



CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM MATEMÁTICA E
COMPUTACIONAL

Problemas de cobertura máxima e de cobertura de *hub* sob condição de incerteza com alocação única e custo fixo

Fernando Félix Oliveira e Silva

Belo Horizonte
Março de 2026

Fernando Félix Oliveira e Silva

Problemas de cobertura máxima e de cobertura de *hub* sob condição de incerteza com alocação única e custo fixo

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Modelagem Matemática e Computacional do CEFET-MG, como parte dos requisitos necessários para obtenção do título de Doutor em Modelagem Matemática e Computacional.

Área de Concentração: Modelagem Matemática e Computacional

Linha de Pesquisa: Sistemas Inteligentes

Orientadora: Profa. Dra. Elisangela Martins de Sá

Coorientador: Prof. Dr. Sérgio Ricardo de Souza

Belo Horizonte - MG
Março de 2026

S586p Silva, Fernando Félix Oliveira
Problemas de cobertura máxima e de cobertura de hub sob condição de incerteza com alocação única e custo fixo / Fernando Félix Oliveira e Silva. – 2026.
103 f.

Tese (doutorado) apresentada ao Programa de Pós-Graduação em Modelagem Matemática e Computacional.

Orientadora: Elisangela Martins de Sá.

Coorientador: Sérgio Ricardo de Souza.

Tese (doutorado) – Centro Federal de Educação Tecnológica de Minas Gerais.

1. Redes de computadores – Teses. 2. Incerteza (Economia) – Teses.
3. Otimização matemática – Teses. 4. Meta-heurística – Teses. 5. Ciência da computação – Matemática – Teses. 6. Aplicações matemáticas em ciência da computação – Teses. 7. Otimização robusta – Teses. I. Sá, Elisangela Martins de. II. Souza, Sérgio Ricardo de. III. Centro Federal de Educação Tecnológica de Minas Gerais. IV. Título.

CDD 519.6



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
COORDENAÇÃO DO PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM MATEMÁTICA E COMPUTACIONAL

**“PROBLEMAS DE COBERTURA MÁXIMA E DE COBERTURA DE HUB
SOB CONDIÇÃO DE INCERTEZA COM ALOCAÇÃO ÚNICA E CUSTO
FIXO”.**

Tese de Doutorado apresentada por **Fernando Félix Oliveira e Silva**, em 03 de março de 2026, ao Programa de Pós-Graduação em Modelagem Matemática e Computacional do CEFET-MG, e aprovada pela banca examinadora constituída pelos professores:

Profª. Drª. Elisangela Martins de Sá (Orientadora)
Centro Federal de Educação Tecnológica de Minas Gerais

Documento assinado digitalmente
gov.br **ELISANGELA MARTINS DE SA**
Data: 05/03/2026 07:12:23-0300
Verifique em <https://validar.itl.gov.br>

Prof. Dr. Sérgio Ricardo de Souza (Coorientador)
Centro Federal de Educação Tecnológica de Minas Gerais

Documento assinado digitalmente
gov.br **SERGIO RICARDO DE SOUZA**
Data: 05/03/2026 19:39:48-0300
Verifique em <https://validar.itl.gov.br>

Prof. Dr. Fabricio Alves Oliveira
Instituto Federal Catarinense

Documento assinado digitalmente
gov.br **FABRICIO ALVES OLIVEIRA**
Data: 04/03/2026 10:32:54-0300
Verifique em <https://validar.itl.gov.br>

Profª. Drª. Nayane Carvalho Freitas
Universidade Federal do Recôncavo da Bahia

Documento assinado digitalmente
gov.br **NAYANE CARVALHO FREITAS**
Data: 04/03/2026 13:17:44-0300
Verifique em <https://validar.itl.gov.br>

Prof. Dr. Marcone Jamilson Freitas Souza
Universidade Federal de Ouro Preto

Documento assinado digitalmente
gov.br **MARCONE JAMILSON FREITAS SOUZA**
Data: 04/03/2026 13:45:21-0300
Verifique em <https://validar.itl.gov.br>

Prof. Dr. Gustavo Campos Menezes
Centro Federal de Educação Tecnológica de Minas Gerais

Documento assinado digitalmente
gov.br **GUSTAVO CAMPOS MENEZES**
Data: 09/03/2026 09:01:09-0300
Verifique em <https://validar.itl.gov.br>

Visto e permitida à impressão,

Prof. Dr. Alisson Marques da Silva
Coordenador do Programa de Pós-Graduação Stricto Sensu em
Modelagem Matemática e Computacional

Dedico este trabalho a meus pais Félix
Ferreira da Silva (*in memoriam*) e Maria
Aparecida de Oliveira e Silva

Agradecimentos

Primeiramente, gostaria de manifestar minha gratidão a Deus.

Agradeço de coração à minha orientadora, Professora Elisangela Martins de Sá, por sua orientação inestimável, paciência, apoio constante e por ter aceitado me orientar. Seus conselhos, críticas construtivas e sua postura forte, mas serena foram essenciais para o desenvolvimento deste trabalho. Professora Elisangela, sou verdadeiramente grato pelo seu comprometimento e dedicação. Também agradeço ao meu coorientador Sérgio Ricardo de Souza por todas as contribuições e paciência.

Também sou grato aos membros da banca examinadora, pela disposição em avaliar esta tese.

Minha gratidão se estende aos professores e colegas do CEFET, que, com suas aulas e discussões, desempenharam um papel significativo na formação deste trabalho. Em especial, agradeço aos Professores Marcone, Beth, Albens e Flávio pelo apoio e pelas discussões estimulantes que ampliaram minha compreensão do tema e aos colegas Jardel, Romário, Leo, Marcos, Fábio, Alexandre e Anderson.

Sou igualmente grato a instituição onde trabalho, a UNIMONTES pelo suporte e apoio, que foram fundamentais para a realização desta pesquisa. Agradeço também aos meus colegas e amigos pelo incentivo e apoio contínuo ao longo de toda a minha trajetória acadêmica.

Não posso deixar de expressar minha profunda gratidão à minha família. Aos meus pais, Félix (*In Memoriam*) e Maria Aparecida, por sua compreensão, amor incondicional e apoio constante. Agradeço também a minha esposa, Kelly Jaciara, pelo carinho e paciência durante este processo intenso. Seu apoio emocional, amor, compreensão e incentivo foram indispensáveis.

Gostaria de dedicar um agradecimento aos meus irmãos, Daniel e Lívia. Vocês são uma parte fundamental da minha vida e sou profundamente grato por ter irmãos tão maravilhosos e por todo o suporte, alegria, momentos e carinho que vocês sempre me ofereceram.

Agradeço à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo apoio financeiro.

Finalmente, agradeço a todos que, de alguma forma, contribuíram para a realização deste trabalho. Cada um de vocês desempenhou um papel essencial, e serei eternamente grato por isso.

Resumo

Esta tese tem como objetivo estudar e propor algoritmos para problemas de localização de *hubs*, com foco nos problemas de cobertura, não capacitados, com alocação única e custos fixos, considerando uma rede completa e não permitindo conexões diretas entre nós não *hubs*. Inicialmente, o estudo propõe o problema denominado *Uncapacitated Single Allocation Hub Maximal Covering Problem* (USAHMCP), no qual se busca selecionar os *hubs* a serem abertos e determinar a alocação de cada nó não *hub* a um único *hub*, respeitando um orçamento para abertura e sem pré-fixar o número de *hubs*. O fato de considerar custos fixos de instalação e restrições orçamentárias distingue este trabalho de outros existentes na literatura. Além disso, demonstrou-se que o USApHMCP padrão (versão em que a quantidade de *hubs* é previamente fixada) é um caso particular do problema introduzido e que, portanto, este novo problema é NP-difícil. Para tratar o problema proposto, foram desenvolvidas duas heurísticas baseadas na meta-heurística *Iterated Local Search* com *Variable Neighborhood Descent* (ILS-VND), que se diferenciam quanto ao critério de alocação gulosa. A estratégia de alocação baseada em potencial de cobertura, proposta neste trabalho, apresentou melhor desempenho em comparação à estratégia de alocação baseada em distância, frequentemente utilizada na literatura. Os testes computacionais consideraram instâncias de até 1000 nós, utilizando quatro conjuntos clássicos: *Australian Post* (AP) com até 200 nós, *Civil Aeronautics Board* (CAB) com até 25 nós, *Contreras* com até 500 nós e *URAND* com 1000 nós. As heurísticas propostas foram capazes de atualizar as melhores soluções conhecidas para as instâncias de 1000 nós. Os resultados mostraram que, quando o custo de abertura dos *hubs* é maior em nós com maiores demandas, o problema tende a abrir mais *hubs* que o modelo tradicional. Por outro lado, quando essa diferenciação não ocorre, o número de *hubs* abertos tende a ser o mesmo, ainda que o conjunto de *hubs* selecionados possa variar. Em continuidade a esse estudo, uma extensão do clássico problema *Uncapacitated Single Allocation Hub Covering Problem* (USAHCP) foi proposta. A extensão foi denominada *Uncapacitated Single Allocation Hub Covering Problem with Minimum Coverage* (USAHCPMC) e se diferencia por permitir que nem toda a demanda da rede seja necessariamente totalmente coberta. O problema USAHCPMC visa selecionar os *hubs* a serem abertos e determinar a alocação de cada nó não-*hub* a um único *hub*, minimizando o custo com instalação de *hub* para cobrir pelo menos uma parte previamente fixada do fluxo total da rede. Dessa forma, o USAHCPMC é NP-difícil, já que é mais geral do que o problema USAHCP (que exige a cobertura de toda a demanda e já foi provado ser da classe NP-difícil). Além disso, este trabalho considera cobertura binária e parcial e analisa os impactos de se tratar incerteza no fluxo de demanda do nó de origem até o nó de destino por meio de uma

abordagem robusta. O problema robusto se baseia na modelagem do conjunto de incertezas utilizando restrições de cardinalidade. Um algoritmo ILS-VND foi proposto para resolver instâncias do problema nominal e do problema robusto. Experimentos computacionais, utilizando dois conjuntos de instâncias da literatura, mostraram que o problema proposto neste trabalho pode gerar soluções interessantes para os investidores, devido ao potencial de economias significativas nos custos de abertura de *hubs* ao não cobrir uma pequena parcela do fluxo total da rede. Além disso, experimentos computacionais considerando condições de incerteza mostraram que os custos de instalação de *hubs* aumentam (ou pelo menos permanecem os mesmos) à medida que o nível de robustez aumenta. Nós que não são instalados como *hubs* em cenários de baixa incerteza podem estar na configuração de *hubs* em cenários de maior incerteza. Finalmente, este trabalho mostrou que o modelo com cobertura parcial tende a exigir custos de abertura de *hubs* menores em comparação com o modelo com cobertura binária. Ademais, a heurística implementada conseguiu encontrar boas soluções em tempos computacionais menores do que o *solver* CPLEX.

Palavras-chave: Cobertura de *hub*, Cobertura máxima, Otimização sob incerteza, Meta-heurística, *Variable Neighborhood Descent*, *Iterated Local Search*, Otimização robusta.

Abstract

This thesis aims to study and propose algorithms for hub location problems, with a focus on uncapacitated covering problems with single allocation and fixed costs, considering a complete network and not allowing direct connections between non-hub nodes. Initially, the study proposes the problem called Uncapacitated Single Allocation Hub Maximal Covering Problem (USAHMCP), in which the goal is to select the hubs to be opened and determine the allocation of each non-hub node to a single hub, respecting a budget for opening and without predefining the number of hubs. The fact of considering fixed installation costs and budget constraints distinguishes this work from others in the literature. Furthermore, it was demonstrated that the standard USA ϕ HMCP (the version in which the number of hubs is previously fixed) is a particular case of the introduced problem and that, therefore, this new problem is NP-hard. To address the proposed problem, two heuristics based on the Iterated Local Search metaheuristic with Variable Neighborhood Descent (ILS-VND) were developed, which differ in the greedy allocation criterion. The allocation strategy based on coverage potential, proposed in this work, showed better performance compared to the distance-based allocation strategy frequently used in the literature. The computational tests considered instances of up to 1000 nodes, using four classical sets: Australian Post (AP) with up to 200 nodes, Civil Aeronautics Board (CAB) with up to 25 nodes, Contreras with up to 500 nodes, and URAND with 1000 nodes. The proposed heuristics were capable of updating the best-known solutions for the 1000-node instances. The results showed that, when the opening cost of the hubs is higher in nodes with higher demands, the problem tends to open more hubs than the traditional model. On the other hand, when this differentiation does not occur, the number of opened hubs tends to be the same, although the set of selected hubs may vary. Continuing this study, an extension of the classical Uncapacitated Single Allocation Hub Covering Problem (USAHCP) was proposed. The extension was named Uncapacitated Single Allocation Hub Covering Problem with Minimum Coverage (USAHCPMC) and differs by allowing that not all network demand must necessarily be fully covered. The USAHCPMC problem aims to select the hubs to be opened and to determine the allocation of each non-hub node to a single hub, minimizing the installation cost of hub to cover at least a previously fixed portion of the total network flow. Thus, the USAHCPMC is NP-hard, since it is more general than the USAHCP problem (which requires covering all demand and has already been proven to be NP-hard). In addition, this work considers binary and partial coverage and analyzes the impacts of dealing with uncertainty in the flow from the origin node to the destination node through a robust approach. The robust problem is based on modeling the uncertainty set using cardinality constraints. An

ILS-VND algorithm was proposed to solve instances of the nominal problem and the robust problem. Computational experiments, using two sets of instances from the literature, showed that the problem proposed in this work can generate interesting solutions for investors, due to the potential for significant savings in hub opening costs by not covering a small portion of the total network flow. Furthermore, computational experiments considering uncertainty conditions showed that the installation costs of hubs increase (or at least remain the same) as the level of robustness increases, and nodes that are not installed as hubs in low-uncertainty scenarios may belong to the hub configuration in higher-uncertainty scenarios. Finally, this work showed that the partial coverage model tends to require lower hub opening costs compared to the binary coverage model. Moreover, the implemented heuristic was able to find good solutions in computational times shorter than the CPLEX solver.

Keywords: Hub covering, Maximal covering, Optimization under uncertainty, Meta-heuristic, Variable Neighborhood Descent, Iterated local search, Robust optimization.

Sumário

Lista de Tabelas	xv
Lista de Figuras	xvi
Lista de Algoritmos	xvii
Lista de Símbolos	xviii
Lista de Nomenclatura	xx
1 Introdução	1
1.1 Justificativa	4
1.2 Objetivos	5
1.2.1 Objetivo geral	5
1.2.2 Objetivos específicos	5
1.3 Contribuições	6
1.4 Artigos derivados desta Pesquisa	6
1.4.1 Artigos em periódicos	6
1.4.2 Artigo publicado em anais de congresso	7
1.5 Organização do trabalho	7
2 Revisão bibliográfica e fundamentação teórica	8
2.1 Revisão bibliográfica	8
2.1.1 Problemas de cobertura máxima de <i>hub</i>	9
2.1.2 Problemas de cobertura de conjunto de <i>hub</i>	11
2.1.3 Condições de incerteza em problemas de cobertura e de cobertura máxima de <i>hub</i>	12
2.1.4 Análise da revisão de literatura	13
2.1.4.1 Sumário da revisão de literatura associada ao USAHMCP	13
2.1.4.2 Sumário da revisão de literatura associada ao USAHCPMC	14
2.2 Fundamentação teórica	16
2.2.1 Heurísticas e meta-heurísticas	16
2.2.2 Procedimento <i>Variable Neighborhood Descent</i> (VND)	17
2.2.3 <i>Iterated Local Search</i> (ILS)	18
2.2.4 Problemas de otimização robusta	19
3 Problema de cobertura máxima de <i>hub</i> com alocação única, não capacitado	24
3.1 Formulação e definição do problema USAHMCP	24
3.1.1 Comparação entre o USA p HMCP e o USAHMCP	26
3.2 Algoritmo ILS-VND proposto para o USAHMCP	29
3.2.1 Representação da solução	29

3.2.2	Função de avaliação	30
3.2.3	Construção da solução inicial	30
3.2.4	Estruturas de vizinhança	31
3.2.5	Procedimentos de alocação gulosa	32
3.2.6	Procedimento de busca local para o USAHMCP	33
3.2.7	<i>Algoritmo ILS-VND para solução do USAHMCP</i>	34
3.3	Etapas preliminares para os experimentos computacionais associados ao USAHMCP	37
3.3.1	Detalhes dos experimentos computacionais	37
3.3.2	Avaliação da importância das estruturas de vizinhança	40
3.3.3	Avaliação da ordem de exploração das estruturas de vizinhança	42
3.4	Experimentos computacionais usando as instâncias AP	44
3.4.1	Comparação entre USA p HMCP e USAHMCP	44
3.4.2	Comparação das variantes da heurística ILS-VND com a heurística da literatura aplicada ao USA p HMCP	47
3.4.3	Comparação da heurística com o <i>solver</i> CPLEX aplicado ao USAHMCP	50
3.5	Experimentos computacionais usando as instâncias CAB	52
3.5.1	Experimentos computacionais para o USA p HMCP usando as instâncias CAB	53
3.5.2	Experimentos computacionais para o USAHMCP usando as instâncias CAB	55
3.6	Experimentos computacionais usando as instâncias DS3	57
3.6.1	Teste de sensibilidade para o parâmetro β_{ij}	58
3.6.2	Desempenho das variantes da heurística aplicadas ao USA p HMCP	59
3.6.3	Desempenho das variantes da heurística aplicadas ao USAHMCP	60
3.7	Experimentos computacionais para o USA p HMCP usando as instâncias URAND de 1000 nós	66
3.8	Análise e discussão de resultados do USAHMCP	67
4	Problema de cobertura de <i>hub</i> com alocação única e cobertura mínima sob condições de incerteza	69
4.1	Formulação e definição do USAHCPMC	69
4.1.1	Definição do problema e formulação	69
4.1.2	Inserção de incerteza sobre o fluxo de demanda W_{ij}	72
4.2	Algoritmo proposto para o USAHCPMC	74
4.2.1	Representação da solução e procedimento de alocação gulosa	74
4.2.2	Função de avaliação	74
4.2.3	Construção da solução inicial	75
4.2.4	Estruturas de vizinhança	76
4.2.5	Procedimento de busca local para o USAHCPMC	77
4.2.6	Algoritmo ILS-VND para solução do USAHCPMC	77
4.3	Detalhes dos experimentos computacionais aplicados ao USAHCPMC	79
4.4	Comparação entre o USAHCPMC e o USAHCP	80
4.4.1	Comparação entre USAHCPMC e USAHCP usando a instância APL com 40 nós e $p = 4$	80

4.4.2	Comparação entre o USAHCPMC e o USAHCP usando a instância CAB com 25 nós, $p = 5$ e $\alpha = 0,8$	82
4.5	Comparação das soluções robustas e nominais para diferentes níveis de robustez e de incerteza	84
4.5.1	Resultados para a instância CAB25.2.6	84
4.5.2	Resultados para a instância AP40.4L	86
4.6	Performance do algoritmo heurístico proposto para o USAHCPMC	89
4.6.1	Resultados para as instâncias AP	89
4.6.2	Resultados dos testes usando as instancias CAB	92
4.7	Análise e discussão dos resultados do USAHCPMC	94
5	Considerações finais e direções futuras	96
5.1	Considerações finais	96
5.2	Direções futuras	98
	Referências	99

Lista de Tabelas

2.1	Principais características dos estudos que abordam os problemas de cobertura de <i>hub</i> relacionados ao USAHMCP.	14
2.2	Principais características dos estudos analisados sobre problemas de cobertura de <i>hub</i> relacionados ao USAHCPMC.	15
3.1	Valores para a razão $\bar{\beta}/\bar{d}$ para instâncias $APn.pL$	40
3.2	Valores de <i>gap</i> e tempo quando uma das estruturas de vizinhança é removida.	41
3.3	Resultados computacionais do VND na resolução de instâncias AP para cada ordem de exploração de estruturas de vizinhança	43
3.4	Resultados para as versões USApHMCP e USAHMCP usando o <i>solver</i> CPLEX em instâncias AP com até 50 nós.	45
3.5	Resultados para a heurística e o <i>solver</i> CPLEX aplicados ao problema USApHMCP usando instâncias AP em comparação com o algoritmo de busca tabu (TS) da literatura	49
3.6	Resultados das heurísticas e do <i>solver</i> CPLEX aplicados ao problema USAHMCP na resolução de instâncias APL.	51
3.7	Resultados das heurísticas e do <i>solver</i> CPLEX aplicados ao problema USAHMCP na resolução de instâncias APT.	52
3.8	Resultados das heurísticas e do <i>solver</i> CPLEX aplicados ao problema USApHMCP usando instâncias CAB de 10 e 15 nós.	54
3.9	Resultados de heurísticas e do <i>solver</i> CPLEX aplicados ao problema USApHMCP usando instâncias CAB de 20 e 25 nós.	55
3.10	Resultados das heurísticas aplicadas ao problema USAHMCP usando instâncias CAB com 10 e 15 nós.	56
3.11	Resultados das heurísticas aplicadas ao problema USAHMCP usando instâncias CAB com 20 e 25 nós.	57
3.12	Resultados do impacto de β na cobertura, tempo e número de <i>hubs</i> usando instâncias DS3 e AP.	59
3.13	Resultados das variantes da heurística proposta aplicada ao USApHMCP para instâncias do conjunto de dados DS3 com 50 e 100 nós.	60
3.14	Resultados das variantes da heurística proposta aplicada ao USApHMCP para instâncias do conjunto de dados DS3 com 150 e 200 nós.	61
3.15	Resultados das variantes da heurística proposta aplicada ao USApHMCP para instâncias do conjunto de dados DS3 com 250 e 500 nós.	62
3.16	Resultados da heurística proposta aplicada ao USAHMCP para instâncias do conjunto DS3 com 50 e 100 nós.	63

3.17	Resultados da heurística proposta aplicada ao USAHMCP para instâncias do conjunto DS3 com 150 e 200 nós.	64
3.18	Resultados da heurística proposta aplicada ao USAHMCP para instâncias do conjunto DS3 com 250 e 500 nós.	65
3.19	Resultados da heurística proposta aplicada ao USA ρ HMCP para instâncias URAND de 1000 nós.	66
4.1	Resultados para variações na cobertura mínima em incrementos de 5% da demanda total para o instância AP40.4L usando o CPLEX	80
4.2	Resultados para variações na cobertura mínima em incrementos de 1% da demanda total para o Instância AP40.4L usando CPLEX	81
4.3	Resultados para variações na cobertura mínima em incrementos de 5% da demanda total para o Instância CAB25.5.8 usando o CPLEX	82
4.4	Resultados para variações na cobertura mínima em incrementos de 1% da demanda total para o Instância CAB25.5.8 usando o CPLEX	83
4.5	Resultados para diferentes níveis de incerteza (γ) e variações percentuais do fluxo de demanda aplicados à instância CAB 25.2.6	85
4.6	Resultados para diferentes níveis de incerteza (γ) e variações percentuais do fluxo de demanda aplicados à instância AP40.4L	88
4.7	Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ usando Instâncias APL.	90
4.8	Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ aplicados à instância APL para instâncias em que não foi possível obter o <i>Lower Bound</i>	90
4.9	Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ usando Instâncias APT.	91
4.10	Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ aplicados a Instâncias APT para instâncias em que não foi possível obter o <i>Lower Bound</i>	91
4.11	Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ usando Instâncias CAB	93
4.12	Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ para instâncias CAB em casos em que não foi possível obter um <i>Lower Bound</i>	93

