	330	306	224	212	356	223	296	565
BCSPWR03	118	1288	774	751	1150	1232	740	841	1073	673	736	1128	4063
BCSSTK01	48	851	654	661	749	851	759	554	676	676	618	729	923
BCSSTK04	132	3631	3585	3515	4079	3631	3576	3555	3493	3555	3398	3631	8077
BCSSTK05	153	2449	2313	2454	2540	2449	2353	2360	2465	2414	2360	2449	10002
BCSSTK22	138	2124	891	855	948	898	883	882	870	873	878	968	5820
CAN_144	144	7355	1071	1063	1572	1848	1295	1321	1096	1308	1323	1859	8012
CAN_161	161	3378	3079	3151	3233	3558	2610	2670	2610	2610	2753	2796	10199
DWT_234	234	1765	1363	1413	1801	1900	1460	1472	1274	1295	1155	1838	14372
LUND_A	147	2870	2303	2303	2303	2870	2471	2507	2303	2303	2363	2870	9510
LUND_B	147	2870	2229	2300	2299	2870	2532	2492	2229	2303	2366	2870	9346
NOS1	237	780	467	467	472	624	468	507	468	467	469	624	16663
NOS4	100	766	761	917	892	766	742	781	936	900	852	845	3413
Melhores resultados		5	5	2	0	2	2	1	4	4	3	0	0
$\bar{y} \frac{s-\bar{x}}{\bar{y}}$		-5,99	-6,02	-1,61	0,23	-3,20	-6,57	-5,72	-4,45	-6,42	-2,58	42,32	

Instância	Tempo (s)										
	CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
ASH85	0.00008	0.00002	0.00008	0.00042	503.84300	0.02657	0.00022	0.00038	0.00025	0.00360	0.17278
BCSPWR01	0.00002	0.00001	0.00003	0.00016	527.31500	0.00745	0.00008	0.00011	0.00012	0.00070	0.10229
BCSPWR02	0.00002	0.00002	0.00004	0.00023	635.63000	0.00900	0.00011	0.00015	0.00019	0.00307	0.11034
BCSPWR03	0.00008	0.00003	0.00011	0.00050	500.57900	0.02613	0.00034	0.00061	0.00089	0.02567	0.26830
BCSSTK01	0.00005	0.00001	0.00005	0.00026	621.28400	0.02829	0.00019	0.00020	0.00022	0.00880	0.11412
BCSSTK04	0.00041	0.00006	0.00036	0.00144	500.28800	0.10722	0.00178	0.01205	0.00186	0.02050	0.30567
BCSSTK05	0.00029	0.00007	0.00029	0.00105	500.63300	0.06697	0.00134	0.00955	0.00138	0.03248	0.32391
BCSSTK22	0.00011	0.00003	0.00010	0.00051	500.37100	0.02796	0.00063	0.00178	0.00056	0.02015	0.27752
CAN_144	0.00018	0.00004	0.00014	0.00089	500.47800	0.03813	0.00059	0.00119	0.00063	0.08453	0.42721
CAN_161	0.00014	0.00003	0.00013	0.00080	500.43200	0.07352	0.00076	0.00170	0.00145	0.01367	0.41429
DWT_234	0.00010	0.00003	0.00014	0.00088	500.22600	0.04197	0.00079	0.00142	0.00133	0.01059	0.59239
LUND_A	0.00028	0.00005	0.00021	0.00078	501.48600	0.07169	0.00171	0.00627	0.00059	0.02714	0.42032
LUND_B	0.00030	0.00006	0.00021	0.00078	501.53500	0.07329	0.00118	0.00620	0.00069	0.02720	0.39847
NOS1	0.00015	0.00002	0.00013	0.00068	500.41400	0.02006	0.00118	0.00277	0.00043	0.00867	0.44627
NOS4	0.00009	0.00002	0.00007	0.00044	500.51900	0.03208	0.00027	0.00035	0.00041	0.00392	0.31988
Melhores resultados	1	15	0	0	0	0	0	0	0	0	0

em relação à largura de banda e ao *profile* foram obtidos pela heurística *CSS-band*.

4.2 Instâncias simétricas pequenas

Nesta subseção, apresentam-se os resultados obtidos pelas heurísticas nos testes computacionais nas 35 instâncias simétricas pequenas da base Harwell-Boeing. Essas instâncias possuem de 209 a 1.000 vértices.

A Tabela 2 é dividida em duas partes. Na primeira parte, são mostrados os resultados das heurísticas em relação à largura de banda. Na segunda parte, mostram-se os resultados das heurísticas em relação ao *profile*. Os resultados em relação ao custo computacional, de cada heurística, são mostrados na Tabela 3.

Para a heurística *VNS-Band*, foram utilizados 8s de tempo máximo de execução em todas as 35 instâncias nos testes para a redução de largura de banda. Para os testes com a redução de *profile*, utilizou-se 12s como critério de parada para a heurística *VNS-band*. Esses foram os menores tempos encontradas para que a heurística *VNS-band* obtivesse os melhores resultados nas reduções de largura de banda e de *profile*. Esses tempos foram encontrados empiricamente.

Em relação aos resultados na redução de largura de banda, mostrados na primeira parte da Tabela 2, a heurística que mais vezes obteve o melhor resultado foi a heurística *VNS-band* com 8s como critério de parada. Também, a heurística *VNS-band* foi a heurística que obteve o menor somatório de escore padronizado entre as heurísticas testadas. Então, a heurística *VNS-band* foi considerada a melhor heurística para a redução de largura de banda nas instâncias simétricas pequenas. Ainda, a heurística FNCHC foi a segunda heurística que mais vezes obteve o melhor resultado e apresentou, também, o segundo menor somatório de escore padronizado.

Na redução de *profile*, a heurística que obteve, mais vezes, o melhor resultado foi a heurística GPS. Porém, a heurística que obteve o menor somatório de escore padronizado foi a heurística *VNS-band* com 12s como critério de parada. Vale destacar que a heurística FNCHC, embora não tenha obtido o melhor *profile* em nenhuma instância, foi a heurística que

Tabela 2 Resultados de cada heurística em relação à largura de banda e ao *profile* das simulações nas 35 instâncias simétricas pequenas da base Harwell-Boeing.

Instância	n	β											
		CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (8s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band	
494_BUS	494	428	62	60	87	137	35	41	56	67	107	56	458
662_BUS	662	335	135	139	125	154	51	53	85	88	101	120	622
685_BUS	685	550	102	100	82	128	39	44	84	72	115	57	647
ASH292	292	24	28	29	34	24	19	21	34	25	53	24	278
BCSPWR04	274	265	65	66	77	95	25	26	48	66	72	44	255
BKSPWR05	443	435	59	59	64	59	29	36	49	47	65	50	415
BKSTK06	420	47	50	49	49	47	45	47	47	52	81	47	409
BKSTK19	817	567	22	22	22	28	15	17	18	22	22	22	790
BKSTK20	485	20	20	22	20	20	13	16	19	20	25	18	467
BKSTM07	420	47	54	59	52	47	45	47	50	54	81	47	408
CAN_292	292	282	74	68	67	99	39	44	78	74	121	66	276
CAN_445	445	403	89	87	94	169	58	62	88	85	109	105	426
CAN_715	715	611	133	142	133	157	108	90	126	133	126	121	691
CAN_838	838	837	124	135	138	175	91	94	136	120	217	128	816
DWT_209	209	184	37	39	34	51	23	24	37	30	39	37	197
DWT_221	221	187	16	18	17	22	13	13	19	16	22	19	207
DWT_245	245	115	43	51	55	58	22	22	50	43	60	44	230
DWT_310	310	28	15	16	13	19	12	12	13	13	20	17	296
DWT_361	361	50	25	25	25	19	16	16	14	14	24	19	344
DWT_419	419	356	34	33	33	50	28	28	34	33	47	44	402
DWT_503	503	452	64	63	64	63	48	47	64	64	88	54	487
DWT_592	592	259	42	42	42	56	33	31	36	42	65	52	573
DWT_878	878	519	37	37	37	54	27	37	27	37	70	49	850
DWT_918	918	839	57	54	63	89	38	41	58	52	73	51	885
DWT_992	992	513	65	67	63	117	73	52	36	63	63	57	969
GR_30_30	900	31	59	60	59	31	59	47	49	49	101	31	872
JAGMESH1	936	778	27	27	28	43	31	28	27	27	48	43	903
NOS2	957	4	4	4	4	4	3	4	3	4	4	4	911
NOS3	960	43	81	81	79	43	79	64	83	65	80	43	939
NOS5	468	178	92	91	88	141	64	67	88	90	134	122	452
NOS6	675	30	17	16	16	21	20	19	16	16	29	21	641
NOS7	729	81	65	65	67	81	81	67	65	65	76	81	701
PLAT362	362	249	52	58	53	58	36	36	52	53	87	45	350
PLSKZ362	362	248	36	36	34	31	20	20	33	37	50	25	341
SHERMAN1	1000	100	61	62	57	56	56	57	57	57	96	55	947
Melhores resultados		2	3	1	2	23	9	7	4	0	3	0	0
$\Sigma \frac{\tau_i}{\sigma_i}$		-10,65	-10,29	-10,21	-6,05	-16,06	-15,95	-11,73	-11,73	-5,22	-11,62	109,51	

Instância	n	profile ₀	<i>profile</i>										
			CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (12s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
494_BUS	494	40975	10566	11671	22601	21390	9118	12201	12569	12048	11215	17776	61693
662_BUS	662	45165	32903	32142	42965	43945	22703	21663	18616	30699	17233	34954	118988
685_BUS	685	28261	25606	26422	25787	28252	17198	20397	17715	16532	17865	25249	145227
ASH292	292	4224	3898	3779	5433	4224	3832	4100	4000	4266	4215	4224	31572
BCSPWR04	274	21015	4825	4884	9327	8429	4039	4482	7345	4909	5269	8250	23186
BCSPWR05	443	36248	8825	4884	16041	14727	8331	10013	10188	8656	7503	14993	52418
BKSTK06	420	14691	13238	13193	15086	14691	14380	15062	13287	13200	15056	14691	76841
BKSTK19	817	74051	9594	9589	10950	11351	8087	10413	8721	9598	8828	10678	255853
BKSTK20	485	4309	5069	5056	5303	4229	3603	4234	4416	4378	4006	3869	81118
BKSTM07	420	14691	13816	13809	15323	14691	14380	15047	13411	13786	14923	14691	75255
CAN_292	292	23170	8589	9232	11636	12720	8852	9313	9414	9113	10161	12402	31862
CAN_445	445	22321	22445	22046	27248	28982	20860	21298	20686	21264	20243	23932	77494
CAN_715	715	72423	38449	39476	47761	49300	40879	43445	40364	38540	36709	57989	200931
CAN_838	838	207200	34295	35841	82204	94541	45971	54838	52107	73168	45165	84109	283028
DWT_209	209	9503	3980	4045	4433	4511	3617	3680	4816	3414	3379	4554	16770
DWT_221	221	9910	2011	2080	2210	2411	2002	2077	1997	2019	2063	2225	18355
DWT_245	245	3934	4620	5004	7270	7769	3523	3558	5054	6309	4091	6988	19668
DWT_310	310	2696	2836	2885	2759	3056	2773	2795	2695	2754	2814	2764	36499
DWT_361	361	5084	5139	5139	5337	5030	5022	4902	4699	4699	4940	5030	50067
DWT_419	419	39726	8221	8156	9447	10092	8674	8936	8276	9684	8572	11172	68002
DWT_503	503	35914	14816	14881	17101	17226	18163	16862	14783	14802	15381	17441	102643
DWT_592	592	28805	10983	11373	12043	12964	13050	12634	11703	10984	11088	20018	136471
DWT_878	878	26055	21034	21044	22026	24141	20318	21543	19120	21161	21856	21903	297043
DWT_918	918	108355	22187	21835	27339	28234	24657	24562	21940	20983	21760	24671	325236
DWT_992	992	262306	37136	37612	37980	38892	43720	36112	33076	36926	36894	36106	434021
GR_30_30	900	26970	33872	34250	35496	26970	33313	31328	27916	29520	36049	26970	312960
JAGMESH1	936	37240	21817	21805	22150	22458	24818	22181	21817	21817	22136	22458	318013
NOS2	957	3180	1907	1907	1912	2544	1908	2082	1908	1907	1909	2544	27961
NOS3	960	39101	47536	46778	48938	39101	47713	44862	42278	41276	40306	39101	405513
NOS5	468	27286	25228	25100	28509	29073	23708	24232	27632	28180	24732	30619	89723
NOS6	675	16229	9305	9305	9357	9523	10587	9821	9305	9305	9301	9523	148014
NOS7	729	53144	34110	34110	34704	53144	49692	35296	34110	34110	34826	53144	190627
PLAT362	362	45261	13366	13786	14363	15110	9948	10184	13366	14936	14062	11765	57533
PLSKZ362	362	43090	6643	6599	7265	5043	5423	5123	6522	6948	6318	5325	45268
SHERMAN1	1000	34740	26673	26757	27054	27330	27421	27431	26109	26997	28167	27536	218119
Melhores resultados		5	6	0	3	7	0	9	5	6	2	0	0
$\Sigma \frac{\tau_i}{\sigma_i}$		-12,75	-12,41	-6,54	-6,13	-13,60	-13,11	-12,27	-11,91	-13,39	-7,35	109,45	

obteve o terceiro menor somatório de escore padronizado. Já a heurística GPS, apresentou o sexto menor somatório de escore padronizado, ficando

atrás das heurísticas *VNS-band*, BL, FNCHC, CMr-GL e BeL-GL. Como a heurística *VNS-band* obteve o menor somatório de escore padronizado, essa heurística foi considerada a melhor heurística na redução de *profile* nas instâncias simétricas pequenas.

Tabela 3 Resultados de cada heurística em relação aos tempos médios, em segundos, de execução das simulações nas 35 instâncias simétricas pequenas da base Harwell-Boeing.

Instância	CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band		FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
					(8s)	(12s)						
494.BUS	0.00030	0.00013	0.00030	0.00278	8.01115	12.01460	0.11930	0.00184	0.00532	0.00871	0.08981	1.59015
662.BUS	0.00049	0.00013	0.00054	0.00418	8.01106	12.01460	0.19709	0.00354	0.00516	0.01813	2.04695	3.35895
685.BUS	0.00065	0.00015	0.00054	0.00373	8.00967	12.00960	0.20994	0.00431	0.00792	0.00878	0.31447	3.23729
ASH292	0.00026	0.00009	0.00031	0.00015	8.01350	12.02710	0.09472	0.00133	0.00193	0.00360	0.01937	0.73219
BCSPWR04	0.00020	0.00006	0.00024	0.00150	8.01258	12.00490	0.08774	0.00093	0.00245	0.00189	0.10597	0.74097
BCSPWR05	0.00022	0.00015	0.00046	0.00216	8.01152	12.02330	0.11853	0.00151	0.00364	0.00594	0.04451	1.31506
BSSSTK06	0.00121	0.00020	0.00080	0.00309	8.01484	12.01660	0.24036	0.00992	0.00514	0.00364	0.25516	1.48727
BSSSTK19	0.00079	0.00021	0.00085	0.00323	8.01750	12.01630	0.18535	0.00941	0.02591	0.00663	0.53780	5.03361
BSSSTK20	0.00039	0.00007	0.00043	0.00182	8.01318	12.01650	0.08821	0.00350	0.00391	0.00714	0.53863	1.66873
BCSSTM07	0.00106	0.00015	0.00073	0.00299	8.01383	12.01140	0.22470	0.00789	0.00338	0.00312	0.24463	1.55799
CAN.292	0.00030	0.00010	0.00087	0.00231	8.01043	12.02410	0.11556	0.00205	0.00262	0.00416	0.25688	0.55259
CAN.445	0.00054	0.00016	0.00042	0.00349	8.00986	12.01210	0.21571	0.00376	0.00296	0.00725	1.07809	1.49066
CAN.715	0.00133	0.00017	0.00285	0.00553	8.00848	12.01690	0.25288	0.00854	0.01361	0.00716	2.70695	3.54888
CAN.838	0.00145	0.00041	0.00139	0.01137	8.01337	12.00920	0.36517	0.01118	0.01964	0.02074	2.13967	5.88495
DWT.209	0.00020	0.00006	0.00027	0.00103	8.01087	12.01990	0.08045	0.00091	0.00164	0.00215	0.06519	0.47427
DWT.221	0.00028	0.00008	0.00025	0.00094	8.01254	12.02810	0.06564	0.00156	0.00247	0.00156	0.07156	0.67145
DWT.245	0.00018	0.00005	0.00024	0.00127	8.01335	12.01810	0.08183	0.00083	0.00146	0.00273	0.11240	0.66980
DWT.310	0.00025	0.00009	0.00038	0.00126	8.01318	12.01950	0.08185	0.00383	0.00908	0.00115	0.06047	0.92464
DWT.361	0.00029	0.00006	0.00030	0.00139	8.01045	12.02200	0.11941	0.00363	0.00684	0.00258	0.03272	1.02529
DWT.419	0.00057	0.00011	0.00047	0.00208	8.01108	12.01390	0.15123	0.00402	0.00486	0.00209	0.50116	1.03497
DWT.503	0.00079	0.00023	0.00061	0.00308	8.00890	12.02450	0.25259	0.00648	0.02181	0.00321	0.16225	2.02977
DWT.592	0.00063	0.00016	0.00076	0.00274	8.00978	12.01480	0.20212	0.00768	0.01973	0.00220	1.06711	2.02271
DWT.878	0.00068	0.00018	0.00154	0.00506	8.01239	12.01670	0.32332	0.01528	0.02766	0.00663	1.38669	6.16999
DWT.918	0.00084	0.00022	0.00147	0.00466	8.01000	12.02050	0.34543	0.01241	0.03426	0.00927	0.40884	6.58047
DWT.992	0.00163	0.00036	0.00314	0.00657	8.01118	12.01550	0.52678	0.03642	0.07698	0.02726	0.44736	8.43124
GR.30.30	0.00025	0.00022	0.00316	0.00032	8.01269	12.01590	0.36400	0.01394	0.01141	0.01476	0.03762	7.29861
JAGMESSH1	0.00082	0.00017	0.00068	0.00408	8.00928	12.01540	0.27140	0.01456	0.00625	0.00146	3.29223	6.66841
NOS2	0.00067	0.00017	0.00053	0.00251	8.01091	12.01490	0.07614	0.00722	0.02009	0.00190	0.07515	7.06462
NOS3	0.00174	0.00031	0.00366	0.00051	8.01534	12.02200	0.54996	0.03008	0.01873	0.06605	0.05944	7.56294
NOS5	0.00077	0.00021	0.00071	0.00357	8.01442	12.01160	0.30422	0.00483	0.00636	0.00614	1.20624	1.68894
NOS6	0.00038	0.00013	0.00049	0.00245	8.01380	12.02580	0.17752	0.00630	0.01452	0.00078	0.18980	3.23702
NOS7	0.00053	0.00019	0.00053	0.00419	8.01123	12.02390	0.34139	0.00925	0.00450	0.00217	2.15392	4.29754
PLAT362	0.00074	0.00017	0.00075	0.00276	8.01369	12.02000	0.22143	0.00567	0.01049	0.00483	0.21445	1.02139
PLSKZ362	0.00030	0.00010	0.00034	0.00203	8.01281	12.01960	0.09231	0.00283	0.00684	0.00227	0.14088	1.01867
SHERMAN1	0.00114	0.00033	0.00068	0.00430	8.01266	12.02390	2.23921	0.01246	0.00923	0.00698	0.15679	9.30472
Melhores resultados	0	35	0	0	0	0	0	0	0	0	0	0

Como nos resultados apresentados na seção 4.1, a heurística mais rápida, nos testes com a 35 matrizes assimétricas, foi a heurística BeL-GL. Essa heurística foi a mais rápida em todas as instâncias. A heurística mais lenta foi a heurística *VNS-band* em suas duas versões: com tempo máximo de 8s e com o tempo máximo de 12s. Ainda, os piores resultados nas reduções de largura de banda e de *profile* foram apresentados pela heurística *CSS-band*.

4.3 Instâncias assimétricas muito pequenas

Nesta subseção, apresentam-se os resultados obtidos pelas heurísticas nos testes computacionais nas 18 instâncias assimétricas muito pequenas da base Harwell-Boeing. Essas instâncias possuem de 30 a 199 vértices.

A Tabela 4 é dividida em três partes. Na primeira parte, têm-se os resultados das heurísticas em relação à largura de banda. Na segunda parte, mostram-se os resultados das heurísticas em relação ao *profile*. Finalmente, na terceira parte, têm-se os resultados das heurísticas em relação ao custo computacional.

Para a heurística *VNS-Band*, foram utilizados 500s de tempo máximo de execução em todas as 18 instâncias nos testes para as reduções de largura de banda e de *profile*. Utilizou-se 500s para mostrar que a heurística *VNS-band*, nem com o tempo máximo estabelecido neste trabalho, obteve resultados melhores, nas reduções de largura de banda e de *profile*, que a heurística FNCHC.

Especificamente, para a instância *GENT113*, utilizou-se a heurística *VNS-band* com 5s como critério de parada. Isso porque, com mais tempo de execução, o executável dessa heurística retorna resultados inconsistentes. Com isso, utilizou-se 5s como critério de parada, pois é o maior tempo em que é possível rodar o executável da heurística *VNS-band*, com a instância *GENT113*, sem que o executável retorne resultados inconsistentes.

Como é mostrado na primeira parte da Tabela 4, a heurística que mais vezes obteve o melhor resultado, na redução de largura de banda, foi a heurística *VNS-band*. Essa heurística obteve os melhores resultados em 13 das 18 instâncias. Entretanto, a heurística que apresentou o menor escore

Tabela 4 Resultados de cada heurística em relação à largura de banda, ao *profile* e ao custo computacional das simulações nas 18 instâncias assimétricas muito pequenas da base Harwell-Boeing.

Instância	n	β											
		CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band	
ARCI30	130	125	129	129	125	125	63	63	101	100	78	125	98
CURTIS54	54	44	50	50	44	44	16	10	12	13	16	44	44
FS.183.1	183	181	175	156	141	181	60	63	149	113	117	181	167
GENT113	113	101	111	111	125	94	27 ¹	39	34	44	38	94	104
GRE.115	115	101	96	97	98	101	23	24	35	33	39	101	106
GRE.185	185	60	39	45	41	60	21	22	21	23	33	60	172
IBM32	32	26	25	26	27	26	14	11	18	18	17	26	26
IMPCOL.B	59	43	54	49	53	43	24	20	41	28	28	43	54
IMPCOL.C	137	91	110	101	109	91	30	32	45	58	65	91	126
LNS.131	131	111	93	95	98	111	20	37	29	31	41	111	113
MCCA	180	65	121	108	78	65	37	38	54	52	73	65	170
PORES1	30	11	14	11	9	11	7	7	9	8	9	11	25
STEAM3	80	43	7	7	7	43	11	7	7	10	8	43	75
WEST0132	132	434	91	91	125	94	32	35	62	47	57	94	122
WEST0156	156	147	151	150	148	147	36	38	65	54	80	147	144
WEST0167	167	158	166	166	165	158	34	35	69	65	84	158	155
WILL57	57	44	20	19	23	44	14	7	9	14	8	44	49
WILL199	199	169	186	174	189	169	65	68	104	108	137	169	182
Melhores resultados		1	1	1	0	13	-22,89	7	2	0	0	0	0
$\Sigma \frac{k-i}{n}$		12,41	10,35	11,23	12,04	-22,23	-22,89	-12,00	-13,67	-11,59	12,04	24,32	

Instância	n	$profile_0$	<i>profile</i>										
			CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
ARCI30	130	9624	9175	9143	9624	9123	7718	6085	6657	6569	5595	9624	9873
CURTIS54	54	715	978	1011	715	749	995	672	701	662	710	715	1829
FS.183.1	183	23064	17259	18923	23064	17582	10079	11445	14677	17047	13791	23064	18540
GENT113	113	4419	3289	3287	4419	3264	2820 ¹	3230	2846	2917	3014	4419	6867
GRE.115	115	5960	4211	4357	5960	4226	2981	3066	3344	3350	3472	5960	7387
GRE.185	185	9698	7039	7255	9698	7272	5541	5656	5560	5700	5732	9698	23366
IBM32	32	599	450	462	599	483	469	400	488	386	429	599	574
IMPCOL.B	59	1192	1618	1624	1192	1406	1259	1232	1683	1349	1555	1192	2146
IMPCOL.C	137	3975	6217	5485	3975	5823	3921	4486	5132	4684	4965	3975	9967
LNS.131	131	6090	3375	4095	6090	3497	2894	3107	2473	2704	3015	6090	6997
MCCA	180	4015	6963	7218	4015	9117	6535	6008	5817	5684	7941	4015	23331
PORES1	30	434	359	366	434	300	276	311	330	353	308	434	639
STEAM3	80	3440	848	848	3440	848	1216	848	848	860	852	3440	5336
WEST0132	132	5329	7884	8040	5329	5230	4129	4340	5446	4504	4769	5329	9837
WEST0156	156	12197	7704	7746	12197	7056	5939	5888	8161	6449	7324	12197	13165
WEST0167	167	8390	10080	9959	8390	8070	5597	5893	7526	8351	8599	8390	14847
WILL57	57	714	612	625	714	470	639	432	437	590	443	714	2139
WILL199	199	25398	18990	18905	25398	17631	14801	15528	16287	18013	17591	25398	23957
Melhores resultados		1	1	1	2	1	9	3	2	1	2	0	0
$\Sigma \frac{k-i}{n}$		0,61	1,66	8,80	-3,96	-12,97	-14,83	-8,26	-10,35	-9,61	8,80	40,11	

Instância	n	Tempo (s)										
		CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
ARCI30	0,00014	0,00003	0,00013	0,00171	500,36700	0,04518	0,00085	0,00228	0,00329	0,01619	0,27576	
CURTIS54	0,00004	0,00001	0,00007	0,00026	512,93500	0,02045	0,00016	0,00023	0,00026	0,00323	0,10114	
FS.183.1	0,00011	0,00003	0,00013	0,00535	500,17900	0,07077	0,00079	0,00318	0,00349	0,33894	0,42969	
GENT113	0,00007	0,00002	0,00011	0,00066	5,09341 ¹	0,18503	0,00061	0,00192	0,00066	0,04220	0,31037	
GRE.115	0,00008	0,00002	0,00013	0,00080	500,53800	0,05219	0,00041	0,00049	0,00056	0,05500	0,22651	
GRE.185	0,00014	0,00007	0,00016	0,00121	500,99500	0,09378	0,00117	0,00221	0,00123	0,02162	0,43835	
IBM32	0,00002	0,00001	0,00003	0,00019	554,71500	0,01514	0,00010	0,00012	0,00012	0,00306	0,08609	
IMPCOL.B	0,00004	0,00001	0,00005	0,00073	552,06900	0,02528	0,00024	0,00032	0,00042	0,01656	0,09577	
IMPCOL.C	0,00006	0,00003	0,00013	0,00076	500,36000	0,06834	0,00075	0,00076	0,00110	0,00913	0,40642	
LNS.131	0,00005	0,00002	0,00007	0,00069	500,51500	0,28147	0,00038	0,00056	0,00096	0,06512	0,33030	
MCCA	0,00039	0,00005	0,00021	0,00240	500,27000	0,11936	0,00248	0,01136	0,00360	0,08900	0,44040	
PORES1	0,00003	0,00001	0,00003	0,00016	517,13500	0,01390	0,00015	0,00020	0,00012	0,00151	0,07414	
STEAM3	0,00015	0,00002	0,00020	0,00076	716,82600	0,03620	0,00059	0,00310	0,00024	0,00308	0,12844	
WEST0132	0,00010	0,00002	0,00009	0,00088	500,39000	0,05658	0,00078	0,00092	0,00104	0,02160	0,31351	
WEST0156	0,00008	0,00003	0,00012	0,00099	500,39500	0,08212	0,00055	0,00084	0,00132	0,01444	0,28191	
WEST0167	0,00012	0,00002	0,00013	0,00137	500,31900	0,07043	0,00087	0,00138	0,00087	0,01572	0,29481	
WILL57	0,00005	0,00001	0,00007	0,00035	536,42800	0,01585	0,00026	0,00057	0,00036	0,00515	0,10329	
WILL199	0,00011	0,00004	0,00016	0,00233	500,21500	0,14719	0,00104	0,00154	0,00278	0,20027	0,26639	
Melhores resultados	0	18	0	0	0	0	0	0	0	0	0	0

¹ Resultado obtido com a heurística VNS-Band(5s).

padronizado foi a heurística FNCHC. A heurística VNS-band apresentou o segundo menor escore padronizado. Mesmo com 500s de execução, a heurística VNS-band não obteve resultados melhores que os resultados da heurística FNCHC em relação à redução de largura de banda.