Lista de Figuras

3.1	Ilustração da rede para a solução ótima para instância AP com 5 nós para o $USApHMCP$ e para o $USAHMCP$	28
3.2	Ilustração da rede para a solução ótima para instância AP com 10 nós e $p = 5$ para o $USApHMCP$ e para o $USAHMCP$ com custos fixos do tipo L e do tipo T.	46
3.3	Ilustração da rede para a solução ótima para instância AP com 25 nós e $p = 4$ para o $USApHMCP$ e para o $USAHMCP$ com custos fixos do tipo L e do tipo T.	47
4.1	Valores de custo mínimo usando CPLEX com a instância AP40.4L.	81
4.2	Tempo de execução usando CPLEX para a instância AP40.4L.	82
4.3	Valores de custo mínimo usando CPLEX com a instância CAB25.5.8.	83
4.4	Tempo de execução usando CPLEX para a instância CAB25.5.8.	84
4.5	Valores da função objetivo e do tempo computacional do <i>solver</i> CPLEX para diferentes valores de variação de $\hat{W}_{ij}$ e de γ para uma instância CAB com 25 nós, $p = 2$ e $\alpha = 0,6$	85
4.6	Redes relacionadas à solução ótima da instância CAB25.2.6, com γ variando de 0,0 a 0,9, onde o tamanho dos nós representa a amplitude dos custos de abertura e a intensidade da cor representa a amplitude do fluxo de demanda no nó.	87
4.7	Valores da função objetivo e do tempo de execução do <i>solver</i> CPLEX para diferentes valores de variação do fluxo de demanda e de γ para a instância APL com 40 nós e $p = 4$	89

Lista de Algoritmos

1	VND Básico	18
2	ILS Básico	19
3	VND Cíclico para o USAHMCP	34
4	Procedimento de perturbação para o USAHMCP	35
5	ILS-VND para o USAHMCP	36
6	VND Cíclico para o USAHCPMC	77
7	Procedimento de perturbação para o USAHCPMC	78
8	ILS-VND para o USAHCPMC	79

Lista de Símbolos

D : Demanda mínima a ser coberta

$DS3$: Conjunto de instâncias de Contreras

FO^* : Melhor valor de função objetivo

N : Conjunto de nós

W_{ij} : Fluxo do nó de origem até o nó de destino

Γ : Parâmetro para nível de robustez do problema robusto

α : Fator de desconto no custo de transporte por meio de conexões entre *hubs*

$\bar{f}$: Custo médio de instalação de *hub*

$\bar{p}$: Quantidade de *hubs* da solução corrente

β_{ij} : Limiar de cobertura do par de origem e destino i, j

χ : Fator de coleta do nó de origem até o *hub*

δ : Fator de distribuição do *hub* até o nó de destino

γ : Razão entre Γ e cardinalidade dos parâmetros incertos

$\hat{W}_{ij}$: Parâmetro de nível de incerteza associado ao problema robusto

λ_{ij} : Parâmetro auxiliar de limite superior associado a a_{ij}^{km} e s_{ij}^{km}

$\mathcal{N}_i$: Estrutura de vizinhança i

$\overline{FO}$: Média dos valores de função objetivo

$\overline{gap}$: Média dos valores de *gap*

a_{ij}^{km} : Parâmetro de cobertura associada a cobertura binária

b : Orçamento total disponível para instalação de *hubs*

c_{ij}^{km} : Custo unitário de transporte de um nó de origem i para o nó de destino j passando pelos *hubs* k e m

d : Percentual da demanda total correspondente a D

d_{ij} : Distância entre os nós i e j

f_k : Custo fixo de abertura de um *hub* no nó k

gap^* : Melhor valor de *gap*

n : Quantidade de nós

p : Parâmetro da instância

s_{ij}^{km} : Parâmetro de cobertura associada a cobertura parcial

x_{ij} : Variável binária de decisão

z_{ij} : Variável não negativa de decisão

Lista de Nomenclatura

AP Australian Post

CAB Civil Aeronautics Board

DBA Distance Based Allocation heuristic

FO Valor de função Objetivo

GA Genetic Algorithm

GVNS General Variable Neighborhood Search

HCP Hub Covering Problem

HLP Hub Location Problem

HMCP Hub Maximal Covering Problem

ILS Iterated Local Search

OPH Ordering Heuristic Procedure

SA Simulated Annealing

TS Tabu Search

USAHCP *Uncapacitated Single Allocation Hub Covering Problem*

USAHCPMC Uncapacitated Single Allocation Hub Covering Problem with Minimum Coverage

USAHMCP Uncapacitated Single Allocation Hub Maximal Covering Problem

VBA Volume Based Allocation heuristic

VND Variable Neighborhood Descent

VNS Variable Neighborhood Search

Capítulo 1

Introdução

Em uma rede *hub-and-spoke*, os fluxos originários de diferentes nós de origem são enviados para nós *hub*, sendo consolidados e enviados para o nó de destino por meio de um caminho contendo apenas *hubs* como nós intermediários. Assim, os *hubs* são instalações especiais da rede que servem como pontos para consolidação, transbordo e desconsolidação do fluxo. Devido à consolidação de fluxos originários e destinados a diferentes nós e ao tráfego desse fluxo consolidado por meio de conexões entre *hubs*, pode ser possível obter economia de escala no transporte por meio de conexões entre *hubs*.

Os problemas de localização de *hub* (*Hub Location Problem* – HLPs) focam principalmente em determinar quais nós serão *hubs* de um conjunto de potenciais *hubs* e possuem diversas aplicações. Aplicações do problema de localização de *hub* foram observadas em sistemas de companhias aéreas, telecomunicações e logística, dentre outros. Por exemplo, em sistemas de companhias aéreas, o problema pode ser usado para identificar os aeroportos que servem como centros de conexão para voos domésticos e internacionais, a fim de minimizar o custo ou o número de voos para atender a uma determinada demanda (Bryan & O’Kelly, 1999). Em sistemas de telecomunicações, o problema pode ser usado para identificar os nós responsáveis pela transferência de dados na rede *backbone* e por enviá-los para a rede local (Chung et al., 1992). Em sistemas de logística, o problema pode ser usado para identificar a localização de centros de distribuição que minimizam o custo total de transporte (Ernst & Krishnamoorthy, 1996).

Diversas variantes de problemas de localização de *hub* são conhecidas na literatura. Os problemas podem ter um número predeterminado de *hubs* (problemas *p-hubs*) ou não. No último caso, a quantidade de *hubs* é uma variável no problema, que é determinada, em alguns casos, com base em um custo fixo para abrir cada *hub*. Quando um nó não *hub* pode ser atendido por apenas um *hub*, é chamado de problema de alocação única; por outro lado, se um nó não *hub* pode ser atendido por mais de

um *hub*, é chamado de problema de alocação múltipla. Quando nós e arcos não têm limite de capacidade de fluxo, o problema é não capacitado; caso contrário, é um problema capacitado. Além disso, os problemas podem se concentrar em uma rede de *hub* completa ou incompleta. Em redes completas, todos os nós *hub* são interconectados, enquanto em redes de *hub* incompletas, há nós *hub* que não são diretamente interconectados. Farahani et al. (2013) discutem as variantes e aplicações do problema de localização de *hub* em mais detalhes. Este trabalho considera um problema de localização de *hub* de alocação única, não capacitado, com uma rede completa.

Com base em uma rede de transporte de passageiros de companhias aéreas, a primeira formulação para um problema de localização de *hub* foi proposta por O’Kelly (1987) para o problema *p-hub* mediana de alocação única. Este problema de localização de *hub* envolve selecionar p nós para serem *hubs* e alocar cada nó não *hub* para um único *hub*, de forma que a soma dos custos de transporte entre os pontos de origem e destino seja minimizada. Uma limitação deste problema é que ele ignora o custo de abertura dos *hubs*. Para abordar esta limitação, O’Kelly (1992) propôs o problema de localização de *hub* com custos fixos, que determina a localização dos *hubs* com objetivo de minimizar o custo total dado pela soma dos custos de transporte e custos para instalar os *hubs*, assumindo que o número de *hubs* é uma variável no problema.

Campbell (1994) introduziu três novos tipos de problemas de localização de *hub*: (i) problemas *p-hub* centro; (ii) problemas de cobertura de conjunto de *hub*; e (iii) problemas de cobertura máxima de p *hubs*. Os problemas *p-hub* centro focam em minimizar o tempo ou a distância máxima, enquanto os outros dois tipos de problemas são problemas de cobertura de *hub*. Nesse tipo de problema, o custo pode representar distância ou tempo de transporte. Assim, os problemas de cobertura de conjunto de *hub* buscam determinar os nós *hubs* para minimizar o custo de abertura dos *hubs* e garantir que toda a demanda será coberta, enquanto que os problemas de cobertura máxima de p *hub* visam maximizar a demanda total coberta com um número predeterminado de *hubs*.

Em problemas de cobertura de *hub*, inicialmente, havia apenas a noção de cobertura binária, na qual a demanda de um nó de origem para um nó de destino é coberta se, e somente se, o custo do caminho que conecta esses dois nós, que contém no máximo dois *hubs* e tem apenas *hubs* como nós intermediários, for menor ou igual a um valor especificado. Peker & Kara (2015) introduziram a noção de cobertura parcial em problemas de cobertura de *hub*. Nesse tipo de cobertura, assume-se um grau de cobertura do par origem e destino usando uma função de decaimento decrescente com imagem no intervalo $(0, 1)$ considerando o tamanho ou custo do caminho. Os autores trabalharam com problema de cobertura máxima de *hub* apresentando

novos modelos para os casos com alocação única e múltipla.

Este trabalho aborda o problema de cobertura máxima de *hub* de alocação única não capacitado (*Uncapacitated single-allocation hub maximum coverage problem* - USAHMCP). Todos os estudos analisados que tratam dessa versão do problema de cobertura, como será apresentado na Seção 2.1.1 de revisão de literatura, assumem que o número de *hubs* é predeterminado, o que é conhecido como problema de cobertura máxima de p *hub* de alocação única não capacitado (USApHMCP). No entanto, a suposição de que o número de *hubs* a serem instalados é igual a p considera indiretamente que há um orçamento disponível para instalar p *hubs*, independentemente de sua localização. Porém, em algumas aplicações práticas, o custo de instalação de *hubs* pode variar significativamente de uma região para outra. Assim, o orçamento disponível pode ser suficiente para abrir um determinado conjunto de *hubs*, mas insuficiente para abrir o mesmo número de *hubs* em diferentes regiões.

Desta forma, inicialmente, este trabalho propõe o USAHMCP com custos fixos, que, por uma questão de simplicidade, será chamado de USAHMCP no restante deste texto. Este problema assume que há um custo fixo de instalação para cada nó *hub*, que pode variar de nó para nó, e que há um orçamento disponível, o que limita o custo total para instalação de *hubs*. Desta forma, o número de *hubs* que serão abertos é uma variável de decisão. É importante notar que, esta versão do problema, que considera o orçamento disponível para abertura de *hubs*, tem a vantagem de ser uma versão mais geral, no sentido de que o USApHMCP pode ser visto como um caso particular deste problema.

Este trabalho também trata do problema *Uncapacitated Single Allocation Hub Covering Problem with Minimum Coverage* (USAHCPMC) com objetivo de minimizar o custo total com abertura de *hubs* e não exigindo que 100% da demanda seja necessariamente coberta. Critérios de cobertura binária e parcial são analisados. Os estudos que abordam essa versão do problema assumem que toda a demanda deve ser coberta e são conhecidos na literatura como *uncapacitated single allocation hub covering problem* (USAHCP). Vale ressaltar que o USAHCP é um caso particular do USAHCPMC.

Em aplicações práticas, o custo de instalação dos *hubs* para cobrir toda a demanda pode ser consideravelmente superior ao custo de cobrir um valor um pouco menor da demanda total. Assim, pode ser interessante para o investidor conhecer possíveis economias que se obtém ao deixar de cobrir uma pequena parcela da demanda total. Ademais, como os valores numéricos associados ao problema podem não ser sempre tão precisos e alterações na rede podem gerar um alto custo para os investidores, considerar a incerteza nos dados do problema pode gerar configurações de rede menos onerosas e mais confiáveis. Por isso, esse trabalho também considera as incertezas no

fluxo de demanda do nó de origem até o nó de destino considerando a abordagem de otimização robusta de [Bertsimas & Sim \(2003\)](#).

Para resolver estes problemas, formulações de programação linear inteira mistas são apresentadas. Além disso, como pode não ser possível resolver instâncias de grande dimensão dos problemas usando um *solver* como o CPLEX, também foram propostos algoritmos baseados na meta-heurística *Iterated Local Search* (ILS) combinada com o procedimento de busca local *Variable Neighborhood Descent* (VND) para o USAHMCP (capítulo 3) e para USAHCPMC (capítulo 4), a fim de encontrar soluções de alta qualidade para os problemas em um tempo computacional aceitável.

Para o USAHMCP, duas heurísticas foram propostas, a primeira é baseada na estratégia de alocação ao *hub* mais próximo e a segunda em uma alocação baseada no potencial de cobertura, que foi proposta pela primeira vez neste trabalho. Para testar a escalabilidade das heurísticas propostas aplicadas ao USAHMCP, instâncias da literatura com até 1000 nós foram usadas. Para o USAHCPMC, apenas a heurística com alocação baseada em potencial de cobertura foi utilizada e os testes consideraram instâncias de até 200 nós.

1.1 Justificativa

Os problemas de cobertura e de cobertura máxima de *hub* possuem muitas aplicações práticas e são cruciais em diversas áreas, como logística, transporte, telecomunicações e planejamento, de forma que a eficiência na alocação de recursos e na cobertura de áreas geográficas é essencial para otimizar operações e reduzir custos. Este trabalho visa contribuir para o desenvolvimento de novos problemas de cobertura de *hub*, mais genéricos e com grande potencial de aplicações práticas, além da proposição de modelos matemáticos para eles. A tese propõe (i) um problema de cobertura máxima de *hub* que considera o orçamento para abertura de *hub*, em vez de fixar previamente uma quantidade de *hubs* a serem abertos; (ii) um problema de cobertura de *hub* que considera incerto o fluxo de demanda de um nó para outro e também considera uma demanda mínima a ser coberta, em vez de exigir a cobertura total da rede. Assim, (i) e (ii) resultam em contribuições significativas para a literatura acadêmica, ampliando o conhecimento teórico e prático nesse campo. Em instâncias com números de nós maiores ou iguais a 150, o *solver* CPLEX se mostrou inviável para resolução dos problemas, visto que estes problemas pertencem à classe NP-difícil e, assim, as heurísticas implementadas podem contribuir com boas soluções em bons tempos computacionais.

1.2 Objetivos

1.2.1 Objetivo geral

O objetivo geral desta tese é estudar os problemas de cobertura máxima e de cobertura de *hub* (este último considerando e não considerando condições de incerteza) não capacitado, com alocação única e com custo fixo em rede completa, bem como desenvolver modelos matemáticos e propor algoritmos meta-heurísticos para solucionar os problemas.

1.2.2 Objetivos específicos

Para alcançar o objetivo geral deste trabalho, têm-se os seguintes objetivos específicos.

- Propor formulação matemática para tratar o USAHMCP;
- Avaliar e estudar estruturas de vizinhanças e métodos heurísticos utilizados em problemas de cobertura e de cobertura máxima de *hub*;
- Propor algoritmos para tratar o USAHMCP;
- Comparar os resultados alcançados com os resultados da literatura;
- Propor formulação matemática para tratar o USAHCPMC e sua versão robusta considerando o fluxo de demanda como incerto;
- Estudar e propor algoritmo para o problema USAHCPMC em sua versão nominal e também robusta considerando o fluxo do nó de origem até o nó de destino incerto;
- Analisar os impactos nas soluções ao variar o valor mínimo da demanda a ser coberta na versão nominal do USAHCPMC;
- Analisar os impactos nas soluções robustas ao variar o nível de incerteza e o nível de robustez.
- Analisar o desempenho da heurística proposta aplicada à versão robusta do USAHCPMC considerando critério de cobertura binária e também de cobertura parcial.

1.3 Contribuições

Este trabalho possui as seguintes contribuições:

- (i) A introdução de um novo problema de localização de *hubs*, o USAHMCP, com base no orçamento disponível para abertura de *hubs* em vez de um número predeterminado de *hubs*. Segundo o nosso conhecimento, este é o primeiro estudo em que o USAHMCP é abordado;
- (ii) A proposição de uma formulação de programação linear inteira mista para modelar o problema USAHMCP;
- (iii) A proposição de duas variantes de um algoritmo baseado em ILS e VND que encontram boas soluções em tempos computacionais razoáveis para resolver instâncias do USAHMCP e, conseqüentemente, para o caso particular, USApHMCP. As variantes diferem no procedimento guloso para alocar nós não *hub* a *hubs* na busca local. Uma variante emprega o procedimento de alocação para o *hub* mais próximo, comumente usado para vários tipos de HLPs de alocação única, e a outra variante emprega um critério guloso com base no potencial de cobertura, proposto neste trabalho;
- (iv) A introdução de uma nova classe de problema de cobertura de *hub* que se diferencia por não exigir a cobertura total da demanda (*Uncapacitated Single Allocation Hub Covering Problem with Minimum Coverage* - USAHCPMC). Até onde é de nosso conhecimento, esta é a primeira vez que esta versão do problema é abordada;
- (v) A proposição de uma formulação de programação linear inteira mista para modelar o problema USAHCPMC e sua versão robusta com fluxo de demanda incerto;
- (vi) A proposição de algoritmo heurístico para tratar incerteza nos dados de fluxo de demanda do problema USAHCPMC.

1.4 Artigos derivados desta Pesquisa

A seguir são listados os trabalhos derivados da presente pesquisa.

1.4.1 Artigos em periódicos

1. Título: *A GVNS algorithm applied to the uncapacitated single allocation hub maximal covering problem with fixed costs*

- Autores: Fernando Félix Oliveira e Silva, Elisangela Martins de Sá, Sérgio Ricardo de Souza
 - Periódico: *Operational Research: An International Journal*.
 - Trabalho aceito para publicação
2. Título: *An ILS-VND algorithm for the robust uncapacitated single allocation hub covering problem with minimum coverage*
- Autores: Fernando Félix Oliveira e Silva, Elisangela Martins de Sá, Sérgio Ricardo de Souza
 - Periódico: *International Journal of Logistics Systems and Management*.
 - Trabalho aceito para publicação

1.4.2 Artigo publicado em anais de congresso

1. SILVA, Fernando.; SÁ, Elisangela Martins de.; SOUZA, Sergio Ricardo de. *Uma metaheurística Iterated Local Search para o problema p-hub de máxima cobertura não capacitado com alocação única. ANAIS DO LIV SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL*. 2022, Juiz de Fora (MG). DOI: 10.59254/sbpo-2022-157502. Disponível em: <https://proceedings.science/sbpo/sbpo-2022/trabalhos/uma-metaheuristica-iterated-local-search-para-o-problema-p-hub-de-maxima-cobertu?lang=pt-br>.

1.5 Organização do trabalho

Esta tese está organizada da seguinte forma: o Capítulo 2 apresenta uma revisão da literatura de problemas de cobertura e cobertura máxima de *hub* e condições de incerteza nesses dois problemas, bem como apresenta a fundamentação teórica sobre condições de incerteza em problemas de otimização. Além disso, fundamenta a teoria sobre heurística e meta-heurísticas, em especial apresentando os algoritmos VND e ILS. O Capítulo 3 apresenta as definições, notação e formulação relacionadas ao USAHMCP e descreve a representação da solução, função de avaliação, estruturas de vizinhanças, algoritmos, resultados e conclusões relacionados ao USAHMCP. O Capítulo 4 apresenta parâmetros e formulação para as versões nominal e robusta do USAHCPMC, bem como descreve os experimentos computacionais realizados e conclusões relacionados ao problema USAHCP com e sem condições de incerteza. Finalmente, o Capítulo 5 apresenta as conclusões gerais e possíveis linhas de pesquisa futuras.

Capítulo 2

Revisão bibliográfica e fundamentação teórica

Este capítulo apresenta uma revisão da literatura sobre problemas de cobertura de *hub*, bem como os fundamentos teóricos associados a problemas dessa classe. O capítulo está organizado em duas seções. A Seção 2.1 é dividida em quatro subseções. A Subseção 2.1.1 apresenta uma revisão de problemas de cobertura máxima de *hub* (HMCP). A Subseção 2.1.2 aborda trabalhos relacionados ao problema de cobertura de conjunto de *hub* (*Hub Covering Problem* – HCP). A Subseção 2.1.3 trata estudos sobre condições de incerteza em problemas de cobertura de *hub*. A Subseção 2.1.4 analisa a literatura e apresenta tabelas, resumindo as características dos problemas abordados nos artigos relevantes para o Capítulo 3 (Subseção 2.1.4.1) e para o Capítulo 4 (Subseção 2.1.4.2). Na Seção 2.2, é apresentada a fundamentação teórica para este trabalho. A Subseção 2.2.1 revisa os principais conceitos de heurísticas e metaheurísticas. A Subseção 2.2.2 aborda os conceitos de VND e a Subseção 2.2.3 apresenta o ILS. Esta parte do capítulo foi baseada no livro [Gendreau et al. \(2010\)](#). Por fim, a Subseção 2.2.4 apresenta a fundamentação teórica do estudo de condições de incerteza em parâmetros dos problemas de otimização.

2.1 Revisão bibliográfica

Para facilitar a apresentação, as abreviações para os principais tipos de problemas serão definidas da seguinte forma. Imediatamente antes da terminologia referente a cada tipo de problema (HMCP e HCP), serão apresentados nesta sequência: (i) o tipo de restrição de capacidade, em que “U” será usado para problemas não capacitados e “C” para problemas capacitados; (ii) o tipo de alocação, em que “SA” corresponde à alocação única e “MA” à alocação múltipla; e (iii) se o número de *hubs* é prede-

terminado, é feita a inclusão de “ p ” logo após a especificação do tipo de alocação; caso contrário, nada é especificado. Assim, o problema USA p HMCP é um HMCP não capacitado com alocação única e um número predefinido de *hubs*.

2.1.1 Problemas de cobertura máxima de *hub*

Campbell (1994) introduziu o USA p HMCP e o UMA p HMCP e também apresentou as primeiras formulações matemáticas para esses problemas. Posteriormente, Marianov et al. (1999) abordaram um UMA p HMCP que considera um ambiente competitivo. Nesse caso, a demanda de um par de nós de origem e destino é coberta se houver um caminho entre os dois nós com um custo de transporte menor que o custo de transporte por meio de companhias aéreas concorrentes. Os autores propuseram uma formulação e uma heurística baseada na metaheurística Busca Tabu (Glover, 1989) para resolver instâncias do problema. Eles também apresentaram resultados de experimentos computacionais usando o conjunto de dados do *Australian Post* (AP) com 20, 25, 40 e 50 nós, encontrando boas soluções.

Weng et al. (2006) propuseram um novo modelo matemático e heurísticas com base nas metaheurísticas Busca Tabu e Algoritmo Genético (Holland, 1975) para o UMA p HMCP. Experimentos computacionais foram conduzidos usando o conjunto de dados AP e uma instância dos fluxos de carga aérea chinesa entre 82 cidades em 2002. Os autores concluíram que as duas heurísticas funcionam bem para o problema e que, enquanto o Algoritmo Genético encontrou melhores resultados, o procedimento de Busca Tabu mostrou melhor eficiência computacional. Qu & Weng (2009) aplicaram uma abordagem evolucionária baseada em reconexão por caminhos para o UMA p HMCP. Os autores apresentaram os resultados de experimentos computacionais usando o conjunto de dados AP, concluindo que a heurística teve um bom desempenho.

Hwang & Lee (2012) apresentaram uma formulação de programação inteira para o USA p HMCP e duas heurísticas, as quais chamaram de *Distance Based Allocation heuristic* (DBA) e *Volume Based Allocation heuristic* (VBA). As duas heurísticas diferem em sua estratégia de alocação, em que uma é baseada na distância e a outra é baseada no volume de demanda. Os autores compararam o desempenho de suas heurísticas com o desempenho do *solver* CPLEX, usando instâncias derivadas do conjunto de dados *Civil Aeronautics Board* (CAB) e concluíram que as heurísticas encontram boas soluções em tempos computacionais razoáveis.