Em relação ao *profile*, a heurística VNS-*band* também foi a heurística que obteve o melhor resultado mais vezes. Porém, a heurística que apresentou o menor escore padronizado foi a heurística FNCHC. Em relação ao custo computacional, a heurística mais rápida foi a heurística BeL-GL. A heurística mais lenta foi a heurística VNS-*band*.

4.4 Instâncias assimétricas pequenas

Nesta subseção, apresentam-se os resultados obtidos pelas heurísticas nos testes computacionais nas 45 instâncias assimétricas pequenas da base Harwell-Boeing. Essas instâncias possuem de 207 a 1.104 vértices.

Para a heurística VNS-*Band*, foram utilizados 9s de tempo máximo de execução em todas as 45 instâncias nos testes para a redução de largura de banda. Para os testes com a redução de *profile*, utilizou-se 12s como critério de parada para a heurística VNS-*band*. Esses foram os menores tempos encontrados para que a heurística VNS-*band* obtivesse os melhores resultados nas reduções de largura de banda e de *profile*. Esses tempos foram encontrados empiricamente.

Especificamente, na instância *SHERMAN4*, o executável da heurística VNS-*band* retorna resultados errados independente do tempo passado como critério de parada. Nos testes mostrados na seção 4.3, um comportamento parecido desse executável foi notado na instância *GENT113*. Porém, nessa instância, o executável do VNS-*band* executa até 5s. Então, utilizou-se, somente na instância *SHERMAN4*, uma outra versão do executável da heurística VNS-*band*. Essa versão é um executável 32 *bits* e esse teste, em específico, teve que ser realizado em uma outra máquina

com sistema operacional 32 *bits*. Para detalhes da máquina utilizada, veja a subseção 3.5, na página 82.

Tabela 5 Resultados de cada heurística em relação a largura de banda e ao *profile* das simulações nas 45 instâncias assimétricas pequenas da base Harwell-Boeing.

Instância	n	$profile_0$	β										
			CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band(9s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
BP_0	822	820	819	819	815	820	250	259	422	439	541	820	789
BP_200	822	820	814	821	811	820	271	285	508	483	609	820	796
BP_400	822	820	814	804	806	820	279	296	530	530	620	820	795
BP_600	822	820	813	814	811	820	286	304	541	523	630	820	798
BP_800	822	820	818	817	817	820	359	309	523	523	600	820	800
BP_1000	822	820	818	814	820	820	371	315	603	489	679	820	799
BP_1200	822	820	820	819	815	820	356	317	554	491	630	820	795
BP_1400	822	820	816	811	816	820	358	324	636	475	681	820	796
BP_1600	822	820	815	815	813	820	301	322	609	476	673	820	799
FS_541.1	541	540	540	540	540	540	270	270	529	529	354	540	518
FS_680.1	680	600	404	358	30	600	25	17	25	20	29	600	629
FS_760.1	760	740	175	732	72	740	52	38	62	40	71	740	731
GRE_216A	216	36	36	36	41	36	28	21	21	21	33	36	203
GRE_343	343	49	49	49	53	49	57	28	28	28	40	49	327
GRE_512	512	64	64	64	66	64	21	36	36	36	61	64	490
HOR_131	434	421	93	93	101	421	55	60	104	73	121	421	417
IMPCOLA	207	167	165	165	127	167	33	53	58	48	66	167	194
IMPCOLD	425	406	416	418	418	406	42	41	85	61	81	406	404
IMPCOLE	225	92	190	188	219	92	42	43	65	75	109	92	214
JPWH_991	991	197	928	975	921	197	232	163	152	152	192	197	950
LNSP_511	511	57	450	485	447	57	46	57	55	55	97	57	486
MBEACXC	496	490	484	485	484	490	316	427	481	481	465	490	491
MBEAFLW	496	490	484	484	484	490	316	426	481	481	465	490	492
MBEAUSE	496	490	490	488	490	490	285	429	482	480	419	490	490
MCFE	765	187	734	241	247	187	126	133	178	179	258	187	754
NNC261	261	64	40	38	37	64	24	26	36	36	49	64	246
NNC666	666	262	67	71	69	262	61	46	66	63	88	262	642
ORSIRR_2	886	554	160	160	157	554	95	99	132	111	206	554	854
PORES_3	532	77	14	13	16	77	23	14	13	15	24	77	510
SAYLR1	238	14	15	15	15	14	14	14	14	15	15	14	224
SAYLR3	1000	100	59	58	57	100	56	57	57	55	104	100	941
SHERMAN4	1104	368	41	41	29	368	36 ¹	31	27	28	46	368	1027
SHL_0	663	661	660	660	658	661	233	252	425	412	374	661	623
SHL_200	663	661	661	661	656	661	239	256	437	426	360	661	628
SHL_400	663	662	660	658	658	662	234	250	465	425	385	662	635
STEAM1	240	146	50	50	50	146	44	46	50	50	63	146	231
STEAM2	600	331	63	63	63	331	63	63	63	63	67	331	588
STR_0	363	359	357	357	346	359	122	127	192	173	256	359	351
STR_200	363	359	348	344	815	820	129	137	233	199	267	820	351
STR_600	363	359	351	353	352	359	138	142	262	221	288	359	352
WEST0381	381	363	370	367	372	363	124	162	328	269	295	363	367
WEST0479	479	388	494	465	473	388	125	133	304	237	313	388	459
WEST0497	497	416	465	416	494	416	88	96	189	269	221	416	476
WEST0655	655	564	561	640	649	564	169	173	382	257	408	564	630
WEST0989	989	855	903	884	841	855	379	237	403	323	560	855	953
Melhores resultados			1	2	1	1	30	14	7	5	0	1	0
$\Sigma \frac{v_i}{n}$			20,84	18,09	18,54	26,50	-54,72	-52,11	-22,82	-28,85	-17,35	26,50	65,41

¹ Resultado proveniente do executável da heurística VNS-Band(9s) 32bits.

Como pode ser observado na Tabela 5, a heurística que obteve o melhor resultado em relação à largura de banda mais vezes foi a heurística VNS-band. Também, a heurística VNS-band foi a heurística que apresentou o menor escore padronizado. O segundo menor escore foi obtido pela heurística FNCHC, que também foi a segunda heurística no número de melhores resultados. Vale ressaltar que, é nítida a superioridade dos resultados das heurísticas VNS-band e FNCHC nas instâncias assimétricas pequenas. Isso porque essas duas heurísticas foram propostas para reduzir

a largura de banda de matrizes simétricas e assimétricas.

Tabela 6 Resultados de cada heurística em relação ao *profile* das simulações nas 45 instâncias assimétricas pequenas da base Harwell-Boeing.

Instância	n	<i>profile</i> _n	<i>profile</i>										
			CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band(12s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
BP_0	822	313542	232492	249292	235809	313542	204085	215330	272394	248221	249988	313542	367471
BP_200	822	356678	276452	270970	276640	356678	235129	242323	290703	273874	280548	356678	402470
BP_400	822	372881	287618	271263	277420	372881	243716	250851	335155	303001	283621	372881	405921
BP_600	822	390312	291410	284213	293314	390312	252441	255028	336451	337004	295578	390312	410346
BP_800	822	400827	311978	303241	284372	400827	270730	264568	311143	290406	291020	400827	418244
BP_1000	822	411136	301913	298198	301295	411136	304830	271438	312766	313475	310747	411136	426792
BP_1200	822	419750	302688	283637	303672	419750	264080	268572	314490	307522	311502	419750	430840
BP_1400	822	416507	303342	303749	309504	416507	299650	279842	322162	316360	313794	416507	433218
BP_1600	822	418320	301596	302505	308094	418320	273213	276168	307868	322252	317037	418320	435456
FS_541.1	541	231254	157999	156902	161360	231254	143919	158608	220229	160727	164501	231254	222494
FS_680.1	680	230957	154209	156566	157707	230957	9567	13727	16817	15296	16084	230957	223733
FS_760.1	760	419828	56918	61380	59194	419828	4212	44868	60737	47978	49130	419828	419364
GRE_216A	216	7130	7388	7526	7353	7130	6379	6378	6252	6252	6332	7130	28293
GRE_343	343	15608	16104	16362	16031	15608	14037	13932	13494	13494	13681	15608	73448
GRE_512	512	30715	31586	32027	31763	30715	40527	27490	26250	26250	26672	30715	161253
HOR_131	434	108080	46568	44424	41452	108080	37611	38024	42731	44934	39140	108080	155563
IMPCOLA	207	5291	9706	9756	5851	5291	6633	8982	7301	8132	8356	5291	22582
IMPCOLD	425	19078	21433	21268	18868	19078	16965	19363	21980	20916	21928	19078	108628
IMPCOLE	225	8971	11244	10111	13970	8971	8586	8638	8870	8757	10510	8971	32312
JPWH_991	991	139359	247683	260674	255225	139359	274052	178925	153998	172468	158827	139359	617513
LNSP_511	511	33577	49094	47964	28029	33577	29840	27493	26776	27097	30979	33577	140758
MBEACXC	496	173863	164128	168420	181029	173863	173180	183140	185763	184889	167115	173863	201549
MBEAFLW	496	173863	172341	178597	171448	173863	173180	183759	185763	184889	167115	173863	203572
MBEAUSE	496	161473	165406	161786	153244	161473	152734	169708	176638	177858	161272	161473	184658
MCFE	765	78957	160071	113848	89225	78957	113677	115725	115859	116427	137029	78957	534213
NNC261	261	9383	11034	11200	10719	9383	8281	9040	11068	9558	10629	9383	46056
NNC666	666	37389	48829	48621	46248	37389	45515	41677	47857	45891	46137	37389	300427
ORSIRL_2	886	211572	156553	156206	144929	211572	135203	131915	149219	135201	137841	211572	577745
PORES_3	532	55932	10690	11123	10995	55932	9337	11104	10777	10737	11202	55932	200447
SAYLR1	238	6298	4998	4998	5139	6298	5957	5520	4998	4998	5010	6298	36737
SAYLR3	1000	69490	52843	56117	52575	69490	55003	53004	53150	52213	57663	69490	441649
SHERMAN4	1104	269743	33748	34036	34316	269743	33627 ¹	22295	21181	22049	21203	269743	419220
SHL_0	663	148668	108431	108008	96751	148668	138287	127217	141750	139926	117021	148668	195984
SHL_200	663	153482	112165	112082	103117	153482	140442	128214	165441	136033	117823	153482	196907
SHL_400	663	159588	109703	112147	114794	159588	143280	129004	159558	148933	127843	159588	201811
STEAM1	240	33051	15060	15060	16257	33051	16539	16212	15060	15084	15864	33051	50325
STEAM2	600	196168	69416	69488	61704	196168	64328	63685	61704	61704	62312	196168	328472
STR_0	363	67460	47623	47672	47374	67460	45259	47301	49611	52381	53681	67460	80962
STR_200	363	80640	58220	58088	59215	80640	51366	50375	58039	59332	59162	80640	90055
STR_600	363	87176	64123	58716	63831	87176	54285	55059	58628	65187	61994	87176	90508
WEST0381	381	70091	86994	86525	83604	70091	67238	73202	81202	84393	82995	70091	97921
WEST0479	479	84283	95668	100341	98934	84283	64739	63125	88591	88014	81677	84283	138152
WEST0497	497	69403	90686	89189	92078	69403	39566	42426	51786	47002	46587	69403	129716
WEST0655	655	159257	139047	139740	135904	159257	114802	112860	165510	159705	144307	159257	258930
WEST0989	989	250490	375258	419762	376481	250490	231017	233155	330835	285486	296197	250490	573230
Melhores resultados			4	2	3	4	20	7	8	6	1	4	0
$\sum \frac{t_i}{n}$			-14,89	-15,09	-17,56	18,37	-30,75	-28,16	-6,84	-11,70	-20,59	18,37	108,83

¹ Resultado proveniente do executável da heurística VNS-Band(12s) 32bits.

Similarmente aos resultados em relação à largura de banda, apresentados na Tabela 5, a heurística VNS-band apresentou os melhores resultados na redução de *profile*, que são mostrados na Tabela 6. A heurística VNS-band foi a heurística que obteve, mais vezes, o melhor resultado. Ainda, essa heurística apresentou o menor escore padronizado.

Pode-se observar na Tabela 7 que, da mesma forma que nos outros conjuntos de instâncias, a heurística mais rápida foi a heurística BeL-GL. A mais lenta foi a heurística VNS-band com os dois tempos utilizados, 8s e 12s.

Tabela 7 Resultados de cada heurística em relação aos tempos médios, em segundos, de execução das simulações nas 45 instâncias assimétricas pequenas da base Harwell-Boeing.

Instância	CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band		FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
					(9s)	(12s)						
BP_0	0.00046	0.00028	0.00059	0.03122	9.01101	12.11430	0.72333	0.00805	0.01394	0.04619	10.9737	5.04834
BP_200	0.00070	0.00027	0.00060	0.03766	9.01290	12.11040	0.77538	0.01258	0.01540	0.05913	26.1421	4.85287
BP_400	0.00057	0.00022	0.00063	0.04351	9.01530	12.11820	0.76745	0.01019	0.01399	0.07177	30.0366	5.16874
BP_600	0.00082	0.00021	0.00073	0.04939	9.01280	12.11280	0.74581	0.01029	0.01145	0.07460	30.6494	4.59449
BP_800	0.00079	0.00019	0.00078	0.04718	9.01725	12.11820	0.77311	0.01045	0.01327	0.05441	30.4342	4.89230
BP_1000	0.00083	0.00018	0.00090	0.05947	9.01216	12.11430	0.80888	0.01014	0.01566	0.09430	32.2775	4.93832
BP_1200	0.00077	0.00023	0.00076	0.06162	9.01550	12.11920	0.80673	0.00934	0.01334	0.05735	28.7140	5.21761
BP_1400	0.00081	0.00021	0.00081	0.06137	9.02008	12.12010	0.82667	0.00941	0.01141	0.09985	28.0810	4.98693
BP_1600	0.00068	0.00027	0.00081	0.05708	9.01137	12.11320	0.80806	0.01065	0.01174	0.09464	27.2418	5.08025
FS_541.1	0.00051	0.00010	0.00040	0.02658	9.02082	12.12240	0.20655	0.01954	0.03533	0.04804	0.41081	2.16609
FS_680.1	0.00037	0.00006	0.00027	0.00336	9.01060	12.11150	0.17512	0.00426	0.00662	0.00405	0.12872	3.07982
FS_760.1	0.00070	0.00011	0.00064	0.00618	9.01532	12.12090	0.40112	0.00830	0.00779	0.00697	0.36369	3.92669
GRE_216A	0.00022	0.00005	0.00014	0.00107	9.02610	12.11650	0.08698	0.00175	0.00386	0.00074	0.03562	0.47035
GRE_343	0.00028	0.00005	0.00022	0.00186	9.01485	12.11640	0.15027	0.00332	0.00712	0.00092	0.17124	0.95727
GRE_512	0.00035	0.00010	0.00033	0.00289	9.01092	12.12930	0.23382	0.00638	0.01413	0.00196	0.30448	1.97609
HOR_131	0.00070	0.00019	0.00054	0.00456	9.01586	12.11820	0.33836	0.00372	0.00390	0.00845	1.38752	1.51304
IMPCOLA	0.00014	0.00004	0.00014	0.00158	9.01719	12.11080	0.07314	0.00105	0.00130	0.00168	0.02356	0.51433
IMPCOLD	0.00019	0.00006	0.00040	0.00302	9.01639	12.11580	0.16637	0.00362	0.00420	0.00452	0.13316	1.38153
IMPCOLE	0.00014	0.00003	0.00016	0.00183	9.01731	12.12960	0.09095	0.00174	0.00551	0.00356	0.33301	0.60196
JPWH_991	0.00074	0.00021	0.00116	0.01129	9.01940	12.11670	0.25023	0.01028	0.01212	0.01624	0.81226	7.26773
LNSP_511	0.00028	0.00005	0.00031	0.00326	9.01310	12.10960	0.24249	0.00425	0.00448	0.00394	0.67530	1.90546
MBEACXC	0.00068	0.00075	0.00525	0.68729	9.06807	12.15370	2.71791	0.10215	0.22174	1.48127	132.095	2.34597
MBEAFLW	0.00679	0.00083	0.00528	0.68282	9.06011	12.15120	2.71398	0.10133	0.22022	1.45529	132.721	2.31394
MBEAUSE	0.00555	0.00010	0.00379	0.57547	9.05699	12.15650	0.23382	0.00638	0.01413	0.00196	119.409	1.97609
NNC261	0.00042	0.00008	0.00027	0.00146	9.03795	12.13700	0.10887	0.00229	0.00428	0.00218	0.06117	0.66376
MCFE	0.00282	0.00023	0.00191	0.04380	9.01755	12.12490	1.03463	0.03807	0.16950	0.07967	5.17640	4.85498
NNC666	0.00049	0.00023	0.00056	0.00467	9.01350	12.10930	0.29576	0.00565	0.01064	0.00780	1.00060	2.93249
ORSRR_2	0.00082	0.00019	0.00113	0.00907	9.01444	12.11440	0.57990	0.00769	0.01107	0.01281	0.50638	5.38558
PORES_3	0.00048	0.00013	0.00034	0.00284	9.01790	12.11470	0.17307	0.00650	0.00432	0.00216	0.11005	2.06139
SAYLR1	0.00013	0.00003	0.00015	0.00023	9.02262	12.12760	0.08067	0.00107	0.00079	0.00039	0.00306	0.59436
SAYLR3	0.00060	0.00016	0.00051	0.00499	9.01580	12.11770	2.13638	0.00800	0.00735	0.00593	0.14012	7.99298
SHERMAN4	0.00046	0.00018	0.00072	0.00410	9.01699 ¹	12.12140 ²	0.32052	0.00800	0.00524	0.00319	0.50591	10.39652
SHL_0	0.00032	0.00010	0.00038	0.01752	9.01451	12.10860	0.19918	0.00393	0.01276	0.01906	9.62097	2.88522
SHL_200	0.00032	0.00016	0.00039	0.01956	9.01106	12.11390	0.19452	0.00323	0.01039	0.02265	10.8522	3.01915
SHL_400	0.00030	0.00017	0.00038	0.01860	9.01018	12.11110	0.21131	0.00398	0.01037	0.02254	9.79234	2.79648
STEAM1	0.00053	0.00010	0.00047	0.00234	9.02253	12.11310	0.22305	0.00292	0.00242	0.00265	0.24201	0.58608
STEAM2	0.00133	0.00024	0.00108	0.00954	9.02204	12.12140	0.70515	0.01826	0.01245	0.00928	3.95201	3.31327
STR_0	0.00028	0.00012	0.00029	0.00675	9.01288	12.11270	0.23600	0.00489	0.00546	0.00940	0.69059	1.13044
STR_200	0.00069	0.00011	0.00043	0.00920	9.01154	12.10890	0.30776	0.00434	0.00612	0.01661	2.03940	1.04132
STR_600	0.00049	0.00011	0.00046	0.01315	9.01081	12.10950	0.32812	0.00581	0.00940	0.02201	2.27952	1.15622
WEST0381	0.00039	0.00006	0.00032	0.00923	9.01090	12.11110	0.41265	0.00753	0.00452	0.01805	1.04006	0.93019
WEST0479	0.00037	0.00007	0.00035	0.00823	9.01282	12.12330	0.33187	0.00384	0.00793	0.01417	1.76949	1.56818
WEST0497	0.00031	0.00009	0.00034	0.00695	9.01452	12.11410	0.21657	0.00376	0.00646	0.00985	0.48501	1.49981
WEST0655	0.00071	0.00016	0.00049	0.01791	9.01121	12.11460	0.56168	0.00534	0.01090	0.02734	0.40966	2.79211
WEST0989	0.00076	0.00016	0.00103	0.02667	9.01467	12.11400	0.84681	0.00960	0.01385	0.05563	12.7455	7.45910
Melhores resultados	0	45	0	0	0	0	0	0	0	0	0	0

¹ Resultado proveniente do executável da heurística VNS-Band(9s) 32bits.

² Resultado proveniente do executável da heurística VNS-Band(12s) 32bits.

4.5 Análise dos resultados dos testes nas instâncias da Harwell-Boeing

Mostra-se, na Tabela 8, as posições de cada heurística em relação aos resultados da redução de largura de banda, da redução de *profile* e do custo computacional. Considerou-se os resultados em todas as 113 matrizes da Harwell-Boeing. Para a largura de banda e o *profile*, foram somados os números de “vitórias” de cada heurística, ou seja, o número de vezes em que cada heurística obteve o melhor resultado. Para o custo computacional, somou-se os tempos de cada heurística nas 113 instâncias da Harwell-Boeing.

Pode-se observar, na Tabela 8, que a heurística que mais vezes obteve o melhor resultado nas reduções de largura de banda e de *profile* foi a heurística VNS-*band*. Porém, o custo computacional dessa heurística foi o pior. Entretanto, como mostrado nas seções 4.2 e 4.4, pode-se obter boas reduções de largura de banda e de *profile* com a heurística VNS-*band* com poucos segundos. Não é necessário utilizar 500s como Mladenovic et al. (2010) utilizaram.

Algumas heurísticas merecem destaque. A heurística BeL-GL foi a heurística mais rápida e obteve resultados razoáveis nas reduções de largura de banda e de *profile*. A heurística CMr-GL, a segunda heurística mais rápida, também apresentou resultados razoáveis nas reduções de largura de banda e de *profile*. A heurística GPS obteve bons resultados nas reduções de largura de banda e de *profile* e foi relativamente rápida.

Tabela 8 Posições, de cada heurística, em relação ao número de “vitórias” obtidas nas 113 matrizes da base Harwell-Boeing. São apresentadas as respectivas posições em relação à largura de banda (β), ao *profile* e ao custo computacional (t).

Heurística	β	<i>profile</i>	t (s)
CMr-GL	6°	4°	2°
BeL-GL	5°	5°	1°
hGPHH	8°	10°	3°
WBRA-GL	9°	8°	6°
VNS- <i>Band</i>	1°	1°	11°
FNCHC	2°	6°	8°
GPS	3°	2°	4°
GGPS	4°	3°	5°
BL	10°	7°	7°
WBRA	7°	9°	9°
CSS- <i>Band</i>	11°	11°	10°

Na Tabela 9, mostra-se, em cada conjunto de instância, a heurística

que obteve o menor somatório do escore padronizado. Observa-se que, no conjunto das matrizes muito pequenas simétricas e assimétricas, a heurística que obteve as melhores reduções de largura de banda e de *profile* foi a heurística FNCHC. Nas instâncias pequenas, simétricas e assimétricas, a heurística que obteve os menores somatórios de escore padronizado, para largura de banda e *profile*, foi a heurística VNS-*band*. Ainda, pode ser observado que a heurística VNS-*band* obteve esses resultados com um custo computacional relativamente baixo, o maior tempo foi 12s.

Ao todo, são 80 matrizes pequenas e 33 matrizes muito pequenas que são divididas em simétricas e assimétricas. Como o número de matrizes pequenas é maior que o número de matrizes muito pequenas, pode-se considerar que a heurística VNS-*band* obteve as melhores reduções de largura de banda e de *profile* nas matrizes da base Harwell-Boeing.

Tabela 9 Heurísticas que obtiveram os menores somatórios de escore padronizado ($\sum \frac{x-\bar{x}}{\sigma}$), em cada conjunto de instância. São apresentados os resultados em relação à largura de banda (β) e *profile*.

Instâncias	Quantidade	Tipo	β	<i>profile</i>
Muito pequenas	15	Simétricas	FNCHC	FNCHC
	18	Assimétricas		
Pequenas	35	Simétricas	VNS- <i>Band</i> (8s)	VNS- <i>Band</i> (12s)
	45	Assimétricas	VNS- <i>Band</i> (9s)	

Os resultados mostrados na Tabela 8 e 9 corroboram com os resultados obtidos por Mladenovic et al. (2010). Podemos reafirmar que a heurística VNS-*band* é a heurística no estado da arte para a redução de largura de banda. Em relação ao *profile*, a heurística VNS-*band* também foi a melhor. Porém, não se pode afirmar que ela é o estado da arte para a redução de *profile*.

5 SIMULAÇÕES COM RESOLUÇÃO DE SISTEMAS DE EQUAÇÕES LINEARES

Neste Capítulo, mostram-se os resultados dos testes com a resolução de SELs pelo MGC preconditionado pelo método de Jacobi (MGCPJ) e pela fatoração incompleta de Cholesky com zero *fill-in* (MGC-IC0). Para isso, foram utilizadas três conjuntos de SELs, conforme descrito na seção 3.3. Para cada conjunto de instâncias, resolveu-se os SELs pelo MGCPJ e pelo MGC-IC0 com e sem as reduções de largura de banda e de *profile* pelas heurísticas para a redução de largura de banda.

Para os testes com a resolução de SELs, todas as heurísticas para a redução de largura de banda, implementadas neste trabalho, foram utilizadas, com a exceção da heurística WBRA de Esposito et al. (1998). Isso porque essa heurística apresentou muita ocupação de memória e custo computacional elevado em relação às outras heurísticas. Por exemplo, em uma instância com, aproximadamente, 30.000 vértices, a heurística WBRA ocupou cerca de 5 GiB de memória. Isso ocorreu pois a heurística WBRA gera as estruturas de nível enraizadas de todos os vértices do grafo. Em seguida, todas as estruturas de nível enraizadas que apresentarem largura de nível menor que a largura de banda original serão armazenadas. Como as matrizes dos SELs utilizadas neste trabalho possuem largura de banda original grande, todas as estruturas de nível enraizadas são armazenadas. Com isso, a ocupação de memória, pela heurística WBRA, é grande e tornou a utilização dessa heurística impraticável em instâncias maiores que as instâncias da Harwell-Boeing.

Na heurística VNS-*Band*, inicialmente, utilizou-se o tempo máximo de execução de um segundo. Na tentativa de se obterem melhores resultados

na resolução de SELs com a renumeração obtida pela heurística *VNS-Band*, utilizou-se, também, outros valores para o tempo máximo de execução.