Peker & Kara (2015) propuseram a classificação da cobertura em duas categorias: cobertura binária e cobertura parcial. Na cobertura binária, qualquer par origem-destino é considerado coberto se o custo do caminho da origem ao destino for menor

ou igual a um valor dado, e não coberto, caso contrário. Por outro lado, na cobertura parcial, um grau de cobertura dos pares origem-destino é definido com base em uma função de decaimento não crescente, considerando o custo do caminho. Além de introduzir o conceito de cobertura parcial para o HLP, [Peker & Kara \(2015\)](#) apresentaram novos modelos para o USApHMCP e o UMApHMCP, com um menor número de variáveis e restrições que os modelos existentes. Além disso, os autores demonstraram que ambos os problemas são NP-difíceis. O artigo também apresentou resultados de experimentos computacionais usando os conjuntos de dados CAB e *Turkish network*.

[Janković et al. \(2017\)](#) propuseram novos modelos matemáticos para o USApHMCP e para o UMApHMCP. Os autores mostraram que o novo modelo matemático para o UMApHMCP tem melhores resultados computacionais do que aqueles apresentados em [Peker & Kara \(2015\)](#), mas o mesmo não se aplica ao USApHMCP. [Janković et al. \(2017\)](#) também propuseram um algoritmo *General Variable Neighborhood Search* ([Mladenović & Hansen, 1997](#)) para o USApHMCP e um algoritmo *Basic Variable Neighborhood Search* para o UMApHMCP, apresentando resultados de experimentos computacionais usando instâncias do conjunto de dados AP com até 200 nós e instâncias URAND com até 1000 nós. [Silva & Cunha \(2017\)](#) aplicaram uma heurística de Busca Tabu em instâncias com 200 nós do conjunto de dados AP e em instâncias CAB para o problema USApHMCP. Os autores usam a estratégia de alocação baseada no *hub* mais próximo em sua busca local com base em trocas entre um nó não *hub* e um *hub*.

Uma aplicação prática interessante do USApHMCP pode ser vista em [Chanta & Sangsawang \(2019\)](#), em que aplicaram o problema para determinar a localização de estações ferroviárias visando maximizar a cobertura de passageiros. Eles afirmam que tal projeto é parte do plano do governo na Tailândia. Para resolver instâncias desse problema, eles usam uma heurística baseada em *Simulated Annealing* ([Kirkpatrick et al., 1983](#)).

Mais recentemente, [Maleki et al. \(2023\)](#) introduziram um novo modelo para o UMApHMCP, que, segundo eles, é superior aos modelos previamente conhecidos para esse problema, pois tem um número muito menor de variáveis binárias. Os autores testaram a solução de seu modelo usando o *solver* Gurobi em instâncias AP, demonstrando bons tempos de execução para instâncias com até 100 nós.

Outras versões do HMCP incluem, por exemplo, versões com r -alocação e versões baseadas em uma abordagem multiobjetivo. Em um HLP de r -alocação, proposto por [Yaman \(2011\)](#), cada nó não *hub* pode ser alocado a no máximo r *hubs*. [Janković & Stanimirović \(2017\)](#) abordaram um HMCP não capacitado com r -alocação e um número predeterminado de *hubs*. No campo multiobjetivo, [Madani et al. \(2018\)](#) abordaram uma versão biobjetiva do HMCP que visa maximizar a cobertura esperada

e minimizar o congestionamento, assumindo um número predeterminado de *hubs* e alocação única. Os autores apresentam um modelo linear e um não linear, além de proporem um algoritmo *Nondominated Sorting Genetic Algorithm II (NSGA II)* para resolver instâncias do problema.

2.1.2 Problemas de cobertura de conjunto de *hub*

Conforme mencionado no Capítulo 1, [Campbell \(1994\)](#) introduziu o USAHCP e o UMAHCP. O autor apresentou formulações lineares e não lineares para o UMAHCP e apresentou indicações para adaptar essas formulações para o USAHCP. Os problemas propostos pelo autor assumem que há um custo de instalação para cada *hub* e focam em minimizar o custo total de instalação dos *hubs*.

Vários outros trabalhos abordaram versões do HCP que visam minimizar o número de *hubs* instalados. Dentre esses trabalhos, [Kara & Tansel \(2003\)](#) propuseram uma formulação de programação não-linear inteira mista para o USAHCP. Além disso, os autores propõem uma linearização para essa formulação e para a formulação não linear introduzida por [Campbell \(1994\)](#). Neste trabalho, os autores também mostraram que o problema é NP-difícil. Os autores aplicaram seu modelo para resolver instâncias do conjunto de dados CAB e mostraram que seu novo modelo linearizado obtém melhores resultados. [Wagner \(2008\)](#) propuseram formulações matemáticas com resultados de tempos computacionais melhores para o UMAHCP e o USAHCP com base na formulação proposta por [Kara & Tansel \(2003\)](#). Os autores abordaram versões do problema com tempo de transporte dependente e independente da quantidade de fluxo enviado do nó de origem até o nó de destino. [Hamacher & Meyer \(2006\)](#) analisaram a relação entre o problema de centro de p *hubs* e os problemas de cobertura de conjunto de *hubs* e propuseram uma nova formulação com base nessa relação. [Ernst et al. \(2011\)](#) melhoraram a formulação de programação inteira existente para o USAHCP incluindo novas variáveis e desigualdades que se tornam facetas do poliedro. Além disso, os autores desenvolveram uma nova formulação para o problema.

Outros autores que estudaram versões do HCP são [Karimi & Bashiri \(2011\)](#), que propuseram uma formulação quadrática para resolver o problema de cobertura do conjunto de *hubs* com alocação única não capacitado e utilizam uma abordagem de linearização do modelo, de forma que o tempo de viagem (custo ou distância) entre dois nós origem-destino não exceda um limite dado. Duas heurísticas de ordenação (*Ordering Heuristic Procedure* – OHP1 e OHP2) foram aplicadas ao problema. [Alumur et al. \(2012\)](#) consideraram alocação única sem restrições de capacidade e rede completa para investigar o problema de minimização de custos na cobertura de *hubs*. O problema envolve a instalação simultânea de dois tipos de *hubs* e conexões entre

hubs para transporte terrestre e aéreo, assegurando entregas dentro de prazos específicos. Considerando o raio de cobertura como uma variável de decisão do modelo, [Ebrahimi-zade et al. \(2016\)](#) propuseram um modelo de programação linear inteira mista para um problema de cobertura de conjunto de *hubs* de alocação única multi-período e utilizam um algoritmo genético (*Genetic Algorithm* - GA) ([Holland, 1975](#)) modificado para resolvê-lo.

Alguns trabalhos abordaram versões do HCP com redes de *hub* incompletas que, além de definir a localização dos *hubs* e a alocação de nós não *hub* a *hubs*, também devem determinar quais conexões entre *hubs* serão instaladas. [Calik et al. \(2009\)](#) estudaram o problema de cobertura de *hub* de alocação única em redes de *hub* incompletas. Os autores propuseram uma formulação de programação inteira para modelar o problema e apresentaram uma heurística baseada na metaheurística de Busca Tabu. Eles testaram sua heurística usando as instâncias CAB e as instâncias *Turkish Network* do transporte de carga turco. [Alumur & Kara \(2009\)](#) propuseram um novo modelo matemático para o HCP considerando redes de *hub* incompletas com foco em aplicações em transporte de carga.

2.1.3 Condições de incerteza em problemas de cobertura e de cobertura máxima de *hub*

Considerando incerteza em problemas de cobertura e cobertura máxima de *hubs*, [Davari et al. \(2013\)](#) estudaram redes incompletas considerando incerta a localização dos pontos de demanda. A incerteza foi modelada por variáveis *Fuzzy*. Os autores resolveram instâncias CAB por meio do algoritmo *Variable Neighborhood Search* (VNS). Já [Mohammadi et al. \(2014\)](#) introduziram um modelo matemático de localização de hubs sustentável (no sentido de que minimiza impactos ambientais) que considera custos incertos. Os dados incertos foram tratados utilizando uma abordagem de programação mista *Fuzzy-Estocástica*. Técnicas heurísticas baseadas em *simulated annealing* (SA) e *imperialist competitive algorithm* (ICA) foram utilizadas para resolver o problema. Os autores consideraram diversos parâmetros como incertos, a saber: custo unitário do transporte de fluxo, tempo unitário de transporte, custo fixo de abertura de um *hub*, capacidade de fluxo de coleta nos *hubs*, custo máximo de coleta/distribuição, número médio de veículos que circulam pelo *hub* por hora, velocidade média, potência do veículo e coeficiente de capacidade de armazenamento nos *hubs*.

[Sadeghi et al. \(2015\)](#) estudaram o problema de cobertura de *hub* considerando incertas as capacidades das conexões e resolveram o problema usando um algoritmo metaheurístico de *Evolution Differential* (DE) ([Storn & Price, 1997](#)). Os autores

utilizaram uma abordagem estocástica para minimizar o custo total de transporte. Wang & Qin (2020) estudaram o problema de cobertura, cobrindo 100% da demanda e de cobertura máxima de *hub*, considerando os fluxos e tempos de viagem como valores incertos. Os autores propuseram modelos de programação estocástica com restrição de probabilidade e utilizaram um algoritmo meta-heurístico híbrido chamado GA-VNS combinando a meta-heurística *Variable Neighborhood Search* (VNS) com *Genetic Algorithm* (GA).

Alguns autores abordaram problemas de cobertura de *hubs* em condições de incerteza, incluindo novas restrições aos problemas clássicos. Ghaffari-Nasab et al. (2015) abordaram problemas de cobertura de *hubs* com capacidade limitada, em que os fluxos de mercadorias entre pares de nós de clientes são incertos. O problema visa determinar o número, a localização e a capacidade dos *hubs*, com foco na minimização do custo total. Uma abordagem de otimização robusta é empregada para lidar com a incerteza. Bidoki et al. (2021) apresentaram um modelo matemático multiperíodo, considerando competitividade, preços, incerteza da demanda e congestionamento. Os autores tratam a incerteza estocasticamente, com foco em questões marítimas, assumindo que os produtos são perecíveis, as capacidades dos *hubs* são limitadas e o congestionamento ocorre se a demanda exceder um valor definido. Eles propuseram um modelo matemático não linear para abordar o problema e um procedimento baseado em um *Genetic Algorithm* (GA). Shang et al. (2025) abordaram o problema da cobertura hierárquica de *hubs* para minimização de custos com demanda incerta em um sistema de transporte urbano abrangente. Inicialmente foi apresentado um modelo determinístico e posteriormente os autores trataram o problema da incerteza adotando informações de distribuição e, nos casos em que essas informações são inexistentes, apresentaram um modelo robusto. Além disso, eles propuseram um procedimento baseado em *Genetic Algorithm* para abordar o problema.

2.1.4 Análise da revisão de literatura

Esta subseção analisa a revisão de literatura pertinente para o USAHMCP na subseção 2.1.4.1, bem como para o USAHCPMC na subseção 2.1.4.2.

2.1.4.1 Sumário da revisão de literatura associada ao USAHMCP

A Tabela 2.1 apresenta um resumo das principais características dos estudos apresentados nas Seções 2.1.1 e 2.1.2, que são os mais associados ao problema apresentado no Capítulo 3 desta tese. A tabela inclui o tipo de alocação, se o número de *hubs* é predefinido ou não (# de *hubs* predefinido), o tipo de problema e o método de resolução utilizado. A coluna “Método” indica se o trabalho propôs um modelo matemático

(“modelo”), um algoritmo heurístico (“heurística”), ou tanto um modelo quanto uma heurística (“ambos”).

Tabela 2.1: Principais características dos estudos que abordam os problemas de cobertura de *hub* relacionados ao USAHMCP.

Artigos	Tipo de alocação	# de <i>hubs</i> predefinido	Tipo de problema	Método
Campbell (1994)	ambos	ambos	HMCP e HCP	modelo
Marianov et al. (1999)	MA	sim	HMCP	ambos
Kara & Tansel (2003)	SA	não	HCP	modelo
Weng et al. (2006)	MA	sim	HMCP	ambos
Wagner (2008)	ambos	não	HCP	modelo
Hamacher & Meyer (2006)	SA	não	HCP	modelo
Calik et al. (2009)	SA	não	HCP	ambos
Alumur & Kara (2009)	SA	não	HCP	modelo
Qu & Weng (2009)	MA	sim	HMCP	heurística
Ernst et al. (2011)	SA	não	HCP	modelo
Hwang & Lee (2012)	SA	sim	HMCP	ambos
Peker & Kara (2015)	ambos	sim	HMCP	modelo
Janković et al. (2017)	ambos	sim	HMCP	ambos
Silva & Cunha (2017)	SA	sim	HMCP	heurística
Maleki et al. (2023)	MA	sim	HMCP	modelo
Janković & Stanimirović (2017)	rA	sim	HMCP	ambos
Madani et al. (2018)	SA	sim	HMCP (bi-objetivo)	ambos
Chanta & Sangsawang (2019)	SA	sim	HMCP	ambos
Karimi & Bashiri (2011)	ambos	não	HMCP e HCP	ambos
Alumur et al. (2012)	SA	não	HCP	modelo
Ebrahimi-zade et al. (2016)	SA	não	HCP	ambos
Trabalho sobre USAHMCP	SA	não	HMCP	ambos

De acordo com a Tabela 2.1, conclui-se que apenas a presente tese aborda um problema HMCP em que o número de *hubs* não é predefinido. Adicionalmente, além deste trabalho, apenas Campbell (1994) e Karimi & Bashiri (2011) consideraram um problema de cobertura de *hub* no qual o número de *hubs* é determinado pelo modelo e há um custo fixo para a instalação de cada *hub*. Entretanto, Campbell (1994) e Karimi & Bashiri (2011) consideraram uma versão sem número predefinido de *hubs* apenas para a versão HCP. Todos os outros trabalhos, nos quais o número de *hubs* não é determinado antecipadamente, abordaram uma versão do HCP baseada na minimização do número de *hubs*, sem considerar o custo de instalação desses *hubs*. Portanto, nenhum dos estudos citados possui características idênticas a este trabalho.

Também é importante destacar que o USA p HMCP, que assume um número predefinido de *hubs*, é um caso particular do problema abordado neste trabalho. Portanto, o algoritmo proposto neste estudo também pode ser aplicado para resolver instâncias do USA p HMCP.

2.1.4.2 Sumário da revisão de literatura associada ao USAHCPMC

A Tabela 2.2 resume as principais características dos estudos apresentados nas Seções 2.1.2 e 2.1.3, que são os mais associados ao trabalho apresentado no Capítulo 4 desta tese. A tabela mostra o tipo de problema e se o autor considerou que toda a

demanda deveria ser totalmente coberta ou não. Além disso, a tabela indica se o estudo considerou ou não a incerteza nos parâmetros. A coluna $< 100\%D$ indica com “sim” se o estudo considera que nem toda a demanda precisa necessariamente ser coberta e com “não”, caso contrário. A coluna Tipo de cobertura indica se a cobertura é binária e/ou parcial. A coluna Método indica o método utilizado para resolver o problema. Finalmente, a coluna Fonte de incerteza indica qual parâmetro foi considerado incerto no estudo.

Na coluna de fonte de incerteza, as notações LC, W_{ij} , t e CC indicam incerteza na localização dos pontos de demanda, no fluxo de demanda, no tempo de viagem entre nós e na capacidade da conexão entre nós, respectivamente. No estudo de Mohammadi et al. (2014), vários coeficientes foram considerados incertos, conforme a Subseção 2.1.3.

Tabela 2.2: Principais características dos estudos analisados sobre problemas de cobertura de *hub* relacionados ao USAHCPMC.

Artigos	Tipo de problema	Aborda incerteza	$< 100\%D$	Fonte de incerteza	Tipo de cobertura	Método
(Campbell, 1994)	HMCP e HCP	Não	Sim	-	binária	-
(Kara & Tansel, 2003)	HCP	Não	Não	-	binária	<i>solver</i>
(Wagner, 2008)	HCP	Não	Não	-	binária	<i>solver</i>
(Hamacher & Meyer, 2006)	HCP	Não	Não	-	binária	<i>Binary Search</i>
(Ernst et al., 2011)	HCP	Não	Não	-	binária	<i>solver</i>
(Calik et al., 2009)	HCP	Não	Não	-	binária	Heurística (TS)
(Alumur & Kara, 2009)	HCP	Não	Não	-	binária	<i>solver</i>
(Karimi & Bashiri, 2011)	HCP	Não	Não	-	binária	Heurística (OHP)
(Alumur et al., 2012)	HCP	Não	Não	-	binária	Relaxação LP
(Ebrahimi-zade et al., 2016)	HCP	Não	Não	-	binária	Heurística (GA)
(Davari et al., 2013)	HCP	<i>Fuzzy</i>	Não	LC	binária	Heurística (VNS)
(Mohammadi et al., 2014)	HCP	<i>Fuzzy-Estocástico</i>	Não	Várias	binária	Heurística (SA/ICA)
(Sadeghi et al., 2015)	HCP	Estocástico	Não	CC	binária	Heurística (DE)
(Wang & Qin, 2020)	HCP	Estocástico	Não	W_{ij} e t	binária	Heurística (GA-VNS)
(Shang et al., 2025)	HCP	Robusta	Não	W_{ij}	binária	Heurística (GA)
(Bidoki et al., 2021)	HCP	Estocástico	Não	W_{ij}	binária	Não linear
(Ghaffari-Nasab et al., 2015)	HCP	Robusta	Não	W_{ij}	binária	<i>solver</i>
Trabalho sobre USAHCPMC	HCP	Robusta	Sim	W_{ij}	ambas	Heurística (ILS-VND)

De acordo com a Tabela 2.2, observa-se que apenas a presente tese aborda o HCP no qual a demanda não precisa ser totalmente coberta. Portanto, nenhum dos estudos analisados apresenta características idênticas às do presente estudo.

É importante destacar que a ideia de considerar que nem toda a demanda precisa ser coberta, embora não abordada nos problemas de cobertura em *hubs*, não é inédita. No contexto dos problemas de localização de instalações, Daskin & Owen (1999) apresentam o *Partial Set Covering Location Problem*, cujo objetivo é selecionar instalações a serem abertas de modo a minimizar o custo total de instalação, garantindo que ao menos uma demanda mínima seja coberta.

Vale ressaltar que existe uma relação direta entre o USAHCPMC e o USAHCP tradicional, que exige a cobertura integral da demanda, de modo que o problema proposto neste trabalho em seu capítulo 4 pode ser visto como uma generalização

do USAHCP, e o algoritmo desenvolvido para o USAHCPMC pode ser aplicado à resolução de instâncias do USAHCP convencional.

2.2 Fundamentação teórica

Esta seção está organizada em quatro subseções, contendo conceitos sobre metaheurísticas (Seção 2.2.1), VND (Seção 2.2.2), ILS (Seção 2.2.3) e otimização robusta (Seção 2.2.4).

2.2.1 Heurísticas e meta-heurísticas

Dentre as formas de resolver um problema de otimização, existem: métodos exatos e métodos heurísticos. Os métodos exatos são geralmente fundamentados na modelagem matemática e garantem a geração da solução ótima, mas podem exigir um tempo computacional demasiadamente grande para encontrar uma solução ótima. Por outro lado, os métodos heurísticos não garantem a geração da solução ótima, mas, em geral, conseguem encontrar boas soluções em tempos computacionais satisfatórios. Alguns problemas de otimização, como os problemas tratados nesta tese, são NP-difíceis, não sendo conhecido nenhum algoritmo que o resolva em tempo polinomial, o que justifica a abordagem heurística.

Segundo Gendreau et al. (2010), as heurísticas construtivas constroem uma solução elemento a elemento. As heurísticas de refinamento (ou heurísticas de busca local) partem de uma solução completa e são baseadas na noção de vizinhança. A partir da definição de algum movimento e dada uma solução corrente, os vizinhos desta solução são gerados pela realização do movimento sobre a solução corrente e compõem a vizinhança dessa solução. Um movimento é definido como sendo uma modificação aplicada à solução corrente, produzindo uma nova solução dentro da vizinhança definida para o problema, enquanto que vizinhança é o conjunto de todas as soluções que podem ser obtidas a partir da solução atual por meio de um único movimento permitido.

Vale ressaltar que o movimento depende do problema. Por exemplo, para um problema de localização de *hubs* com alocação única, pode-se definir um movimento como sendo a troca de alocação de algum nó não *hub*. Todas as soluções que podem ser obtidas a partir de uma solução corrente pela troca de alocação são vizinhas dela.

Uma solução s_0 é um ótimo local quando nenhum de seus vizinhos retorna um valor de função objetivo maior que a função objetivo aplicada em s_0 para problemas de maximização (ou menor, para problemas de minimização). Uma solução s_0 é um ótimo global se nenhuma outra solução factível (solução que satisfaz todas as

restrições) retorna um valor de função objetivo maior para problemas de maximização (ou menor, para problemas de minimização) que a função objetivo aplicada em s_0 .

As heurísticas de busca local tentam melhorar uma solução corrente. A nova solução, gerada a partir de uma solução e um movimento, é escolhida segundo alguma estratégia. As principais estratégias são:

- (i) *Best improvement*: Essa estratégia analisa todas as soluções vizinhas e escolhe a melhor para ser a nova solução corrente. Esse processo é feito até que não se consiga melhorar mais a solução.
- (ii) *First improvement*: Esse método explora as soluções vizinhas e adota a primeira que for melhor que a solução atual.
- (iii) *Random improvement*: Esse método explora aleatoriamente soluções vizinhas, enquanto houver melhoria.

No procedimento de busca local, há um risco da solução ficar presa em ótimos locais e, para tentar evitar isto, as meta-heurísticas são propostas (Glover & Kochenberger, 2003). Segundo Blum & Roli (2003), as meta-heurísticas são baseadas em busca local ou busca populacional. São exemplos de meta-heurísticas baseadas em busca local: Busca Tabu (Glover, 1989), *Simulated Annealing* (Kirkpatrick et al., 1983), *Iterated Local Search* (ILS) (Lourenço et al., 2003) e são exemplos de meta-heurísticas baseadas em busca populacional: Algoritmos Genéticos (Holland, 1975), Colônia de Formigas (Dorigo et al., 1996). Esta tese utiliza a meta-heurística ILS.

2.2.2 Procedimento *Variable Neighborhood Descent* (VND)

O método *Variable Neighborhood Descent* (VND) é um procedimento de busca local, proposto por Mladenović & Hansen (1997), que explora diferentes estruturas de vizinhanças. Ele utiliza os seguintes princípios básicos: (i) um ótimo local com relação a uma vizinhança não necessariamente corresponde a um ótimo com relação a outra vizinhança; (ii) um ótimo global corresponde a um ótimo local para todas as estruturas de vizinhança; e, para muitos problemas; (iii) ótimos locais com relação a uma vizinhança são relativamente próximos. Pesquisas recentes que utilizam o procedimento VND ou VNS incluem Vincent et al. (2024), Guo et al. (2023) e Vukićević et al. (2023).

O Algoritmo 1 mostra um procedimento VND básico, em que max é a quantidade de estruturas de vizinhança. De forma geral, o algoritmo começa com uma vizinhança específica, inicia um *loop* na linha 4 e seleciona um vizinho dentro dessa vizinhança. A partir da linha 6, se uma melhoria no valor de função objetivo é encontrada dentro da

vizinhança atual, a solução é atualizada e retorna para primeira vizinhança; porém, caso a busca na vizinhança atual não consiga encontrar melhorias no valor de função objetivo, o VND passa para a próxima vizinhança (linha 10) no conjunto definido e esse processo é repetido até que todas as vizinhanças tenham sido exploradas sem que haja melhoria, ou até que uma condição de parada seja atendida.

Algoritmo 1 VND Básico

```

1: Entrada: Solução  $s$ 
2: Saída: Solução  $s$ 
3:  $k \leftarrow 1$ 
4: while  $k \leq \max$  do
5:    $\bar{s} \leftarrow$  busca de um vizinho melhor na vizinhança  $\mathcal{N}_k$  de  $s$ 
6:   if valor de função objetivo melhorou then
7:      $s \leftarrow \bar{s}$ 
8:      $k \leftarrow 1$ 
9:   else
10:     $k \leftarrow k + 1$ 
11:   end if
12: end while
13: Retorne  $s$ .
```

A ordem de exploração da vizinhança após alguma melhora diferencia os tipos de VND. No *Sequential* VND (VND básico apresentado no Algoritmo 1), se houver melhoria em uma vizinhança, retorna-se à primeira delas; caso contrário, passa-se para a vizinhança seguinte. Já no *Pipe* VND, se houver melhoria em uma vizinhança, permanece-se nela; caso contrário, passa-se para a vizinhança seguinte. Por outro lado, o *Cyclic* VND passa para a vizinhança seguinte, independentemente de haver ou não melhora na solução atual. Neste trabalho, é utilizado o *Cyclic* VND.