Na heurística *CSS-Band*, utilizou-se, como critério de parada, dez iterações e uma partícula carregada a cada 1.000 vértices nas instâncias com até 50.000 vértices. Nas instâncias com 50.000 a 499.999 vértices, utilizou-se uma iteração e uma partícula carregada a cada 10.000 vértices. Para instâncias com mais de 500.000 vértices, utilizou-se uma iteração e uma partícula carregada a cada 20.000 vértices.

Nas seções 5.1 e 5.2, são descritas as estruturas das tabelas dos resultados com as simulações do MGCPJ e MGC-IC0, respectivamente. Os resultados da resolução de SELs oriundos da discretização da condução do calor são mostrados na seção 5.3. Os resultados da resolução de SELs oriundos da discretização da equação de Laplace são mostrados na seção 5.4.

5.1 Estruturas das Tabelas dos Resultados das Simulações com o MGCPJ

A Tabela 10, mostrada na subseção 5.3.1, a Tabela 12, mostrada na subseção 5.4.1.1 e a Tabela 14, mostrada na subseção 5.4.2.1, referentes aos resultados do MGCPJ, são estruturadas como a seguir. Na primeira coluna, n é a dimensão da matriz de coeficientes do respectivo SEL ou o número de vértices do grafo. Na segunda coluna, há o nome da heurística utilizada para as reduções de largura de banda e *profile*. Note que, na primeira linha da segunda coluna, de cada instância, há a heurística “Sem renumeração”. Isso significa que nenhuma heurística foi utilizada para as reduções de largura de banda e de *profile*. Com isso, o SEL foi resolvido pelo MGCPJ em sua

forma original, isto é, sem a reordenação da matriz de coeficientes. Com esse resultado é possível verificar se a redução de largura de banda da matriz de coeficientes, por uma heurística, reduz o custo computacional do MGCPJ.

Os resultados em relação à largura de banda e ao *profile* são mostrados na terceira e quarta colunas, respectivamente, da Tabela 10. Na quinta e sexta colunas, são apresentados, respectivamente, os resultados das heurística em relação ao custo computacional em segundos e a ocupação de memória em *mebibytes*. O número de iterações e o custo computacional, em segundos, do MGCPJ são mostrados na sétima e oitava colunas. Os desvios padrões e os coeficientes de variação, referentes ao custo computacional do MGCPJ, são mostrados na nona e décima colunas. Na última coluna, têm-se os percentuais de reduções dos tempos totais de execução da heurística correspondente, mais os tempos de execução do MGCPJ, em relação ao tempo de execução do MGCPJ sem reduções de largura de banda e de *profile*.

Na tabelas, para cada instância, os melhores resultados são destacados em negrito. Isso é feito para os resultados em relação à largura de banda, ao *profile*, ao custo computacional da heurística, ao custo computacional do MGCPJ e à porcentagem de redução do custo computacional do MGCPJ.

5.2 Estruturas das Tabelas dos Resultados das Simulações com o MGC-IC0

A Tabela 11, mostrada na subseção 5.3.2, a Tabela 13, mostrada na subseção 5.4.1.2 e a Tabela 15, mostrada na subseção 5.4.2.2, referentes aos resultados do MGC-IC0, são estruturadas como a seguir. Na primeira

coluna das tabelas, n é a dimensão da matriz de coeficientes do respectivo SEL ou o número de vértices do grafo. Na segunda coluna, há o nome da heurística utilizada para as reduções de largura de banda e *profile*. Note que, na primeira linha da segunda coluna, de cada instância, há a heurística “Sem renumeração”. Isso significa que nenhuma heurística foi utilizada para as reduções de largura de banda e de *profile*. Com isso, o SEL foi resolvido pelo MGC-IC0 em sua forma original, isto é, sem a reordenação da matriz de coeficientes.

Os resultados em relação à largura de banda e ao *profile* são mostrados na terceira e quarta colunas respectivamente. Na quinta e sexta colunas, são apresentados, respectivamente, os resultados das heurística em relação ao custo computacional em segundos e a ocupação de memória em *mebibytes*. Os custos computacionais do IC0 e do MGC condicionado são mostrados, separadamente, na sétima e oitava colunas. Decidiu-se discriminar esses dados, pois as renumerações produzidas pelas heurísticas também afetam o custo computacional do IC0.

O número de iterações e o custo computacional total, em segundos, do MGC-IC0 são mostrados na nona e décima colunas. Os desvios padrões e os coeficientes de variação, referentes ao custo computacional total do MGC-IC0, são mostrados na décima primeira e décima segunda colunas. Na última coluna, têm-se os percentuais de reduções dos tempos totais de execução da heurística correspondente, mais os tempos de execução do MGC-IC0, em relação ao tempo de execução do MGC-IC0 sem reduções de largura de banda e de *profile*.

Na tabelas, para cada instância, os melhores resultados são destacados em negrito. Isso é feito para os resultados em relação à

largura de banda, ao *profile*, ao custo computacional da heurística, ao custo computacional do IC0, ao custo computacional do MGC preconditionado, ao custo computacional total do MGC-IC0 e à porcentagem de redução do custo computacional do MGC-IC0.

5.3 Simulações Com instâncias oriundas da discretização da condução do calor

Nesta seção, apresentam-se os resultados dos testes computacionais nos SELs resultantes da discretização da condução do calor por volumes finitos. Esses SELs foram gerados pelo projeto computacional de Oliveira (2014a). Foram utilizadas oito instâncias (SELs) com tamanhos de 4.846 a 965.545 incógnitas. Executou-se cada heurística e o resolutor três vezes em cada instância.

Na subseção 5.3.1, são mostrados os resultados das simulações do MGCPJ com a redução de largura de banda. Os resultados com o MGC-IC0, com a redução de largura de banda, são apresentados na subseção 5.3.2. As ocupações de memória das heurísticas são detalhadas na subseção 5.3.3. Comparações dos resultados dos resolutores MGCPJ e MGC-IC0 são realizadas na subseção 5.3.4.

5.3.1 Simulações com o método dos gradientes conjugados preconditionado pelo método de Jacobi

Nesta subseção, são apresentados os resultados da resolução de SELs, oriundos da discretização da condução do calor, pelo MGCPJ com as reduções de largura de banda e de *profile*. Executou-se cada heurística e o MGCPJ três vezes. Os resultados médios dessas execuções são apresentados

na Tabela 10.

Em relação à heurística *VNS-band*, como pode ser observado na Tabela 10, em cada instância, foram utilizados diferentes tempos de execução. Porém, em todas as oito instâncias, os melhores resultados da heurística *VNS-Band*, em relação à redução do custo computacional do MGCPJ, foram obtidos com um segundo. Ainda, como descrito na seção 3.4, na página 74, o executável da heurística *VNS-band* aceita matrizes com dimensão de até 500.000. Com isso, não foi possível executar a heurística *VNS-band* na última instância, com 965.545 vértices.

Devido ao grande custo computacional das heurísticas BL e *CSS-band* e aos resultados ruins obtidos na redução do custo computacional do MGCPJ, essas heurísticas não foram executadas nas duas últimas instâncias, com 492.853 e 965.545 vértices. Também, pelos mesmos motivos, as heurísticas GPS e GGPS não foram executadas na última instância, com 965.545 vértices.

Na Tabela 10, pode-se observar que, nos resultados em relação às ocupações de memória, as heurísticas CMr-GL, BeL-GL, hGPHH e BL não apresentaram ocupação de memória. Em relação às heurísticas CMr-GL, BeL-GL e hGPHH, esses resultados foram obtidos, pois essas heurísticas não precisam de nenhuma estrutura auxiliar para realizar a renumeração dos vértices do grafo, embora atributos extras tenham sido implementados na classe vértice. Em relação à heurística BL, essa heurística faz uso de algumas estruturas extras, porém, a ocupação de memória não é significativa e foi considerada como 0.

A heurística que mais vezes obteve as melhores reduções de largura de banda e de *profile*, foi a GPS. Porém, o custo computacional dessa

Tabela 10 Resultados referentes às execuções do MGCPJ e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da condução calor.

n	Heurística	β	<i>profile</i>	Heurística		MGCPJ		σ	C_v	Redução	
				t(s)	m.(MiB)	iter.	t(s)				
4846	Sem renum.	4769	9116750	-	-	319	9,97	0,04	0,46%	-	
	CMr-GL	161	468110	0,005	0	319	9,55	0,19	2,01%	4,16%	
	BeL-GL	163	467970	0,003	0	319	10,21	0,07	0,66%	-2,43%	
	hGPHH	161	518637	0,005	0	319	9,46	0,05	0,5%	5,07%	
	WBRA-GL	171	441298	0,063	1,03	319	9,49	0,04	0,46%	4,21%	
	VNS-Band	(1s)	154	499797	1,054	44,51	319	10,06	0,15	1,50%	-11,43%
		(2s)	154	499797	2,049	44,51	319	9,78	0,07	0,71%	-18,65%
		(5s)	154	499797	5,045	44,51	319	9,81	0,03	0,33%	-48,99%
		(10s)	154	499797	10,053	44,51	319	9,82	0,08	0,89%	-99,30%
	FNCHC	121	437962	2,287	2,22	319	10,04	0,08	0,86%	-23,67%	
	GPS	140	406081	0,166	0,32	319	9,83	0,19	1,90%	-0,30%	
	GGPS	130	422281	0,273	0,56	319	9,83	0,22	2,31%	-1,30%	
	BL	233	417653	0,545	0	319	10,27	0,05	0,48%	-8,42%	
	CSS-Band	4765	8596483	2,360	22,16	319	10,93	0,27	2,51%	-33,29%	
10728	Sem renum.	10626	45314579	-	-	462	37,35	0,36	0,96%	-	
	CMr-GL	270	1579179	0,02	0	462	34,19	0,06	0,17%	8,41%	
	BeL-GL	274	1590382	0,01	0	462	34,69	0,07	0,20%	7,09%	
	hGPHH	275	1768638	0,02	0	462	33,44	0,04	0,12%	10,45%	
	WBRA-GL	306	1680767	0,22	2,58	462	34,36	0,11	0,32%	7,41%	
	VNS-Band	(1s)	552	1746660	1,14	135,62	462	34,73	0,40	1,14%	3,96%
		(2s)	552	1746660	2,14	135,62	462	35,09	0,13	0,37%	0,32%
		(5s)	552	1746660	5,23	135,62	462	35,16	0,16	0,46%	-8,14%
		(10s)	552	1746660	10,15	135,62	462	34,92	0,14	0,40%	-20,67%
	FNCHC	208	1637305	6,12	2,49	462	35,74	0,98	2,74%	-12,07%	
	GPS	207	1358676	0,63	1,54	462	34,69	0,23	0,66%	5,43%	
	GGPS	226	1477595	1,00	1,38	462	35,35	0,63	1,79%	2,68%	
	BL	398	1365197	6,38	0	462	35,96	0,12	0,32%	-13,36%	
	CSS-Band	10625	42631422	4,71	79,92	462	42,80	0,78	1,82%	-27,20%	
23367	Sem renum.	23167	216212086	-	-	671	124,39	0,53	0,42%	-	
	CMr-GL	313	4664523	0,07	0	671	110,69	2,01	0,82%	10,96%	
	BeL-GL	314	4662697	0,05	0	671	112,37	2,34	2,04%	9,62%	
	hGPHH	314	5129407	0,08	0	671	106,32	0,26	0,25%	14,46%	
	WBRA-GL	681	5618347	0,75	7,73	671	109,46	1,84	1,68%	11,40%	
	VNS-Band	(1s)	1564	8889127	1,54	371,10	671	113,53	1,74	1,53%	7,49%
		(2s)	1564	8889127	2,52	371,10	671	112,99	0,27	0,24%	7,14%
		(5s)	1564	8889127	5,51	371,10	671	111,84	1,62	1,44%	5,66%
		(10s)	1564	8889127	10,52	371,10	671	113,83	0,33	0,29%	0,03%
	FNCHC	333	5519183	15,64	2,96	671	111,14	0,64	0,57%	-1,92%	
	GPS	293	4221479	3,57	3,34	671	112,56	3,50	3,12%	6,64%	
	GGPS	314	5007391	5,30	3,51	671	111,47	2,72	2,45%	6,13%	
	BL	465	4296542	11,20	0	671	116,66	0,29	0,25%	-2,79%	
	CSS-Band	23183	202741325	94,78	549,09	671	149,49	1,59	1,05%	-96,37%	
50592	Sem renum.	50461	1020411959	-	-	970	391,19	0,13	0,03%	-	
	CMr-GL	647	17502999	0,26	0	970	348,44	10,71	3,07%	10,87%	
	BeL-GL	650	17518537	0,19	0	970	345,01	6,23	1,81%	11,76%	
	hGPHH	640	19173913	0,26	0	970	335,17	0,48	0,14%	14,25%	
	WBRA-GL	706	18363549	2,61	15,98	970	334,85	2,12	0,63%	13,72%	
	VNS-Band	(1s)	7377	47268027	3,03	967,10	970	355,55	8,48	2,38%	8,34%
		(4s)	7377	47268027	6,88	967,10	970	368,95	2,74	0,74%	3,93%
		(9s)	7377	47268027	10,91	967,10	970	357,99	7,04	1,97%	5,69%
		(15s)	7377	47268027	16,91	967,10	970	363,49	7,91	2,18%	2,75%
	FNCHC	479	17779307	43,05	5,55	970	349,03	2,05	0,59%	-0,23%	
	GPS	466	14149442	24,65	5,51	970	357,03	10,56	2,96%	2,43%	
	GGPS	499	15922368	55,64	7,72	970	341,25	2,13	0,63%	-1,46%	
	BL	961	14447154	491,74	0	970	361,89	2,08	0,58%	-118,21%	
	CSS-Band	50287	951213645	954,92	2701,00	970	474,22	10,54	2,22%	-265,33%	

heurística não compensa a redução do custo computacional do MGCPJ. Ainda, a redução do custo computacional do MGCPJ não é proporcional às reduções de largura de banda e de *profile*. Como nos testes nas instâncias

Tabela 10 Conclusão.

n	Heurística	β	profile	Heurística		MGCPJ		σ	C_V	Redução	
				t(s)	m.(MiB)	iter.	t(s)				
108683	Sem renum.	108216	4725435534	-	-	1398	1238,76	29,02	2,34%	-	
	CMr-GL	868	54290089	0,25	0	1398	1043,15	2,57	0,25%	15,77%	
	BeL-GL	868	54309266	0,12	0	1398	1055,17	1,41	0,13%	14,81%	
	hGPHH	868	59697870	0,26	0	1399	1062,85	5,69	0,53%	14,18%	
	WBRA-GL	756	56646417	9,52	26,55	1400	1046,99	5,02	0,48%	14,71%	
	VNS-Band	(1s)	16676	157588720	8,64	2293,76	1398	1067,87	2,34	0,22%	13,09%
		(9s)	16676	157588720	17,45	2293,76	1398	1101,91	1,38	0,13%	9,64%
		(18s)	16676	157588720	27,53	2293,76	1398	1087,70	2,19	0,20%	9,97%
		(27s)	16676	157588720	37,55	2293,76	1398	1116,26	0,36	0,03%	6,86%
	FNCHC	738	55924986	116,36	11,63	1398	1045,24	4,70	0,45%	6,23%	
	GPS	642	48744729	139,22	12,62	1398	1070,03	5,69	0,53%	2,38%	
	GGPS	736	54051275	225,84	23,33	1398	1048,35	0,79	0,08%	2,86%	
	BL	1262	45836037	617,56	0	1398	1117,72	0,68	0,06%	-40,08%	
CSS-Band	108390	99051906	369,55	586,52	1400	1479,50	11,81	0,80%	-49,26%		
232052	Sem renum.	231672	21652820640	-	-	2034	3814,37	0,33	0,01%	-	
	CMr-GL	1471	183551861	0,58	0	2039	3370,62	5,59	0,17%	11,62%	
	BeL-GL	1471	183479875	0,31	0	2038	3382,99	5,79	0,17%	11,30%	
	hGPHH	1471	200679371	0,61	0	2030	3352,40	20,39	0,61%	12,09%	
	WBRA-GL	1972	177829270	30,03	65,74	2029	3284,84	15,75	0,48%	13,09%	
	VNS-Band	(1s)	17036	313594920	33,49	5048,32	2030	3379,30	33,09	0,98%	10,52%
		(34s)	17036	313594920	67,01	5048,32	2030	3412,60	30,29	0,89%	8,77%
	FNCHC	1145	182638143	287,81	22,40	2031	3340,45	18,27	0,55%	4,88%	
	GPS	1104	148697458	585,48	25,42	2032	3422,30	2,65	0,08%	-5,07%	
	GGPS	1243	170494162	1236,95	43,79	2031	3337,75	9,26	0,28%	-19,93%	
	BL	2048	153659093	47604,01	0	2038	3589,31	1,11	0,03%	-1242,12%	
	CSS-Band	231606	2857086021	1551,79	841,35	2039	4709,74	6,28	0,13%	-64,16%	
	492853	Sem renum.	492100	97893937993	-	-	2927	11619,20	11,49	0,10%	-
CMr-GL		1805	510308269	1,48	0	2925	10440,46	12,96	0,22%	10,13%	
BeL-GL		1811	510450511	0,93	0	2932	10241,11	8,86	0,09%	11,85%	
hGPHH		1814	557734280	1,57	0	2932	10395,07	16,07	0,15%	10,52%	
WBRA-GL		2671	590763978	101,77	222,88	2935	10539,18	37,65	0,36%	8,42%	
VNS-Band		(1s)	41845	1083304907	137,67	10844,16	2936	10580,39	11,12	0,11%	7,76%
		(138s)	26192	959280807	273,79	10844,16	2935	11096,60	2,46	0,02%	2,14%
FNCHC		1660	583349111	749,03	74,19	2932	10233,68	14,51	0,14%	5,48%	
GPS		1454	460254575	2700,88	49,85	2932	10516,23	25,40	0,34%	-13,75%	
GGPS		1734	571294474	4759,91	107,57	2932	10297,02	16,08	0,16%	-29,59%	
BL		-	-	-	-	-	-	-	-	-	-
CSS-Band		-	-	-	-	-	-	-	-	-	-
965545		Sem renum.	964827	377848438952	-	-	4358	34061,44	11,67	0,03%	-
	CMr-GL	2201	1332260636	2,21	0	4346	31978,36	37,83	0,12%	6,11%	
	BeL-GL	2453	1447960230	1,14	0	4360	32247,29	39,08	0,12%	5,32%	
	hGPHH	2315	1495703784	2,37	0	4370	32392,77	15,64	0,05%	4,89%	
	WBRA-GL	3603	1469674194	259,95	256,07	4362	31901,95	19,41	0,06%	5,58%	
	VNS-Band ¹	-	-	-	-	-	-	-	-	-	-
	FNCHC	2189	1562324154	1636,39	139,75	4365	32511,67	30,43	0,09%	-0,25%	
	GPS	-	-	-	-	-	-	-	-	-	-
	GGPS	-	-	-	-	-	-	-	-	-	-
	BL	-	-	-	-	-	-	-	-	-	-
	CSS-Band	-	-	-	-	-	-	-	-	-	-

¹ A heurística VNS-Band executa em instâncias de até 500000 vértices.

da Harwell-Boeing, mostrados no Capítulo 4, a heurística mais rápida foi a heurística BeL-GL.

Em relação ao MGCPJ, observa-se que não há variação no número de iterações. Os gráficos das reduções dos custos computacionais do MGCPJ são mostrados nas Figuras 13, 14 e 15. Na Figura 13, são apresentadas as porcentagens de todas as 10 heurísticas testadas. Com essa figura, nota-se,

claramente, que a heurística BL e a heurística CSS-*band* foram as piores para a redução do custo computacional do MGCPJ.

Na Figura 14, foi removido as heurísticas BL e CSS-*band*. Com isso, é possível observar que as heurísticas GPS, GGPS, FNCHC e VNS-*band* obtiveram resultados piores que os resultados das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL. O gráfico dessas quatro heurísticas é mostrado na Figura 15.

Conforme mostrado no gráfico da Figura 15, as heurísticas que apresentaram as melhores reduções foram as heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL. Essas heurísticas, além de gerarem reordenações em que há mais coerência de *cache*, possuem baixo custo computacional.

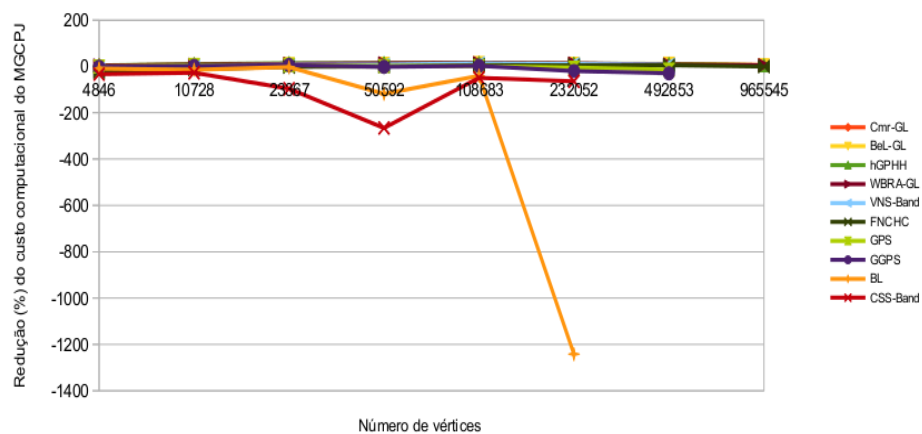


Figura 13 Gráfico das porcentagens das reduções do custo computacional do MGCPJ de todas as 10 heurísticas testadas.

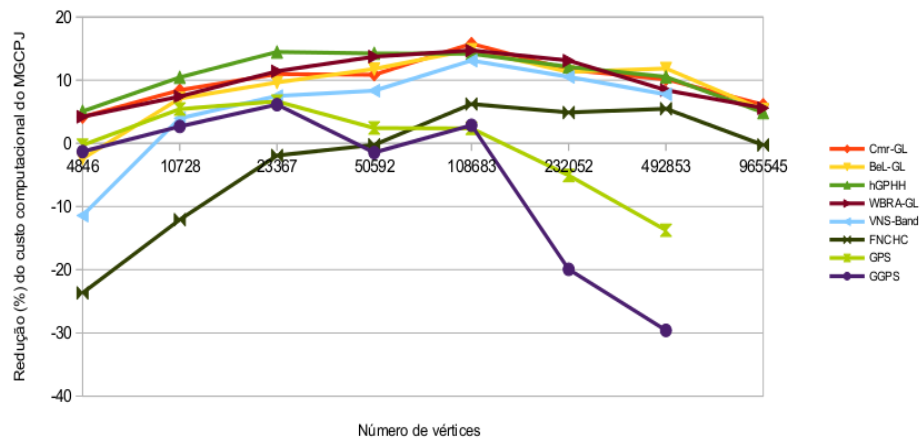


Figura 14 Gráfico das porcentagens das reduções do custo computacional do MGCPJ sem os resultados das heurísticas BL e CSS-Band.

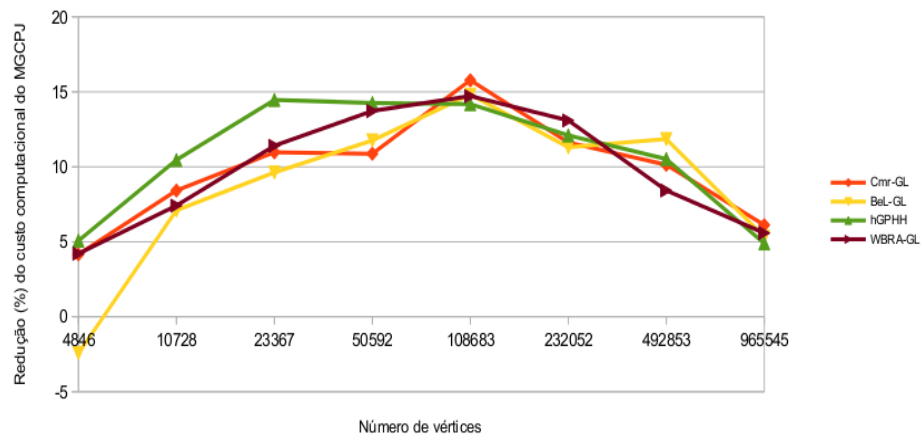


Figura 15 Gráfico das porcentagens das reduções do custo computacional do MGCPJ das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.

5.3.2 Simulações com o método dos gradientes conjugados preconditionado pelo método incompleto de Cholesky com zero *fill-in*

Nesta subseção, são apresentados os resultados da resolução de SELs oriundos da discretização da condução do calor, pelo MGC-IC0 com as

reduções de largura de banda e de *profile*. Executou-se cada heurística e o MGC-IC0 três vezes. Os resultados, médios, dessas execuções são apresentados na Tabela 11.

Devido ao grande custo computacional envolvido nas simulações realizadas neste trabalho, os testes nas instâncias com 492.853 e 965.545 vértices não haviam terminado até a data da entrega da dissertação. Com isso, mostram-se, na Tabela 11, os resultados do MGC-IC0 com todas as instâncias excetuando-se as instâncias com 492.853 e 965.545 vértices. Também, só foi possível executar a heurística *VNS-band* com 1s de tempo máximo.

Assim como nos resultados do MGCPJ, apresentados na seção 5.3.1, a heurística GPS foi a que obteve as melhores reduções de largura de banda e de *profile*. Também, em relação ao consumo de memória, não houve diferença com os resultados apresentados na subseção 5.3.1. Ainda, a heurística mais rápida foi a heurística BeL-GL. Esses resultados já eram esperados, pois as instâncias são as mesmas.

Mesmo a heurística GPS ter apresentado as melhores reduções de largura de banda e de *profile*, a redução do custo computacional do MGC-IC0 não foi relevante. Isso porque o custo computacional dessa heurística não compensa a redução do custo computacional do MGC-IC0. Além disso, as reduções do custo computacional do MGC-IC0 não são proporcionais às reduções de largura de banda e de *profile*.