2.2.3 Iterated Local Search (ILS)

O algoritmo *Iterated Local Search* (ILS), apresentado em Lourenço et al. (2003), é uma meta-heurística que consiste em explorar o espaço de soluções por meio de perturbações em ótimos locais, seguidas de busca local. A perturbação modifica a solução corrente, guiando a busca para uma solução intermediária. A perturbação precisa ser suficientemente forte para permitir que a busca local explore diferentes soluções e fraca o suficiente para evitar um reinício aleatório.

O ILS combina intensificação com diversificação e, enquanto a intensificação consiste em explorar a região atual de busca e é obtida fazendo-se busca local na solução corrente, a diversificação consiste em mudar a região de busca e é obtida aumentando-se gradativamente a quantidade de perturbações na solução ótima local corrente. O

Algoritmo 2 ILS Básico

```

1: Entrada:  $ILSmax$  e  $s_0$ 
2:  $s \leftarrow$  Busca local ( $s_0$ )
3:  $Iter \leftarrow 0$  (Contador de iterações sem melhoria)
4:  $nivel \leftarrow 1$ 
5: while  $Iter \leq ILSmax$  do
6:    $Iter \leftarrow Iter + 1$ 
7:    $\bar{s} \leftarrow$  perturbação( $s, nivel$ )
8:    $s_1 \leftarrow$  busca local sobre  $\bar{s}$ 
9:   if valor de função objetivo melhorou then
10:     $s \leftarrow s_1$ 
11:     $Iter \leftarrow 0$ 
12:     $nivel \leftarrow 1$ 
13:   else
14:     $nivel \leftarrow nivel + 1$ 
15:   end if
16: end while
17: Retorne  $s$ .

```

aumento gradativo da quantidade de perturbações é determinado pelo nível de perturbação. Quanto maior o nível da perturbação, mais “distante” a solução não perturbada está da solução perturbada. Um exemplo de perturbação de nível 1 seria dois movimentos aleatórios utilizando uma estrutura de vizinhança.

Um pseudocódigo para o ILS é apresentado no Algoritmo 2. De forma geral, o algoritmo ILS básico tem, como entrada, o parâmetro $ILSmax$, que determina a quantidade máxima de iterações sem melhora que o algoritmo irá realizar, e uma solução inicial (s_0).

Inicialmente, na linha 2 é feita uma busca local sobre a solução inicial e, enquanto o critério de parada não for satisfeito (linha 5), o número de iterações é incrementado em uma unidade e na linha 7 uma perturbação é aplicada na solução corrente respeitando o nível atual. Na linha 8 é feito um procedimento de busca local sobre a solução perturbada. A partir da linha 9, se a solução encontrada for melhor que a solução corrente, a solução corrente é atualizada, a quantidade de iterações sem melhoria é zerada e retorna-se ao primeiro nível de perturbação. Caso a solução não seja melhor que a melhor solução encontrada até então, o nível é incrementado em uma unidade na linha 14.

2.2.4 Problemas de otimização robusta

A otimização sob incertezas é uma técnica que considera que certos parâmetros de entrada do problema podem apresentar variabilidade ou não ser totalmente co-

nhcidos. Segundo [Ben-Tal & Nemirovski \(2000\)](#), soluções ótimas de problemas de otimização linear (sem considerar condições de incerteza) podem se tornar ineficazes se os valores dos parâmetros do problema forem perturbados. Isto é, considerar condições de incerteza em dados de entrada que podem apresentar variação é essencial para encontrar soluções mais confiáveis.

O primeiro trabalho formal que tratou explicitamente condições de incerteza em problema de otimização foi [Dantzig \(1955\)](#). O autor formulou um modelo de programação linear no qual alguns parâmetros (como os parâmetros da matriz e o vetor de recursos) eram aleatórios com distribuições de probabilidade conhecidas. Mais tarde, os trabalhos de [Soyster \(1973\)](#) e [Ben-Tal & Nemirovski \(2000\)](#) consolidaram a teoria de otimização robusta. [Soyster \(1973\)](#) propôs um modelo de otimização linear que considera os dados incertos dentro de um conjunto convexo, oferecendo alta proteção contra incertezas e soluções conservadoras e [Ben-Tal & Nemirovski \(2000\)](#) desenvolveram modelos menos conservadores, baseados em conjuntos elipsoidais de incerteza.

O problema nominal de otimização (sem considerar condições de incerteza) admite que os parâmetros do modelo (valores presentes na função objetivo e/ou os valores presentes nas restrições e/ou termos independentes) são conhecidos e representados por seus valores nominais. A incerteza em problemas de otimização é comumente abordada por meio de três abordagens: (i) otimização robusta, em que a incerteza pode ser representada por um conjunto finito de cenários ou modelada com base em intervalos predefinidos nos quais os valores dos parâmetros incertos podem estar, (ii) otimização estocástica, na qual os parâmetros incertos seguem distribuições de probabilidade conhecidas e (iii) otimização *fuzzy*, que representa o parâmetro incerto como um número *fuzzy*, ou seja, a incerteza é representada por uma função de pertinência, de modo que a incerteza vem da imprecisão subjetiva ou da falta de informações exatas.

A abordagem utilizada neste trabalho é a otimização robusta de [Bertsimas & Sim \(2003\)](#), que não considera que todos os valores dos parâmetros incertos serão iguais ao pior caso (como ocorre com [Soyster \(1973\)](#)). Mais precisamente, considere o problema nominal:

$$\min \sum_{j=1}^n c_j x_j \quad (2.1)$$

$$\text{s. a: } \sum_{j=1}^n a_{ij} x_j \geq b_i \quad \forall i = 1, 2, \dots, m \quad (2.2)$$

$$x_j \geq 0 \quad (2.3)$$

Seja J_i o conjunto dos valores incertos da restrição i e $\tilde{a}_{ij} \in [a_{ij} - \hat{a}_{ij}, a_{ij} + \hat{a}_{ij}]$, o parâmetro incerto para cada $j \in J_i$. Considere também Γ_i um parâmetro que controla e determina a variação total máxima dos valores dos parâmetros incertos, de forma que $0 \leq \Gamma_i \leq |J_i|$, em que $|J_i|$ é a cardinalidade de J_i . Quando $\Gamma_i = 0$, temos o problema nominal; quando $\Gamma_i = |J_i|$, temos o problema de Soyster (1973); e quando $0 < \Gamma_i < |J_i|$, é assumido que o valor de $\lfloor \Gamma_i \rfloor$ parâmetros corresponderá ao pior caso e o valor de um parâmetro a_{it} , para $t \in J_i$, irá variar $(\Gamma_i - \lfloor \Gamma_i \rfloor)\hat{a}_{it}$ do seu valor nominal. Aqui, $\lfloor \Gamma_i \rfloor$ é a parte inteira de Γ_i .

O conjunto de incertezas associado aos parâmetros incertos na restrição i de (2.2) é

$$\mathcal{U}_i(\Gamma_i) = \left\{ \tilde{a}_{ij} \in \mathbb{R}^n : \tilde{a}_{ij} = a_{ij} + s_{ij}\hat{a}_{ij} \text{ para } s_{ij} \in [-1, 1], \sum_{j \in J_i} |s_{ij}| \leq \Gamma_i, \forall j \in J_i \right\} \quad (2.4)$$

A contraparte robusta associada ao problema nominal (2.1), (2.2) e (2.3) é dada por (2.1), (2.3) e

$$\sum_{j=1}^n a_{ij}x_j + \min_{S_i \cup \{t_i\} \subseteq J_i: |S_i| = \lfloor \Gamma_i \rfloor} \left\{ \sum_{j \in S_i} -\hat{a}_{ij}x_j - (\Gamma_i - \lfloor \Gamma_i \rfloor)\hat{a}_{it_i}x_j \right\} \geq b_i, \quad \forall i \quad (2.5)$$

O Exemplo 1 ilustra o funcionamento da restrição (2.5) em um exemplo com cinco variáveis de decisão.

Exemplo 1. Considere o seguinte problema nominal de otimização linear:

$$\min \quad 5x_1 + 10x_2 + 8x_3 + 12x_4 + 4x_5 \quad (2.6)$$

$$s. \ a: \quad 10x_1 + 15x_2 + 20x_3 + 25x_4 + 30x_5 \geq 100 \quad (2.7)$$

$$x_j \geq 0, \quad j = 1, 2, \dots, 5 \quad (2.8)$$

Considere incertos os valores de coeficientes das variáveis presentes na restrição (2.7) (observe que $|J_i| = 5$) e uma variação de 5% sobre seus valores nominais. Assumindo que $\Gamma = 20\%$ de $|J_i|$, temos que $\Gamma = 1$. Assim, apenas um dos valores estará no pior caso. Se a solução corrente for $x_1 = x_2 = x_3 = x_4 = x_5 = 1$, o valor $30x_5 = 30$ é o maior valor incerto e 5% de 30 é igual a 1.5 e dessa forma a solução apresentada (antes factível), agora se torna infactível, visto que $10 + 15 + 20 + 25 + (30 - 1.5) = 98.5 < 100$.

Por outro lado, para $\Gamma = 1$ e solução corrente $x_1 = x_5 = 1$, $x_2 = x_4 = 2$ e $x_3 = 3$,

o valor $20x_3 = 60$ será o maior e como 5% de 20 é igual a 1, o valor da expressão (2.7) para essa solução será $10(1) + 15(2) + 19(3) + 25(2) + 30(1) = 177$.

Para finalizar o exemplo numérico, considere $\Gamma = 100\%$ de $|J_i|$ ($\Gamma = 5$, temos o problema de Soyster (1973)) e uma variação de 10% sobre os valores nominais dos parâmetros presentes na restrição (2.7). Se a solução corrente for $x_1 = x_5 = 1$, $x_2 = x_4 = 2$ e $x_3 = 3$, o valor da expressão (2.7) será $9(1) + 13,5(2) + 18(3) + 22,5(2) + 27(1) = 162$.

Perceba que o aumento de Γ , assim como o aumento da variação dos valores nominais, tende a dificultar a factibilidade de uma solução. $\diamond$

A minimização interna na restrição (2.5) torna o problema composto pelas expressões (2.1), (2.3) e (2.5) não linear, mas a minimização interna pode ser dada por

$$\nu_i(x) = \min_{S_i \cup \{t_i\} \subseteq J_i: |S_i| = \lfloor \Gamma_i \rfloor} \left\{ \sum_{j \in S_i} -\hat{a}_{ij}x_j - (\Gamma_i - \lfloor \Gamma_i \rfloor)\hat{a}_{it_i}x_j \right\},$$

que pode ser representada da seguinte forma para cada $i = 1, 2, \dots, m$

$$\nu_i(x) = \min \sum_{j \in J_i} -\hat{a}_{ij}x_j s_{ij} \tag{2.9}$$

$$\text{s. a: } \sum_{j \in J_i} s_{ij} \leq \Gamma_i \tag{2.10}$$

$$0 \leq s_{ij} \leq 1 \quad \forall j \in J_i \tag{2.11}$$

Este problema possui a seguinte forma canônica para problemas de minimização

$$\nu_i(x) = \min \sum_{j \in J_i} -\hat{a}_{ij}x_j s_{ij} \tag{2.12}$$

$$\text{s. a: } \sum_{j \in J_i} -s_{ij} \geq -\Gamma_i \tag{2.13}$$

$$-s_{ij} \geq -1 \quad \forall j \in J_i \tag{2.14}$$

$$s_{ij} \geq 0 \quad \forall j \in J_i. \tag{2.15}$$

O dual do problema descrito por (2.12)-(2.15) é dado por

$$\beta(x) = \max \quad -\Gamma_i \phi_i - \sum_{j \in J_i} \rho_{ij} \tag{2.16}$$

$$\text{s. a: } -\phi_i - \rho_{ij} \leq -\hat{a}_{ij}x_j \quad \forall j \in J_i \quad (2.17)$$

$$\phi_i \geq 0 \quad (2.18)$$

$$\rho_{ij} \geq 0 \quad \forall j \in J_i. \quad (2.19)$$

Pela teoria da dualidade, a função objetivo do problema dual em conjunto com suas restrições podem ser usadas para substituir a minimização interna da restrição (2.5). Observe que se esta restrição é atendida para os valores ótimos das variáveis duais, então ela será atendida para qualquer valor viável para as variáveis duais. Desta forma, é possível obter um modelo robusto linearizado sem a inclusão da função de maximização.

Assim, o modelo robusto linearizado pode ser dado por

$$\min \sum_{j=1}^n c_j x_j \quad (2.20)$$

$$\text{s. a: } \sum_{j=1}^n a_{ij}x_j - \Gamma_i \phi_i - \sum_{j \in J_i} \rho_{ij} \geq b_i, \quad \forall i \quad (2.21)$$

$$\phi_i + \rho_{ij} \geq \hat{a}_{ij}x_j \quad \forall j \in J_i \quad (2.22)$$

$$x_j \geq 0 \quad \forall j = 1, \dots, n \quad (2.23)$$

$$\rho_{ij} \geq 0 \quad \forall i = 1, \dots, m, j \in J_i \quad (2.24)$$

$$\phi_i \geq 0 \quad \forall i = 1, \dots, m \quad (2.25)$$

Essa formulação incorpora a incerteza de forma controlada no modelo. O parâmetro Γ determina o nível de proteção desejado e o modelo resultante mantém estrutura linear, mantendo a tratabilidade computacional.

Capítulo 3

Problema de cobertura máxima de *hub* com alocação única, não capacitado

Este capítulo aborda uma definição do USAHMCP, uma formulação para ele e uma comparação do problema com o USA p HMCP (Seção 3.1). Além disso, apresenta a metodologia heurística utilizada para solução do USAHMCP (Seção 3.2) e detalhes e experimentos computacionais preliminares (Seção 3.3), bem como os resultados computacionais usando as instâncias AP (Seção 3.4), CAB (Seção 3.5), DS3 (Seção 3.6) e URAND de 1000 nós (Seção 3.7). Por fim, uma conclusão do capítulo é apresentada na Seção 3.8.

3.1 Formulação e definição do problema USAHMCP

Esta seção aborda o problema de cobertura máxima de *hub* de alocação única, não capacitado e com custos fixos (USAHMCP). Este problema visa determinar quais nós serão *hubs* e como nós não *hubs* serão alocados aos *hubs* para maximizar a demanda coberta. Ele assume que:

- (i) não há limite de capacidade nos *hubs* ou nos arcos da rede;
- (ii) cada nó deve ser alocado a um único *hub*;
- (iii) há um custo de abertura de cada *hub*;
- (iv) o custo total de abertura dos *hubs* não deve exceder o orçamento definido para esta finalidade;

- (v) há um fator de economia de escala constante, que representa o desconto no transporte em conexões entre dois *hubs*;
- (vi) as redes de *hubs* são completas, ou seja, há uma conexão direta entre todos os pares de *hubs*;
- (vii) conexões diretas entre nós não *hubs* não são permitidas;
- (viii) uma demanda é considerada coberta se o custo de transporte da origem ao destino for menor que um limite pré-estabelecido.

A formulação para o USAHMCP usa a seguinte notação. Seja $G = (N, A)$ um grafo em que N representa o conjunto de nós (sendo $n = |N|$), que também representa o conjunto de possíveis *hubs*, e A é o conjunto de arcos. Seja W_{ij} o fluxo originado no nó $i \in N$ que deve ser enviado para o nó de destino $j \in N$. Além disso, f_k representa o custo fixo de abertura de um *hub* no nó $k \in N$ e b representa o orçamento total disponível para instalar *hubs*.

Seja d_{ij} o comprimento do arco $(i, j) \in A$, que representa a distância entre os nós i e j . Seja também:

- (i) α : fator de desconto no custo de transporte por meio de conexões entre *hubs*;
- (ii) χ : fator de coleta; e
- (iii) δ : fator de distribuição, sendo $\alpha < \chi$ e $\alpha < \delta$.

O custo unitário de transporte de um nó de origem i para o nó de destino j , passando pelos *hubs* k e m , é dado por:

$$c_{ij}^{km} = \chi d_{ik} + \alpha d_{km} + \delta d_{mj}.$$

Seja β_{ij} o limiar de cobertura do par de origem e destino $i, j \in N$, no qual a demanda de i para j é coberta se o custo total da rota que conecta esses dois nós for menor ou igual a β_{ij} . Além disso, a_{ij}^{km} é um parâmetro de cobertura, que indica se a demanda de i para j passando por *hubs* nos nós k e m é coberta ou não. O parâmetro de cobertura a_{ij}^{km} é igual a 1 se $c_{ij}^{km} \leq \beta_{ij}$, e igual a zero, caso contrário. O parâmetro auxiliar de limite superior

$$\lambda_{ij} = \max_{km} \{a_{ij}^{km}, \quad \forall i, j \in N\}$$

também será usado na formulação.

Esta formulação usa as seguintes variáveis de decisão. A variável binária x_{ik} é igual a 1 se o nó i for alocado ao *hub* k e 0, caso contrário. Além disso, x_{ii} é 1 se um

hub for instalado no nó i e 0, caso contrário. A variável não negativa z_{ij} representa a fração coberta da demanda originada no nó i e destinada ao nó de destino j .

A formulação para o USAHMCP, baseada em [Peker & Kara \(2015\)](#), é dada por:

$$\max \sum_{i \in N} \sum_{j \in N} W_{ij} z_{ij} \quad (3.1)$$

$$\text{s. a: } \sum_{k \in N} f_k x_{kk} \leq b \quad (3.2)$$

$$\sum_{k \in N} x_{ik} = 1 \quad \forall i \in N \quad (3.3)$$

$$x_{ik} \leq x_{kk} \quad \forall i, k \in N \quad (3.4)$$

$$z_{ij} \leq \sum_{k \in N} a_{ij}^{km} x_{ik} + \lambda_{ij}(1 - x_{jm}), \quad \forall i, j, m \in N \quad (3.5)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \quad (3.6)$$

$$z_{ij} \geq 0 \quad \forall i, j \in N. \quad (3.7)$$

A função objetivo (3.1), a ser maximizada, representa a demanda total coberta, considerando todos os pares de origem e destino. A restrição (3.2) garante que o custo total de instalação de *hubs* não excederá o orçamento disponível. As restrições (3.3) impõem que cada nó seja alocado a exatamente um *hub*. As restrições (3.4) garantem que os nós podem ser alocados apenas a *hubs* abertos. As restrições (3.5) determinam o limite superior para a fração de fluxo entre os nós de origem e destino que é coberta. Note que $z_{ij} = 1$ se houver um caminho $i - k - m - j$ com custo total menor ou igual a β_{ij} ($a_{ij}^{km} = 1$ e, consequentemente, $\lambda_{ij} = 1$) e se o nó i for alocado ao *hub* k ($x_{ik} = 1$) e j for alocado ao *hub* m ($x_{jm} = 1$); caso contrário, $z_{ij} = 0$. Por fim, as expressões (3.6) e (3.7) apresentam o domínio das respectivas variáveis de decisão. A variável z_{ij} é definida como não negativa, mas, como mencionado anteriormente, seu valor será sempre 0 ou 1.

3.1.1 Comparação entre o USA p HMCP e o USAHMCP

O USA p HMCP se diferencia do USAHMCP (Seção 3.1) por fixar previamente o número de *hubs*, enquanto que o USAHMCP determina essa quantidade por meio de uma restrição orçamentária. O modelo matemático do USA p HMCP proposto por [Peker & Kara \(2015\)](#) é formado pelas expressões (3.1), (3.3)-(3.7) e $\sum_{k \in N} x_{kk} = p$ que determina a abertura de exatamente p *hubs*. Uma ilustração da diferença entre os dois problemas é dada por meio de um pequeno exemplo com uma instância adaptada do conjunto de dados *Australian Post* (AP) com cinco nós. Considere os seguintes dados da instância: $n = 5$, $\chi = 1$, $\alpha = 0,75$, $\delta = 1$, a matriz de fluxo de demanda

entre os pares de nós de origem e destino dada por:

$$W_{ij} = \begin{pmatrix} 17,7 & 12,5 & 7,9 & 13,2 & 6,4 \\ 15,7 & 11,7 & 8,4 & 15,4 & 7,3 \\ 7,3 & 6,9 & 7,0 & 15,2 & 7,1 \\ 23,7 & 24,3 & 26,1 & 67,4 & 35,3 \\ 7,1 & 7,6 & 8,6 & 24,3 & 13,8 \end{pmatrix}$$

e as coordenadas dos nós: nó 1:(29,4; 17,8), nó 2:(29,3; 31,8), nó 3:(32,0; 38,7), nó 4:(30,3; 46,3), nó 5:(28,1; 50,6). Usando as coordenadas dos nós, a matriz Euclidiana com a distância entre todos os pares de nós é dada por:

$$d_{ij} = \begin{pmatrix} 0 & 14 & 21 & 28,5 & 32,8 \\ 14 & 0 & 7,4 & 14,5 & 18,8 \\ 21 & 7,4 & 0 & 7,8 & 12,5 \\ 28,5 & 14,5 & 7,8 & 0 & 4,8 \\ 32,8 & 18,8 & 12,5 & 4,8 & 0 \end{pmatrix}$$

Considere também o valor de $\beta_{ij} = \beta$ sendo 26,4 e que os cinco nós da rede são nós de demanda e também são nós candidatos para serem instalados como *hubs*.

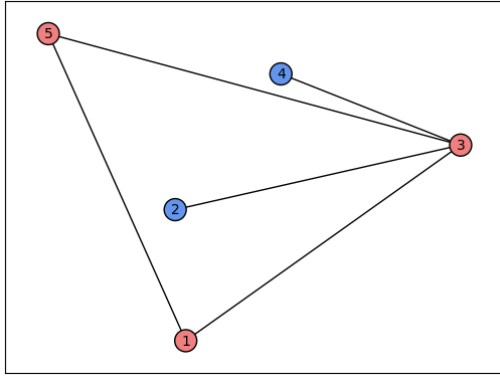
Para o USApHMCP, assume-se que devem ser instalados três *hubs*. Para o USAHMCP, têm-se os custos de abertura de *hub* sendo $f_1 = 29$, $f_2 = 28$, $f_3 = 30$, $f_4 = 24$ e $f_5 = 26$ e orçamento fixado em 82,2 (que corresponde a três vezes o valor da média dos custos de instalação de *hub*). Com esses dados, a cobertura máxima é dada por 397,9 para os dois problemas.

A solução ótima dessa instância do USAHMCP abre os *hubs* 1, 2 e 4 ao custo de 81. Já a solução ótima para o USApHMCP abre os *hubs* 1, 3 e 5. Note que a configuração de *hubs* abertos no USApHMCP não é uma solução factível para o USAHMCP, já que seu custo de abertura, igual a 85, ultrapassa o orçamento de 82,2. Vale ressaltar que, neste caso, ambos os problemas conseguiram cobertura máxima.

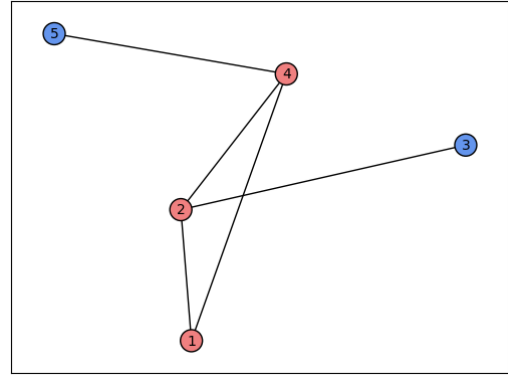
Note também que não é possível abrir *hubs* nos nós 1, 2 e 3 para o USAHMCP, porque, neste caso, o custo total de instalação também excederia o orçamento disponível. Portanto, o orçamento e os custos fixos de instalação não especificam diretamente que uma solução viável deve corresponder a um número predefinido de *hubs*. Desta forma, o conjunto de soluções viáveis para o USAHMCP e o USApHMCP pode ser completamente diferente.