Em relação à redução do custo computacional do MGC-IC0, há alguns pontos importantes que precisam ser destacados. As renumerações dos vértices pelas heurísticas também influenciam no custo computacional do IC0 e não somente do MGC preconditionado. Especificamente, o custo

Tabela 11 Resultados referentes às execuções do MGC-IC0 e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da condução calor.

n	Heurística	β	$profile$	Heurística		IC0 t(s)	MGC t(s)	MGC-IC0		σ	C_v	Red.
				t(s)	m.(MiB)			iter.	t(s)			
4846	Sem renum.	4769	9116750	-	-	2,29	5,81	144	8,10	0,02	0,26%	-
	CMr-GL	152	425223	0,005	0	3,24	3,50	92	6,74	0,02	0,34%	16,72%
	BeL-GL	165	448906	0,002	0	3,31	3,65	95	6,96	0,06	0,80%	14,04%
	hGPHH	152	471590	0,006	0	3,29	4,04	106	7,33	0,01	0,19%	9,43%
	WBRA-GL	198	495965	0,069	1,03	3,31	3,89	101	7,20	0,02	0,33%	10,26%
	VNS-Band (1s)	154	499797	1,054	44,51	3,25	3,68	97	6,93	0,01	0,09%	1,43%
	FNCHC	126	460851	2,297	2,22	3,25	3,78	98	7,03	0,04	0,62%	-15,14%
	GPS	140	406081	0,166	0,32	3,24	3,59	94	6,83	0,01	0,21%	13,63%
	GGPS	136	409134	0,397	0,56	3,31	3,61	94	6,92	0,01	0,07%	9,67%
	BL	233	417653	0,533	0	3,30	3,89	100	7,19	0,07	1,10%	4,65%
	CSS-Band	4709	8470608	0,588	19,59	2,54	5,51	132	8,05	0,04	0,61%	-6,64%
	10728	Sem renum.	10626	45314579	-	-	11,77	20,69	206	32,46	0,27	0,82%
CMr-GL		242	1504910	0,014	0	16,71	11,36	133	28,07	0,28	1,01%	13,48%
BeL-GL		241	1455453	0,007	0	15,86	11,77	137	27,63	0,11	0,42%	14,86%
hGPHH		249	1645616	0,015	0	15,32	12,96	152	28,28	0,10	0,34%	12,83%
WBRA-GL		300	1641665	0,203	2,58	15,74	12,44	145	28,18	0,22	0,77%	12,56%
VNS-Band (1s)		552	1746660	1,148	135,62	16,78	12,46	145	29,24	0,51	1,75%	6,38%
FNCHC		211	1646613	5,968	2,49	15,69	12,37	144	28,06	0,06	0,22%	-4,83%
GPS		207	1358676	0,658	1,54	16,60	11,40	133	28,00	0,11	0,38%	11,71%
GGPS		230	1540354	1,122	2,57	15,88	11,62	135	27,50	0,26	0,96%	11,82%
BL		398	1365197	5,895	0	15,82	12,27	141	28,09	0,06	0,22%	-4,70%
CSS-Band		10602	42658429	5,030	83,38	12,02	19,51	192	31,53	0,44	1,39%	-12,63%
23367		Sem renum.	23167	216212086	-	-	71,05	70,76	302	141,81	2,21	1,56%
	CMr-GL	367	4913698	0,05	0	93,52	36,80	192	130,32	0,45	0,35%	8,07%
	BeL-GL	355	4786316	0,02	0	93,43	38,12	198	131,55	0,73	0,56%	7,22%
	hGPHH	376	5521227	0,05	0	90,82	41,63	219	132,45	0,29	0,22%	6,57%
	WBRA-GL	671	5597225	0,71	7,73	92,58	39,99	208	132,57	0,78	0,59%	6,02%
	VNS-Band (1s)	1564	8889127	1,50	371,10	93,56	43,36	223	136,92	0,58	0,42%	2,39%
	FNCHC	309	5319151	15,32	2,96	92,59	40,00	208	132,59	0,97	0,73%	-4,30%
	GPS	293	4221479	3,65	3,34	93,42	37,32	194	130,74	0,45	0,35%	5,23%
	GGPS	317	5194437	5,22	3,51	92,75	38,86	202	131,61	0,11	0,08%	3,51%
	BL	465	4296542	11,19	0	95,57	41,64	213	137,21	0,69	0,51%	-4,65%
	CSS-Band	23127	203145287	38,63	549,09	69,96	66,75	276	136,71	2,80	2,05%	-23,64%
	50592	Sem renum.	50461	1020411959	-	-	383,76	220,27	431	604,03	0,83	0,14%
CMr-GL		553	16182346	0,11	0	479,33	115,46	278	594,79	4,01	0,67%	1,51%
BeL-GL		541	16420191	0,06	0	478,28	118,42	283	596,70	3,01	0,50%	1,20%
hGPHH		547	18159371	0,12	0	471,42	128,22	314	599,64	0,15	0,03%	0,71%
WBRA-GL		706	18363549	2,49	15,98	478,02	125,58	301	603,60	2,57	0,43%	-0,34%
VNS-Band (1s)		7377	47268027	2,89	967,10	490,37	130,10	309	620,47	0,69	0,11%	-3,20%
FNCHC		507	18166254	42,46	5,55	479,22	122,11	299	601,32	2,94	0,49%	-6,58%
GPS		466	14149442	24,22	5,51	480,09	115,99	276	596,07	4,22	0,71%	-2,69%
GGPS		497	16985652	48,51	7,72	478,83	120,15	289	598,98	2,99	0,50%	-7,20%
BL		961	14447154	493,21	0	494,69	124,66	291	619,35	2,17	0,35%	-84,19%
CSS-Band		50349	953617481	391,62	2701,00	400,99	216,97	398	617,91	4,38	0,71%	-67,13%

computacional do IC0 aumentou com as reduções de largura de banda e $profile$. Como a reordenação dos vértices altera a coerência de $cache$, possivelmente isso influenciou no custo computacional do IC0. Também, nota-se que o número de iterações apresentados na nona coluna da Tabela 11, diferentemente dos resultados apresentados na subseção 5.3.1, varia consideravelmente dependendo da heurística para redução de largura de banda utilizada.

Tabela 11 Conclusão.

n Heurística	β	$profile$	Heurística		IC0	MGC	MGC-IC0	σ	C_p	Red.	
			t(s)	m.(MiB)	t(s)	t(s) iter.	t(s)				
Sem renum.	108216	4725435534	-	-	1963,07	702,21	627	2665,28	3,56	0,13%	-
CMr-GL	911	55350648	0,25	0	2184,50	351,83	400	2536,34	6,95	0,27%	4,83%
BeL-GL	907	55649180	0,13	0	2166,44	357,33	408	2523,77	2,04	0,08%	5,31%
hGPHH	885	59355159	0,27	0	2177,27	399,50	452	2576,77	3,39	0,13%	3,31%
WBRA-GL	1764	61146321	9,31	26,55	2164,43	375,23	428	2539,66	3,87	0,15%	4,36%
VNS- <i>Band</i> (1s)	16676	157588720	8,65	2293,76	2203,43	395,10	442	2598,53	4,17	0,16%	2,18%
FNCHC	756	59671449	116,22	11,63	2164,09	376,48	430	2540,57	7,29	0,29%	0,32%
GPS	642	48744729	139,04	12,62	2167,24	354,53	402	2521,77	4,27	0,17%	0,17%
GGPS	743	54261170	226,47	23,33	2151,90	363,15	416	2515,06	2,10	0,08%	-2,86%
BL	1262	45836037	609,52	0	2258,58	388,54	428	2647,11	16,28	0,61%	-22,19%
CSS- <i>Band</i>	108332	110396292	471,52	596,52	2179,06	700,35	575	2879,42	12,76	0,44%	-25,73%
Sem renum.	231672	21652820640	-	-	9338,85	2167,06	908	11505,91	19,05	0,17%	-
CMr-GL	1231	168178362	0,55	0	9726,02	1080,58	573	10806,60	21,12	0,20%	6,07%
BeL-GL	1275	167101014	0,28	0	9731,16	1125,55	595	10856,70	8,85	0,08%	5,64%
hGPHH	1228	180118365	0,63	0	9810,81	1239,59	654	11050,40	3,65	0,03%	3,95%
WBRA-GL	1958	184101780	30,14	65,74	9751,17	1177,30	621	10928,47	13,14	0,12%	4,76%
VNS- <i>Band</i> (1s)	17036	313594920	33,43	5048,32	9941,82	1220,63	636	11162,45	17,78	0,16%	2,69%
FNCHC	1148	185662203	286,27	22,40	9728,52	1170,46	618	10898,98	19,92	0,18%	2,79%
GPS	1104	148697458	584,63	25,42	9757,33	1108,14	580	10865,47	13,71	0,13%	0,49%
GGPS	1190	172865013	1297,69	43,79	9694,34	1121,97	595	10816,30	11,49	0,11%	-5,28%
BL	2048	153659093	47698,40	0	10286,48	1219,04	621	11472,19	6,46	0,06%	-414,26%
CSS- <i>Band</i>	231315	2916471097	4486,55	841,35	11253,46	2207,07	830	13470,53	29,48	0,22%	-56,07%

Em geral, nos resultados apresentados na Tabela 11, se há reduções de largura de banda e de *profile*, o número de iterações do MGC-IC0 também diminuiu. Possivelmente, as reordenações geradas pelas heurísticas fazem com que a matriz preconditionadora gerada pelo IC0 seja melhor condicionada. Com isso, o MGC-IC0 converge com menos iterações e, consequentemente, com menor custo computacional.

Verifica-se, na coluna dos custos computacionais do IC0 da Tabela 11, que, com a exceção das duas primeiras instâncias, o custo computacional do preconditionador IC0 foi maior que o custo computacional do MGC preconditionado. Isso se deve ao comportamento quadrático do algoritmo do IC0 nessas instâncias.

Mostram-se, nas Figuras 16, 17 e 18, os gráficos dos percentuais de redução do custo computacional do MGC-IC0. Similarmente aos resultados com o MGCPJ mostrados na subseção 5.3.1, as heurísticas BL e CSS-*band* obtiverem os piores percentuais de redução do custo computacional do MGC-IC0.

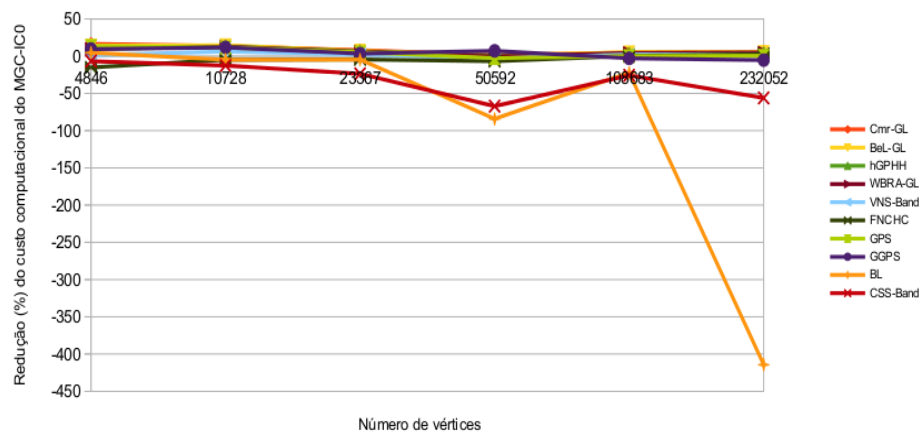


Figura 16 Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 de todas as 10 heurísticas testadas.

Analisando-se o gráfico da Figura 17, é possível considerar que as heurísticas GPS, GGPS, FNCHC e VNS-band obtiveram percentuais de redução do custo computacional do MGC-IC0 piores que os percentuais das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL. Observando-se a Figura 18, verifica-se que a heurística CMr-GL obteve resultados dos percentuais de redução do custo computacional do MGC-IC0 ligeiramente melhores que os resultados das demais heurísticas. Apenas nas instâncias com 10.728 e 108.683 vértices, a heurística CMr-GL não obteve o maior percentual de redução do custo do MGC-IC0. Ainda, observa-se que, na instância com 50.592 vértices, o percentual de redução do custo computacional MGC-IC0 foi menor dos que os percentuais apresentados nas outras instâncias. Isso ocorreu porque, provavelmente, as reordenações geradas por essas quatro heurísticas não geraram boa coerência de *cache* e localidade e memória como nas outras instâncias.

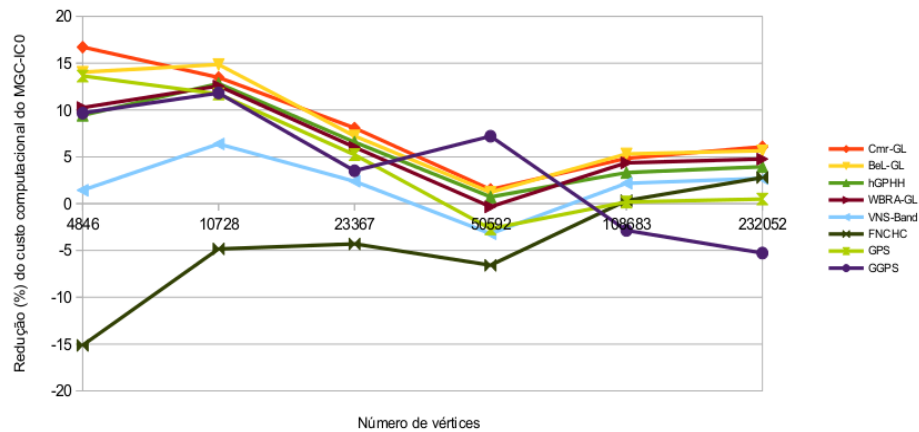


Figura 17 Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 sem os resultados das heurísticas BL e CSS-Band.

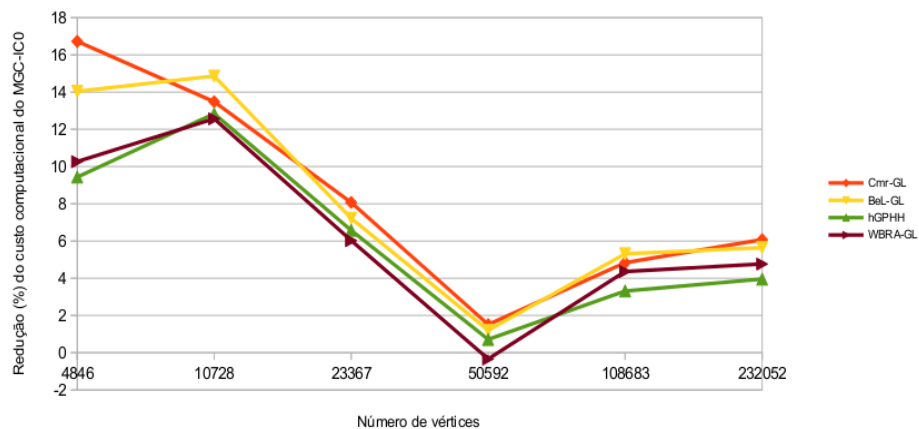


Figura 18 Gráfico das percentuais das reduções do custo computacional do MGC-IC0 das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.

5.3.3 Ocupações de memória

Como descrito, as heurísticas CMr-GL, BeL-GL, hGPHH e BL não apresentaram ocupações de memória significativas. Então, mostram-se, nesta subseção, os gráficos dos consumos das heurísticas WBRA-GL, VNS-band, FNCHC, GPS, GGPS e CSS-band. A ocupação de memória dessas

heurísticas é a mesma nos testes com o MGCPJ e com o MGC-IC0. Embora nos testes com o MGCPJ é utilizada uma estrutura de dados diferente da estrutura de dados utilizada nos testes com o MGC-IC0, as heurísticas utilizam, para ambos os testes, a mesma estrutura de dados (para detalhes, veja a subseção 3.4.1). Com isso, o mesmo consumo de memória é observado nos dois testes.

O gráfico das ocupações de memória das heurísticas WBRA-GL, VNS-*band*, FNCHC, GPS, GGPS e CSS-*band* é mostrado na Figura 19. Pode-se observar que a heurística VNS-*band* é a heurística com o maior consumo de memória. Na heurística CSS-*band*, utilizou-se diferentes números de PCs. Isso justifica o comportamento anormal de ocupação de memória da heurística CSS-*band*.

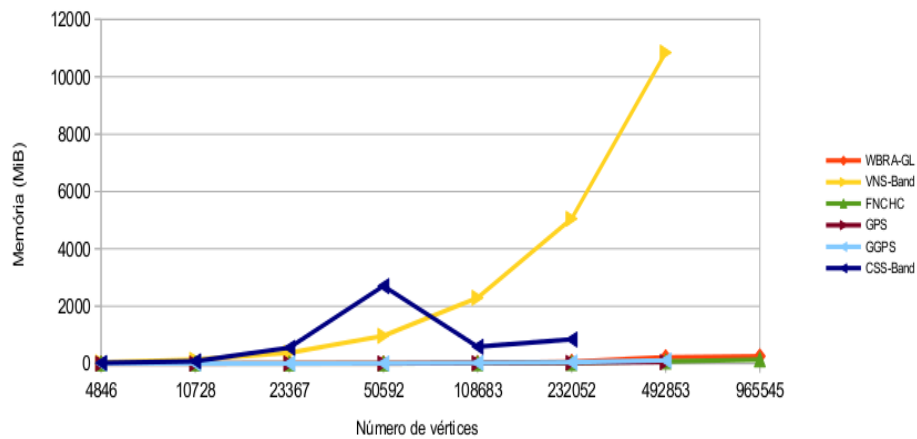


Figura 19 Gráfico das ocupações de memória, em *mebibytes*, das heurísticas WBRA-GL, VNS-*Band*, FNCHC, GPS, GGPS e CSS-*Band* nas instâncias oriundas da discretização da equação da condução do calor.

Na Figura 20, tem-se o mesmo gráfico de ocupações de memória mostrado na Figura 19. Porém, na Figura 20, não são apresentadas as ocupações de memória das heurísticas VNS-*band* e CSS-*band*. Com isso, é

possível observar, com mais detalhes, os comportamentos das ocupações de memória das heurísticas WBRA-GL, FNCHC, GPS e GGPS. Entre essas heurísticas, a WBRA-GL é a heurística que apresentou a maior ocupação de memória.

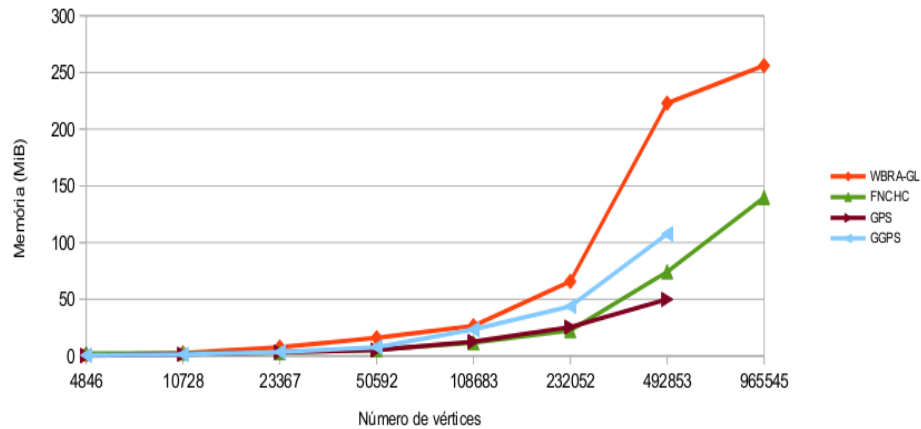


Figura 20 Gráfico das ocupações de memória, em *mebibytes*, das heurísticas WBRA-GL, FNCHC, GPS e GGPS nas instâncias oriundas da discretização da equação da condução do calor.

5.3.4 Comparação dos resultados dos resolutores MGCPJ e MGC-IC0

Apresenta-se, na Figura 21, o gráfico dos custos computacionais, em segundos, do MGCPJ e do MGC-IC0. Como só foi possível obter, até a data da entrega deste trabalho, resultados com o MGC-IC0 com instâncias de até 232.052 vértices, os resultados do MGCPJ, com as instâncias com 492.853 e 965.545 vértices, não são mostrados.

Pode-se considerar que a heurística CMr-GL obteve as melhores reduções do custo computacional do MGC-IC0. Em relação às simulações com o MGCPJ, não se pode afirmar que a heurística CMr-GL foi a

melhor heurística na redução do custo computacional do MGCPJ. Porém, a heurística CMr-GL foi uma das melhores juntamente com as heurísticas BeL-GL, hGPHH e WBRA-GL. Com isso, os resultados mostrados na Figura 21 são referentes aos resolvers MGCPJ e do MGC-IC0 com reduções de largura e banda e de *profile* pela heurística CMr-GL.

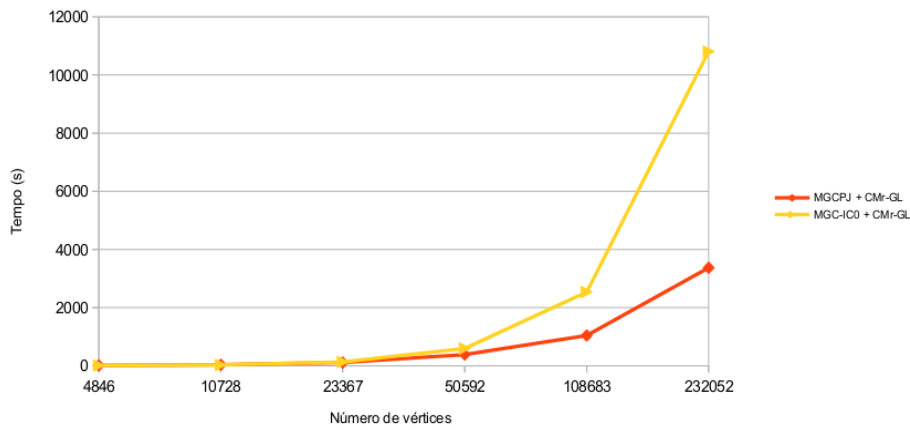


Figura 21 Custos computacionais, em segundos, nas instâncias com até 232052 vértices, do MGCPJ e do MGC-IC0. Mostram-se os resultados dos resolvers MGCPJ e MGC-IC0 com as reduções de largura de banda e *profile* pela heurísticas CMr-GL.

Pode-se observar na Figura 21, que o custo computacional do MGC-IC0 + CMr-GL foi maior que o custo computacional do MGCPJ + CMr-GL. Isso ocorreu por causa do alto custo computacional do IC0. A redução no número de iterações do MGC pelo IC0 não compensou o custo computacional do IC0.

5.4 Simulações com instâncias oriundas da discretização equação de Laplace

Nesta seção, apresentam-se os resultados dos testes computacionais na resolução dos SELs resultantes da discretização da equação de Laplace por volumes finitos. Foram utilizados dois conjuntos de instâncias: SELs numerados aleatoriamente e SELs numerados pela curva de Sierpiński modificada (OLIVEIRA; KISCHINHEVSKY, 2008).

Na subseção 5.4.1, apresentam-se os resultados dos testes computacionais dos SELs numerados aleatoriamente. Os resultados dos testes computacionais dos SELs numerados pela curva de Sierpiński modificada são mostrados na subseção 5.4.2.

5.4.1 Instâncias numeradas de forma aleatória

Nos testes apresentados nesta subseção, utilizou-se sete SELs numerados de forma aleatória, com tamanhos de 7.322 a 277.118 incógnitas, provenientes da discretização da equação de Laplace por volumes finitos. Executou-se cada heurística e o resolutor três vezes em cada instância.

Na subseção 5.4.1.1, são mostrados os resultados das simulações do MGCPJ com a redução de largura de banda. Os resultados com o MGC-IC0, com a redução de largura de banda, são apresentados na subseção 5.4.1.2. As ocupações de memória das heurísticas são detalhadas na subseção 5.4.1.3. Comparações dos resultados dos resolvers MGCPJ e MGC-IC0 são realizadas na subseção 5.4.1.4.

5.4.1.1 Simulações com o método dos gradientes conjugados precondicionado pelo método de Jacobi

Nesta subseção, são apresentados os resultados da resolução de SELs, oriundos da discretização da equação de Laplace obtida de forma aleatória pelo MGCPJ com as reduções de largura de banda e de *profile*. Executou-se cada heurística e o resolutor MGCPJ três vezes. Os resultados médios dessas execuções são apresentados na Tabela 12.

Em relação à heurística VNS-*band*, como pode ser observado na Tabela 12, em cada instância, foram utilizados diferentes tempos de execução. Porém, em todas as oito instâncias, os melhores resultados da heurística VNS-*Band*, em relação à redução do custo computacional do MGCPJ, foram obtidos com um segundo.

Nos resultados apresentados na Tabela 12, a heurística que mais vezes obteve a melhor redução de largura de banda foi a heurística FNCHC. A heurística FNCHC só não obteve a melhor redução de largura de banda em duas das seis instâncias. Em relação ao *profile*, a heurística que mais vezes obteve o menor *profile* foi a heurística BL. De forma análoga aos resultados apresentados anteriormente, a heurística mais rápida foi a heurística BeL-GL.

Em relação à ocupação de memória, as heurísticas CMr-GL, BeL-GL, hGPHH e BL não apresentaram resultados significativos. Observa-se, ainda, que as heurísticas com maiores ocupações de memória são as heurísticas VNS-*band* e CSS-*band*.

Diferentemente dos resultados apresentados na seção 5.3, não se obteve redução no custo computacional do MGCPJ com as reduções de largura de banda e de *profile*. Verifica-se, na última coluna da Tabela 12,

Tabela 12 Resultados referentes às execuções do MGCPJ e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.

n	Heurística	β	$profile$	Heurística		MGCPJ		σ	C_v	Redução
				t(s)	m.(MiB)	iter.	t(s)			
7322	Sem renum.	7248	16083808	-	-	498	10,39	0,02	0,16%	-
	CMr-GL	80	396652	0,005	0	498	11,12	0,04	0,37%	-7,07%
	BeL-GL	80	396709	0,003	0	498	12,54	0,07	0,55%	-20,72%
	hGPHH	80	406461	0,005	0	498	11,18	0,03	0,29%	-7,65%
	WBRA-GL	152	411821	0,052	0,73	498	12,61	0,08	0,61%	-21,85%
	VNS-Band	(1s) 1599	966638	1,061	75,90	498	11,40	0,31	2,73%	-19,92%
		(2s) 1599	966638	2,06	75,90	498	11,65	0,01	0,02%	-31,95%
	FNCHC	75	444803	2,273	0,54	498	12,51	0,05	0,36%	-42,25%
	GPS	78	404414	0	0,24	498	12,16	0,40	3,26%	-20,89%
	GGPS	79	397534	0,415	1,28	498	12,46	0,04	0,30%	-23,87%
	BL	152	407458	0,189	0	498	12,77	0,03	0,20%	-24,74%
	CSS-Band	7190	16103602	1,079	40,91	498	13,57	0,06	0,48%	-41,00%
15944	Sem renum.	15902	76482022	-	-	745	33,74	0,21	0,63%	-
	CMr-GL	121	1149442	0,02	0	745	38,65	0,04	0,10%	-14,61%
	BeL-GL	122	1149179	0,01	0	745	42,96	0,005	0,01%	-27,36%
	hGPHH	124	1231692	0,02	0	745	38,56	0,29	0,76%	-14,34%
	WBRA-GL	175	1271533	0,11	1,29	745	36,08	0,15	0,42%	-7,26%
	VNS-Band	(1s) 3916	5108940	1,19	196,70	745	40,01	0,49	1,23%	-22,11%
		(2s) 3916	5108940	2,19	196,70	745	39,01	0,17	0,45%	-22,11%
		(3s) 3916	5108940	3,20	196,70	745	39,42	0,01	0,01%	-26,31%
	FNCHC	113	1321180	5,85	1,26	745	42,89	0,11	0,26%	-44,46%
	GPS	118	1154030	1,58	0,51	745	41,63	1,38	3,31%	-28,06%
	GGPS	118	1210195	3,55	2,83	745	42,85	0,07	0,17%	-37,52%
	BL	212	1144254	1,09	0	745	43,65	0,07	0,16%	-32,60%
CSS-Band	15749	77021429	8,56	192,70	745	50,72	0,33	0,65%	-75,70%	
34238	Sem renum.	34059	357518296	-	-	1069	105,36	0,41	0,39%	-
	CMr-GL	194	3411077	0,04	0	1069	113,75	0,90	0,79%	-8,00%
	BeL-GL	192	3413781	0,02	0	1069	122,25	0,32	0,26%	-16,05%
	hGPHH	192	3759478	0,04	0	1069	113,30	0,11	0,10%	-7,57%
	WBRA-GL	272	3750387	0,29	2,9	1069	113,50	0,54	0,48%	-8,00%
	VNS-Band	(1s) 2726	6767128	2,66	490,55	1069	116,21	1,23	1,06%	-12,82%
		(3s) 2726	6767128	3,66	490,55	1069	123,07	0,31	0,25%	-20,28%
	FNCHC	192	3913543	15,44	3,95	1069	121,94	0,38	0,31%	-30,39%
	GPS	191	3545656	9,72	1,57	1069	118,49	0,59	0,50%	-21,69%
	GGPS	170	3415253	19,69	5,15	1069	121,60	0,11	0,09%	-34,10%
	BL	334	3282297	4,31	0	1069	125,07	0,29	0,23%	-22,80%
	CSS-Band	33923	359144453	67,28	910,68	1069	154,73	0,63	0,41%	-110,71%
75542	Sem renum.	75490	1744941733	-	-	1540	327,95	0,38	0,11%	-
	CMr-GL	274	12086129	0,09	0	1540	361,89	0,62	0,17%	-10,38%
	BeL-GL	273	12107123	0,04	0	1540	372,46	0,81	0,22%	-13,58%
	hGPHH	277	13793938	0,10	0	1540	360,27	0,22	0,06%	-9,89%
	WBRA-GL	436	13766127	0,88	6,72	1540	372,62	1,02	0,27%	-13,89%
	VNS-Band	(1s) 21310	42564399	3,70	1198,13	1540	364,72	1,35	0,37%	-12,34%
		(4s) 21310	42564399	6,66	1198,13	1540	383,35	0,53	0,14%	-18,92%
	FNCHC	269	13978910	40,81	7,20	1540	360,65	1,66	0,46%	-22,42%
	GPS	272	12086603	41,53	3,60	1540	371,00	1,52	0,41%	-25,79%
	GGPS	271	12405895	86,66	10,31	1540	390,82	0,44	0,11%	-45,60%
	BL	460	11175444	42,51	0	1540	395,90	0,64	0,16%	-33,68%
	CSS-Band	74879	1747422045	692,60	3819,18	1540	496,82	1,33	0,27%	-262,68%

que as renumerações geradas por todas as heurísticas aumentaram o custo computacional do MGCPJ em todas as instâncias.

Era esperado que, com as reduções de largura de banda e de $profile$, fossem obtidas reduções do custo computacional do MGCPJ. Especificamente, os SELs apresentados nesta subseção foram montados de

Tabela 12 Conclusão.

n	Heurística	β	<i>profile</i>	Heurística		MGCPJ		σ	C_v	Redução
				t(s)	m.(MiB)	iter.	t(s)			
101780	Sem renum.	101583	3169282786	-	-	2173	630,79	0,36	0,06%	-
	CMr-GL	405	21399542	0,13	0	2173	682,61	1,81	0,26%	-8,24%
	BeL-GL	404	21404400	0,06	0	2173	699,69	0,09	0,01%	-10,93%
	hGPHH	407	24041332	0,14	0	2173	684,12	0,30	0,04%	-8,48%
	WBRA-GL	697	24026871	1,53	7,50	2173	698,40	0,32	0,05%	-10,96%
	VNS-Band (1s)	5207	25033097	5,58	1638,40	2173	685,04	2,07	0,30%	-9,49%
	(6s)	5207	25033097	10,63	1638,40	2173	737,53	0,86	0,12%	-18,61%
	FNCHC	391	26974311	59,59	9,82	2173	698,63	1,41	0,20%	-20,20%
	GPS	405	21399542	73,12	4,89	2173	693,82	4,57	0,66%	-21,58%
	GGPS	400	21727818	153,20	15,46	2173	734,50	8,73	1,19%	-40,73%
	BL	745	19394495	4385,66	0	2173	741,47	0,50	0,07%	-712,81%
CSS-Band	101333	3160566736	1638,03	611,37	2173	944,45	2,03	0,22%	-309,40%	
192056	Sem renum.	191738	11329772559	-	-	2383	1304,62	0,72	0,06%	-
	CMr-GL	360	42578191	0,24	0	2382	1436,77	3,39	0,24%	-10,15%
	BeL-GL	363	42634929	0,12	0	2383	1471,47	0,22	0,02%	-12,79%
	hGPHH	364	48308977	0,27	0	2383	1429,24	0,12	0,01%	-9,57%
	WBRA-GL	617	48295345	3,25	13,65	2383	1466,81	2,59	0,18%	-12,68%
	VNS-Band (1s)	11142	99018771	16,49	3194,93	2383	1442,61	6,79	0,47%	-11,84%
	(17s)	11142	99018771	32,50	3194,93	2383	1535,75	0,87	0,06%	-20,21%
	FNCHC	348	48496246	114,76	20,96	2383	1469,44	1,65	0,11%	-21,43%
	GPS	371	41541059	256,01	10,05	2383	1474,96	3,57	0,24%	-32,68%
	GGPS	363	42925208	530,82	27,84	2383	1467,54	0,63	0,04%	-53,18%
	BL	621	40149530	349,55	0	2383	1580,14	0,19	0,01%	-47,91%
CSS-Band	191446	2737568773	793,50	1124,89	2383	1998,90	5,49	0,27%	-114,04%	
277118	Sem renum.	277019	23512579029	-	-	2771	2235,58	2,78	0,12%	-
	CMr-GL	421	74726891	0,35	0	2771	2382,51	5,18	0,22%	-6,69%
	BeL-GL	424	74852621	0,17	0	2771	2428,08	0,76	0,03%	-8,62%
	hGPHH	427	84714895	0,38	0	2771	2328,01	3,04	0,13%	-4,15%
	WBRA-GL	713	84585649	6,20	18,47	2771	2420,37	0,21	0,01%	-8,54%
	VNS-Band (1s)	12132	97666318	32,12	4618,20	2771	2397,35	6,11	0,26%	-8,67%
	(33s)	12132	97666318	64,18	4618,20	2771	2516,44	0,54	0,02%	-15,43%
	FNCHC	424	86076670	183,16	26,65	2771	2425,58	0,74	0,03%	-16,69%
	GPS	399	72378558	510,31	16,37	2771	2458,52	12,28	0,50%	-32,80%
	GGPS	420	75610158	1054,65	20,70	2771	2585,99	7,44	0,29%	-62,85%
	BL	793	66880423	401,85	0	2771	2613,54	0,85	0,03%	-34,88%
CSS-Band	276285	23509305627	1606,08	1679,64	2771	3314,23	26,97	0,81%	-120,09%	

forma aleatória. Com isso, esperava-se que, com a reordenações realizadas pelas heurísticas para a redução de largura de banda, fossem obtidas mais coerência de *cache*. Porém, aparentemente, as ordenações originais das instâncias possuem mais coerência de *cache*.

5.4.1.2 Simulações com o método dos gradientes conjugados preconditionado pelo método incompleto de Cholesky com zero *fill-in*

Nesta subseção, são apresentados os resultados da resolução de SELs oriundos da discretização da equação de Laplace obtida de forma aleatória pelo MGC-IC0 com as reduções de largura de banda e de *profile*. Executou-

se cada heurística e o MGC-IC0 três vezes. Os resultados médios dessas execuções são apresentados na Tabela 13.

Em relação à heurística VNS-*band*, como pode ser observado na Tabela 13, em cada instância, foram utilizados diferentes tempos de execução. Porém, na maioria das instâncias, os melhores resultados da heurística VNS-*Band*, em relação à redução do custo computacional do MGCPJ, foram obtidos com um segundo. Somente na instância com 192.056 vértices a melhor redução do custo computacional do MGC-IC0, pela heurística VNS-*band*, não foi obtida com 1s, mas com 36s.

Como pode ser observado na Tabela 13, de forma análoga aos resultados do MGCPJ mostrados na subseção 5.4.1.1, a heurística que mais vezes obteve a melhor redução de largura de banda foi a FNCHC. Também, a heurística que mais reduziu o *profile* foi a BL e a heurística mais rápida foi a BeL-GL. Em relação ao consumo de memória, os mesmos resultados dos testes com o MGCPJ, mostrados na subseção 5.4.1.1, foram obtidos.

Observa-se que, com as reduções de largura de banda e de *profile*, o número de iterações do MGC-IC0 reduziu consideravelmente. Possivelmente, as reordenações geradas pelas heurísticas fazem com que a matriz preconditionadora gerada pelo IC0 seja melhor condicionada. Com isso, o MGC-IC0 converge com menos iterações e, conseqüentemente, com menor custo computacional. Por isso, diferentemente dos resultados apresentados na subseção 5.4.1.1, houve redução no custo computacional do MGC-IC0 com as reduções de largura de banda e de *profile*.

As renumerações dos vértices, pelas heurísticas, também influenciam no custo computacional do IC0 e não somente do MGC preconditionado. Como a reordenação dos vértices altera a coerência de *cache*, possivelmente

Tabela 13 Resultados referentes às execuções do MGC-IC0 e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.

<i>n</i>	Heurística	β	<i>profile</i>	Heurística		IC0 t(s)	MGC t(s)	MGC-IC0 iter.	σ	C_v	Red.	
				t(s)	m.(MiB)							
7322	Sem renum.	7248	16083808	-	-	3,36	7,66	271	11,01	0,01	0,05%	-
	CMr-GL	80	397878	0,005	0	4,17	5,19	201	9,36	0,01	0,12%	14,94%
	BeL-GL	80	398377	0,003	0	4,16	5,56	202	9,72	0,01	0,06%	11,69%
	hGPHH	80	406461	0,006	0	4,18	5,36	208	9,55	0,02	0,24%	13,21%
	WBRA-GL	154	412219	0,051	0,73	4,18	5,64	205	9,82	0,04	0,43%	10,25%
	VNS-Band (1s)	1599	966638	1,061	75,90	4,15	5,59	214	9,74	0,07	0,67%	1,90%
	(2s)	1599	966638	2,068	75,90	4,16	5,53	214	9,68	0,03	0,30%	-6,70%
	FNCHC	71	423722	2,222	0,54	4,16	5,63	205	9,79	0,02	0,18%	-9,08%
	GPS	78	404414	0,362	0,24	4,18	5,46	203	9,65	0,11	1,12%	10,17%
	GGPS	80	403391	0,323	1,28	4,17	5,61	204	9,78	0,02	0,27%	8,26%
	BL	152	407458	0,151	0	4,12	5,55	201	9,67	0,01	0,11%	10,81%
	CSS-Band	7185	15979344	0,870	40,91	3,36	7,69	271	11,05	0,10	0,87%	-8,27%
15944	Sem renum.	15902	76482022	-	-	19,38	25,79	396	45,17	0,08	0,17%	-
	CMr-GL	120	1149124	0,02	0	24,45	16,56	290	41,01	0,04	0,11%	9,37%
	BeL-GL	124	1148009	0,01	0	24,58	17,87	291	42,45	0,04	0,09%	6,21%
	hGPHH	121	1219929	0,02	0	24,20	17,26	304	41,47	0,07	0,16%	8,35%
	WBRA-GL	176	1233347	0,11	1,29	24,71	18,76	305	43,46	0,04	0,10%	3,75%
	VNS-Band (1s)	3916	5108940	1,18	196,70	24,29	18,72	319	43,10	0,28	0,64%	2,19%
	(2s)	3916	5108940	2,19	196,70	24,18	18,24	319	42,42	0,19	0,45%	1,24%
	FNCHC	113	1315311	5,73	1,26	24,50	18,47	301	42,97	0,13	0,31%	2,47%
	GPS	118	1154030	3,61	0,51	24,62	17,42	290	42,04	0,59	1,40%	-0,84%
	GGPS	117	1210422	1,79	2,83	24,66	18,57	302	43,22	0,21	0,48%	0,57%
	BL	212	1144254	1,09	0	24,99	17,95	291	42,95	0,14	0,33%	2,72%
	CSS-Band	15724	76687757	8,06	192,70	18,64	25,60	395	44,24	0,56	1,26%	-15,53%
34238	Sem renum.	34609	357518296	-	-	108,04	81,50	565	189,53	0,60	0,32%	-
	CMr-GL	192	3413888	0,04	0	125,85	50,13	411	175,98	0,47	0,27%	7,13%
	BeL-GL	191	3416318	0,02	0	126,75	52,36	412	179,11	0,75	0,42%	5,49%
	hGPHH	193	3756415	0,04	0	125,15	52,87	434	178,02	0,03	0,02%	6,05%
	WBRA-GL	267	3750738	0,29	2,9	127,10	54,89	431	181,99	0,67	0,37%	3,83%
	VNS-Band (1s)	2726	6767128	1,67	490,55	127,16	54,85	444	182,01	0,31	0,17%	3,09%
	(2s)	2726	6767128	2,66	490,55	134,93	54,20	444	189,13	0,14	0,07%	-1,19%
	FNCHC	170	3910584	15,03	3,95	126,38	55,36	436	181,73	0,76	0,42%	-3,81%
	GPS	191	3545656	9,61	1,57	126,88	52,55	418	179,43	0,80	0,45%	0,26%
	GGPS	192	3415253	19,63	5,15	125,86	52,04	411	177,90	0,36	0,21%	-4,22%
	BL	334	3282297	4,30	0	129,89	53,82	419	183,70	0,01	0,09%	0,81%
	CSS-Band	33953	359290792	68,63	910,68	107,35	81,42	562	188,77	2,80	1,48%	-35,81%
75542	Sem renum.	75490	1744941733	-	-	590,68	268,22	815	858,91	1,15	0,13%	-
	CMr-GL	275	12096902	0,09	0	621,62	164,43	607	786,05	0,65	0,08%	8,47%
	BeL-GL	275	12128940	0,04	0	624,21	168,84	616	793,05	1,40	0,18%	7,66%
	hGPHH	277	13793938	0,10	0	624,75	179,41	663	804,16	0,14	0,02%	6,36%
	WBRA-GL	459	13739132	0,87	6,72	627,12	175,33	639	802,45	1,52	0,19%	6,47%
	VNS-Band (1s)	21310	42564399	3,69	1198,13	636,68	176,04	642	812,72	2,15	0,26%	4,95%
	(4s)	21310	42564399	6,69	1198,13	628,72	175,63	642	804,35	0,73	0,09%	5,57%
	(7s)	21310	42564399	10,72	1198,13	650,73	174,77	642	825,50	0,95	0,12%	2,64%
	FNCHC	266	13848181	41,49	7,20	629,77	176,63	644	806,40	2,68	0,33%	1,28%
	GPS	272	12086603	41,44	3,60	623,99	165,54	607	789,52	0,42	0,05%	3,25%
	GGPS	271	12405895	86,21	10,31	625,03	166,47	604	791,50	1,45	0,18%	-2,19%
	BL	460	11175444	42,77	0	642,44	171,11	612	813,55	0,83	0,10%	0,30%
CSS-Band	75031	1747217131	692,02	3819,18	590,65	268,80	819	859,44	7,56	0,88%	-80,63%	

isso influenciou no custo computacional do IC0.

A heurística CMr-GL obteve os melhores resultados em relação à redução do computacional MGC-IC0. Isso é mostrado nas Figuras 22, 23 e 24. Na Figura 22, nota-se que as heurísticas CSS-band e BL foram as heurísticas com as piores reduções do custo computacional do MGC-IC0.

Tabela 13 Conclusão.

n	Heurística	β	profile	Heurística		IC0	MGC	MGC-IC0	σ	C_v	Red.	
				t(s)	m.(MiB)							t(s)
101780	Sem renum.	101583	3169282786	-	-	1093,56	405,13	907	1498,68	0,13	0,01%	-
	CMr-GL	407	21336387	0,13	0	1126,20	248,99	670	1375,19	0,64	0,05%	8,23%
	BeL-GL	406	21346316	0,06	0	1127,17	253,62	676	1380,79	0,71	0,05%	7,86%
	hGPHH	408	23967214	0,13	0	1123,87	265,42	716	1389,29	1,81	0,13%	7,29%
	WBRA-GL	697	24026871	1,53	7,50	1128,75	262,94	701	1391,70	2,27	0,16%	7,04%
	VNS-Band (1s)	5207	25033097	5,58	1638,40	1139,72	255,56	686	1395,28	0,74	0,05%	6,53%
	VNS-Band (6s)	5207	25033097	14,67	1638,40	1134,55	252,38	686	1386,93	2,51	0,18%	6,48%
	VNS-Band (15s)	5207	25033097	21,74	1638,40	1133,88	252,30	686	1386,18	2,67	0,19%	6,06%
	FNCHC	399	25629738	59,89	9,82	1133,36	264,52	705	1397,88	0,97	0,07%	2,73%
	GPS	405	21399542	73,37	4,89	1124,59	249,59	668	1374,18	3,46	0,25%	3,41%
	GGPS	402	21624122	152,69	15,46	1128,08	250,57	665	1378,65	2,65	0,19%	-2,18%
	BL	745	19394495	4383,70	0	1151,55	262,36	683	1413,92	4,32	0,31%	-286,84%
CSS-Band	101300	3161769330	227,12	611,37	1093,12	405,07	906	1498,20	9,21	0,61%	-15,12%	
192056	Sem renum.	191738	11329772559	-	-	4115,15	1045,04	1215	5160,18	10,96	0,21%	-
	CMr-GL	360	42577946	0,24	0	4029,35	641,39	909	4670,73	3,90	0,08%	9,48%
	BeL-GL	360	42627911	0,11	0	4036,74	659,31	927	4696,05	4,22	0,09%	8,99%
	hGPHH	364	48325681	0,26	0	4056,77	704,00	1000	4760,77	3,95	0,08%	7,74%
	WBRA-GL	616	48302407	3,26	13,65	4062,07	695,39	975	4757,46	3,26	0,07%	7,74%
	VNS-Band (1s)	11142	99018771	16,56	3194,93	4055,66	704,80	992	4760,46	8,18	0,17%	7,43%
	VNS-Band (17s)	11142	99018771	35,42	3194,93	4042,63	696,15	992	4738,78	6,55	0,14%	7,48%
	VNS-Band (36s)	11142	99018771	53,51	3194,93	4018,57	689,57	992	4708,15	3,75	0,08%	7,72%
	VNS-Band (54s)	11142	99018771	72,55	3194,93	4045,34	693,92	992	4739,26	3,93	0,08%	5,98%
	FNCHC	344	48743268	115,23	20,96	4055,82	699,79	982	4755,61	7,46	0,16%	5,61%
	GPS	371	41541059	257,39	10,05	4047,62	648,79	911	4696,41	7,84	0,17%	4,00%
	GGPS	363	42925208	530,00	27,84	4040,40	644,60	904	4685,00	1,46	0,03%	-1,06%
BL	621	40149530	349,47	0	4142,02	681,53	934	4823,55	17,22	0,36%	-0,25%	
CSS-Band	191322	2731669814	1345,01	1124,89	4101,31	1046,61	1211	5147,92	15,01	0,29%	-25,83%	
277118	Sem renum.	277019	23512579029	-	-	8562,78	1765,92	1430	10328,70	3,22	0,03%	-
	CMr-GL	420	74697812	0,34	0	8371,11	1075,95	1066	9447,06	7,03	0,07%	8,53%
	BeL-GL	419	74759303	0,17	0	8387,01	1095,76	1078	9482,77	2,16	0,02%	8,19%
	hGPHH	425	84682829	0,38	0	8415,59	1177,67	1168	9593,26	5,38	0,06%	7,12%
	WBRA-GL	714	84487732	6,16	18,47	8421,79	1163,83	1146	9585,62	3,27	0,03%	7,13%
	VNS-Band (1s)	12132	97666318	32,30	4618,20	8430,11	1149,17	1134	9579,28	11,61	0,12%	6,94%
	VNS-Band (33s)	12132	97666318	65,12	4618,20	9009,83	1133,83	1134	10143,66	15,71	0,15%	1,16%
	FNCHC	412	85849403	182,40	26,65	8418,54	1161,37	1144	9579,91	8,70	0,09%	5,48%
	GPS	399	72378558	511,51	16,37	8395,41	1087,38	1070	9482,79	23,37	0,25%	3,24%
	GGPS	422	75150257	1051,71	20,70	8411,78	1068,69	1047	9480,48	6,09	0,06%	-1,97%
	BL	793	66880423	397,85	0	8639,69	1138,35	1093	9778,04	2,86	0,03%	1,48%
	CSS-Band	276240	2054061697	3994,77	1679,64	8579,13	1764,89	1421	10344,02	46,55	0,45%	-38,82%

Removeu-se, da Figura 24, os resultados das heurísticas CSS-band e BL, com isso, pode-se verificar os resultados das oito heurísticas restantes. Observa-se que, as heurísticas GGPS, GPS, FNCHC e VNS-band obtiveram resultados piores que os resultados das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.

Mostra-se, na Figura 23, os resultados das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL em relação aos percentuais das reduções dos custos computacionais do MGC-IC0. Observa-se, claramente, que a heurística CMr-GL obteve as melhores reduções seguida pelas heurísticas BeL-GL, hGPHH e WBRA-GL.

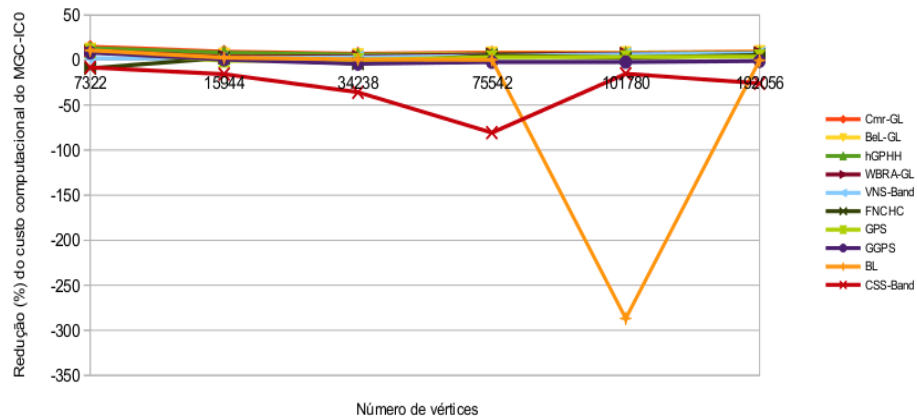


Figura 22 Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 de todas as 10 heurísticas testadas.

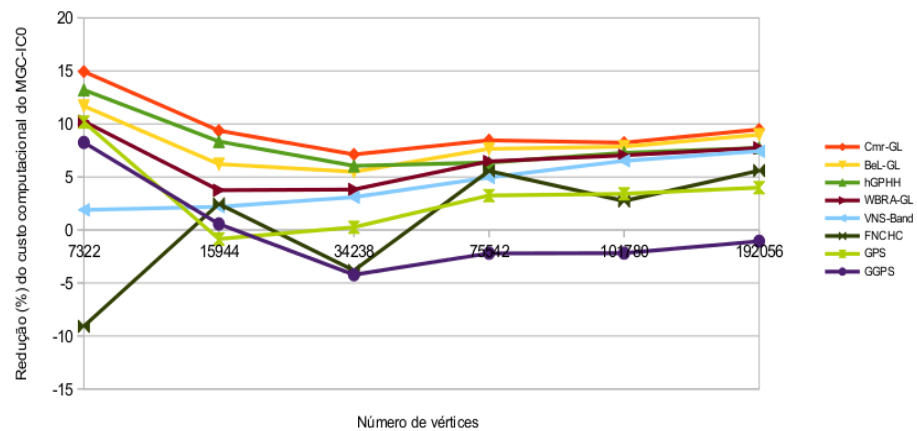


Figura 23 Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 sem os resultados das heurísticas BL e CSS-Band.

5.4.1.3 Ocupações de memória

Como descrito, as heurísticas CMr-GL, BeL-GL, hGPHH e BL não apresentaram ocupações de memória significativas. Então, mostram-se, nesta subseção, os gráficos dos consumos das heurísticas WBRA-GL, VNS-band, FNCHC, GPS, GGPS e CSS-band. A ocupação de memória dessas

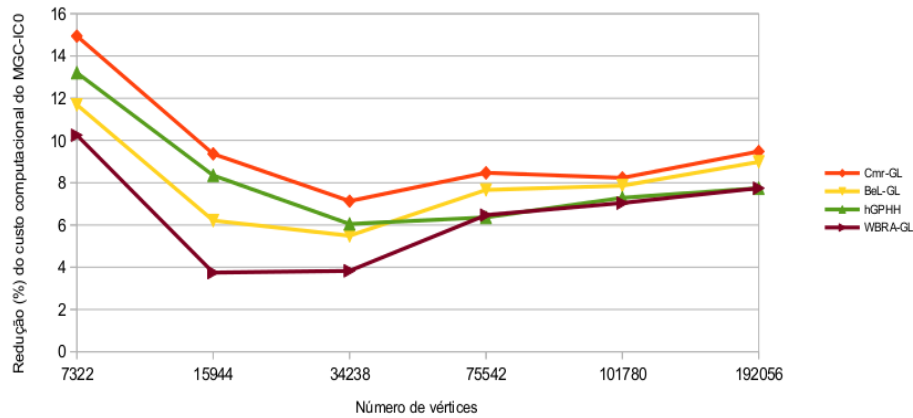


Figura 24 Gráfico das percentuais das reduções do custo computacional do MGC-IC0 das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.

heurísticas é a mesma nos testes com o MGCPJ e com o MGC-IC0. Embora nos testes com o MGCPJ seja utilizada uma estrutura de dados diferente da estrutura de dados utilizada nos testes com o MGC-IC0, as heurísticas utilizam, para ambos os testes, a mesma estrutura de dados (para detalhes, veja a subseção 3.4.1). Com isso, o mesmo consumo de memória é observado nos dois testes.

O gráfico das ocupações de memória das heurísticas WBRA-GL, VNS-band, FNCHC, GPS, GGPS e CSS-band é mostrado na Figura 25. Pode-se observar que a heurística VNS-band, é a heurística com o maior consumo de memória. Na heurística CSS-band, utilizou-se diferentes números de PCs. Isso justifica o comportamento anormal de ocupação de memória da heurística CSS-band.

Na Figura 26, tem-se o mesmo gráfico de ocupações de memória mostrado na Figura 25, porém não são apresentados as ocupações de memória das heurísticas VNS-band e CSS-band. Com isso, é possível observar com mais detalhes os comportamentos das ocupações de memória

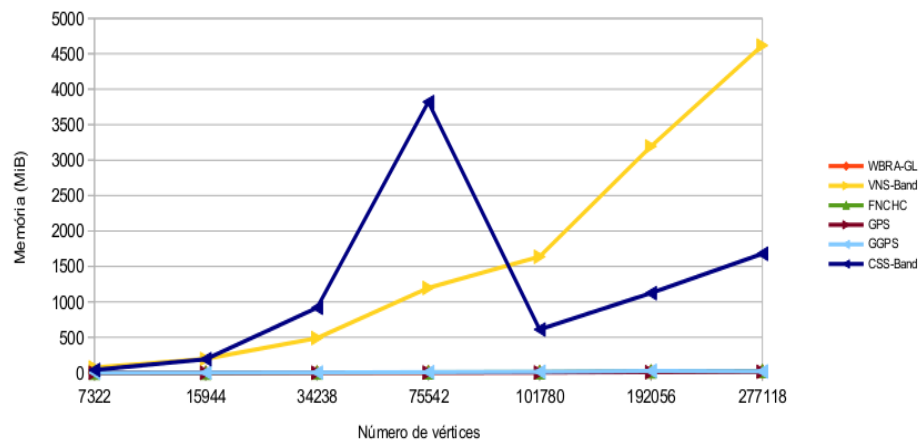


Figura 25 Gráficos das ocupações de memória, em *mebibytes*, das heurísticas WBRA-GL, VNS-Band, FNCHC, GPS, GGPS e CSS-Band nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.

das heurísticas WBRA-GL, FNCHC, GPS e GGPS. Entre essas heurísticas a heurística GGPS é a heurística que apresentou a maior ocupação de memória.