As Figuras 3.1a e 3.1b ilustram as redes *hub-and-spoke* que representam as soluções ótimas para a instância de teste para o USApHMCP e o USAHMCP. Os *hubs* são representados por círculos vermelhos, enquanto os nós não *hubs* são representados

por círculos azuis.



(a) Rede para USApHMCP.



(b) Rede para USAHMCP.

Figura 3.1: Ilustração da rede para a solução ótima para instância AP com 5 nós para o USApHMCP e para o USAHMCP.

Pode-se notar que as redes, na Figura 3.1, representando as soluções do USApHMCP e do USAHMCP, apesar de terem o mesmo número de *hubs*, têm *hubs* em nós diferentes. Observe que os nós 2 e 4 são *hubs* na Figura 3.1b, mas não são *hubs* na Figura 3.1a. Porém, é importante ressaltar que, apesar disso, as duas redes resultam no valor de cobertura máxima igual a 397,9. É importante lembrar que a solução da rede mostrada na Figura 3.1b possui custo total de abertura de *hub* menor que a outra rede. Outro ponto interessante a destacar é que os nós não *hubs* nem sempre são alocados ao *hub* mais próximo. Por exemplo, o nó 2, na Figura 3.1a, é alocado ao *hub* 3, em vez de ser alocado ao *hub* 1, que é o *hub* mais próximo.

A Proposição 3.1.1 a seguir mostra que o USApHMCP é um caso particular do USAHMCP. Além disso, como o USApHMCP é um problema NP-difícil reconhecido, o USAHMCP também é um problema NP-difícil.

Proposição 3.1.1. *O USAHMCP, modelado pelas expressões (3.1)-(3.7), é NP-difícil e tem como caso particular o USApHMCP.*

Demonstração. Considere um caso particular do problema USAHMCP em que $f_k = 1$, para todo $k \in N$ e que $b = p$. Portanto, neste caso, a expressão (3.2) é equivalente a

$$\sum_{k \in N} x_{kk} \leq p. \quad (3.8)$$

Suponha que a solução ótima do USAHMCP abra $\underline{p}$ *hubs*, isto é, $\sum_{k \in N} x_{kk} = \underline{p}$, sendo $\underline{p} \leq p$. Neste caso, é possível atingir uma cobertura máxima com $\underline{p}$ *hubs*. Se $\underline{p} < p$, então qualquer solução que abra $(p - \underline{p})$ novos *hubs* também será uma solução ótima para o USAHMCP. Observe que esta solução com $(p - \underline{p})$ *hubs* adicionais

satisfaz à condição:

$$\sum_{k \in N} x_{kk} = p, \quad (3.9)$$

Portanto, uma solução ótima para o USA p HMCP, modelada pelas expressões (3.1), (3.3)-(3.7) e (3.9) em [Peker & Kara \(2015\)](#), pode ser encontrada através deste caso particular do USAHMCP.

Finalmente, suponha por contradição que o USAHMCP não é NP-difícil, então seu caso particular (USA p HMCP) também não seria NP-difícil. Isso é uma contradição, pois o USA p HMCP foi provado ser NP-difícil por [Peker & Kara \(2015\)](#). Portanto, o USAHMCP também é NP-difícil. $\square$

Conforme demonstrado, o USA p HMCP é uma caso particular do USAHMCP, assim, algoritmos implementados para o USAHMCP podem ser utilizados para resolver o USA p HMCP.

3.2 Algoritmo ILS-VND proposto para o USAHMCP

Esta seção descreve o algoritmo proposto para resolver instâncias do USAHMCP. Este algoritmo é baseado na meta-heurística *Iterated Local Search* (ILS), que aplica iterativamente uma busca local seguida por perturbações, para evitar que a busca fique presa em ótimos locais. Para executar a busca local, o algoritmo emprega o método *Variable Neighborhood Descent* (VND). Além disso, são apresentadas duas variantes do algoritmo, que diferem apenas na estratégia de alocação dos nós não *hub* a nós *hub*.

A seção está organizada da seguinte forma. A Seção 3.2.1 descreve como uma solução é representada. Em seguida, a função de avaliação é apresentada na Seção 3.2.2. A Seção 3.2.3 apresenta o procedimento para construir uma solução inicial. A Seção 3.2.4 detalha as estruturas de vizinhança usadas na busca local e a Seção 3.2.5 detalha os procedimentos de alocação gulosa usados em três estruturas de vizinhança. A Seção 3.2.6 apresenta o procedimento de busca local com base em VND. Por fim, a Seção 3.2.7 apresenta o algoritmo ILS-VND.

3.2.1 Representação da solução

Uma solução s é representada por uma lista com três elementos, $s = (\bar{p}, y, h)$, contendo duas listas, y e h , e o escalar $\bar{p}$. A lista $y = (y_1, \dots, y_n)$ possui n elementos inteiros, em que y_i indica a qual *hub* o nó i está alocado. A lista h é uma lista de inteiros de tamanho n , que contém em suas primeiras $\bar{p}$ posições os *hubs* instalados e nas posições de $\bar{p} + 1$ até n , os nós não *hubs*, em que $h = (h_1, \dots, h_{\bar{p}}, \dots, h_n)$. A lista

h com n elementos facilita a identificação de quais nós são *hubs* e quais nós não são *hubs*, o que é bastante útil para o procedimento de busca local usado.

Por exemplo, se o conjunto de nós N for tal que $N = \{1, 2, 3, 4, 5\}$, uma solução possível poderia ser dada por $\bar{p} = 2$ e as listas $h = (3, 5, 1, 2, 4)$ e $y = (5, 3, 3, 5, 5)$. Então, as duas primeiras posições de h mostram os *hubs* 3 e 5 abertos, e a lista y indica que os nós 1, 4 e 5 são alocados ao *hub* 5 e os nós 2 e 3 são alocados ao *hub* 3.

3.2.2 Função de avaliação

Como a representação da solução não proíbe soluções inviáveis, a avaliação da solução pode ser usada para evitar soluções nas quais o custo total de instalação de *hubs* excede o orçamento. Assim, quando a solução é viável, a função de avaliação retorna o valor da função objetivo, e caso a solução seja inviável, a função de avaliação retorna o valor 0. Portanto, dada uma solução $s = (\bar{p}, h, y)$, a função de avaliação é:

$$f(s) = \begin{cases} \sum_{i \in N} \sum_{j \in N} W_{ij} a_{ij}^{y_i y_j}, & \text{se } \sum_{i=1}^{\bar{p}} f_{h_i} \leq b \\ 0 & \text{caso contrário} \end{cases}$$

Observe que a ordem de complexidade da função de avaliação é $\mathcal{O}(n^2)$.

3.2.3 Construção da solução inicial

A solução inicial é obtida por um procedimento guloso que seleciona um único nó para se tornar um *hub* com base na razão entre fluxo e custo fixo, em que o fluxo associado a um nó é dado pela soma das demandas nele originadas e das demandas destinadas a esse nó. Assim, para cada nó $i \in N$, calcula-se a razão: r_i , dada por:

$$r_i = \frac{\sum_{j \in N} (W_{ij} + W_{ji})}{f_i}$$

Após calcular essas razões, os nós em N são classificados em ordem decrescente de valores de r_i e colocados na lista h . Essa ordenação é usada como uma ordem de exploração em algumas partes do algoritmo. O primeiro elemento na lista será o único *hub* da solução, ou seja, $\bar{p} = 1$. Finalmente, todos os nós são alocados a esse único *hub* instalado.

3.2.4 Estruturas de vizinhança

O espaço de solução do USAHMCP é explorado por meio de quatro tipos de movimentos aplicados à solução $s = (\bar{p}, y, h)$. Cada movimento define uma vizinhança específica, composta pelas soluções que podem ser obtidas ao se aplicar essa modificação à solução atual. Cada estrutura de vizinhança é detalhada a seguir:

- (i) Vizinhança de abertura de *hub* ($\mathcal{N}_1$): O conjunto de soluções que podem ser obtidas adicionando um novo *hub* à solução corrente s . Portanto, os vizinhos de uma solução com $\bar{p}$ *hubs* têm $\bar{p} + 1$ *hubs*. Além disso, essa estrutura de vizinhança assume que uma alocação gulosa é feita após a abertura de um novo *hub*, resultando em uma nova solução. Os procedimentos de alocação gulosa considerados neste capítulo são detalhados na Seção 3.2.5. Observe que uma solução tem $n - \bar{p}$ vizinhos nesta vizinhança;
- (ii) Vizinhança de realocação ($\mathcal{N}_2$): O conjunto de soluções que podem ser obtidas modificando a alocação de um nó não *hub*. Assim, um nó não *hub* alocado a um *hub* é alocado a outro *hub* aberto. O número de vizinhos gerados dessa vizinhança é igual a $(n - \bar{p})(\bar{p} - 1)$, visto que não há necessidade de realocar nós *hubs*;
- (iii) Vizinhança de substituição de *hub* ($\mathcal{N}_3$): Conjunto de soluções que podem ser obtidas pelo fechamento de um *hub* aberto e pela abertura de um *hub* fechado. A implementação dessa estrutura de vizinhança assume que, após a substituição, uma alocação gulosa de todos os nós não *hub* a *hubs* abertos é realizada. O número de vizinhos gerados a partir dessa vizinhança é igual a $\bar{p}(n - \bar{p})$;
- (iv) Vizinhança de substituição de *hub* com uma busca local de realocação ($\mathcal{N}_4$): Conjunto de soluções que podem ser obtidas fechando-se um *hub* aberto anteriormente e abrindo-se um *hub* que foi fechado. Esta estrutura de vizinhança é uma combinação das estruturas $\mathcal{N}_3$ e $\mathcal{N}_2$. Ela representa o conjunto de soluções que podem ser obtidas através de:
 - (a) fechamento de um *hub* aberto e abertura de um *hub* fechado;
 - (b) aplicação de uma alocação gulosa na solução parcial obtida na etapa (a);
 - (c) aplicação de uma busca local usando a estrutura de vizinhança $\mathcal{N}_2$ na solução resultante da etapa (b).

A complexidade desta estrutura é maior que as demais, devido à alocação gulosa e à busca local, que aumentam sua ordem de complexidade.

Note que as estruturas de vizinhança estão apresentadas de acordo com o número de vizinhos, o que não é tão simples, pois algumas vizinhanças dependem do número de *hubs* abertos. O número de vizinhos na vizinhança de substituição de *hub* pode ser maior que na vizinhança de realocação (solução com 2 *hubs* para instâncias de 10 nós), mas, em outros casos, ocorre o oposto (solução com 4 *hubs* para instâncias de 10 nós). Assumindo que n tende a ser maior que $\bar{p}^2$, o ordenamento das vizinhanças do menor para o maior número de vizinhos seria: $\mathcal{N}_1$, $\mathcal{N}_2$, $\mathcal{N}_3$ e $\mathcal{N}_4$.

É importante destacar que não é necessário considerar nenhuma estrutura de vizinhança baseada apenas no fechamento de *hubs*, pois o fechamento de um *hub* nunca resulta em uma melhoria na cobertura.

3.2.5 Procedimentos de alocação gulosa

Conforme mencionado na seção anterior, ao executar um movimento relacionado às estruturas de vizinhança $\mathcal{N}_1$, $\mathcal{N}_3$ e $\mathcal{N}_4$, é necessário atualizar a solução com uma nova alocação de todos os nós não *hub* ao novo conjunto de *hubs*. Foram considerados dois tipos de alocação gulosa:

- (i) alocação com base no *hub* mais próximo;
- (ii) alocação com base no potencial de cobertura.

Essas variações de alocação levam a duas variantes diferentes da heurística, descritas a seguir:

- (i) A primeira alocação gulosa aloca um nó não *hub* ao *hub* aberto mais próximo. Seja $H = \{h_1, h_2, \dots, h_{\bar{p}}\}$ o conjunto de *hubs* abertos. Assim, o nó i será alocado ao *hub* $k^* \in H$, em que:

$$k^* = \arg \min_{k \in H} \{d_{ik}\}.$$

- (ii) O segundo tipo de alocação gulosa realiza alocações com base no potencial de cobertura. Neste caso, o nó i será alocado ao *hub* k^* que poderia resultar na melhor cobertura possível, caso todos os outros nós fossem alocados aos *hubs* mais promissores.

Assim, seja N_{ik} o conjunto de nós de destino para os quais existe um caminho com origem em i , passando por k (com i alocado a k) e por no máximo outro *hub* m cujo custo total não exceda o limiar de cobertura. Este conjunto é dado por:

$$N_{ik} = \{j \in N : \exists m \in H, \text{ para } a_{ij}^{km} = 1\}.$$

O potencial de cobertura P_{ik} da alocação de i ao *hub* k é dado por:

$$P_{ik} = \sum_{j \in N_{ik}} (W_{ij} + W_{ji}).$$

Desta forma, a alocação gulosa baseada no potencial de cobertura aloca o nó não *hub* i ao *hub* k^* com maior potencial de cobertura, dado por:

$$k^* = \arg \max_{k \in H} \{P_{ik}\}.$$

Esta estratégia de alocação tem uma complexidade $O(n^2\bar{p}^2)$ no pior caso, uma vez que, no laço principal, para cada nó não-*hub* i e para cada *hub* k , para cada nó não-*hub* j e para cada *hub* m , verifica-se se o caminho de i para j passando pelos *hubs* k e m está coberto ou não.

É importante destacar que a alocação baseada no *hub* mais próximo é a estratégia de alocação mais aplicada para o USApHMCP, conforme apontam [Calik et al. \(2009\)](#); [Hwang & Lee \(2012\)](#); [Silva & Cunha \(2017\)](#). Além disso, essa estratégia de alocação requer menor esforço computacional que a alocação baseada no potencial de cobertura, que requer o cálculo de P_{ik} sempre que uma alocação gulosa é realizada. No entanto, a alocação baseada no potencial de cobertura tem a vantagem de considerar diretamente as informações de possíveis coberturas na decisão de alocação, o que pode resultar em alocações com maior cobertura que o outro tipo de alocação. Segundo o nosso conhecimento, este é o primeiro trabalho que considera essa estratégia de alocação gulosa.

3.2.6 Procedimento de busca local para o USAHMCP

O procedimento de busca local é baseado no método *Variable Neighborhood Descent* (VND), proposto por [Mladenović & Hansen \(1997\)](#). Como descrito na Seção 2.2.2, o VND é um algoritmo de descida baseado em vizinhança, que explora várias vizinhanças em vez de apenas uma.

O algoritmo proposto usa o VND cíclico, descrito em [Hansen et al. \(2019\)](#), para executar a busca local. Este procedimento começa com uma solução inicial, como a solução corrente, e uma dada estrutura de vizinhança. Então, iterativamente, o melhor vizinho da solução corrente na vizinhança corrente é identificado. Se o melhor vizinho for melhor que a solução corrente, o melhor vizinho se torna a solução corrente. Então, o algoritmo passa para a próxima estrutura de vizinhança, independentemente de ter havido uma melhora na solução corrente.

O Algoritmo 3 apresenta um pseudocódigo do procedimento de busca local. Este algoritmo usa as quatro estruturas de vizinhança apresentadas na Seção 3.2.4 e recebe como entrada uma determinada solução. Então, inicializa-se o valor de k , que representa o índice para a estrutura de vizinhança corrente, e de $count$, que representa um contador para o número de estruturas de vizinhança exploradas sem melhoria. Na linha 5, o *loop* principal é executado enquanto $count$ não for maior que o número de vizinhanças, ou seja, enquanto a solução corrente não for um ótimo local em relação às quatro estruturas de vizinhança. O *loop* principal consiste em encontrar o melhor vizinho da solução corrente na vizinhança $\mathcal{N}_k$ (linha 6). A partir da linha 7, se o melhor vizinho tiver um valor de função objetivo melhor que a solução corrente, então a solução corrente é atualizada, redefinindo o valor de $count$ para zero. Se não houver melhoria na execução da linha 7, então $count$ é incrementado. O algoritmo passa para a próxima vizinhança (linha 13) independentemente de melhoria.

Algoritmo 3 VND Cíclico para o USAHMCP

```

1: Entrada: Solução  $s$ 
2: Saída: Solução  $s$ 
3:  $k \leftarrow 1$ 
4:  $count \leftarrow 0$ 
5: while  $count < |\mathcal{N}|$  do
6:    $\bar{s} \leftarrow$  melhor vizinho de  $s$  na vizinhança  $\mathcal{N}_k$ 
7:   if  $f(\bar{s}) > f(s)$  then
8:      $s \leftarrow \bar{s}$ 
9:      $count \leftarrow 0$ 
10:  else
11:     $count \leftarrow count + 1$ 
12:  end if
13:   $k \leftarrow (k \bmod 4) + 1$ 
14: end while
15: Retorne  $s$ .
```

3.2.7 Algoritmo ILS-VND para solução do USAHMCP

Conforme mencionado na Seção 2.2.3, o algoritmo ILS combina intensificação com diversificação. Neste capítulo, a etapa de intensificação é realizada pelo VND (Seção 3.2.6) e a diversificação será detalhada no próximo parágrafo. Mais precisamente, o algoritmo ILS utilizado neste capítulo funciona da seguinte forma. Uma solução obtida a partir de uma busca local é perturbada, gerando uma nova solução. Em seguida, a solução perturbada passa novamente por um procedimento de busca local. Se este novo ótimo local for melhor que a solução corrente, então ele se torna a nova solução corrente. Caso contrário, ocorre um aumento no nível de perturbação. O

Algoritmo 4 Procedimento de perturbação para o USAHMCP

-
- 1: Entrada: solução s , nível de perturbação i
 - 2: Saída: solução perturbada s'
 - 3: $k \leftarrow i + 1$;
 - 4: Selecionar aleatoriamente k *hubs* da solução s ;
 - 5: Selecionar aleatoriamente k nós não *hubs* da solução s ;
 - 6: Substituir os nós *hubs* do passo 4 pelos nós escolhidos no passo 5;
 - 7: $s' \leftarrow$ alocação gulosa de todos os nós para os novos *hubs*;
 - 8: Retorne s'
-

parâmetro ILSMAX é responsável pelo critério de parada, limitando o aumento do nível, de forma a não acontecer um reinício aleatório.

O algoritmo proposto ILS-VND usa substituições aleatórias de *hubs* abertos por nós não *hub* para executar a perturbação, seguido de uma alocação gulosa. O número de substituições é definido pelo nível de perturbação, em que $\text{perturbação}(s, \text{nível})$ retorna a solução resultante de uma perturbação de nível i na solução s , que é dada pela troca aleatória de $i + 1$ *hubs* por $i + 1$ nós não *hubs*.

Por exemplo, se $N = \{1, 2, 3, 4, 5, 6\}$, $\bar{p} = 3$ e $h = (2, 4, 5, 1, 3, 6)$, uma perturbação de nível 1 poderia resultar em $h_1 = (1, 3, 5, 2, 4, 6)$ com a substituição dos *hubs* 2 e 4 pelos nós não *hub* 1 e 3. Uma perturbação de nível 2 poderia resultar em $h_2 = (1, 3, 6, 2, 4, 5)$ com a substituição de 3 *hubs*. Em todos os casos, o número de *hubs* permanece o mesmo. Perceba que não é possível realizar uma perturbação de nível 3, já que implicaria em uma troca de quatro *hubs*, mas $\bar{p} = 3$. O procedimento de perturbação está descrito no Algoritmo 4.

O Algoritmo 5 descreve o pseudocódigo do ILS usado. Este algoritmo tem, como entrada, o parâmetro ILSMAX, que representa o número máximo de iterações. Além disso, o contador dado por iter é usado para contar o número de iterações do ILS. Então, enquanto o critério de parada não é atingido (linha 7), o algoritmo realiza uma perturbação (linha 8) na solução corrente, seguida de uma busca local sobre essa solução perturbada na linha 9. A partir da linha 10, se a solução obtida após aplicar a busca local à solução perturbada for melhor que a melhor solução encontrada até então, a melhor solução encontrada é atualizada e o nível de perturbação retorna ao valor inicial. Caso contrário (a partir da linha 14), o nível é aumentado em uma unidade. Além disso, a cada iteração, o contador de iterações é incrementado em uma unidade. Para definir os critérios de parada, o número máximo de iterações (dado por ILSMAX) e também o número de *hubs* da solução corrente (dado por $\bar{p}$) são utilizados para evitar que o número de substituições resulte em uma reinicialização aleatória ou que o número de substituições seja maior que a quantidade corrente de *hubs*, por exemplo, se a solução corrente possui $\bar{p} = 3$, então não é possível realizar a

Algoritmo 5 ILS-VND para o USAHMCP

```

1: Entrada: ILSMAX
2: Saída:  $s$ 
3:  $s_0 \leftarrow$  Solução inicial gulosa; ▷ Seção 3.2.3
4:  $s \leftarrow VND(s_0)$ ; ▷ Algoritmo 3
5:  $iter \leftarrow 1$ ;
6:  $nivel \leftarrow 1$ ;
7: while  $iter \leq ILSMAX$  e  $nivel \leq \bar{p} - 2$  do
8:    $s_1 \leftarrow perturbação(s, nivel)$ ; ▷ Algoritmo 4
9:    $s_2 \leftarrow VND(s_1)$ ; ▷ Algoritmo 3
10:  if  $f(s_2) > f(s)$  then
11:     $f(s) \leftarrow f(s_2)$ 
12:     $s \leftarrow s_2$ 
13:     $nivel \leftarrow nivel + 1$ 
14:  else
15:     $nivel \leftarrow nivel + 1$ 
16:  end if
17:   $iter \leftarrow iter + 1$ 
18: end while
19: Retorne  $s$ 

```

troca de 4 *hubs*.

3.3 Etapas preliminares para os experimentos computacionais associados ao USAHMCP

Esta seção apresenta os experimentos computacionais realizados associados à solução do USAHMCP. A Seção 3.3.1 detalha as instâncias, os parâmetros do algoritmo e as configurações dos experimentos computacionais. A Seção 3.3.2 avalia a importância de cada estrutura de vizinhança, enquanto a Seção 3.3.3 avalia as possíveis ordens para explorar estruturas de vizinhança no VND.

3.3.1 Detalhes dos experimentos computacionais

Os algoritmos foram codificados usando a linguagem C++ e executados em um *notebook* com um processador Intel core i7 10750H 2.60 GHz e 16 GB de RAM, usando o sistema operacional Windows. Os modelos foram implementados usando a biblioteca *Concert Technologies* e resolvidos usando a versão acadêmica 20.1.0.0 do *software* IBM ILOG CPLEX. Os experimentos com as heurísticas usam o parâmetro $ILSMAX = 3$.

Os testes apresentam os resultados obtidos com a heurística proposta, considerando as duas estratégias de alocação, usando quatro conjuntos de dados da literatura. A heurística proposta é aplicada à solução de instâncias do USAHMCP e do USA p HMCP. Este último é abordado nos termos propostos na Proposição 3.1.1, transformando-o, assim, em um caso particular do USAHMCP, em que o orçamento é igual ao número de *hubs* e os custos fixos de instalação de *hubs* são iguais a 1.

Os resultados da aplicação da heurística ILS-VND são comparados com os resultados da literatura para o USA p HMCP e com os resultados da aplicação do *solver* CPLEX para o USAHMCP.

Neste trabalho, a menos que a seção explicita de forma diferente, para problemas de maximização, o *gap* foi calculado da seguinte forma:

$$gap = 100 \left(\frac{FO_{CPLEX} - FO_{Heur}}{FO_{Heur}} \right)$$

Em que FO_{CPLEX} é o valor de função objetivo obtido pelo *solver* CPLEX e FO_{Heur} é o valor de função objetivo obtido pela heurística. O valor de *gap* é tal que, quanto mais próximo de zero, melhor é.

Os experimentos computacionais utilizaram os seguintes conjuntos de dados:

- (i) *Australian Post* (AP): proposto por Ernst & Krishnamoorthy (1996);
- (ii) *Civil Aeronautics Board* (CAB), proposto por O’Kelly (1987);

- (iii) as instâncias propostas por [Contreras et al. \(2011\)](#); e
- (iv) conjunto de instâncias URAND com 1000 nós, proposto por [Ilić et al. \(2010\)](#).