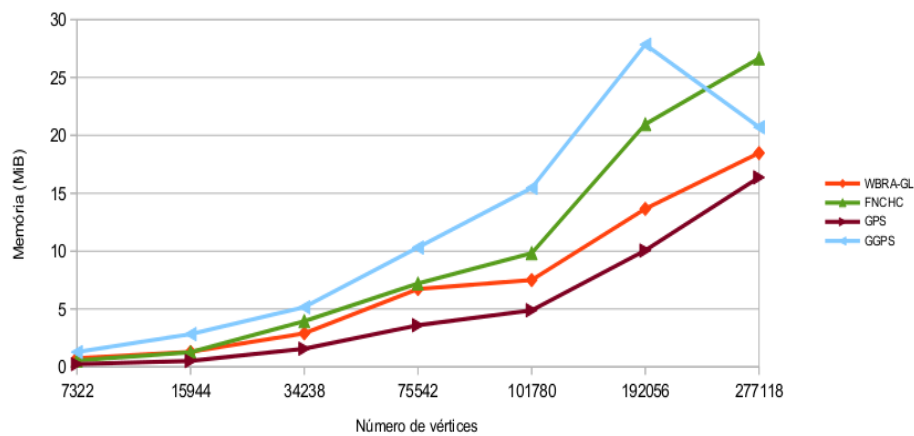


Figura 26 Gráficos das ocupações de memória, em *mebibytes*, das heurísticas WBRA-GL, FNCHC, GPS e GGPS nas instâncias oriundas da discretização da equação de Laplace numeradas de forma aleatória.

Particularmente, a heurística GGPS consumiu mais memória do que as outras heurísticas, possivelmente, por causa do número de estruturas de nível enraizadas construídas. O número de estruturas de nível enraizadas geradas por essa heurística depende do número de vértices pseudo-periféricos escolhidos na etapa inicial dessa heurística. Isso pode variar de instância para instância. Então, provavelmente, muitos vértices pseudo-periféricos foram escolhidos nas instâncias dos testes mostrados na Tabela 13.

5.4.1.4 Comparação dos resultados dos resolutores MGCPJ e MGC-IC0

Apresenta-se, na Figura 27 o gráfico dos custos computacionais em segundos, do MGCPJ e do MGC-IC0. Pode-se considerar que a heurística CMr-GL obteve as melhores reduções do custo computacional do MGC-IC0. Em relação às simulações com o MGCPJ, não se obtiveram reduções no custo computacional do MGCPJ com as heurísticas para redução de banda. Com isso, os resultados mostrados na Figura 27 são referentes aos resolutores MGCPJ sem as reduções de largura de banda e de *profile* e do MGC-IC0 com reduções de largura e banda e de *profile* pela heurística CMr-GL.

Pode-se observar na Figura 27, que o custo computacional do MGC-IC0 + CMr-GL foi maior que o custo computacional do MGCPJ. Isso ocorreu por causa do alto custo computacional do IC0. A redução no número de iterações do MGC pelo IC0 não compensou o custo computacional do IC0.

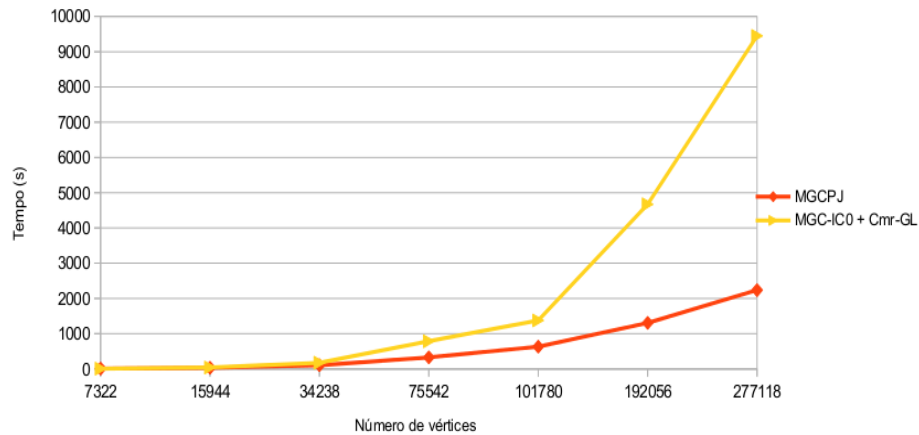


Figura 27 Gráfico dos custos computacionais, em cada instância, dos resolvers MGCPJ e MGC-IC0. Utilizaram-se os resultados da melhor combinação para cada resolutor, ou seja, utilizou-se os resultados do MGCPJ sem redução de largura de banda e utilizou-se os resultados do MGC-IC0 com redução de largura de banda pela heurística CMr-GL.

5.4.2 Instâncias numeradas pela curva de sierpiński modificada

Nos testes apresentados nesta subseção, utilizou-se SELs numerados pela curva de Sierpiński modificada (OLIVEIRA; KISCHINHEVSKY, 2008), com tamanhos de 16.922 a 1.015.004 incógnitas, provenientes da discretização da equação de Laplace por volumes finitos. Executou-se cada heurística e o resolutor três vezes em cada instância.

Na subseção 5.4.2.1, são mostrados os resultados das simulações do MGCPJ com a redução de largura de banda. Os resultados com o MGC-IC0, com a redução de largura de banda, são apresentados na subseção 5.4.2.2. As ocupações de memória das heurísticas são detalhadas na subseção 5.4.2.3. Comparações dos resultados dos resolvers MGCPJ e MGC-IC0 são realizadas na subseção 5.4.2.4.

5.4.2.1 Simulações com o método dos gradientes conjugados preconditionado pelo método Jacobi

Nesta subseção, são apresentados os resultados da resolução de SELs oriundos da discretização da equação de Laplace obtidos pela curva de Sierpiński modificada, pelo MGCPJ com as reduções de largura de banda e de *profile*. Executou-se cada heurística e o resolutor MGCPJ três vezes. Os resultados médios dessas execuções são apresentados na Tabela 14.

Em relação à heurística VNS-*band*, como pode ser observado na Tabela 14, em cada instância, foram utilizados diferentes tempos de execução. Porém, em todas as oito instâncias, os melhores resultados da heurística VNS-*Band*, em relação à redução do custo computacional do MGCPJ, foram obtidos com um segundo. Ainda, como descrito na seção 3.4, na página 74, o executável da heurística VNS-*band* aceita matrizes com dimensão de até 500.000. Com isso, não foi possível executar a heurística VNS-*band* nas últimas instâncias, com 750.446 e 1.015.004 vértices.

Nos resultados apresentados na Tabela 14, a heurística que obteve, na maioria das instâncias, a melhor redução de largura de banda foi a heurística FNCHC. Em relação ao *profile*, a heurística que mais vezes obteve o menor *profile* foi a heurística BL. Porém, a redução do custo computacional do MGCPJ não é proporcional às reduções de largura de banda e de *profile*. Ainda, de forma análoga aos resultados apresentados anteriormente, a heurística mais rápida foi a heurística BeL-GL.

Em relação à ocupação de memória, as heurísticas CMr-GL, BeL-GL, hGPHH e BL não apresentaram resultados significativos. Observa-se, ainda, que as heurísticas com maiores ocupações de memória são as heurísticas VNS-*band* e CSS-*band*.

Tabela 14 Resultados referentes às execuções do MGCPJ e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas pela curva de Sierpiński modificada.

n	Heurística	β	Heurística		MGCPJ			σ	C_v	Redução	
			<i>profile</i>	t(s)	m.(MiB)	iter.	t(s)				
16922	Sem renum.	16921	1710910	-	-	767	51,20	0,02	0,04%	-	
	CMr-GL	115	1252527	0,012	0	767	52,56	0,73	1,40%	-2,68%	
	BeL-GL	117	1253216	0,005	0	767	52,79	0,40	0,76%	-5,08%	
	hGPHH	119	1321688	0,016	0	767	51,08	0,07	0,14%	0,20%	
	WBRA-GL	178	1327087	0,181	1,55	767	53,03	0,07	0,14%	-3,93%	
	VNS-Band	(1s)	4756	2393029	1,287	144,70	767	51,57	0,19	0,37%	-3,24%
		(2s)	4756	2393029	2,298	144,70	767	55,31	0,02	0,04%	-4,74%
	FNCHC		114	1372628	8,212	2,02	767	50,95	0,04	0,09%	-15,55%
	GPS		115	1252527	3,248	0,46	767	52,70	0,24	0,47%	-9,28%
	GGPS		115	1321081	2,311	4,12	767	52,92	0,13	0,25%	-7,87%
BL		224	1235707	0,698	0	767	53,73	0,30	0,56%	-6,31%	
CSS-Band		16746	85797563	13,445	140,91	767	55,39	1,06	1,91%	-34,43%	
39716	Sem renum.	39715	6309342	-	-	1144	187,88	0,04	0,02%	-	
	CMr-GL	195	4376986	0,05	0	1144	210,13	0,52	0,25%	-11,87%	
	BeL-GL	198	4380812	0,02	0	1144	211,13	0,20	0,10%	-12,39%	
	hGPHH	192	4770829	0,05	0	1144	209,49	0,31	0,15%	-11,53%	
	WBRA-GL	280	4764403	0,57	3,35	1385	259,05	0,36	0,14%	-38,18%	
	VNS-Band	(1s)	5863	9979067	2,32	327,22	1144	211,90	1,11	0,52%	-14,02%
		(3s)	5863	9979067	4,33	327,22	1144	214,33	0,52	0,24%	-16,38%
	FNCHC		189	5021600	22,89	3,35	1144	211,14	1,00	0,47%	-24,56%
	GPS		180	4464634	9,05	0,77	1144	213,40	0,65	0,31%	-18,40%
	GGPS		194	4391324	19,90	3,08	1144	213,65	4,54	2,13%	-24,31%
BL		335	4156848	6,63	0	1144	221,48	0,15	0,07%	-21,41%	
CSS-Band		39346	480512986	341,561	537,93	1144	257,20	1,69	0,66%	-281,69%	
68414	Sem renum.	68413	14882117	-	-	1514	430,09	0,03	0,01%	-	
	CMr-GL	238	9598308	0,08	0	1514	480,92	1,09	0,23%	-11,84%	
	BeL-GL	240	9605665	0,03	0	1514	485,29	1,02	0,21%	-12,84%	
	hGPHH	236	10705920	0,08	0	1514	480,96	0,39	0,08%	-11,85%	
	WBRA-GL	316	10702498	1,22	5,15	1514	473,28	1,20	0,25%	-10,33%	
	VNS-Band	(1s)	516	17030717	4,59	557,40	1514	483,22	0,93	0,19%	-13,42%
		(5s)	516	17030717	8,61	557,40	1514	506,31	0,18	0,03%	-19,72%
	FNCHC		233	11325816	42,02	5,12	1514	484,70	1,66	0,34%	-22,47%
	GPS		225	9751463	28,90	1,45	1514	495,22	0,29	0,06%	-21,86%
	GGPS		233	9781546	68,94	6,18	1514	507,94	0,07	0,01%	-34,13%
BL		440	8910920	30,69	0	1514	515,09	0,43	0,08%	-26,90%	
CSS-Band		67862	1432183654	1962,01	2042,96	1514	590,52	3,97	0,67%	-493,49%	
105764	Sem renum.	105763	29560801	-	-	1846	815,79	0,04	0,04%	-	
	CMr-GL	311	18180951	0,13	0	1846	899,11	2,09	0,23%	-10,23%	
	BeL-GL	307	18183664	0,05	0	1846	904,16	2,29	0,25%	-10,84%	
	hGPHH	309	20753083	0,14	0	1846	900,93	1,75	0,20%	-10,45%	
	WBRA-GL	523	20706588	2,33	8,25	2210	1071,93	0,31	0,03%	-31,68%	
	VNS-Band	(1s)	2809	33762228	9,38	857,20	1846	901,06	0,36	0,04%	-11,60%
		(10s)	2809	33762228	18,26	857,20	1846	930,94	0,15	0,02%	-16,35%
	FNCHC		289	21067109	69,00	9,36	1846	905,39	2,95	0,33%	-19,44%
	GPS		299	18336159	60,95	2,06	1846	927,15	0,80	0,09%	-21,12%
	GGPS		306	18163269	136,02	7,47	1846	898,52	0,29	0,03%	-26,81%
BL		483	16959146	112,25	0	1846	968,00	0,40	0,04%	-32,42%	
CSS-Band		105406	3418070351	305,06	494,47	1846	1114,79	3,78	0,34%	-74,05%	

Como nos resultados apresentados na subseção 5.4.1.1, não se obteve redução no custo computacional do MGCPJ com as reduções de largura de banda e de *profile*. Verifica-se, na última coluna da Tabela 14, que as renumerações geradas por todas as heurísticas, com exceção da heurística CMr-GL na instância com 16.922 vértices, aumentaram o custo computacional do MGCPJ em todas as instâncias.

Tabela 14 Conclusão.

n	Heurística	Heurística			MGCPJ			σ	C_v	Redução	
		β	<i>profile</i>	t(s)	m.(MiB)	iter.	t(s)				
237086	Sem renum.	237085	115804392	-	-	2611	2629,25	8,69	0,33%	-	
	CMr-GL	391	56621430	0,29	0	2612	2865,81	7,98	0,28%	-9,08%	
	BeL-GL	392	56725752	0,12	0	2611	2869,30	1,04	0,04%	-9,13%	
	hGPHH	393	64411087	0,30	0	2612	2852,31	3,88	0,14%	-8,50%	
	WBRA-GL	671	64251019	6,87	15,47	2612	2815,44	0,52	0,02%	-7,34%	
	VNS- <i>Band</i>	(1s)	1783	92303199	40,84	1911,67	2612	2858,49	5,97	0,21%	-10,27%
		(41s)	1783	92303199	81,09	1911,67	2612	3054,09	1,74	0,06%	-19,24%
	FNCHC	388	64592246	176,95	20,38	2612	2858,73	1,31	0,05%	-15,46%	
	GPS	392	56476790	392,34	12,85	2612	2965,49	5,43	0,18%	-27,71%	
	GGPS	389	57030510	820,90	16,20	2612	2869,19	11,37	0,40%	-40,35%	
	BL	718	52953332	698,29	0	2612	3100,47	0,90	0,03%	-44,48%	
CSS- <i>Band</i>	236418	2682971255	1657,82	1107,41	2612	3636,71	25,22	0,69%	-101,37%		
467504	Sem renum.	467503	382386929	-	-	3446	6972,41	2,65	0,04%	-	
	CMr-GL	448	130166482	0,57	0	3446	7433,93	3,43	0,05%	-6,63%	
	BeL-GL	445	130547416	0,24	0	3446	7444,73	4,57	0,06%	-6,68%	
	hGPHH	445	149299971	0,60	0	3446	7422,67	6,17	0,08%	-6,46%	
	WBRA-GL	721	148990820	16,10	25,97	3446	7402,24	3,32	0,04%	-6,40%	
	VNS- <i>Band</i>	(1s)	5001	227019725	154,17	4903,50	3446	7409,42	2,46	0,03%	-8,48%
		(155s)	5001	227019725	307,20	4903,50	3446	7695,53	11,25	0,15%	-14,78%
	FNCHC	449	156037685	371,87	39,59	3446	7416,14	6,92	0,09%	-11,70%	
	GPS	455	129684974	1593,12	26,78	3446	7617,91	2,46	0,03%	-32,11%	
	GGPS	438	132748941	3228,37	29,45	3446	8033,11	7,73	0,10%	-61,51%	
	BL	862	119436630	895,21	0	3446	8070,87	3,74	0,04%	-28,59%	
CSS- <i>Band</i>	466181	2526782462	9693,08	3644,05	3446	9489,89	6,99	0,07%	-175,13%		
750446	Sem renum.	750445	911516500	-	-	4246	13659,77	5,64	0,07%	-	
	CMr-GL	461	224589050	0,93	0	4245	14562,95	4,39	0,03%	-6,61%	
	BeL-GL	462	225259320	0,38	0	4245	14673,63	2,30	0,02%	-7,42%	
	hGPHH	471	257304543	0,96	0	4246	14629,19	7,03	0,05%	-7,18%	
	WBRA-GL	727	257141087	27,66	41,23	4246	15194,03	0,54	0,01%	-11,43%	
	VNS- <i>Band</i>	-	-	-	-	-	-	-	-	-	
	FNCHC	715	304070803	615,49	60,35	4245	14675,77	0,23	0,01%	-11,94%	
	GPS	-	-	-	-	-	-	-	-	-	
	GGPS	-	-	-	-	-	-	-	-	-	
	BL	-	-	-	-	-	-	-	-	-	
	CSS- <i>Band</i>	-	-	-	-	-	-	-	-	-	
1015004	Sem renum.	1015003	1580908606	-	-	4569	20178,33	2,41	0,01%	-	
	CMr-GL	461	316976358	1,35	0	4569	21208,57	8,08	0,04%	-5,11%	
	BeL-GL	463	317942317	0,65	0	4569	21268,53	12,01	0,05%	-5,41%	
	hGPHH	465	363241451	1,43	0	4569	21117,00	20,10	0,09%	-4,66%	
	WBRA-GL	726	362058495	38,81	54,83	4569	22303,33	11,73	0,05%	-10,72%	
	VNS- <i>Band</i>	-	-	-	-	-	-	-	-	-	
	FNCHC	455	365943729	819,904	87,53	4569	21173,57	10,56	0,05%	-9,01%	
	GPS	-	-	-	-	-	-	-	-	-	
	GGPS	-	-	-	-	-	-	-	-	-	
	BL	-	-	-	-	-	-	-	-	-	
	CSS- <i>Band</i>	-	-	-	-	-	-	-	-	-	

Era esperado que, com as reduções de largura de banda e de *profile*, fossem obtidas reduções do custo computacional do MGCPJ. Especificamente, os SELs apresentados nesta subseção foram montados por uma curva de preenchimento de espaço. Aparentemente, quando um SEL é montado por uma curva de preenchimento de espaço, há mais coerência de *cache*. Esses resultados também foram observados por Abreu (2014).

5.4.2.2 Simulações Com o método dos gradientes conjugados precondicionado pelo método incompleto de Cholesky com zero *fill-in*

Nesta subseção, são apresentados os resultados da resolução de SELs oriundos da discretização da equação de Laplace obtidos pela curva de Sierpiński modificada, pelo MGC-IC0 com as reduções de largura de banda e de *profile*. Executou-se cada heurística e o MGC-IC0 três vezes. Os resultados médios dessas execuções são apresentados na Tabela 15.

Devido ao grande custo computacional envolvido nas simulações realizadas neste trabalho, os testes nas instâncias com 750.446 e com 1.015.004 vértices não haviam terminado até a data da entrega desta dissertação. Com isso, mostram-se, na Tabela 15, os resultados do MGC-IC0 com todas as instâncias excetuando-se as instâncias com 750.446 e com 1.015.004 vértices. Também, só foi possível executar a heurística VNS-*band* com 1s de tempo máximo.

Como pode ser verificado na Tabela 15, as heurísticas GPS, GGPS, BL e CSS-*band* não obtiveram resultados bons na redução do custo computacional do MGC-IC0 nas instâncias com até 237.086 vértices. Com isso, para se economizar tempo de processamento, decidiu-se não utilizar essas heurísticas nas instâncias com mais de 237.086 vértices.

Como pode ser verificado na Tabela 15, as heurísticas que obtiveram as melhores reduções de largura de banda foram as heurísticas GPS, GGPS e FNCHC. Em relação ao *profile*, a heurística que obteve os melhores resultados foi a heurísticas BL. A redução do custo computacional do MGC-IC0 não é proporcional às reduções de largura de banda e de *profile*. Com isso, como o custo computacional dessas heurísticas é,

Tabela 15 Resultados referentes às execuções do MGC-IC0 e aos SEL com renumeração dos vértices nas instâncias oriundas da discretização da equação de Laplace numeradas pela curva de Sierpiński modificada.

n	Heurística	β	<i>profile</i>	Heurística		IC0	MGC	MGC-IC0		σ	C_v	Red.
				t(s)	m.(MiB)			t(s)	iter.			
16922	Sem renum.	16921	1710910	-	-	33,24	25,79	295	59,06	0,14	0,25%	-
	CMr-GL	115	1253675	0,018	0	32,69	26,09	299	58,78	0,22	0,38%	0,44%
	BeL-GL	115	1255425	0,007	0	32,70	26,28	300	58,98	0,04	0,07%	0,12%
	hGPHH	118	1322460	0,019	0	32,72	27,14	311	59,86	0,08	0,13%	-1,39%
	WBRA-GL	176	1326189	0,176	1,55	32,88	27,25	311	60,13	0,49	0,81%	-2,11%
	VNS- <i>Band</i> (1s)	4756	2393029	1,296	144,70	32,92	28,06	319	60,98	0,16	0,28%	-5,45%
	FNCHC	114	1364365	8,278	2,02	32,93	27,35	312	60,28	0,30	0,49%	-16,08%
	GPS	115	1252527	3,276	0,46	32,71	26,19	299	58,90	0,15	0,26%	-5,28%
	GGPS	115	1321081	4,121	4,12	32,88	27,02	308	59,90	0,26	0,44%	-8,40%
	BL	224	1235707	0,700	0	32,74	26,85	303	59,59	0,11	0,19%	-2,08%
	CSS- <i>Band</i>	16723	86692655	11,73	140,91	26,73	37,82	411	64,55	1,10	1,70%	-29,16%
	39716	Sem renum.	38715	6309342	-	-	191,46	92,62	437	283,72	0,11	0,04%
CMr-GL		195	4377424	0,04	0	195,05	92,86	446	287,91	0,13	0,05%	-1,49%
BeL-GL		198	4380812	0,02	0	195,34	93,58	448	288,92	0,48	0,16%	-1,84%
hGPHH		197	4773104	0,04	0	194,25	97,22	467	291,46	0,34	0,12%	-2,74%
WBRA-GL		280	4762429	0,56	3,35	195,02	95,60	458	290,63	0,27	0,09%	-2,63%
VNS- <i>Band</i> (1s)		5963	9979067	2,31	327,22	194,72	96,58	460	291,30	0,49	0,17%	-3,49%
FNCHC		189	4885951	22,66	3,35	195,15	96,38	462	291,54	0,17	0,06%	-10,74%
GPS		180	4464634	8,78	0,77	194,75	95,50	457	290,26	0,19	0,07%	-5,40%
GGPS		192	4453408	19,21	3,08	195,67	91,15	435	286,82	0,40	0,14%	-7,86%
BL		335	4156848	5,60	0	195,47	95,56	452	291,03	0,81	0,28%	-4,55%
CSS- <i>Band</i>		39424	473525061	22,47	537,93	153,84	137,88	608	291,71	0,86	0,30%	-10,74%
68414		Sem renum.	68413	14882117	-	-	621,85	205,26	567	827,12	0,52	0,06%
	CMr-GL	238	9598308	0,08	0	624,12	218,25	589	842,37	0,33	0,04%	-1,85%
	BeL-GL	241	9608251	0,04	0	625,43	220,47	593	845,90	1,58	0,19%	-2,28%
	hGPHH	238	10712844	0,09	0	622,24	232,47	628	854,70	2,07	0,24%	-3,35%
	WBRA-GL	315	10703573	1,22	5,15	625,37	230,36	620	855,73	1,08	0,13%	-3,61%
	VNS- <i>Band</i> (1s)	516	17030717	4,58	557,40	624,00	232,83	626	856,84	0,65	0,07%	-4,15%
	FNCHC	229	11045900	42,28	5,12	626,18	233,80	629	859,99	1,41	0,16%	-9,09%
	GPS	225	9751463	28,94	1,45	624,54	220,68	593	845,22	0,85	0,10%	-5,69%
	GGPS	235	9719428	67,55	6,18	625,21	218,51	587	843,72	1,92	0,23%	-10,17%
	BL	440	8910920	30,86	0	632,88	225,83	600	858,71	0,82	0,10%	-7,55%
	CSS- <i>Band</i>	68078	1423971272	99,24	2042,96	481,60	328,12	797	809,73	3,02	0,37%	-9,90%
	105764	Sem renum.	105763	29560801	-	-	1708,76	396,11	691	2104,87	1,82	0,09%
CMr-GL		306	18161623	0,13	0	1612,34	406,62	705	2018,96	2,23	0,11%	4,08%
BeL-GL		306	18183385	0,07	0	1609,20	417,28	721	2026,48	1,23	0,06%	3,72%
hGPHH		308	20752577	0,15	0	1604,55	456,04	790	2060,59	3,18	0,15%	2,10%
WBRA-GL		522	20711096	2,28	8,25	1608,44	442,96	766	2051,40	1,31	0,06%	2,43%
VNS- <i>Band</i> (1s)		2809	33762228	9,20	857,20	1604,61	445,50	769	2050,11	1,67	0,08%	2,16%
FNCHC		291	20811211	68,14	9,36	1607,47	445,04	770	2052,51	3,57	0,17%	-0,75%
GPS		299	18336159	61,50	2,06	1610,69	414,16	713	2024,85	0,86	0,04%	0,88%
GGPS		310	18521099	137,95	7,47	1612,18	407,44	702	2019,62	4,70	0,23%	-2,50%
BL		483	16959146	112,89	0	1640,94	430,68	733	2071,62	1,34	0,06%	-3,78%
CSS- <i>Band</i>		105266	3356556041	349,91	494,47	1310,76	626,19	964	1936,95	8,11	0,42%	-8,64%

relativamente, grande, essas heurísticas não obtiveram as melhores reduções do custo computacional do MGC-IC0. Isso porque os tempos gastos por essas heurísticas, nas reduções de largura de banda e *profile*, não foram compensados pelas reduções dos tempos de execuções do MGC-IC0.