Para simplificar, o terceiro conjunto de dados será chamado de DS3 a partir de agora.

Para o USAHMCP, o orçamento b é dado pelo custo fixo médio de instalação de *hubs* $\bar{f}$, calculado como a média aritmética dos custos fixos, multiplicado por um parâmetro inteiro p , isto é, $b = \bar{f} \times p$. Vale a pena mencionar que p é um parâmetro para o USA p HMCP. Além disso, como o parâmetro b está sendo definido em função de um parâmetro p , ele também será considerado como um parâmetro nos experimentos para o USAHMCP.

As instâncias AP estão disponíveis em: <https://andreas-ernst.github.io/Mathprog-ORlib/info/readmeAP.html> e possuem até 200 nós. Essas instâncias fornecem a matriz de fluxo de origem-destino, coordenadas dos nós e custos fixos para instalação de *hubs*. Esta tese considera, para estas instâncias, um fator de coleta igual a 1, um fator de desconto igual a 0,75 e um fator de distribuição também igual a 1.

O conjunto de dados AP fornece dois tipos de custos fixos para instalação de *hubs*:

- (i) custos *tight*, representados como T, em que os custos fixos são maiores em nós com fluxos maiores; e
- (ii) custos fixos *loose*, representados como L, que não diferenciam custos fixos com base no volume de fluxo do nó.

A notação $APn.pL$ e $APn.pT$ indica uma instância com n nós, parâmetro p , com custo fixo do tipo L e T, respectivamente. A matriz de distância é calculada usando a distância euclidiana entre os nós. Como as instâncias AP não fornecem os valores de β_{ij} , foi utilizado um único valor β para todos os nós. Este valor foi obtido como o resultado da função objetivo para o problema p -Hub Centro de Alocação Única não Capacitado (USA p HCP) e representado como $\bar{\beta}$, que é o valor tradicionalmente usado na literatura. Esses valores foram fornecidos por [Peker & Kara \(2015\)](#), para $n < 25$, e por [Meyer et al. \(2009\)](#), para $n \geq 25$. Os mesmos valores de β_{ij} foram usados em [Silva & Cunha \(2017\)](#).

As instâncias CAB possuem até 25 nós e também estão disponíveis em: <https://andreas-ernst.github.io/Mathprog-ORlib/info/readmeAP.html>. As instâncias CAB fornecem informações sobre o número de nós, parâmetro p , matriz de fluxo origem-destino e coordenadas dos nós. Nesta tese foram utilizados fatores de coleta e distribuição iguais a 1, e o fator de desconto $\alpha \in \{0,4; 0,6; 0,8\}$. A matriz de distância é calculada usando a distância euclidiana entre os nós. Diferentemente das

instâncias AP, as instâncias CAB não possuem um custo fixo para a instalação do *hub*. Assim, nessa tese, foram utilizados os valores de custo fixo empregados em de Camargo et al. (2008), que podem ser encontrados em <https://github.com/fabricioneo/plcml/tree/main>. A notação $CABn.p.\alpha$ indica uma instância CAB com n nós, parâmetro p e um fator de desconto α . Os valores para β_{ij} também não estão presentes nas instâncias CAB e foram definidos novamente como um único valor β , fornecido originalmente por Kara & Tansel (2003). Esses valores para β_{ij} também foram usados em outros estudos (Hwang & Lee, 2012; Peker & Kara, 2015; Silva & Cunha, 2017).

As instâncias DS3 propostas em Contreras et al. (2011) estão disponíveis em <https://users.encs.concordia.ca/~icontrer/web/instances.html>. Essas instâncias têm três níveis diferentes de magnitude para a quantidade de fluxo originado em um determinado nó:

- (i) nós de nível baixo (LL);
- (ii) nós de nível médio (ML); e
- (iii) nós de nível alto (HL).

Os fluxos de saída total dos nós LL, ML e HL estão nos intervalos $[1, 10]$, $[10, 100]$ e $[100, 1000]$, respectivamente. Este conjunto de dados apresenta três classes de instâncias:

- (a) classe I (representada por i), tendo de 50 a 500 nós, com um incremento de 50 (exceto para 300 nós);
- (b) classe II (representada por j), tendo 50, 100, 150 e 200 nós;
- (c) classe III (representada por k), tendo 50, 100, 150 e 200 nós.

As classes I, II e III fornecem o número de nós, coordenadas de nós, matriz de fluxo e custo fixo para instalação de *hubs*. Os valores de p usados para este trabalho foram 2, 3, 4 e 5. Os fatores de coleta (χ), desconto (α) e distribuição (δ) foram, respectivamente, $\chi = 1$, $\alpha = 0,75$ e $\delta = 1$.

Como este é o primeiro trabalho que usa instâncias do conjunto DS3 no problema de cobertura de *hubs*, os valores β_{ij} não estão disponíveis para essas instâncias. Portanto, os experimentos usam um único valor $\bar{\beta}$ para todos os β_{ij} , assim como feito para o conjunto de dados AP e CAB. O valor $\bar{\beta}$ para o conjunto DS3 foi calculado com base no valor $\bar{\beta}$ para as instâncias AP. Para determinar um valor $\bar{\beta}$ para o conjunto DS3, as frações $\bar{\beta}/\bar{d}$ das instâncias $APn.pL$ foram computadas, sendo $\bar{d}$ a distância média entre os nós. Os resultados para diferentes valores de n e p estão na Tabela 3.1.

Analisando-a, pode-se notar que a média das razões diminui à medida que o número de *hubs* aumenta, o que é esperado, já que um maior número de *hubs* tende a reduzir a distância de um nó i qualquer a um nó j genérico, passando pelos *hubs* k e m .

Para instâncias DS3, o valor de β_{ij} usado foi o valor médio da fração multiplicado pelas distâncias médias. Assim, para instâncias com 2 *hubs*, o valor β_{ij} usado é dado por $\bar{\beta} = 2,88 \times \bar{d}$, enquanto para instâncias com 3 *hubs*, o valor de β_{ij} usado foi dado por $\bar{\beta} = 2,64 \times \bar{d}$. A notação $DS3nw.p$ indica uma instância do conjunto DS3 com n nós da classe w e parâmetro p .

Tabela 3.1: Valores para a razão $\bar{\beta}/\bar{d}$ para instâncias $APn.pL$.

n	p			
	2	3	4	5
10	2,37	2,04	1,91	1,91
20	2,45	2,31	2,06	2,02
25	2,70	2,37	2,31	2,31
40	3,05	2,87	2,58	2,46
50	3,23	2,97	2,61	2,50
100	3,15	2,90	2,69	2,60
200	3,24	3,05	2,85	2,78
Média	2,88	2,64	2,43	2,37

Também foi utilizado o conjunto de instâncias URAND com 1000 nós, propostas por Ilić et al. (2010), que fornecem o número de nós, valor de p , o fluxo origem-destino e as coordenadas dos nós. A matriz de distância é obtida calculando-se a distância euclidiana entre os nós. A notação $URn.p$ indica uma instância URAND com n nós e parâmetro p . O custo para abrir um *hub* não está presente na instância. Portanto, os testes com as instâncias do conjunto URAND consideraram apenas o USAHMC, o que significa que todos os custos de abertura de *hub* foram iguais a 1. Os experimentos usam os valores $\beta_{ij} = \bar{\beta}$ apresentados em Janković et al. (2017), que são iguais a 75% do valor da função objetivo encontrado para o problema p -hub centro de alocação múltipla não capacitado no trabalho de Brimberg et al. (2017).

3.3.2 Avaliação da importância das estruturas de vizinhança

Essa seção apresenta um conjunto de experimentos, conduzidos para avaliar a importância de cada estrutura de vizinhança apresentadas na Seção 3.2.4. Esses experimentos visam avaliar o efeito da remoção de qualquer uma dessas estruturas na qualidade da solução e no tempo de execução.

Os experimentos utilizaram instâncias $APn.p$ com 10, 20, 25, 40, 50 e 100 nós e $p \in \{2, 3, 4, 5\}$, resultando em 24 instâncias. Esses experimentos utilizam apenas a

heurística de busca local VND (Algoritmo 3) com uma estratégia de alocação gulosa baseada em distância. Além disso, as estruturas de vizinhança foram exploradas no Algoritmo 3 na mesma ordem em que foram originalmente apresentadas na Seção 3.2.4. Estes experimentos foram limitados ao USApHMCP.

Assim, a versão do algoritmo VND com todas as estruturas de vizinhança foi executada para cada instância. Em seguida, o mesmo experimento foi realizado para a versão do algoritmo VND resultante da remoção de uma estrutura de vizinhança e mantendo as outras três. Para cada versão do algoritmo e cada instância foram obtidos os valores da função objetivo (FO) e do tempo de execução, em segundos. Em seguida, foi calculada a diferença entre o valor de FO da versão com uma estrutura de vizinhança removida e o valor de FO com todas as quatro vizinhanças.

A Tabela 3.2 mostra os resultados dos experimentos computacionais. Os melhores e piores *gaps* médios e tempos de execução foram destacados em negrito. A coluna APn apresenta o *gap* médio e o tempo de execução para instâncias com n nós. As linhas gap_i indicam o *gap* médio referente à diferença relativa entre o valor da função objetivo da versão com todas as estruturas de vizinhança e o valor de FO da versão do algoritmo VND sem a estrutura de vizinhança $\mathcal{N}_i$. O valor de gap_i é dado por:

$$gap_i = 100 \left(\frac{FO_{todas} - FO_i}{FO_{todas}} \right),$$

em que FO_{todas} é o valor de FO para o algoritmo VND com todas as estruturas de vizinhança e FO_i é o valor de FO para o algoritmo VND sem a vizinhança i . Além disso, as linhas T_i indicam o tempo médio de execução, em segundos, da versão do algoritmo sem $\mathcal{N}_i$, enquanto T_{todas} indica o tempo médio de execução com todas as estruturas.

Tabela 3.2: Valores de *gap* e tempo quando uma das estruturas de vizinhança é removida.

	AP10	AP20	AP25	AP40	AP50	AP100	média
gap_1	1,688	0,113	0,0462	0,275	-0,009	0,016	0,355
gap_2	0,264	-0,113	0,014	0,0018	0	0,012	0,0299
gap_3	0	0	0	0	0,028	0,011	0,007
gap_4	0,961	0,524	-0,015	0,059	0,034	-0,028	0,256
T_1	0,009	0,088	0,23	0,788	2,853	22,0678	4,339
T_2	0,00775	0,073	0,1285	0,867	1,88025	22,4575	4,236
T_3	0,0095	0,06975	0,153	0,942	4,981	30,8185	6,162
T_4	0,00575	0,01925	0,02775	0,087	0,1655	1,67925	0,331
T_{todas}	0,0075	0,0725	0,11075	0,65625	1,907	25,57575	4,722

É importante ressaltar que o valor de gap_i negativo indica que a remoção da

estrutura $\mathcal{N}_i$ resultou em soluções melhores, em média, em comparação com a versão com todas as estruturas de vizinhança, enquanto o valor de gap_i positivo indica o contrário, ou seja, que a remoção da estrutura $\mathcal{N}_i$ resultou em soluções piores do que com todas as estruturas de vizinhança.

Analisando esta tabela, nota-se que o gap médio ao retirar cada estrutura de vizinhança é positivo, indicando que a versão com todas as estruturas de vizinhança encontra soluções de melhor qualidade, em média, do que as versões sem uma das estruturas de vizinhança. Considerando que quanto maior o gap , pior será o desempenho da busca local sem a estrutura de vizinhança, pode-se perceber que o pior cenário seria retirar a vizinhança $\mathcal{N}_1$, seguida da vizinhança $\mathcal{N}_4$. A retirada da vizinhança $\mathcal{N}_3$ tem o menor impacto no gap . Porém, pode-se observar que a remoção da vizinhança $\mathcal{N}_3$ é a segunda mais impactante para a instância de 50 nós, e a ausência da vizinhança $\mathcal{N}_2$ é a segunda mais impactante para as instâncias de 100 nós. Além disso, a remoção da vizinhança $\mathcal{N}_2$ melhorou o gap médio das instâncias de 20 nós, a remoção da vizinhança $\mathcal{N}_4$ melhorou o gap médio das instâncias de 25 e 100 nós, e a remoção da vizinhança $\mathcal{N}_1$ melhorou o gap médio das instâncias de 50 nós.

Analisando o tempo de execução médio, pode-se notar que a remoção da vizinhança $\mathcal{N}_3$ aumentou o tempo de execução médio em aproximadamente 31%. Portanto, embora a análise de $gaps$ indique que ela quase não afeta o valor de FO , seu impacto no tempo de execução é bastante significativo. Além disso, a remoção da vizinhança $\mathcal{N}_4$ reduz o tempo médio de execução. Este resultado é esperado, pois, pela análise da complexidade das estruturas de vizinhança (Seção 3.2.4), esta é a vizinhança que exige maior esforço computacional, por incluir uma busca local na avaliação de cada vizinho.

Portanto, os resultados mostram que todas as vizinhanças são importantes, uma vez que a remoção da vizinhança $\mathcal{N}_3$ aumenta o tempo de execução do algoritmo e é a segunda mais impactante para instâncias de 50 nós. A remoção da vizinhança $\mathcal{N}_4$ melhoraria o tempo de execução, mas a vizinhança $\mathcal{N}_4$ tem o segundo maior gap médio geral. A vizinhança $\mathcal{N}_1$ é a que contribui mais positivamente para o gap médio global. Por fim, a remoção da vizinhança $\mathcal{N}_2$ não seria benéfica, pois é a segunda mais impactante para as instâncias de 25 e 100 nós.

3.3.3 Avaliação da ordem de exploração das estruturas de vizinhança

Foi realizado um conjunto de experimentos para avaliar as diferentes ordens de exploração. Todos os experimentos utilizaram apenas o algoritmo VND com alocação gulosa baseada em distância aplicado ao USApHMCP.

Tabela 3.3: Resultados computacionais do VND na resolução de instâncias AP para cada ordem de exploração de estruturas de vizinhança

n.p	1234		1324		1243		1423		1432		1342	
	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]
10.2	0	0,007	0	0,004	0	0,007	0	0,007	0	0,003	0	0,001
20.3	0,45	0,025	0,45	0,028	0,45	0,027	0	0,058	0	0,028	0	0,027
25.4	0,06	0,25	0,06	0,129	0,06	0,267	0,06	0,113	0,06	0,135	0,06	0,128
40.5	0	0,728	0	0,71	0	1,851	0	1,316	0	1,338	0	0,481
50.2	0	0,322	0	0,326	0	0,33	0	0,227	0	0,23	0	0,232
100.4	0,09	26,32	0,141	21,913	0,14	30,912	0,14	23,284	0,14	23,103	0,09	19,264
média	0,1	4,61	0,109	3,85	0,11	5,57	0,03	4,1675	0,03	4,1395	0,025	3,3555
n.p	2134		2143		2413		2431		2341		2314	
	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]
10.2	0	0,002	0	0,001	0	0,001	0	0,001	0	0,002	0	0,002
20.3	0	0,028	0	0,028	0,87	0,009	0,87	0,011	0,87	0,012	0,87	0,012
25.4	0,06	0,134	0,06	0,136	0,63	0,035	0,63	0,034	0,63	0,032	0	0,148
40.5	0	0,485	0	1,314	0,02	1,603	0,08	1,183	0,02	1,6	0,08	1,177
50.2	0	0,23	0	0,232	0	0,234	0	0,24	0	0,241	0	0,236
100.4	0,09	19,377	0,14	23,179	0,06	7,224	0,06	24,362	0,06	7,442	0,06	24,313
média	0,025	3,376	0,03	4,15	0,26	1,52	0,27	4,31	0,26	1,55	0,17	4,31
n.p	3214		3241		3421		3412		3142		3124	
	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]
10.2	0	0,002	0	0,002	0	0,003	0	0,002	0	0,002	0	0,002
20.3	0,87	0,011	0,87	0,011	0,87	0,012	0,87	0,011	0,87	0,011	0,87	0,009
25.4	0	0,151	0,63	0,034	0	0,149	0,63	0,036	0	0,147	0,63	0,032
40.5	0,08	1,164	0,02	1,574	0,02	1,583	0,02	1,583	0,08	1,162	0,08	1,165
50.2	0	0,237	0	0,243	0	0,244	0	0,239	0	0,237	0	0,237
100.4	0,06	24,617	0,06	7,299	0,06	7,253	0,06	7,374	0,06	24,217	0,06	24,542
média	0,17	4,36	0,26	1,53	0,16	1,54	0,26	1,54	0,17	4,296	0,27	4,33
n.p	4312		4321		4231		4213		4123		4132	
	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]	gap	T[s]
10.2	0	0,001	0	0,001	0	0,002	0	0,002	0	0,002	0	0,001
20.3	0,87	0,011	0,87	0,012	0,87	0,009	0,87	0,011	0,87	0,011	0,87	0,008
25.4	0,63	0,032	0	0,15	0	0,146	0	0,148	0,63	0,032	0,63	0,033
40.5	0,08	1,188	0,08	1,166	0,08	1,162	0,02	1,569	0,02	1,588	0,02	1,597
50.2	0	0,243	0	0,238	0	0,242	0	0,237	0	0,234	0	0,236
100.4	0,06	24,216	0,06	24,334	0,06	24,39	0,06	7,255	0,06	7,323	0,06	7,256
média	0,27	4,28	0,17	4,32	0,17	4,33	0,16	1,537	0,26	1,53	0,26	1,52

Esses experimentos foram realizados com instâncias AP com 10 nós e 2 *hubs*, 20 nós e 3 *hubs*, 25 nós e 4 *hubs*, 40 nós e 5 *hubs*, 50 nós e 2 *hubs* e 100 nós e 4 *hubs*, resultando em um total de seis instâncias. Além disso, todas as 24 ordens foram consideradas.

A Tabela 3.3 fornece os *gaps* e o tempo de execução para cada sequência. A sequência “*abcd*” representa uma exploração na seguinte ordem: $\mathcal{N}_a, \mathcal{N}_b, \mathcal{N}_c$ e $\mathcal{N}_d$. A tabela destaca as menores médias de valores de *gap* e de tempo de execução em negrito.

De acordo com a Tabela 3.3, o menor valor médio de *gap* ocorreu nas ordens de exploração 1342 e 2134, ambas atingindo o valor de 0,025. Como está sendo utilizado um algoritmo VND cíclico, vale destacar que as duas ordens de exploração se referem ao mesmo ciclo, diferindo apenas na estrutura de vizinhança em que a busca se inicia. Porém, considerando estas duas ordens, o menor tempo médio ocorreu para a ordem de exploração 1342. Em relação ao tempo, o menor tempo médio de execução, dado por 1,52 segundos, ocorreu nas ordens de exploração 4132 e 2413.

Portanto, a ordem de exploração 1342 será utilizada no restante deste capítulo.

3.4 Experimentos computacionais usando as instâncias AP

Esta seção apresenta os resultados computacionais da aplicação da heurística ILS-VND para a solução dos problemas USAHMCP e USA p HMCP usando instâncias do conjunto de dados AP. Inicialmente, na Subseção 3.4.1, são apresentados os resultados computacionais referentes à solução de instâncias $APn.pL$ e $APn.pT$ com até 50 nós usando o *solver* CPLEX. O objetivo é comparar os resultados dos problemas USA p HMCP e o USAHMCP. A Subseção 3.4.2, por sua vez, apresenta a comparação entre os resultados da heurística ILS-VND proposta, aplicada ao USA p HMCP, com os resultados de [Silva & Cunha \(2017\)](#), que é o estado da arte para essas instâncias. Por fim, a Subseção 3.4.3 mostra a comparação entre os resultados advindos da aplicação da heurística ILS-VND e os referentes ao *solver* CPLEX para instâncias do USAHMCP.

3.4.1 Comparação entre USA p HMCP e USAHMCP

Esses experimentos computacionais usam o *solver* CPLEX para resolver instâncias AP com até 50 nós. Os resultados estão apresentados na Tabela 3.4. Esta tabela mostra os valores da função objetivo (FO), o tempo de execução em segundos ($T[s]$) e o custo de abertura dos *hubs* abertos para o USA p HMCP e o USAHMCP usando instâncias AP nas versões L e T denotadas por $n.p$. Os valores que foram melhores para uma versão do que para outra foram destacados em negrito. Além disso, como os valores de custos fixos não são utilizados para resolver o USA p HMCP, os valores de custos reportados representam uma simulação de custos assumindo que existe um custo fixo para abrir *hubs* do tipo L ou T.

De acordo com a tabela, o USAHMCP, considerando apenas os custos fixos do tipo L, obtém soluções ótimas com os mesmos valores de função objetivo e quantidade de *hubs* que o USA p HMCP. Porém, os custos de abertura de *hubs* são diferentes para sete instâncias, mostrando que a resolução dos dois problemas não resulta nos mesmos conjuntos de *hubs* abertos para essas instâncias. Além disso, o USAHMCP(L) encontra soluções com menores custos de instalação para quatro instâncias, enquanto o USA p HMCP para três. É importante notar também que o tempo de execução dos dois problemas foi bastante semelhante.

O teste estatístico de Wilcoxon pareado ([Wilcoxon, 1945](#)) foi aplicado para verificar se havia evidência estatística de diferença entre os tempos de resolução dos dois

Tabela 3.4: Resultados para as versões USApHMCP e USAHMCP usando o *solver* CPLEX em instâncias AP com até 50 nós.

Inst	USApHMCP				USAHMCP (T)				USAHMCP (L)			
	FO	T[s]	Custo(T)	Custo(L)	FO	T[s]	Custo	$\bar{p}$	FO	T[s]	Custo	$\bar{p}$
10.2	3978,92	0,13	65649	55627	3978,92	0,15	65649	2	3978,92	0,13	55649	2
10.3	3937,3	0,08	97026	82997	3978,92	0,08	122450	4	3937,3	0,08	82997	3
10.4	3954,53	0,09	130690	112608	3978,92	0,05	163509	5	3954,53	0,09	112781	4
10.5	3954,53	0,08	169831	142382	3978,92	0,03	190021	6	3954,53	0,08	141465	5
20.2	3973,2	0,34	60291	48331	3978,92	0,57	86389	3	3973,2	0,35	48331	2
20.3	3973,2	0,75	86389	70270	3978,92	0,30	136047	5	3973,2	0,73	70270	3
20.4	3974,27	0,65	141838	108560	3978,92	0,33	181510	5	3974,27	0,65	108560	4
20.5	3973,2	0,72	167936	130498	3973,2	0,27	227922	7	3973,2	0,72	130498	5
25.2	3976,77	1,22	61852	43008	3978,92	1,74	124173	3	3976,77	1,23	43008	2
25.3	3972,51	1,49	132798	75943	3978,92	1,34	150106	4	3972,51	1,46	75943	3
25.4	3976,68	0,82	173131	99754	3976,68	0,59	244220	6	3976,68	0,79	93765	4
25.5	3976,68	1,61	220919	123704	3976,68	0,88	314353	8	3976,68	1,64	129999	5
40.2	3978,92	4,88	105614	50470	3978,92	8	154295	4	3978,92	4,96	50470	2
40.3	3978,92	4,65	127267	69921	3978,92	4,62	189128	4	3978,92	4,72	69921	3
40.4	3977,28	19,15	278197	111138	3978,92	5,89	308824	6	3977,28	21,03	107501	4
40.5	3977,97	22,1	248564	119795	3977,97	5,94	388145	9	3977,97	23,51	119795	5
50.2	3978,69	472,4	222350	60322	3978,92	27,35	183372	4	3978,69	395,7	54477	2
50.3	3978,42	897,9	155987	70510	3978,92	8,97	272440	6	3978,42	792,6	70510	3
50.4	3978,92	5,44	217747	95212	3978,92	4,83	359136	7	3978,92	5,11	95212	4
50.5	3978,92	5,19	245052	116600	3978,92	7,87	445895	9	3978,92	7,23	116600	5

problemas. Com nível de significância de 0,05, que será utilizado em todos os testes estatísticos realizados a seguir, os resultados do teste não indicaram diferença estatisticamente significativa entre os tempos de execução do USAHMCP e do USApHMCP ($p - \text{valor} = 0,9$). Vale ressaltar que o teste de Shapiro-Wilk (Shapiro & Wilk, 1965) para testar normalidade dos valores de tempo de execução computacional referentes ao USApHMCP retornou p-valor igual a $2,8 \times 10^{-8}$, indicando não normalidade dos dados. Entretanto, o teste de Shapiro-Wilk para os valores referentes ao custo de USAHMCP retornou p-valor igual a 0,28 (os outros dados de custo também encontraram p-valor maior que 0,05), indicando normalidade nos dados de custo. O teste de Bartlett (Bartlett, 1937) encontrou p-valor igual a 5×10^{-9} , indicando que as variâncias são diferentes; e o teste de Welch (Welch, 1947) foi aplicado, não encontrando diferença significativa entre os custos de USApHMCP e USAHMCP (p-valor= 0,96).