Em relação à ocupação de memória, as heurísticas CMr-GL, BeL-GL, hGPHH e BL não apresentaram resultados significativos. Observa-se, ainda, que as heurísticas com maiores ocupações de memória são as

Tabela 15 Conclusão.

n	Heurística	β	profile	Heurística		IC0	MGC	MGC-IC0		σ	C _v	Red.	
				t(s)	m.(MiB)			t(s)	iter.				t(s)
237086	Sem renum.	237085	115804392	-	-	8532,65	1228,89	956	9761,54	4,66	0,05%	-	
	CMr-GL	391	56621430	0,29	0	8223,28	1288,89	991	9512,17	2,56	0,03%	2,55%	
	BeL-GL	392	56725752	0,15	0	8217,34	1319,85	1014	9537,19	5,65	0,06%	2,30%	
	hGPHH	398	64285994	0,34	0	8223,60	1429,04	1102	9652,64	5,12	0,05%	1,11%	
	WBRA-GL	672	64236202	6,87	15,47	8206,33	1402,49	1078	9608,82	7,11	0,07%	1,49%	
	VNS-Band (1s)	1783	92303199	40,81	1911,67	8192,05	1409,12	1085	9601,18	3,82	0,04%	1,22%	
	FNCHC	411	66836889	117,14	20,38	8192,86	1405,03	1081	9597,89	14,00	0,15%	0,48%	
	GPS	392	56476790	329,41	12,85	8211,96	1306,15	1000	9518,11	4,81	0,05%	-0,88%	
	GGPS	388	57902889	821,65	16,20	8233,53	1272,04	971	9505,57	10,43	0,11%	-5,79%	
	BL	718	52953332	705,68	0	8380,82	1360,20	1027	9741,02	12,29	0,13%	-7,02%	
CSS-Band	236309	2846248883	3457,00	1107,41	7244,02	2005,18	1335	9249,20	16,69	0,18%	-30,17%		
467504	Sem renum.	467503	382386929	-	-	32586,49	2948,99	1180	35535,49	8,21	0,02%	-	
	CMr-GL	448	130166482	0,58	0	32069,17	3305,69	1321	35374,85	29,53	0,08%	0,45%	
	BeL-GL	450	130399787	0,31	0	32114,15	3265,03	1304	35379,18	11,09	0,03%	0,44%	
	hGPHH	452	149122278	0,67	0	31984,63	3762,82	1473	35747,46	12,43	0,03%	-0,60%	
	WBRA-GL	722	148936289	16,04	25,97	32106,75	3595,98	1438	35702,73	3,07	0,01%	-0,52%	
	VNS-Band (1s)	5001	227019725	153,50	4903,50	31988,08	3554,27	1412	35542,35	34,78	0,09%	-0,45%	
	FNCHC	439	149803349	372,82	39,59	32101,20	3580,63	1432	35681,83	39,15	0,12%	-1,46%	
	GPS	-	-	-	-	-	-	-	-	-	-	-	-
	GGPS	-	-	-	-	-	-	-	-	-	-	-	-
	BL	-	-	-	-	-	-	-	-	-	-	-	-
CSS-Band	-	-	-	-	-	-	-	-	-	-	-	-	

heurísticas VNS-band e CSS-band.

Observa-se que, nas instâncias com 39.716 e 68.414 vértices, não foram obtidas reduções nos custos computacionais do MGC-IC0 com as reduções de largura de banda e de *profile*. Ainda, nas outras instâncias nota-se que os percentuais das reduções dos custos computacionais do MGC-IC0 não foram tão grandes como, por exemplo, os resultados mostrados na subseção 5.3. Isso ocorreu, possivelmente, pois os SELs foram montados por uma curva de preenchimento de espaço e, com isso, há mais coerência de *cache*.

Um ponto importante a ser observado é que as heurísticas, na maioria das instâncias, não reduziram o custo do MGC condicionado, mas sim do IC0. Além disso, aparentemente, o IC0 gera uma matriz condicionadora melhor quando o SEL é montado por uma curva de preenchimento de espaço, como a curva curva de Sierpiński modificada. Isso porque, com as renumerações realizadas pelas heurísticas, o número de iterações do MGC-IC0 aumentou em quase todas as instâncias.

Mostram-se, nas Figuras 28, 29 e 30, os gráficos dos percentuais das reduções dos custos computacionais do MGC-IC0 pelas heurísticas. Nota-se que as heurísticas BL, CSS-*band* e GGPS foram as heurísticas que apresentaram os piores percentuais da redução do custo computacional do MGC-IC0. Com isso, apresenta-se, na Figura 29, o mesmo gráfico da Figura 28, mas sem os resultados das heurísticas BL, CSS-*band* e GGPS.

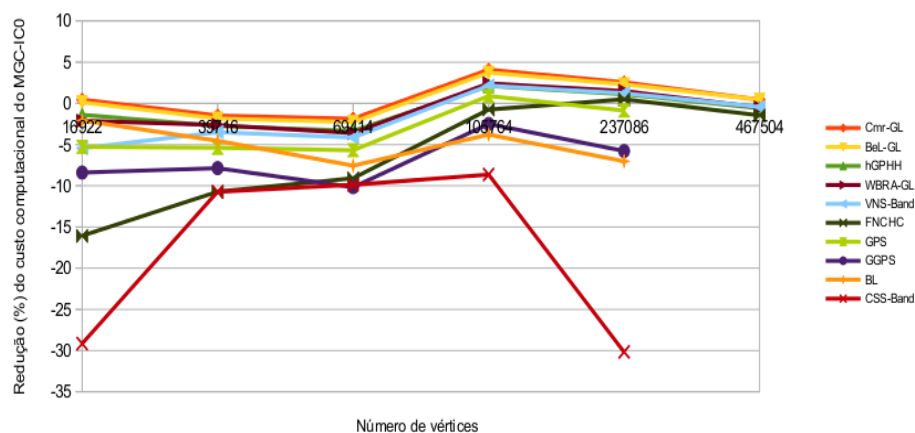


Figura 28 Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 de todas as 10 heurísticas testadas.

Mostra-se, na Figura 29, os resultados das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL em relação aos percentuais das reduções dos custos computacionais do MGC-IC0. Observa-se, que a heurística CMr-GL obteve as melhores reduções. Ainda, pode-se observar que a heurística BeL-GL foi a segunda com as melhores reduções dos custos computacionais do MGC-IC0. Pode-se considerar, também, que a heurística WBRA-GL obteve resultados ligeiramente superiores que a heurística hGPHH na redução do custo computacional do MGC-IC0.

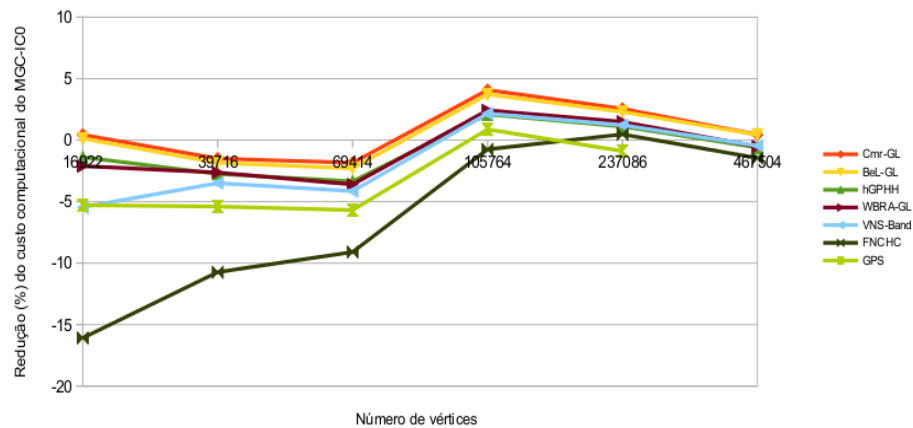


Figura 29 Gráfico dos percentuais das reduções do custo computacional do MGC-IC0 sem os resultados das heurísticas BL, CSS-Band e GGPS.

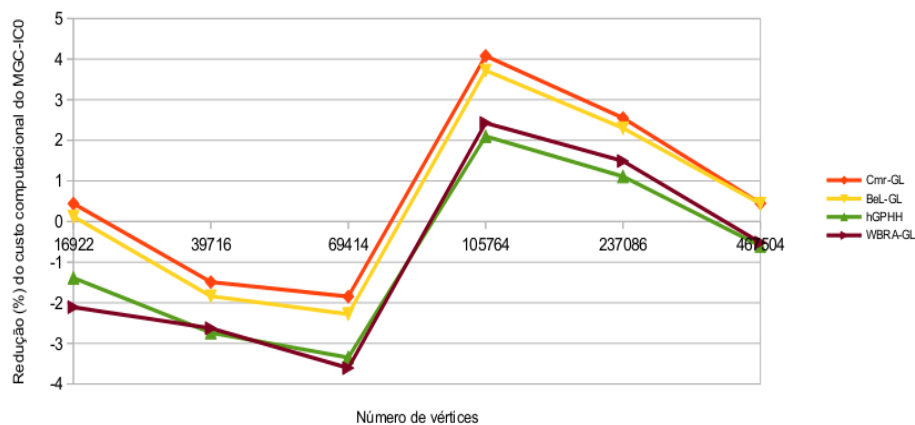


Figura 30 Gráfico das percentuais das reduções do custo computacional do MGC-IC0 das heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL.

5.4.2.3 Ocupações de memória

Como descrito, as heurísticas CMr-GL, BeL-GL, hGPHH e BL não apresentaram ocupações de memória significativas. Então, mostram-se, nesta subseção, os gráficos dos consumos das heurísticas WBRA-GL, VNS-band, FNCHC, GPS, GGPS e CSS-band. A ocupação de memória dessas

heurísticas é a mesma nos testes com o MGCPJ e com o MGC-IC0. Embora nos testes com o MGCPJ seja utilizada uma estrutura de dados diferente da estrutura de dados utilizada nos testes com o MGC-IC0, as heurísticas utilizam, para ambos os testes, a mesma estrutura de dados (para detalhes, veja a subseção 3.4.1). Com isso, o mesmo consumo de memória é observado nos dois testes.

O gráfico das ocupações de memória das heurísticas WBRA-GL, VNS-band, FNCHC, GPS, GGPS e CSS-band é mostrado na Figura 31. Pode-se observar que, a heurística VNS-band, é a heurística com o maior consumo de memória. Na heurística CSS-band, utilizou-se diferentes números de PCs. Isso justifica o comportamento anormal de ocupação de memória dessa heurística.

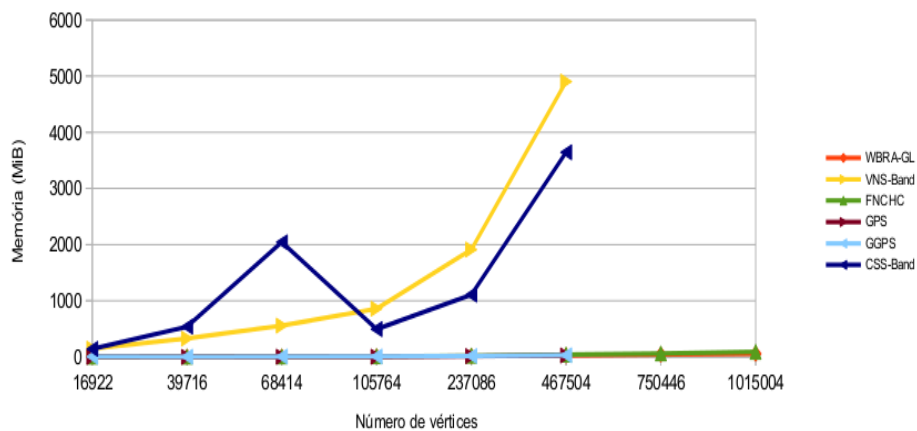


Figura 31 Gráficos das ocupações de memória, em *mebibytes*, das heurísticas WBRA-GL, VNS-Band, FNCHC, GPS, GGPS e CSS-Band nas instâncias oriundas da discretização da equação de Laplace numeradas pela Curva de Sierpiński modificada.

Na Figura 32, tem-se o mesmo gráfico de ocupações de memória mostrado na Figura 31. Porém, na Figura 31, não são apresentados as ocupações de memória das heurísticas VNS-band e CSS-band. Com isso, é

possível observar, com mais detalhes, os comportamentos das ocupações de memória das heurísticas WBRA-GL, FNCHC, GPS e GGPS. Entre essas heurísticas, a FNCHC é a heurística que apresentou a maior ocupação de memória.

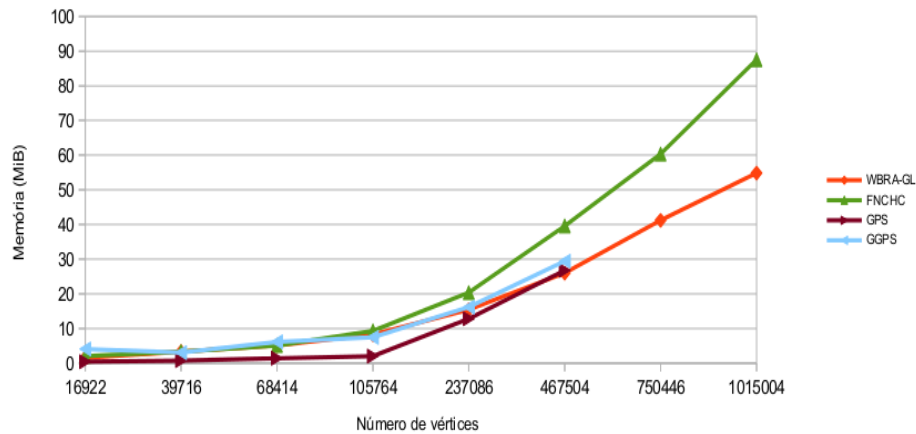


Figura 32 Gráficos das ocupações de memória, em *mebibytes*, das heurísticas WBRA-GL, FNCHC, GPS e GGPS nas instâncias oriundas da discretização da equação de Laplace numeradas pela Curva de Sierpiński modificada.

5.4.2.4 Comparação dos resultados dos resolutores MGCPJ e MGC-IC0

Apresenta-se, na Figura 33, o gráfico dos custos computacionais em segundos do MGCPJ e do MGC-IC0. Pode-se considerar que a heurística CMr-GL obteve as melhores reduções do custo computacional do MGC-IC0. Em relação às simulações com o MGCPJ, não se obtiveram reduções no custo computacional do MGCPJ com as heurísticas para redução de banda. Com isso, os resultados mostrados na Figura 33 são referentes aos resolutores MGCPJ sem as reduções de largura de banda e de *profile* e

do MGC-IC0 com reduções de largura e banda e de *profile* pela heurística CMr-GL.

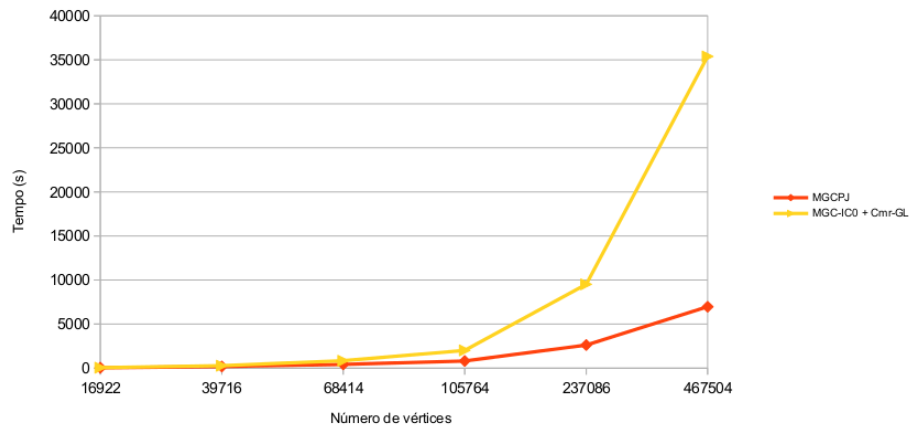


Figura 33 Gráfico dos custos computacionais, em cada instância, dos resolvers MGCPJ e MGC-IC0. Utilizaram-se os resultados da melhor combinação para cada resolutor, ou seja, utilizou-se os resultados do MGCPJ sem redução de largura de banda e utilizou-se os resultados do MGC-IC0 com redução de largura de banda pela heurística CMr-GL.

Pode-se observar na Figura 33, que o custo computacional do MGC-IC0 + CMr-GL foi maior que o custo computacional do MGCPJ. Isso ocorreu por causa do alto custo computacional do IC0. A redução no número de iterações do MGC pelo IC0 não compensou o custo computacional do IC0.

6 CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho, heurísticas para a redução de largura de banda, selecionadas por meio de uma revisão sistemática, foram implementadas e avaliadas em relação às reduções de largura de banda e de *profile*. Também, avaliou-se os efeitos dessas reordenações na redução do custo computacional do MGC preconditionado. Especificamente, utilizou-se o MGCPJ e o MGC-IC0. Além disso, foram propostas duas novas heurísticas, com baixo custo computacional, para as reduções de largura de banda e de *profile*.

Ao todo, foram testadas 11 heurísticas para a redução de largura de banda. Dessas 11 heurísticas, oito foram selecionadas por meio das revisões sistemáticas realizadas por Chagas e Oliveira (2015) e Oliveira e Chagas (2015a). Essas oito heurísticas foram consideradas as melhores heurísticas com boa redução de largura de banda e custo computacional razoável. Entre essas oito heurísticas, há a heurística *VNS-band* que, considerando-se somente a redução de largura de banda, é a heurística no estado da arte no problema. Ainda, a heurística GPS, que é uma das heurísticas mais difundidas na literatura, também foi implementada. As outras duas heurísticas implementadas, as heurísticas WBRA-GL e BeL-GL, foram propostas neste trabalho.

Nas seções seguintes, são realizadas algumas considerações finais. Na seção 6.1, são apresentadas as considerações sobre as simulações com as instâncias da base Harwell-Boeing. Nas seções 6.2 e 6.3, apresentam-se as considerações dos testes realizados com as reduções de largura de banda e de *profile* com os resolutores MGCPJ e MGC-IC0 respectivamente. Apresentam-se, na seção 6.4, as considerações finais sobre as heurísticas propostas neste trabalho. Finalmente, na seção 6.4, propostas de trabalhos

futuros são descritas.

6.1 Considerações sobre as simulações na base Harwell-Boeing

Nas simulações realizadas com as matrizes da base Harwell-Boeing, a heurística que obteve os melhores resultados, em relação às reduções de largura de banda e de *profile*, foi a heurística VNS-*band*. Como pode ser verificado na seção 4.5, na página 96, a heurística VNS-*band* obteve mais vezes as melhores reduções de largura de banda e de *profile*. Além disso, a heurística VNS-*band* obteve os menores somatórios do escore padronizado nas 35 instâncias pequenas simétricas e nas 45 instâncias pequenas assimétricas. Esses dois conjuntos contêm as maiores matrizes da base Harwell-Boeing e, também, possuem 80 das 113 matrizes dessa base. Com isso, esses resultados têm mais relevância que os resultados obtidos nos conjuntos de 15 matrizes simétricas e 18 matrizes assimétricas.

Os resultados obtidos pela heurística VNS-*band*, neste trabalho, corroboram com os resultados obtidos por Mladenovic et al. (2010) e reforçam a condição dessa heurística como estado da arte na redução de largura de banda. Além disso, essa heurística pode ser considerada a melhor heurística para a redução de largura de banda com baixo custo computacional. Isso porque, diferentemente dos testes realizados por Mladenovic et al. (2010), os resultados foram obtidos, pela heurística VNS-*band*, com um limite relativamente baixo para o tempo de execução.

A heurística VNS-*band* pode ser utilizada em sistemas em que as matrizes esparsas são armazenadas por *skyline*. No armazenamento por *skyline*, quanto menores são a largura de banda e o *profile*, menor será a ocupação de memória.

Vale ressaltar, também, que a heurística FNCHC obteve boas reduções de largura de banda e *profile* com baixo custo computacional. Ainda, observa-se que a heurística mais rápida, em todos os quatros conjuntos de matrizes, foi a heurística BeL-GL, proposta neste trabalho. Essa heurística apresentou reduções de largura de banda e *profile* razoáveis.

6.2 Considerações sobre os testes com resolução de SELs pelo MGCPJ

Nas simulações com o MGCPJ, observou-se que não houve variação no número de iterações desse resolutor com as reduções de largura de banda. Isso porque, provavelmente, as reduções de largura de banda e de *profile* não melhora o condicionamento realizado pelo método de Jacobi.

Ainda, nas instâncias oriundas da discretização da equação de Laplace, não foram obtidas reduções nos custos computacionais do MGCPJ com as reduções de largura de banda e de *profile*. Possivelmente, há mais coerência de *cache* nos SELs originais dessas instâncias. Em relação às instâncias obtidas pela curva de preenchimento de espaço, esses resultados são justificáveis. Isso porque há indícios que, se um SEL é montado por uma curva de preenchimento de espaço, então, haverá melhor localidade de memória desses elementos e, conseqüentemente, maior coerência de *cache*. Resultados similares foram obtidos por Abreu (2014).

Em relação aos resultados obtidos nas simulações do MGCPJ com as instâncias oriundas da discretização da condução do calor, houve redução no custo computacional desse resolutor com as reduções de largura de banda e de *profile*. Pode-se considerar que as heurísticas que obtiveram os maiores percentuais de redução do custo computacional do MGCPJ foram

as heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL. Essas heurísticas foram, também, as mais rápidas.

Destacam-se, as heurísticas CMr-GL e BeL-GL que podem ser consideradas as mais adequadas para a redução do custo computacional do MGCPJ em SELs oriundos da discretização, por volumes finitos, da condução do calor. As vantagens dessas heurísticas são mostradas na Tabela 16.

Tabela 16 Vantagens das heurísticas CMr-GL e BeL-GL.

Vantagens	
CMr-GL	BeL-GL
<ul style="list-style-type: none"> • Heurística consolidada presente em bibliotecas e <i>softwares</i>, por exemplo, MATLAB. • Obteve a maior redução do custo computacional do MGCPJ na maior instância, com 965545 vértices. • Apresentou, na maioria das instâncias, melhores reduções de largura de banda e de <i>profile</i> em comparação com os resultados da heurística BeL-GL. 	<ul style="list-style-type: none"> • A busca em largura é amplamente difundida na literatura. • Foi mais rápida em todos os testes realizados. • A implementação é um pouco mais simples que a implementação da heurística CMr-GL. • Obteve melhores resultados na redução do custo computacional do MGCPJ em mais instâncias.

Ainda, verificou-se que a redução do custo computacional do MGCPJ não é proporcional às reduções de largura de banda e de *profile*. Com isso, não compensa reduzir muito a largura de banda se o custo computacional para isso é alto. Mesmo heurísticas com custo computacional razoável, como as heurísticas FNCHC, VNS-*band* e GPS, não obtiveram resultados bons como as heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL, na redução do custo computacional do MGCPJ.

Finalmente, verificou-se que o custo computacional do MGCPJ é menor que o custo computacional do MGC-IC0 na resolução de SELs

oriundos das discretizações, por volumes finitos, da condução do calor e da equação de Laplace. Com isso, o MGCPJ é mais adequado para a resolução desses SELs.

6.3 Considerações sobre os testes com resolução de SELs pelo MGC-IC0

Diferentemente dos resultados com o MGCPJ, obtiveram-se reduções, com as reduções de largura de banda e de *profile*, nos custos computacionais do MGC-IC0 em todas os três conjuntos de SELs. Além disso, houve variação no número de iterações do MGC-IC0 com as reduções de largura de banda e de *profile*. Observou-se, também, que a redução do custo computacional do MGC-IC0 não é proporcional às reduções de largura de banda e de *profile*.

As heurísticas que obtiveram os melhores percentuais de redução do custo computacional do MGC-IC0 foram as heurísticas CMr-GL, BeL-GL, hGPHH e WBRA-GL. Porém, a heurística que obteve os maiores percentuais foi a heurística CMr-GL, nos três conjuntos de SELs. Com isso, a heurística CMr-GL é a mais indicada para se reduzir o custo computacional do MGC-IC0 nos SELs oriundos das discretizações, por volumes finitos, da condução do calor e da equação de Laplace.

As renumerações dos vértices, pelas heurísticas, também influenciam no custo computacional do IC0 e não somente do MGC preconditionado. Como a reordenação dos vértices altera a coerência de *cache*, possivelmente isso influenciou no custo computacional do IC0.

Em relação aos testes nos conjuntos de SELs oriundos da discretização da condução do calor e dos SELs oriundos da discretização

da equação de Laplace obtidos aleatoriamente, com as reduções de largura de banda e de *profile*, o número de iterações do MGC-IC0 reduziu consideravelmente. Possivelmente, as reordenações geradas pelas heurísticas fazem com que a matriz preconditionadora gerada pelo IC0 seja melhor condicionada. Com isso, o MGC-IC0 converge com menos iterações e, conseqüentemente, com menor custo computacional. Com os SELs, oriundos da discretização da equação de Laplace, obtidos pela curva de Sierpiński modificada, o contrário foi observado, ou seja, o número de iterações do MGC-IC0 aumentou com as reduções de largura de banda e de *profile*. Aparentemente, o IC0 gera uma matriz preconditionadora melhor quando o SEL é montado por uma curva de preenchimento de espaço.

6.4 Considerações sobre as heurísticas propostas: BeL-GL e WBRA-GL

As heurísticas WBRA-GL e BeL-GL foram propostas neste trabalho. Essas heurísticas apresentaram reduções de largura de banda e de *profile* razoáveis. Porém, essas heurísticas foram propostas com o intuito serem rápidas, pois o principal motivo de se reduzir a largura de banda de matrizes é se obter redução no custo computacional da resolução de SELs.

Nas simulações em que se obteve redução, com as reduções de largura de banda e de *profile*, no custo computacional do MGCPJ e do MGC-IC0, as heurísticas BeL-GL e WBRA-GL apresentaram bons percentuais de redução dos custos computacionais desses resolutores. Especificamente, esses resultados foram observados nas simulações com o MGCPJ e o MGC-IC0 nas instâncias oriundas da discretização da condução do calor e nas simulações com o MGC-IC0 nas instâncias oriundas das discretizações

da equação de Laplace obtidas de forma aleatória e pela curva de Sierpiński modificada. Nessas casos, os percentuais de redução dos custos computacionais do MGCPJ e do MGC-IC0, pelas heurísticas BeL-GL e WBRA-GL, ficaram entre os quatros melhores percentuais.

Particularmente, a heurística BeL-GL é uma heurística extremamente simples, de fácil implementação e possui baixo custo computacional. Ainda, com os resultados do MGCPJ nas instâncias oriundas da discretização da condução do calor, pode-se considerar que, juntamente com a heurística CMr-GL, a heurística BeL-GL foi a melhor heurística na redução do custo computacional do MGCPJ.

Em relação à heurística WBRA-GL, observou-se que a modificação proposta neste trabalho da heurística WBRA original, diminuiu consideravelmente o custo computacional e a ocupação de memória dessa heurística. Embora apenas uma estrutura de nível seja gerada na heurística WBRA-GL em vez de várias, como na WBRA, a heurística WBRA-GL obteve resultados similares aos resultados da heurística WBRA nas reduções de largura de banda e de *profile*.

Com esses resultados, observa-se que heurísticas para a redução de largura de banda devem ser rápidas, se o intuito é obter redução do custo computacional na resolução de SELs.

6.5 Trabalhos futuros

Como trabalho futuro, pretende-se implementar outros preconditionadores para o MGC. Espera-se implementar o preconditionador *Incomplete LU with Threshold* (ILUT) e avaliar, com as reduções de largura de banda e de *profile*, a redução no custo computacional do MGC. Outros

precondicionadores que podem ser testados são os preconditionadores *Multigrid* algébrico, *Successive Over-Relaxation* e *Symmetric Successive Over-Relaxation*.

Ainda, simulações computacionais com SELs oriundos de outros problemas devem ser realizadas. Para isso, bases de SELs, disponíveis gratuitamente na rede, serão utilizadas.

Também pretende-se verificar se a heurística *VNS-band* é a heurística no estado da arte para a redução de *profile*. Para isso, comparações na redução de *profile*, com as heurísticas identificadas na revisão sistemática realizada por Bernardes e Oliveira (2015), precisam ser feitas.

Espera-se realizar um estudo sobre heurísticas que são paralelizáveis. Especificamente, pretende-se realizar uma revisão sistemática de heurísticas paralelas para a redução de largura de banda.

REFERÊNCIAS

ABBASBANDY, S.; Jafarian, A.; Ezzati, R. Conjugate gradient method for fuzzy symmetric positive definite system of linear equations. **Applied Mathematics and Computation**, v. 171, p. 1184-1191, Dec. 2005.