Considerando os custos fixos do tipo T, verifica-se que o USAHMCP encontra soluções com maior cobertura em 11 das 20 instâncias e com cobertura igual à do USApHMCP e do USAHMCP(L) nas restantes 9 instâncias. O USAHMCP(T) abriu mais *hubs* do que o USApHMCP em 19 casos, geralmente tendo um custo total de abertura de *hubs* mais elevado. Porém, vale ressaltar que a solução ótima da instância AP50.2T do USAHMCP(T) resulta em maior cobertura e menor custo total para abertura de *hubs* do que a solução ótima do USApHMCP, com o cálculo dos custos de abertura utilizando os custos fixos do tipo T. Além disso, o tempo de execução do USAHMCP(T) foi menor que o tempo de execução do USApHMCP e do

USAHMCP(L) para 14 das 20 instâncias. Os resultados do teste de Wilcoxon indicaram diferença significativa entre os valores da função objetivo (p -valor = 0,0038), já o teste de normalidade de Shapiro-Wilk obteve p -valor igual a $1,8 \times 10^{-7}$ indicando não normalidade nos dados de valores de função objetivo. O teste de Welch para comparação dos custos de abertura de *hubs* também indicou diferença significativa entre os custos de abertura de USA p HMCP e USAHMCP(T) ao encontrar p -valor igual a $2,9 \times 10^{-5}$. Portanto, para estes testes com o tipo T, há indícios de que os custos de instalação de *hubs* do problema USA p HMCP foram menores que os custos de instalação de *hubs* do problema USAHMCP.

As Figuras 3.2 e 3.3 descrevem a rede *hub-and-spoke*, representando a solução ótima para as instâncias AP10.5 e AP25.4 para o USA p HMCP e o USAHMCP, com ambos os tipos de custos de abertura de *hub*. Nestas figuras, os *hubs* são representados por círculos vermelhos, enquanto os nós não *hubs* são representados por círculos azuis.

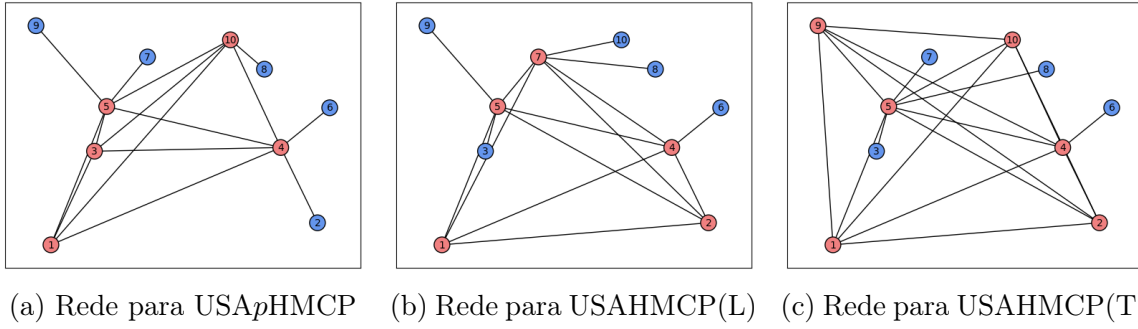
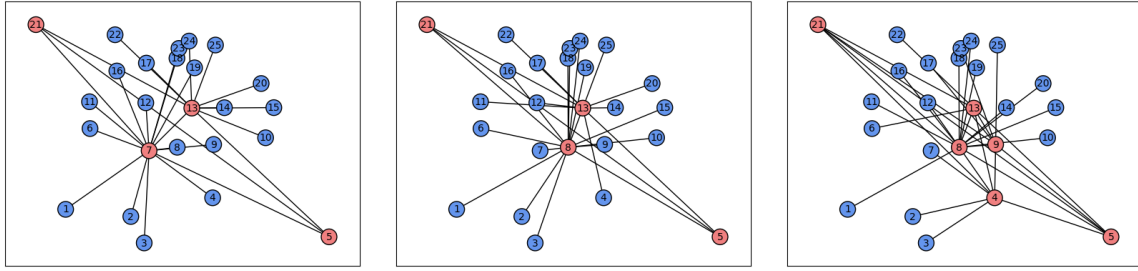


Figura 3.2: Ilustração da rede para a solução ótima para instância AP com 10 nós e $p = 5$ para o USA p HMCP e para o USAHMCP com custos fixos do tipo L e do tipo T.

Pode-se notar que as redes, na Figura 3.2, representando as soluções do USA p HMCP e do USAHMCP(L) (Figuras 3.2a–3.2b), apesar de terem o mesmo número de *hubs*, têm *hubs* em nós diferentes. Além disso, os *hubs* não comuns podem estar razoavelmente distantes. Observe que o nó 2 é um *hub* na Figura 3.2b, mas está distante dos nós 3 e 10, que são *hubs* apenas na Figura 3.2a. Porém, é importante ressaltar que, apesar disso, as duas redes resultam na mesma cobertura (Tabela 3.4). A rede para a solução do USAHMCP(T) (Figura 3.2c) abre uma unidade de *hub* a mais, mas, com exceção do *hub* 9, ela compartilha *hubs* com as outras duas redes. Adicionalmente, é importante lembrar que esta solução corresponde a uma solução com cobertura um pouco maior que as demais, possivelmente devido à abertura do *hub* 9. Outro ponto interessante a destacar é que os nós não-*hub* nem sempre são alocados ao *hub* mais próximo. Por exemplo, o nó 8, na Figura 3.2c, é alocado ao *hub* 5, em vez de ser alocado ao *hub* 10, que é o *hub* mais próximo, ou ao *hub* 4.

Considerando a Figura 3.3, as redes que representam as soluções do USA p HMCP



(a) Rede para USApHMCP (b) Rede para USAHMCP(L) (c) Rede para USAHMCP(T)

Figura 3.3: Ilustração da rede para a solução ótima para instância AP com 25 nós e $p = 4$ para o USApHMCP e para o USAHMCP com custos fixos do tipo L e do tipo T.

e do USAHMCP(L) (Figuras 3.3a–3.3b) são razoavelmente semelhantes, diferindo apenas na escolha dos nós 7 ou 8 para serem *hubs*, que estão muito próximos uns dos outros. No caso da rede USAHMCP(T), na Figura 3.3c, ela possui todos os *hubs* da solução USAHMCP(L) e dois *hubs* adicionais. Porém, vale ressaltar que apesar de possuir dois *hubs* adicionais, o valor de FO ótimo para o USAHMCP(T) é igual ao valor de FO ótimo para o USAHMCP(L), mostrando que a solução representada na Figura 3.3b com custo mais baixo para abertura de *hub* também é, neste caso, uma solução ótima para o USAHMCP(T). Assim, é possível que se obtenha uma solução com a mesma cobertura e com um orçamento menor.

3.4.2 Comparação das variantes da heurística ILS-VND com a heurística da literatura aplicada ao USApHMCP

A heurística implementada para o problema USAHMCP foi aplicada ao caso particular USApHMCP tomando o orçamento igual a p e assumindo todos os custos de abertura de *hubs* iguais a 1 para comparar os resultados com os de Silva & Cunha (2017). Os experimentos computacionais utilizaram instâncias AP com até 200 nós.

Os experimentos consistiram em executar o *solver* CPLEX para obter ou a solução ótima ou um limite superior. Foi estabelecido um limite de tempo de 86400 segundos (24h) para execução do *solver*. Para instâncias em que o CPLEX excedeu a memória disponível, foi considerado o limite superior de 3978,92 para a cobertura. Este valor é igual à soma de todas as demandas de cada instância do conjunto de dados AP, independentemente do seu tamanho.

Todos os experimentos computacionais utilizando a heurística ILS-VND proposta consideram as duas formas de alocação (Subseção 3.2.5). A versão ILS-VND-D refere-se à alocação baseada em distância e a versão ILS-VND-CP refere-se à alocação baseada em potencial de cobertura. Cada heurística foi executada 30 vezes em cada

instância.

Os resultados são apresentados na Tabela 3.5. Nesta tabela, a coluna UB apresenta os limites superiores obtidos pelo *solver* CPLEX ou a demanda total. A coluna *gap* apresenta o *gap* relativo entre o valor da função objetivo da solução encontrada pela heurística Busca Tabu (TS) de Silva & Cunha (2017) e o valor de UB mostrado na própria coluna. Vale ressaltar que, segundo Silva & Cunha (2017), sua heurística de busca Tabu sempre inicia a busca pela mesma solução (encontrada de forma gulosa) e, portanto, por não ser um procedimento estocástico, seu valor de *gap* é único. As colunas $\overline{gap}$ e gap^* apresentam os *gaps* médios e de melhor valor do valor de *FO*, respectivamente, alcançado pelas variantes heurísticas propostas neste trabalho e pelo UB. A coluna T[s] apresenta o tempo de execução em segundos.

Como os experimentos computacionais reportados por Silva & Cunha (2017) utilizaram um processador diferente do utilizado nos experimentos computacionais desta tese, o tempo de execução foi ajustado com base na *CPU mark* de cada processador utilizando a ferramenta *PassMark*, disponível em <https://www.cpubenchmark.net/singleCompare.php>. Como os valores de *CPU mark* dos processadores utilizados por Silva & Cunha (2017) e nesta tese são 24338 e 11918, respectivamente, então o tempo de execução computacional informado na Tabela 3.5 é igual ao tempo computacional informado por Silva & Cunha (2017) multiplicado pelo fator $24338/11918 = 2,04$.

O *solver* CPLEX reportou falta de memória no estágio de carregamento do modelo para as instâncias de 200 nós. Estes casos estão representados na tabela com a notação “Mem”. Além disso, o CPLEX atingiu o tempo limite de 24 horas na resolução da instância de 100 nós com p igual a 2, 3 e 4. Nesse caso, o *solver* encontrou os seguintes limites para as instâncias analisadas: para 2 *hubs*, limite inferior de 3929,91 e limite superior de 3978,92; para 3 *hubs*, limite inferior de 3480,78 e limite superior de 3978,92; e para 4 *hubs*, limite inferior de 3947,24 e limite superior de 3978,92. Na coluna TS *gap*, os valores iguais a zero foram destacados nos casos em que alguma de nossas variantes heurísticas não obteve *gap* zero. Além disso, os melhores tempos (entre todos os métodos) e os menores *gaps* (entre nossas heurísticas) foram destacados em negrito, desde que não houvesse empate.

De acordo com a Tabela 3.5, o valor médio de $\overline{gap}$ e gap^* da versão ILS-VND-D é maior que os valores médios da versão ILS-VND-CP. Assim, considerando os casos testados, pode-se observar uma melhoria nos *gaps* quando se utiliza a alocação gulosa baseada no potencial de cobertura em comparação com a alocação gulosa baseada na distância. Além disso, a alocação baseada no potencial de cobertura alcançou um *gap* médio igual a zero em 17 dos 28 casos, enquanto a alocação baseada na distância, em 9 casos. A alocação baseada no potencial de cobertura alcançou um *gap* para a melhor solução igual a zero em 18 instâncias, enquanto a alocação baseada na distância, em

Tabela 3.5: Resultados para a heurística e o *solver* CPLEX aplicados ao problema USA_pHMCP usando instâncias AP em comparação com o algoritmo de busca tabu (TS) da literatura

Inst	CPLEX		TS		ILS-VND-D			ILS-VND-CP		
	UB	T[s]	gap	T[s]	$\overline{gap}$	gap*	T[s]	$\overline{gap}$	gap*	T[s]
AP10.2L	3978,92	0,06	0	0,03	0	0	0,002	0	0	0,01
AP10.3L	3937,3	0,12	0	0,1	0	0	0,004	0	0	0,015
AP10.4L	3954,53	0,12	0	0,19	0	0	0,003	0	0	0,017
AP10.5L	3954,53	0,13	0	0,27	0	0	0,004	0	0	0,018
AP20.2L	3973,21	0,38	0	0,39	0,12	0,12	0,012	0	0	0,038
AP20.3L	3973,2	0,99	0	1,22	0,43	0,43	0,042	0	0	0,088
AP20.4L	3974,27	0,26	0	2,47	1,29	1,29	0,072	0,42	0,42	0,091
AP20.5L	3973,2	0,39	0	4,1	0,11	0,11	0,136	0	0	0,185
AP25.2L	3976,57	1,4	0	0,94	0,04	0,04	0,02	0	0	0,09
AP25.3L	3972,51	2,44	0	2,999	0,11	0,11	0,1	0	0	0,35
AP25.4L	3976,68	1,36	0	6,02	0	0	0,067	0	0	0,88
AP25.5L	3976,68	1,62	0	10,31	0	0	0,093	0	0	0,94
AP40.2L	3978,92	4,33	0	6,71	0,02	0,02	0,387	0	0	1,36
AP40.3L	3978,92	4,59	0	20,67	0,02	0,02	0,722	0	0	2,73
AP40.4L	3977,28	17,96	0	42,47	0	0	1,347	0	0	3,07
AP40.5L	3977,97	19,83	0	72,66	0,05	0,05	0,746	0,08	0,08	4,04
AP50.2L	3978,69	441,42	0	16,11	0	0	0,289	0	0	2,17
AP50.3L	3978,42	864,11	0	49,29	0,03	0,03	2,337	0	0	3,68
AP50.4L	3978,92	20,21	0	100,06	0,08	0,08	6,443	0,04	0,04	9,49
AP50.5L	3978,92	20,23	0	169,05	0,07	0,07	5,101	0,07	0,02	7,71
AP100.2L	3978,69	24h	0	16,11	0,01	0,01	16,314	0,02	0	14,59
AP100.3L	3978,42	24h	0	49,29	0	0	27,954	0	0	31,17
AP100.4L	3978,92	24h	0	100,06	0,02	0,02	53,126	0,008	0,008	57,88
AP100.5L	3978,92	6541,6	0	169,05	0,02	0,02	113,951	0,005	0,005	111,12
AP200.2L	3978,92	Mem	0	5607,96	0,01	0,01	153,873	0,005	0,005	583,43
AP200.3L	3978,92	Mem	0	15662,6	0,02	0,02	1063,17	0,007	0,007	1216,25
AP200.4L	3978,92	Mem	0	32615,5	0,01	0,01	2280,98	0,001	0,001	2173,43
AP200.5L	3978,92	Mem	0	54532,2	0,01	0,01	1291,34	0,003	0,003	2214,99
Média	-	-	0	3902,1	0,09	0,09	179,24	0,024	0,021	229,99

9. Além disso, em 10 instâncias a Busca Tabu obteve um *gap* igual a zero enquanto o algoritmo ILS-VND atingiu um *gap* estritamente positivo.

Considerando o tempo de execução, o tempo médio de execução para a versão ILS-VND-D é 179,24, enquanto o tempo médio de execução para a versão ILS-VND-CP é 229,99. Ou seja, houve um aumento no tempo de execução da heurística ao utilizar a alocação baseada no potencial de cobertura, o que é plausível, pois a complexidade desse procedimento de alocação é maior. O tempo médio de execução do trabalho de [Silva & Cunha \(2017\)](#) é 3902,1 o que indica que, em média, o tempo da heurística ILS-VND-D é mais de 20 vezes menor que a heurística de Busca Tabu de [Silva & Cunha \(2017\)](#). A heurística ILS-VND-CP, em média, gasta 15 vezes menos tempo que a heurística de [Silva & Cunha \(2017\)](#). Assim, considerando as instâncias testadas, percebe-se que a heurística proposta consegue encontrar boas soluções em tempos

computacionais aceitáveis.

O teste estatístico de normalidade de Shapiro-Wilk (Shapiro & Wilk, 1965) para o *gap* do ILS-VND-D obteve p-valor igual a $6,6 \times 10^{-10}$; o mesmo teste aplicado ao tempo de execução de ILS-VND-D obteve p-valor igual a $1,4 \times 10^{-9}$. Isso indica que os dados não são normais. O teste estatístico Wilcoxon pareado foi aplicado para verificar se havia evidência estatística de diferença entre os tempos de execução das duas variantes. O teste indicou evidências de diferença entre os tempos de execução. Os testes também mostraram evidências de que o tempo de execução da variante ILS-VND-CP, a variante com maior tempo médio de execução, é menor que o tempo de execução da heurística de Silva & Cunha (2017) (p-valor = $7,5 \times 10^{-9}$). Adicionalmente, os testes também mostraram que havia evidência estatística de diferença entre os *gaps* do ILS-VND-CP e do ILS-VND-D (p-valor = 0,00072), e também entre o *gap* da heurística de Silva & Cunha (2017) e o *gap* da heurística ILS-VND-CP (p-valor = $7,5 \times 10^{-9}$).

3.4.3 Comparação da heurística com o *solver* CPLEX aplicado ao USAHMCP

Um conjunto de experimentos foi realizado para comparar as duas versões da heurística proposta com os resultados obtidos utilizando o *solver* CPLEX na resolução de instâncias do USAHMCP. Para as instâncias que o CPLEX demorou mais de 24h para executar, o limite superior do valor de *FO* também foi considerado 3978,92, pois, como já mencionado, uma solução com valor de função objetivo maior que este valor não é possível para estas instâncias. Cada versão da heurística foi executada 30 vezes para cada instância.

Os resultados são apresentados na Tabela 3.6 e na Tabela 3.7. As colunas destas tabelas seguem o mesmo padrão já apontado na subseção anterior. A diferença é que, neste caso, não há trabalhos de literatura para comparação, pois este é o primeiro estudo a abordar esta versão do problema.

Para esses experimentos, o CPLEX reportou falta de memória no estágio de carregamento do modelo para instâncias com 200 nós. Isso foi indicado na tabela com a notação “Mem”, na coluna de tempo do CPLEX. Além disso, na Tabela 3.6, o CPLEX excedeu o tempo limite de 24 horas para a instância de 100 nós com $p = 4$, fornecendo um limite inferior de 3895,54 e um limite superior de 3978,92. Os melhores tempos e *gaps* foram destacados quando não houve empate.

De acordo com essas tabelas, o ILS-VND-D obteve *gap* zero em 30 ocorrências, enquanto o ILS-VND-CP o fez em 37 ocorrências. Da mesma forma que na subseção anterior, em média, o ILS-VND-CP encontra soluções com melhores *gaps* do que

Tabela 3.6: Resultados das heurísticas e do *solver* CPLEX aplicados ao problema USAHMCPL na resolução de instâncias APL.

Inst	CPLEX		ILS-VND-D			ILS-VND-CP		
	UB	T[s]	$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
AP10.2L	3978,92	0,035	0	0	0,001	0	0	0,001
AP10.3L	3937,3	0,07	2,87	1,16	0,001	1,39	0	0,001
AP10.4L	3954,53	0,045	0	0	0,001	0	0	0,01
AP10.5L	3954,53	0,054	0	0	0,01	0	0	0,01
AP20.2L	3973,21	0,575	0,12	0,12	0,01	0	0	0,03
AP20.3L	3973,2	0,373	0,43	0,43	0,03	0	0	0,03
AP20.4L	3974,27	0,226	0,42	0,42	0,06	0	0	0,04
AP20.5L	3973,2	0,156	0	0	0,08	0	0	0,09
AP25.2L	3976,77	1,149	0,006	0,006	0,03	0,006	0,006	0,07
AP25.3L	3972,51	1,878	0,17	0,17	0,09	0	0	0,07
AP25.4L	3976,68	0,878	0	0	0,14	0	0	0,12
AP25.5L	3976,68	0,799	0	0	0,11	0	0	0,17
AP40.2L	3978,92	53,688	0,06	0,04	0,32	0,06	0,06	0,36
AP40.3L	3978,92	4,347	0	0	0,73	0	0	0,58
AP40.4L	3977,28	3,892	0	0	0,94	0,08	0,05	0,82
AP40.5L	3977,97	3,021	0	0	0,03	0	0	1,5
AP50.2L	3978,69	8,345	0,01	0	0,25	0	0	0,39
AP50.3L	3978,42	8,774	0,09	0,013	1,13	0,01	0	1,5
AP50.4L	3978,92	5,368	0,08	0,05	3,45	0,07	0,07	2,8
AP50.5L	3978,92	13,873	0,07	0,07	1,75	0,07	0,07	1,82
AP100.2L	3978,92	2090,02	0,03	0,03	4,52	0,03	0,03	3,59
AP100.3L	3978,92	1982,76	0,06	0,06	13,26	0,05	0,05	11,55
AP100.4L	3978,92	24h	0	0	72,42	0	0	39,68
AP100.5L	3978,92	45632,21	0,02	0,02	86,82	0,01	0,01	29,88
AP200.2L	3978,92	Mem	0,01	0,01	51,27	0	0	69,12
AP200.3L	3978,92	Mem	0,008	0,008	127,94	0,007	0,007	285,1
AP200.4L	3978,92	Mem	0,02	0,02	547,23	0,002	0,002	452,74
AP200.5L	3978,92	Mem	0,009	0,009	598,91	0,003	0,003	572,76
Média	-	-	0,16	0,09	53,98	0,06	0,01	52,67

a variante ILS-VND-D. Porém, diferentemente do ocorrido na subseção anterior, o tempo médio do ILS-VND-D foi maior que o tempo médio da heurística ILS-VND-CP. Percebe-se que para apenas uma instância o gap foi maior que 1. Adicionalmente, o gap médio, considerando as 40 instâncias, foi de 0,077% para o ILS-VND-D e 0,035% para o ILS-VND-CP, indicando que as variantes da heurística encontraram boas soluções para essas instâncias. Além disso, o teste de Shapiro para o gap e tempo do ILS-VND-D (considerando a Tabela 3.6) obteve $p\text{-valor} = 3,4 \times 10^{-9}$ e $p\text{-valor} = 1,4 \times 10^{-9}$, respectivamente, indicando não haver normalidade nos dados de gap e tempo; e é importante ressaltar que, segundo o teste estatístico Wilcoxon

Tabela 3.7: Resultados das heurísticas e do *solver* CPLEX aplicados ao problema USAHMCPT na resolução de instâncias APT.