ABREU, A. A. A. M. de. **Reduções de largura de banda e de perfil de matrizes por mapa auto-organizável**. 2014. 126 p. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2014.

BAECK, T.; FOGEL, D.; MICHALEWICZ, Z. **Handbook of evolutionary computation**. New York: Taylor and Francis, 1997. 1130 p. Disponível em: <<http://books.google.com.br/books?id=n5nuiIZvmpAC>>. Acesso em: 12 dez. 2014.

BENZI, M. Preconditioning techniques for large linear systems: a survey. **Journal of Computational Physics**, v. 182, n. 2, p. 418-477, Nov. 2002.

BERNARDES, J. A. B.; OLIVEIRA, S. L. G. de. A systematic review of heuristics for profile reduction of symmetric matrices. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE, 2015, Reykjavík, Iceland. **Proceedings...** Reykjavík: ICCS, 2015.

BOUTORA, Y.; TAKORABET, N.; IBTIOUEN, R.; MEZANI, S. A new method for minimizing the bandwidth and profile of square matrices for triangular finite elements mesh. **IEEE Transactions on Magnetics**, v. 43, n. 4, p. 1513-1516, Apr. 2007.

BURDEN, R. L.; FAIRES, J. D. **Análise Numérica**. Boston, USA: Thomson Learning, 2003. 736 p.

BURGESS, D. A.; GILES, M. B. Renumbering unstructured grids to improve the performance of codes on hierarchical memory machines. **Advances in Engineering Software**, v. 28, p. 189-201, Feb. 1997.

BURGESS, I. W.; LAI, P. K. F. A new mode renumbering algorithm for bandwidth reduction. **International Journal for numerical methods in engineering**, v. 23, p. 1693-1704, 1986.

BURKARD, R. E.; CELA, E. Linear assignment problems and extensions. In: DU, D.-Z.; PARDALOS, P. M. (Eds.) *Handbook of Combinatorial Optimization: Supplement Volume A*, Kluwer Academic Publishers. New York: Springer US, 1999. p. 75-149.

CHAGAS, G. O. **Uma verificação do método dos gradientes conjugados com redução de largura de banda pelo método Cuthill-McKee reverso com vértice pseudoperiférico pelo algoritmo de George-Liu**. 2013. 51 p. Monografia (Bacharelado em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2013.

CHAGAS, G. O.; OLIVEIRA, S. L. G. de. Metaheuristic-based heuristics for symmetric-matrix bandwidth reduction: a systematic review. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE, 2015, Reykjavík, Iceland. **Proceedings...** Reykjavík: ICCS, 2015.

COPPERSMITH, D.; WINOGRAD, S. Matrix multiplication via arithmetic progressions. **Journal of Symbolic Computation**, v. 9, n. 3, p. 251-280, Mar. 1990.

CORMEN, T. H., et al. **Introduction to Algorithms**. 3^a ed. London, England: The MIT Press, 2009. 1251 p.

COWLING, P., KENDALL, G., SOUBEIGA, E. A hyperheuristic approach to scheduling a sales summit. In: Burke, E., Erben, W. (Ed.). **Practice and Theory of Automated Timetabling III, Lecture Notes in Computer Science**: volume 2079. Berlin: Springer, 2001. p. 176-190.

CUTHILL, E.; MCKEE, J. Reducing the bandwidth of sparse symmetric matrices. In: NATIONAL CONFERENCE, 24., 1969, New York. **Proceedings...** New York: ACM, 1969. p. 157-172.

DUFF, I. S.; GRIMES, R. G.; LEWIS, J. G. **Users' guide for the Harwell-Boeing sparse matrix collection (release I)**. France: CERFACS, 1992. 84 p. (Relatório técnico TR-PA-92-86).

DUFF, I. S.; MEURANT, G. A. The effect of ordering on preconditioned conjugate gradients. **BIT Numerical Mathematics**, Berlin, v. 29, n. 4, p. 635-657, 1989.

ESPOSITO, A.; CATALANO, M. S. F.; MALUCELLI, F.; TARRICONE, L. A new matrix bandwidth reduction algorithm. **Operations Research Letters**, Amsterdam, v. 23, n.3-5, p. 99-107, Oct. 1998.

ESPOSITO, A.; MALUCELLI, F. TARRICONE, L. Bandwidth and profile reuction of sparse matrices: an experimental comparison of new heuristics. In: **ALGORITHMS AND EXPERIMENTS**, 1998, Trento. **Proceedings...** Trento: SIAM, 1998. p. 19-26.

GEORGE, J. A. **Computer implementation of the finite element method**. Stanford: Computer Science Department, 1971. 228 p.

GEORGE, J. A.; LIU, J. W. H. An implementation of a pseudoperipheral node finder. **ACM Transactions on Mathematical Software**, New York, v. 5, n. 3, p. 284-295, Sept. 1979.

GEORGE, A.; LIU, J. W. H. **Computer Solution of Large Sparse Positive Definite Systems**. New York: Prentice-Hall, 1981. 314 p.

GIBBS, N. E.; POOLE, W. G.; STOCKMEYER, P. K. Algorithm for reducing the bandwidth and profile of a sparse matrix. **SIAM Journal on Numerical Analysis**, Philadelphia, v. 13, n. 2, p. 236-250, Apr. 1976.

GOLUB, G. H.; VAN LOAN, C. F. **Matrix Computations**. 3. ed. Baltimore: The Johns Hopkins University Press, 1996. 687 p.

HESTENES, M. R.; STIEFEL, E. Methods of conjugate gradients for solving linear systems. **Journal of Research of the National Bureau of Standards**, Essex, v. 49, n. 36, p. 409-436, Dec. 1952.

JACOBI, C. G. J. Über eine neue Auflösungsart der bei der methode der kleinsten quadrate vorkommenden linearen gleichungen, *astron. Nachrichten*, Stuttgart, v. 22, p. 297, 1845.

JENNINGS, A. A compact storage scheme for the solution of symmetric simulataneous equations. **The Computer Journal**, Oxford, v. 9, n. 3, p. 281-285, 1966.

KAVEH, A. Ordering for bandwidth reduction. **Computer Structure**, Oxford, v. 24, n. 3, p. 413-420, June 1986.

KAVEH, A. **Structural Mechanics: Graph and Matrix Methods**. Baldock, England: Research Studies Press LTD, 2004. 440p.

KAVEH, A.; SHARAFI, P. Ordering for bandwidth and profile minimization problems via charged system search algorithm. **IJST Transactions of Civil Engineering**, Elmsford, v. 36, n. 1, p. 39-52, Jan. 2012.

KAVEH, A.; TALATAHARI, S.; A novel heuristic optimization method: charged system search. **Acta Mechanica**, New York, v.213, n. 1, p. 267-286, Jan. 2010.

KOOHESTANI, B.; POLI, R. A genetic programming approach to the matrix bandwidth-minimization problem. In: 11th INTERNATIONAL CONFERENCE ON PARALLEL PROBLEM SOLVING FROM NATURE, 2011, Berlim. **Proceedings...** Berlim: PPSN, 2011. p. 482-491.

KOZA, J. R. G. P. **On the Programming of Computers by Means of Natural Selection**. Cambridge: MIT Press, 1992. 680 p.

KRYLOV, A. N. On the numerical solution of the equation by which in technical question frequencies of small oscillations of material systems are determined (em russo). **News of Academy of Sciences of the USSR**, v. 7, n 4, p. 491-539, Oct. 1931.

LANCZOS, C. Solutions of systems of linear equations by minimized iterations. **Journal of Research of the National Bureau of Standards**, Washington, v.49, n. 3, p. 33-53, July 1952.

LIM, A.; RODRIGUES, B.; XIAO, F. Heuristics for matrix bandwidth reduction. *European Journal of Operational Research*, Amsterdam, v. 174, n. 1, p. 69-91, Oct. 2006.

LIM, A.; RODRIGUES, B.; XIAO, F. A fast algorithm for bandwidth minimization. **International Journal on Artificial Intelligence Tools**, Singapore, v. 3, n. 3, p. 537-544, June 2007.

LIU, J. W. **On reducing the profile of sparse symmetric matrices**. 1976. 208 p. Tese (Doutorado em Ciência da Computação) - University of Waterloo, fev. 1976.

LIU, W.; SHERMAN, A. H. Comparative analysis of the Cuthill-McKee and reverse Cuthill-McKee ordering algorithms for sparse matrices. **SIAM Journal on Numerical Analysis**, Philadelphia, v. 13, n. 2, p. 198-213, Apr. 1976.

LUENBERGER, D. G.; **Linear and Nonlinear Programming**. Stanford: Springer, 2008. 492 p.

MADZVAMUSE, A.; GEORGE, U. Z. A multilevel Cholesky conjugate gradients hybrid solver for linear systems with multiple right-hand sides. **Finite Elements in Analysis and Design**, Amsterdam, v. 74, n. 3, p. 76-92, Dez. 2013.

MEIJERINK, J. A.; VAN DER VORST, H. A. An iterative solution method for linear systems of which the coefficient matrix is a symmetric

M-matrix. **Mathematics of Computation**, v. 31, n. 137, p. 148-162, Jan. 1977.

MLADENOVIC, N.; HANSEN, P. Variable neighborhood search. **Computers and Operations Research**, New York, v. 24, n. 11, p. 1097-1100, Nov. 1997.

MLADENOVIC, N. et al. Variable neighbourhood search for bandwidth reduction. **European Journal of Operational Research**, Amsterdam, v. 200, n. 1, p. 14-27, Jan. 2010.

OLIVEIRA, F. S. de. **Uma avaliação de movimento de malhas baseado na formulação laplaciana na resolução de equação de calor por discretizações de volumes finitos com refinamento de Delaunay**. 2014. 124 p. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2014a.

OLIVEIRA, S. L. G. de. **Simulações do método Cuthill-McKee reverso em matrizes de sistemas de equações lineares resolvidos pelo método dos gradientes conjugados pré-condicionado por Jacobi** Lavras, Brasil: Editora da UFLA, 2014b.7 p. (Relatório técnico RT2014-01).

OLIVEIRA, S. L. G. de; CHAGAS, G. O. A systematic review of heuristics for symmetric-matrix bandwidth reduction methods not based on metaheuristics (a ser publicado). In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 2015, Porto de Galinhas, Pernambuco, Brasil. **Proceedings...** Porto de Galinhas: SBPO, 2015a.

OLIVEIRA, S. L. G. de; CHAGAS, G. O. Adaptive mesh refinement for finite-volume discretizations with scalene triangles. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE, 2015, Reykjavík, Iceland. **Proceedings...** Reykjavík: ICCS, 2015b.

OLIVEIRA, S. L. G. de; CHAGAS, G. O. **Introdução à heurísticas para redução de largura de banda de matrizes**. São

Carlos, São Paulo, Brasil: SBMAC, 2014. 106 p. Disponível em:
<http://www.sbmac.org.br/arquivos/notas/livro_75.pdf>

OLIVEIRA, S. L. G. de; KISCHINHEVSKY, M. Sierpiński curve for total ordering of a graph-based adaptive simplicial-mesh refinement for finite volume discretizations. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, 2008, Belém, Brasil. **Anais...** Belém: CNMAC, 2008. p. 581-585.

OLIVEIRA, S. L. G. de; KISCHINHEVSKY, M.; TAVARES, J. M. R. S. Novel graph-based adaptive triangular mesh refinement for finite-volume discretizations. **Computer Modeling in Engineering and Science**, Porto, v.95, n. 2, p. 119-141, May 2013.

OLIVEIRA, S. L. G. de; OLIVEIRA, F. S.; CHAGAS, G. O. A novel approach to the weighted Laplacian formulation applied to 2D Delaunay triangulations. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE AND ITS APPLICATIONS, 2015, Banff, Canada. **Proceedings...** Banff: ICCSA, 2015.

PAPADIMITRIOU, C. The NP-completeness of bandwidth minimization problem. **Computing Journal**, Manchester, v. 16, n. 1, p. 177-192, Mar. 1976.

REID, J. K.; SCOTT, J. A. **Reducing the total bandwidth of a sparse unsymmetric matrix**. Chilton Didcot, UK: CCLRC Rutherford Appleton Laboratory, 2005. 16 p. (Relatório técnico RAL-TR-2005-001).

RODRIGUEZ-TELLO, E.; KAO, H. J.; TORRES0-JIMENEZ, J. An improved simulated annealing algorithm for bandwidth minimization. **European Journal of Operational Research**, Amsterdam, v. 185, n. 3, p. 1319-1335, Mar. 2008.

SAAD, Y. **Iterative Methods for Sparse Linear Systems**. 2^a ed. Philadelphia: SIAM, 2003. 517 p.

SHEWCHUK, J. R. **An introduction to the conjugate gradient method without the agonizing pain.** Pittsburgh, USA: School of Computer Science, Carnegie Mellon University, 1994. 64 p. (Relatório técnico ASC-9318163).

STEINER, G.; YEOMANS, J. S. A linear time algorithm for maximum matching in convex, bipartite graphs. **Computers and Mathematics with Applications**, Oxford, v. 31, n. 12, p. 91-96, June 1996.

TARJAN, R. E. **Graph theory and gaussian elimination.** Stanford: Stanford University. 1975. 15 p. (Relatório técnico STAN-CS75-526).

TEMAM, O. de; JALBY, W. Characterizing the behavior of sparse algorithms on caches. In: ACM/IEEE CONFERENCE ON SUPERCOMPUTING, 1992, Minneapolis. **Proceedings...** Minneapolis: SC92, 1992. p. 578-587.

VERSCHOOR, M.; JALBA, A. C. Analysis and performance estimation of the conjugate gradient method on multiple GPUs. **Parallel Computing**, v. 38, p. 552-575, 2012.

WANG, Q.; GUO, Y. C.; SHI, X. W. A generalized GPS algorithm for reducing the bandwidth and profile of a sparse matrix. In: PROGRESS IN ELECTROMAGNETICS RESEARCH, 2009, Massachusetts. **Proceedings...** Massachusetts: EMW Publishing, 2009. p. 121-136.

WILLIAMS, V. V. Multiplying Matrices Faster Than Coppersmith-Winograd. In: FORTY-FOURTH ANNUAL ACM SYMPOSIUM ON THEORY OF COMPUTING. **Proceedings...** New York: ACM, 2012. p. 887-898.

APÊNDICE

APÊNDICE A - Coeficientes de Variação e Desvios Padrões dos Testes nas Instâncias da Harwell-Boeing

Tabela 17 Desvio padrão (σ) e coeficiente de variação (C_v) dos resultados obtidos pelas heurísticas em relação aos tempos médios de execução nas 15 instâncias simétricas muito pequenas da base Harwell-Boeing.

Instância		CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
ASH85	σ	$1 \cdot 10^{-5}$	$6 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-1}$	$8 \cdot 10^{-6}$	$3 \cdot 10^{-3}$	$2 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$3 \cdot 10^{-5}$	$6 \cdot 10^{-2}$
	C_v	13,51%	2,48%	6,38%	0,12%	3,21%	11,58%	10,34%	0,97%	3,75%	7,44%	34,14%
BCSPWR01	σ	$2 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	$3 \cdot 10^{-1}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-3}$	$8 \cdot 10^{-7}$	$5 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$6 \cdot 10^{-6}$	$1 \cdot 10^{-2}$
	C_v	8,18%	16,23%	2,38%	0,25%	7,25%	13,48%	9,17%	4,10%	6,18%	3,63%	11,14%
BCSPWR02	σ	$6 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$5 \cdot 10^{-1}$	$9 \cdot 10^{-6}$	$7 \cdot 10^{-4}$	$1 \cdot 10^{-6}$	$7 \cdot 10^{-6}$	$6 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$2 \cdot 10^{-2}$
	C_v	26,77%	9,19%	6,35%	0,30%	5,62%	7,85%	9,03%	4,35%	5,49%	22,60%	14,42%
BCSPWR03	σ	$1 \cdot 10^{-5}$	$1 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$3 \cdot 10^{-3}$	$4 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$5 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$8 \cdot 10^{-2}$
	C_v	18,18%	33,77%	0,93%	0,17%	3,26%	13,14%	12,67%	8,84%	4,07%	9,66%	29,16%
BCSSTK01	σ	$1 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	$5 \cdot 10^{-4}$	$6 \cdot 10^{-1}$	$3 \cdot 10^{-5}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$3 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-2}$
	C_v	28,17%	12,55%	5,78%	0,37%	7,92%	9,37%	19,65%	10,17%	5,65%	7,17%	19,51%
BCSSTK04	σ	$3 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$6 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$6 \cdot 10^{-5}$	$6 \cdot 10^{-3}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$1 \cdot 10^{-4}$	$8 \cdot 10^{-2}$
	C_v	7,65%	11,69%	2,86%	0,10%	3,42%	5,79%	2,85%	21,36%	5,07%	6,81%	25,50%
BCSSTK05	σ	$1 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$7 \cdot 10^{-5}$	$6 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$5 \cdot 10^{-3}$	$9 \cdot 10^{-6}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-6}$	$8 \cdot 10^{-5}$	$9 \cdot 10^{-2}$
	C_v	3,75%	0,82%	0,23%	0,31%	8,36%	7,77%	12,20%	18,36%	1,12%	7,49%	26,82%
BCSSTK22	σ	$3 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-5}$	$3 \cdot 10^{-3}$	$4 \cdot 10^{-6}$	$6 \cdot 10^{-4}$	$4 \cdot 10^{-6}$	$3 \cdot 10^{-5}$	$9 \cdot 10^{-2}$
	C_v	2,70%	2,08%	1,01%	0,17%	5,22%	9,53%	13,91%	3,49%	4,15%	5,38%	30,66%
CAN_144	σ	$5 \cdot 10^{-5}$	$8 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-5}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-6}$	$1 \cdot 10^{-4}$	$1 \cdot 10^{-1}$
	C_v	29,76%	12,82%	0,23%	0,11%	5,74%	4,73%	5,69%	18,51%	2,01%	10,85%	23,75%
CAN_161	σ	$2 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$3 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$5 \cdot 10^{-3}$	$3 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$1 \cdot 10^{-1}$
	C_v	16,78%	12,85%	2,46%	0,16%	13,12%	7,16%	10,45%	11,20%	2,15%	2,29%	29,84%
DWT_234	σ	$1 \cdot 10^{-5}$	$3 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$4 \cdot 10^{-3}$	$3 \cdot 10^{-6}$	$4 \cdot 10^{-5}$	$3 \cdot 10^{-6}$	$4 \cdot 10^{-5}$	$2 \cdot 10^{-1}$
	C_v	12,79%	3,96%	2,18%	0,07%	1,76%	10,10%	9,57%	2,76%	2,33%	4,45%	39,86%
LUND_A	σ	$1 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$6 \cdot 10^{-5}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-4}$	$50 \cdot 10^{-3}$	$1 \cdot 10^{-5}$	$8 \cdot 10^{-4}$	$3 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$1 \cdot 10^{-1}$
	C_v	4,62%	1,19%	0,23%	0,15%	9,52%	6,99%	21,66%	13,11%	1,24%	6,88%	25,49%
LUND_B	σ	$6 \cdot 10^{-5}$	$3 \cdot 10^{-5}$	$6 \cdot 10^{-5}$	$4 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$5 \cdot 10^{-3}$	$2 \cdot 10^{-6}$	$4 \cdot 10^{-4}$	$2 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$1 \cdot 10^{-1}$
	C_v	20,13%	2,99%	0,21%	0,22%	3,00%	7,51%	4,08%	6,29%	1,13%	5,70%	26,09%
NOS1	σ	$3 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$7 \cdot 10^{-5}$	$9 \cdot 10^{-4}$	$8 \cdot 10^{-7}$	$5 \cdot 10^{-4}$	$2 \cdot 10^{-6}$	$3 \cdot 10^{-5}$	$5 \cdot 10^{-2}$
	C_v	21,06%	2,05%	2,12%	0,16%	6,53%	4,32%	4,56%	18,60%	1,87%	4,62%	11,57%
NOS4	σ	$4 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$7 \cdot 10^{-3}$	$8 \cdot 10^{-6}$	$4 \cdot 10^{-3}$	$2 \cdot 10^{-6}$	$7 \cdot 10^{-5}$	$3 \cdot 10^{-6}$	$3 \cdot 10^{-5}$	$5 \cdot 10^{-2}$
	C_v	4,98%	1,35%	5,21%	0,34%	1,94%	11,09%	8,86%	20,25%	3,50%	7,11%	14,17%

Tabela 19 Desvio padrão (σ) e coeficiente de variação (C_v) dos resultados obtidos pelas heurísticas em relação aos tempos médios de execução nas 18 instâncias assimétricas muito pequenas da base Harwell-Boeing.

Instância		CMr-GL	BeL-GL	hGPHH	WBRA-GL	VNS-Band (500s)	FNCHC	GPS	GGPS	BL	WBRA	CSS-Band
ARC130	σ	$2 \cdot 10^{-5}$	$2 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$4 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	$4 \cdot 10^{-4}$	$4 \cdot 10^{-5}$	$7 \cdot 10^{-2}$
	C_v	16,54%	7,89%	3,45%	11,32%	0,28%	4,52%	18,87%	4,99%	13,11%	0,24%	25,48%
CURTIS54	σ	$4 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$6 \cdot 10^{-6}$	$3 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	$5 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$8 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-3}$
	C_v	8,89%	20,20%	4,33%	2,29%	0,35%	8,52%	3,48%	1,95%	2,99%	0,68%	2,43%
FS_183.1	σ	$3 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$	$3 \cdot 10^{-5}$	$3 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	$4 \cdot 10^{-4}$	$1 \cdot 10^{-1}$
	C_v	2,70%	9,66%	2,87%	20,52%	0,26%	1,91%	3,45%	9,50%	3,53%	0,16%	31,78%
GENT113	σ	$4 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$8 \cdot 10^{-6}$	$3 \cdot 10^{-5}$	$5 \cdot 10^{-3}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-4}$	$2 \cdot 10^{-5}$	$6 \cdot 10^{-5}$	$6 \cdot 10^{-2}$
	C_v	5,28%	13,61%	4,10%	1,22%	0,92%	2,48%	2,50%	7,12%	3,59%	0,14%	20,24%
GRE_115	σ	$2 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$6 \cdot 10^{-6}$	$7 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$5 \cdot 10^{-5}$	$3 \cdot 10^{-5}$	$7 \cdot 10^{-5}$	$1 \cdot 10^{-4}$	$6 \cdot 10^{-2}$
	C_v	2,76%	15,45%	4,70%	8,70%	0,71%	3,29%	12,49%	6,34%	13,36%	0,25%	25,36%
GRE_185	σ	$2 \cdot 10^{-5}$	$5 \cdot 10^{-6}$	$5 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-5}$	$5 \cdot 10^{-4}$	$4 \cdot 10^{-5}$	$1 \cdot 10^{-1}$
	C_v	14,95%	7,16%	2,77%	12,64%	0,23%	2,12%	20,24%	0,89%	38,48%	0,17%	33,96%
IBM32	σ	$2 \cdot 10^{-6}$	$1 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$5 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	$6 \cdot 10^{-6}$	$7 \cdot 10^{-6}$	$9 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$3 \cdot 10^{-3}$
	C_v	10,33%	13,15%	6,69%	12,15%	0,43%	8,88%	5,98%	5,59%	7,06%	0,35%	3,10%
IMPCOL.B	σ	$3 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$6 \cdot 10^{-5}$	$6 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$9 \cdot 10^{-5}$	$8 \cdot 10^{-3}$
	C_v	8,31%	14,30%	5,48%	8,20%	0,41%	5,25%	8,50%	6,71%	2,60%	0,53%	8,32%
IMPCOL.C	σ	$1 \cdot 10^{-5}$	$2 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$2 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-4}$	$3 \cdot 10^{-5}$	$2 \cdot 10^{-2}$
	C_v	19,08%	8,40%	2,92%	1,69%	0,17%	4,32%	4,73%	1,57%	9,39%	0,29%	4,65%
LNS_131	σ	$3 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$5 \cdot 10^{-6}$	$8 \cdot 10^{-6}$	$3 \cdot 10^{-3}$	$6 \cdot 10^{-3}$	$8 \cdot 10^{-6}$	$4 \cdot 10^{-5}$	$7 \cdot 10^{-5}$	$5 \cdot 10^{-5}$	$8 \cdot 10^{-2}$
	C_v	5,61%	8,47%	7,60%	1,35%	0,16%	1,97%	2,02%	8,78%	7,35%	0,08%	25,08%
MCCA	σ	$7 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$6 \cdot 10^{-6}$	$3 \cdot 10^{-4}$	$4 \cdot 10^{-2}$	$5 \cdot 10^{-3}$	$3 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$5 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$1 \cdot 10^{-1}$
	C_v	1,95%	22,68%	2,88%	12,48%	0,11%	4,40%	12,27%	15,50%	14,01%	0,52%	29,49%
PORES1	σ	$5 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$5 \cdot 10^{-6}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$9 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$5 \cdot 10^{-3}$
	C_v	13,52%	14,89%	5,62%	3,14%	1,22%	11,44%	13,56%	4,78%	39,30%	0,93%	6,95%
STEAM3	σ	$5 \cdot 10^{-5}$	$2 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$1 \cdot 10^{-5}$	$5 \cdot 10^{-4}$	$5 \cdot 10^{-6}$	$4 \cdot 10^{-4}$	$3 \cdot 10^{-3}$
	C_v	31,93%	6,16%	6,85%	22,46%	2,06%	4,30%	2,56%	16,32%	2,09%	13,96%	2,65%
WEST0132	σ	$9 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$1 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-3}$	$1 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$7 \cdot 10^{-5}$	$4 \cdot 10^{-5}$	$8 \cdot 10^{-2}$
	C_v	9,14%	17,55%	3,78%	14,06%	0,14%	2,52%	18,78%	18,16%	6,67%	0,17%	27,00%
WEST0156	σ	$4 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$4 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$9 \cdot 10^{-6}$	$8 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$3 \cdot 10^{-4}$	$8 \cdot 10^{-2}$
	C_v	4,74%	15,39%	3,39%	1,54%	0,18%	4,44%	1,63%	0,98%	1,45%	0,20%	29,84%
WEST0167	σ	$3 \cdot 10^{-6}$	$4 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$1 \cdot 10^{-4}$	$3 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$2 \cdot 10^{-4}$	$3 \cdot 10^{-5}$	$3 \cdot 10^{-5}$	$3 \cdot 10^{-4}$	$9 \cdot 10^{-2}$
	C_v	2,72%	17,84%	2,54%	11,56%	0,27%	4,52%	21,31%	2,32%	3,39%	0,22%	31,34%
WILL57	σ	$7 \cdot 10^{-6}$	$2 \cdot 10^{-6}$	$3 \cdot 10^{-6}$	$5 \cdot 10^{-5}$	$1 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	$6 \cdot 10^{-5}$	$3 \cdot 10^{-5}$	$4 \cdot 10^{-3}$
	C_v	12,18%	17,86%	4,75%	12,77%	0,33%	3,20%	6,37%	4,42%	16,87%	0,55%	3,83%
WILL199	σ	$4 \cdot 10^{-6}$	$1 \cdot 10^{-5}$	$4 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-3}$	$6 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$8 \cdot 10^{-5}$	$2 \cdot 10^{-4}$	$1 \cdot 10^{-2}$
	C_v	3,42%	32,49%	2,38%	8,32%	0,22%	3,79%	1,73%	1,37%	2,90%	0,08%	5,44%