Inst	CPLEX		ILS-VND-D			ILS-VND-CP		
	UB	T[s]	$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
AP10.2T	3978,92	0,15	0	0	0,001	0	0	0,001
AP10.3T	3978,92	0,08	0	0	0,001	0	0	0,01
AP10.4T	3978,92	0,05	0,63	0,61	0,001	0,63	0,61	0,01
AP10.5T	3978,92	0,03	0,61	0,61	0,001	0,61	0,61	0,001
AP20.2T	3978,92	0,57	0,14	0,14	0,02	0	0	0,05
AP20.3T	3978,92	0,30	0	0	0,02	0	0	0,02
AP20.4T	3978,92	0,33	0,12	0,12	0,07	0,12	0,12	0,09
AP20.5T	3973,2	0,27	0	0	0,07	0	0	0,08
AP25.2T	3978,92	1,74	0	0	0,01	0	0	0,02
AP25.3T	3978,92	1,34	0	0	0,12	0,16	0,16	0,10
AP25.4T	3976,68	0,59	0	0	0,05	0	0	0,06
AP25.5T	3976,68	1,61	0	0	0,09	0	0	0,10
AP40.2T	3978,92	4,88	0	0	0,09	0	0	0,08
AP40.3T	3978,92	4,65	0	0	0,99	0	0	1,69
AP40.4T	3978,92	19,15	0	0	0,94	0	0	1,37
AP40.5T	3977,97	22,1	0	0	0,06	0	0	0,3
AP50.2T	3978,92	472,4	0,05	0,04	0,15	0,03	0,02	0,16
AP50.3T	3978,92	897,9	0	0	1,93	0	0	1,72
AP50.4T	3978,92	5,44	0,08	0,08	1,05	0,02	0,02	1,65
AP50.5T	3978,92	5,19	0,15	0,15	1,74	0,15	0,15	2,02
AP100.2T	3978,92	690,02	0	0	13,52	0	0	14,59
AP100.3T	3978,92	382,79	0	0	45,26	0	0	38,55
AP100.4T	3978,92	298,65	0	0	56,42	0	0	61,68
AP100.5T	3978,92	231,91	0,005	0,005	83,02	0	0	63,88
AP200.2T	3978,92	Mem	0	0	111,92	0	0	119,12
AP200.3T	3978,92	Mem	0	0	395,64	0	0	397,1
AP200.4T	3978,92	Mem	0	0	3712,3	0	0	852,74
AP200.5T	3978,92	Mem	0,001	0,001	691,01	0,001	0,001	650,76
Média	-	-	0,064	0,063	182,73	0,061	0,06	78,86

pareado, não houve evidência estatística de diferença entre os tempos de execução (p-valor= 0,53) e *gaps* (p-valor= 0,06) das duas heurísticas. Situação semelhante acontece com os testes estatísticos da Tabela 3.7.

3.5 Experimentos computacionais usando as instâncias CAB

Utilizando instâncias CAB, foram realizados testes para avaliar as duas versões da heurística proposta para a resolução de instâncias do USA p HMCP (Seção 3.5.1)

e do USAHMCP (Seção 3.5.2). Para o USA p HMCP, os resultados dos experimentos computacionais foram comparados com os resultados de Silva & Cunha (2017). Para o USAHMCP, as versões da heurística proposta foram comparadas com os resultados do *solver* CPLEX. Os experimentos usam 48 instâncias CAB com até 25 nós. Cada variante heurística foi executada 30 vezes para cada instância.

3.5.1 Experimentos computacionais para o USA p HMCP usando as instâncias CAB

Esta seção compara os resultados de experimentos computacionais utilizando as duas versões da heurística proposta, utilizando o *solver* CPLEX e os resultados reportados por Silva & Cunha (2017). Os resultados para as instâncias CAB com 10 e 15 nós são apresentados na Tabela 3.8 e os resultados para as instâncias CAB com 20 e 25 nós são apresentados na Tabela 3.9. O tempo de execução do algoritmo TS foi ajustado conforme descrito anteriormente. Na coluna *gap* para o TS, valores zero foram destacados caso alguma versão da nossa heurística não encontrasse um *gap* igual a zero para aquela instância. Além disso, os melhores valores de *gap* (entre nossas heurísticas) foram destacados quando não houve empate.

De acordo com as Tabelas 3.8 e 3.9, os valores médios de $\overline{gap}$ e gap^* para o ILS-VND-D foram iguais a 0,4% e 0,32%, respectivamente, e para a versão ILS-VND-CP, iguais a 0,17% e 0,11%. A variante ILS-VND-CP não apresenta *gaps* acima de 1,3%, enquanto a variante ILS-VND-D apresenta $\overline{gap}$ acima de 3%, por exemplo, para a instância CAB15.4.6. Considerando os casos testados, pode-se observar uma melhoria no *gap* quando se utiliza a alocação gulosa baseada no potencial de cobertura em comparação com a alocação gulosa baseada na distância. Além disso, o ILS-VND-CP obteve valor zero para $\overline{gap}$ e gap^* em 34 e 40, respectivamente, das 48 instâncias testadas, enquanto o ILS-VND-D obteve valor zero para $\overline{gap}$ e gap^* em 24 e 26 instâncias. Portanto, esses testes indicam uma superioridade da alocação baseada em potencial de cobertura em relação a alocação baseada em distância.

O teste de normalidade de Shapiro para o *gap* de ILS-VND-D obteve p-valor = $3,3 \times 10^{-8}$ e o mesmo teste para o tempo de ILS-VND-D obteve p-valor = $1,1 \times 10^{-7}$ indicando que não há normalidade nesses dados. De acordo com o teste estatístico pareado de Wilcoxon, houve evidência estatística de diferença entre os tempos de execução e os *gaps* do TS de Silva & Cunha (2017), ILS-VND-D e ILS-VND-CP. Mais precisamente, com relação aos testes entre *gaps*, o valor mínimo de p-valor foi 0,014 no teste entre Silva & Cunha (2017) e ILS-VND-CP e o valor máximo de p-valor $4,3 \times 10^{-5}$ para o teste entre Silva & Cunha (2017) e ILS-VND-D. Com relação ao tempo, o valor mínimo e máximo de p-valor foi $1,7 \times 10^{-9}$ e $2,97 \times 10^{-9}$,

Tabela 3.8: Resultados das heurísticas e do *solver* CPLEX aplicados ao problema USA ρ HMCP usando instâncias CAB de 10 e 15 nós.

Inst	CPLEX		TS		ILS-VND-D			ILS-VND-CP		
	UB	T[s]	<i>gap</i>	T[s]	$\overline{gap}$	<i>gap</i> *	T[s]	$\overline{gap}$	<i>gap</i> *	T[s]
CAB10.2.4	999026	0,15	0	0,041	0	0	0,002	0	0	0,009
CAB10.2.6	999026	0,16	0	0,041	0	0	0,002	0	0	0,009
CAB10.2.8	999026	0,09	0	0,018	1,35	1,17	0,001	0	0	0,009
CAB10.3.4	990542	0,12	0	0,06	0	0	0,004	0	0	0,001
CAB10.3.6	984530	0,12	0	0,06	0	0	0,008	0	0	0,008
CAB10.3.8	999026	0,11	0	0,06	0	0	0,004	0	0	0,009
CAB10.4.4	991270	0,09	0	0,16	0	0	0,008	0	0	0,01
CAB10.4.6	990542	0,08	0	0,16	0	0	0,006	0	0	0,01
CAB10.4.8	990542	0,07	0	0,12	0	0	0,008	0	0	0,011
CAB10.5.4	992622	0,08	0	0,18	0	0	0,009	0	0	0,013
CAB10.5.6	992622	0,09	0	0,27	0	0	0,008	0	0	0,013
CAB10.5.8	999026	0,06	0	0,18	0,86	0,86	0,007	0,86	0,86	0,011
CAB15.2.4	2364942	0,54	0	0,1	0	0	0,012	0	0	0,022
CAB15.2.6	2364942	0,17	0	0,06	0	0	0,009	0	0	0,017
CAB15.2.8	2364942	0,19	0	0,12	0	0	0,009	1,29	1,29	0,016
CAB15.3.4	2364942	0,29	0	0,39	0	0	0,029	0,19	0	0,078
CAB15.3.6	2364942	0,48	0	0,39	0,17	0,17	0,014	0	0	0,064
CAB15.3.8	2364942	0,25	0	0,39	0	0	0,015	0	0	0,065
CAB15.4.4	2349684	0,18	0	0,67	0,73	0,73	0,02	0	0	0,06
CAB15.4.6	2364942	0,14	0	0,67	3,16	1,92	0,021	1,29	1,29	0,061
CAB15.4.8	2320434	0,15	0	0,69	1,29	1,29	0,034	0	0	0,074
CAB15.5.4	2364942	0,14	0	1,2	0	0	0,033	0	0	0,073
CAB15.5.6	2320434	0,14	0	1,2	1,16	0	0,015	0,75	0	0,085
CAB15.5.8	2320434	0,16	0	1,2	0	0	0,019	0	0	0,079
Média	-	0,17	0	0,35	0,36	0,26	0,012	0,18	0,14	0,034

respectivamente.

Segundo as Tabelas 3.8 e 3.9, os tempos médios obtidos foram iguais a 0,06 e 0,14, para as versões ILS-VND-D e ILS-VND-CP, respectivamente. Portanto, houve um aumento no tempo de execução da heurística ao utilizar a alocação baseada no potencial de cobertura. O tempo médio de execução do [Silva & Cunha \(2017\)](#) é 1,78, o que indica que, em média, o tempo da heurística ILS-VND-D é mais de 20 vezes menor que o da heurística Busca Tabu do [Silva & Cunha \(2017\)](#) e a heurística ILS-VND-CP requer em média 12 vezes menos tempo que a heurística de [Silva & Cunha \(2017\)](#). De acordo com o teste estatístico Wilcoxon pareado, houve evidência estatística de diferença entre os tempos de execução e *gaps* do TS de [Silva & Cunha \(2017\)](#), ILS-VND-D e ILS-VND-CP.

Portanto, considerando as instâncias testadas, percebe-se que a heurística implementada é capaz de encontrar boas soluções em baixos tempos computacionais.

Tabela 3.9: Resultados de heurísticas e do *solver* CPLEX aplicados ao problema USApHMCP usando instâncias CAB de 20 e 25 nós.

Inst	CPLEX		TS		ILS-VND-D			ILS-VND-CP		
	UB	T[s]	<i>gap</i>	T[s]	$\overline{gap}$	<i>gap</i> *	T[s]	$\overline{gap}$	<i>gap</i> *	T[s]
CAB20.2.4	5754594	1,2	0	0,35	0	0	0,033	0,92	0	0,083
CAB20.2.6	5754594	1,17	0	0,32	0	0	0,039	1,21	1,21	0,089
CAB20.2.8	5754594	1,1	0	0,35	0	0	0,018	0	0	0,078
CAB20.3.4	5754594	1,2	0	1,08	0,16	0	0,045	0,06	0	0,095
CAB20.3.6	5754594	1	0	1,14	0,05	0,05	0,061	0	0	0,091
CAB20.3.8	5754594	0,32	0	1,12	0,57	0,57	0,047	0,38	0	0,097
CAB20.4.4	5754594	0,68	0	2,26	1,84	0,91	0,085	0,32	0,32	0,125
CAB20.4.6	5754594	0,68	0	2,22	0,38	0,38	0,079	0	0	0,139
CAB20.4.8	5754594	0,31	0	2,26	0,82	0,82	0,078	0	0	0,148
CAB20.5.4	5754594	0,38	0	3,75	0,69	0,69	0,173	0,57	0	0,293
CAB20.5.6	5710086	0,26	0	3,75	1,45	1,45	0,108	0	0	0,308
CAB20.5.8	5752254	0,31	0	3,69	0,83	0,74	0,11	0	0	0,31
CAB25.2.4	8540006	4,62	0	0,898	0	0	0,034	0	0	0,24
CAB25.2.6	8540006	2,71	0	0,86	0	0	0,034	0	0	0,35
CAB25.2.8	8540006	1,69	0	0,84	0	0	0,038	0	0	0,32
CAB25.3.4	8540006	2,15	0	3,02	0,04	0,04	0,112	0	0	0,33
CAB25.3.6	8536326	1,84	0	2,77	0	0	0,078	0	0	0,31
CAB25.3.8	8536326	1,4	0	2,73	0,1	0,1	0,076	0	0	0,311
CAB25.4.4	8540006	5,91	0	5,49	0,44	0,44	0,19	0	0	0,32
CAB25.4.6	8531438	1,38	0	5,45	0,33	0,33	0,197	0,1	0,1	0,312
CAB25.4.8	8531438	1,2	0	5,49	0,53	0,53	0,184	0,1	0,1	0,321
CAB25.5.4	8540006	2,31	0	9,22	0,71	0,71	0,258	0	0	0,334
CAB25.5.6	8540006	3,34	0	9,18	0,29	0,29	0,258	0	0	0,37
CAB25.5.8	8525878	1,48	0	9,14	1,38	1,38	0,258	0,04	0,04	0,396
Média	-	1,61	0	3,22	0,44	0,39	0,108	0,15	0,07	0,24

3.5.2 Experimentos computacionais para o USAHMCP usando as instâncias CAB

Os resultados para instâncias CAB com 10 e 15 nós e com 20 e 25 nós estão disponíveis nas Tabelas 3.10 e 3.11, respectivamente. Estas tabelas apresentam os resultados retornados pelo *solver* CPLEX e por cada uma das duas versões da heurística proposta. Os melhores valores de *gap* (entre nossas heurísticas) foram destacados em negrito quando não houve empate.

Com base nas Tabelas 3.10 e 3.11, foram obtidas médias gerais de tempo de execução de 0,6, 0,04 e 0,07 para o *solver* CPLEX, ILS-VND-D e ILS-VND-CP, respectivamente. Os valores médios de $\overline{gap}$ foram iguais a 0,67% e 0,41%, respectivamente, para ILS-VND-D e ILS-VND-CP, enquanto as médias de *gap** foram iguais a 0,54% e 0,26%, respectivamente. O teste de normalidade de Shapiro para o *gap* de

Tabela 3.10: Resultados das heurísticas aplicadas ao problema USAHMCP usando instâncias CAB com 10 e 15 nós.

Inst	CPLEX		ILS-VND-D			ILS-VND-CP		
	UB	T[s]	$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
CAB10.2.4	987490	0,12	0	0	0,001	0	0	0,001
CAB10.2.6	999026	0,11	0	0	0,001	0	0	0,01
CAB10.2.8	999026	0,08	3,55	3,55	0,001	2,95	1,45	0,001
CAB10.3.4	999026	0,05	0,86	0,86	0,001	0,86	0,86	0,01
CAB10.3.6	999026	0,08	1,47	1,47	0,001	1,47	1,47	0,01
CAB10.3.8	999026	0,07	0	0	0,001	0	0	0,01
CAB10.4.4	991270	0,11	2,04	2,04	0,01	0,06	0	0,01
CAB10.4.6	990542	0,07	0	0	0,001	0	0	0,02
CAB10.4.8	990542	0,1	0	0	0,01	0	0	0,02
CAB10.5.4	992622	0,1	0	0	0,01	0	0	0,02
CAB10.5.6	992622	0,14	2,8	0,86	0,001	0	0	0,02
CAB10.5.8	999026	0,06	0,86	0,86	0,01	1,48	0,86	0,02
CAB15.2.4	2364942	0,17	0	0	0,01	0	0	0,04
CAB15.2.6	2364942	0,25	0	0	0,01	0	0	0,01
CAB15.2.8	2364942	0,18	0	0	0,01	0	0	0,02
CAB15.3.4	2364942	0,27	0,73	0,73	0,01	0	0	0,03
CAB15.3.6	2364942	0,58	0	0	0,02	0	0	0,03
CAB15.3.8	2320434	0,31	1,05	0	0,01	0,78	0	0,04
CAB15.4.4	2347738	0,23	2,34	0	0,01	2,94	1,95	0,03
CAB15.4.6	2334920	0,26	1,94	1,94	0,03	0,89	0,62	0,04
CAB15.4.8	2320434	0,24	1,94	1,94	0,01	1,94	1,31	0,03
CAB15.5.4	2308898	0,33	0	0	0,02	0	0	0,04
CAB15.5.6	2320434	0,13	3,45	3,45	0,01	1,84	0	0,03
CAB15.5.8	2320434	0,16	0	0	0,02	0	0	0,04
Média	-	0,18	0,96	0,74	0,009	0,63	0,36	0,022

ILS-VND-D obteve $p\text{-valor} = 5,3 \times 10^{-9}$ e o mesmo teste para o tempo de ILS-VND-D obteve $p\text{-valor} = 2,5 \times 10^{-9}$, indicando que não há normalidade nos dados. De acordo com o teste estatístico pareado de Wilcoxon, houve evidência estatística de diferença entre os tempos de execução das duas heurísticas ($p\text{-valor} = 7,8 \times 10^{-9}$), bem como entre os valores de gap alcançados ($p\text{-valor} = 0,007$).

A heurística ILS-VND-D encontrou gap zero em 28 das 40 instâncias, enquanto a heurística ILS-VND-CP encontrou um gap zero em 34 das 40 instâncias ao observar o gap^* . A heurística ILS-VND-D encontrou um gap maior que 3% para 2 instâncias (instâncias CAB10.2.8 e CAB15.5.6), enquanto a heurística ILS-VND-CP possui todos os $gaps$ abaixo de 3% (maior gap na resolução da instância CAB15.4.4). Pode-se observar que a heurística ILS-VND-CP obteve, em média, um gap médio menor que a heurística ILS-VND-D. No entanto, o tempo médio de execução da heurística ILS-VND-CP quase duplica o tempo médio da heurística ILS-VND-D.

Tabela 3.11: Resultados das heurísticas aplicadas ao problema USAHMCP usando instâncias CAB com 20 e 25 nós.

Inst	CPLEX		ILS-VND-D			ILS-VND-CP		
	UB	T[s]	$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
CAB20.2.4	5754594	0,38	0	0	0,02	0	0	0,06
CAB20.2.6	5754594	1,13	0	0	0,02	0,1	0,1	0,03
CAB20.2.8	5754594	0,47	0	0	0,02	0	0	0,04
CAB20.3.4	5754594	0,32	0	0	0,03	0,35	0,35	0,05
CAB20.3.6	5754594	1,04	0,62	0,62	0,04	0,62	0,62	0,09
CAB20.3.8	5754594	0,31	0,57	0,57	0,05	0	0	0,07
CAB20.4.4	5754594	0,78	1,82	1,82	0,05	0	0	0,09
CAB20.4.6	5754594	0,43	0,31	0	0,02	0,52	0,52	0,06
CAB20.4.8	5710086	0,68	0	0	0,03	0	0	0,09
CAB20.5.4	5747720	0,58	1,48	1,18	0,1	1,07	1,07	0,11
CAB20.5.6	5710086	0,4	1,28	1,28	0,04	0,41	0	0,11
CAB20.5.8	5752254	0,32	1,27	1,27	0,05	0,92	0,52	0,12
CAB25.2.4	8540006	0,82	0	0	0,03	0	0	0,06
CAB25.2.6	8540006	2,13	0	0	0,02	0	0	0,06
CAB25.2.8	8540006	1,65	0	0	0,04	0	0	0,08
CAB25.3.4	8540006	1,41	0	0	0,04	0	0	0,09
CAB25.3.6	8540006	1,5	0	0	0,03	0	0	0,09
CAB25.3.8	8540006	1,68	0,14	0,14	0,08	0	0	0,13
CAB25.4.4	8540006	1,73	0	0	0,06	0	0	0,15
CAB25.4.6	8540006	0,75	0,1	0,1	0,11	0	0	0,22
CAB25.4.8	8540006	1,29	0,5	0,5	0,15	0,16	0	0,21
CAB25.5.4	8540006	1,16	0,38	0	0,19	0	0	0,37
CAB25.5.6	8540006	0,93	0	0	0,28	0	0	0,38
CAB25.5.8	8540006	2,54	0,81	0,81	0,15	0,54	0,54	0,28
Média	-	1,02	0,39	0,35	0,07	0,2	0,16	0,13

Pode-se notar também que, em média, as duas heurísticas gastam mais de 10 vezes menos tempo computacional que o CPLEX, o que corrobora a ideia de bom tempo de execução da heurística.

3.6 Experimentos computacionais usando as instâncias DS3

Esta seção apresenta os resultados dos experimentos usando as instâncias do conjunto de dados DS3. Como este é o primeiro estudo que utiliza este conjunto de instâncias para HMCPs e não existem resultados para o problema de centro de p hubs usando estas instâncias, que foram a fonte dos valores de β_{ij} para as outras instâncias, o procedimento descrito na Seção 3.3.1 foi usado para determinar o va-

lor de β_{ij} . Assim, inicialmente, foram realizados testes de sensibilidade variando o valor de β_{ij} para instâncias com 50 nós dos conjuntos DS3 e APn.pL para validar o valor proposto de $\bar{\beta}$ (Seção 3.6.1). Além disso, foram realizados testes para avaliar o desempenho das duas versões da heurística para os problemas USApHMCP (Seção 3.6.2) e USAHMCP (Seção 3.6.3).

3.6.1 Teste de sensibilidade para o parametro β_{ij}

Para o USAHMCP, foram realizados experimentos computacionais utilizando instâncias de 50 nós e $p = 2$ dos conjuntos APL e DS3 (Classe I) para analisar a sensibilidade do problema ao valor do parâmetro β_{ij} . Valores variados de β_{ij} foram testados usando o CPLEX. Assim, a porcentagem de cobertura, tempo de execução do *solver* e *hubs* abertos foram obtidos. Os resultados estão disponíveis na Tabela 3.12, onde Cob[%] apresenta o percentual de cobertura. Cada instância tem seu valor de $\beta_{ij} = \bar{\beta}$, onde o valor $\bar{\beta}$ para a instância AP foi retirado de Silva & Cunha (2017), enquanto para a instância do conjunto DS3 foi obtido com o procedimento proposto na Seção 3.3.1. Os resultados avaliam outros valores para o raio de cobertura β_{ij} com base em frações do valor $\bar{\beta}$.

Observando a Tabela 3.12 pode-se observar que as soluções para ambas as instâncias possuem 100% de cobertura para frações de $\bar{\beta}$ próximas de 1 (β_{ij} próximos de 90% de $\bar{\beta}$). Além disso, o tempo de execução médio da instância DS3 foi de 98,9 segundos, próximo ao tempo de execução médio da instância AP50.2L, que foi de 86,7 segundos. Em outras palavras, a instância com 50 nós para o novo conjunto de dados, usando o valor β_{ij} proposto, parece exigir um esforço computacional semelhante ao da instância AP usando valores β_{ij} retirados da literatura.

É importante observar que o percentual de cobertura referente à solução ótima para a instância AP50.2L já é superior a 90% mesmo quando β_{ij} é igual a 60% de $\bar{\beta}$. Já para a instância do conjunto DS3, a cobertura obtida para o menor valor de β_{ij} testado fica pouco acima de 70% da demanda total. Uma explicação para essa diferença é que, segundo Contreras et al. (2011), as instâncias AP possuem um número muito pequeno de nós para os quais o fluxo de saída é muito maior do que para outros nós. Portanto, espera-se que mesmo valores pequenos de β_{ij} forneçam uma grande porcentagem de cobertura. Vale ressaltar que ambas as instâncias sempre retornaram uma solução ótima com no máximo 2 *hubs*.

Tabela 3.12: Resultados do impacto de β na cobertura, tempo e número de *hubs* usando instâncias DS3 e AP.

DS3					APL				
	β	Cob[%]	T[s]	<i>hubs</i>		β	Cob[%]	T[s]	<i>hubs</i>
$\bar{\beta}=1451,02$	$0,5\bar{\beta}$	54,54	155	2, 47	$\bar{\beta}=65523$	$0,5\bar{\beta}$	86,27	224	15, 24
	$0,6\bar{\beta}$	74,85	346	2, 47		$0,6\bar{\beta}$	93,28	216	15, 24
	$0,7\bar{\beta}$	91	382	3, 7		$0,7\bar{\beta}$	97,3	187	17, 24
	$0,8\bar{\beta}$	99,82	25	7, 40		$0,8\bar{\beta}$	99,25	162,6	16, 31
	$0,9\bar{\beta}$	100	14	37, 50		$0,9\bar{\beta}$	99,91	24,8	10, 15
	$\bar{\beta}$	100	24	2, 7		$\bar{\beta}$	99,99	8,3	9, 14
	$1,1\bar{\beta}$	100	7	4, 50		$1,1\bar{\beta}$	100	12,3	16, 27
	$1,2\bar{\beta}$	100	16	4, 11		$1,2\bar{\beta}$	100	11,9	3, 17
	$1,3\bar{\beta}$	100	11	11, 30		$1,3\bar{\beta}$	100	9	15, 45
	$1,4\bar{\beta}$	100	9	35		$1,4\bar{\beta}$	100	10,9	3, 50

3.6.2 Desempenho das variantes da heurística aplicadas ao USA_pHMCP

As duas variantes da heurística propostas foram aplicadas ao USA_pHMCP utilizando todas as instâncias DS3 (I, II e III). Cada variante foi executada 30 vezes para instâncias com até 150 nós e apenas 1 vez para instâncias com 200 nós ou mais.

Nestes experimentos, o *solver* CPLEX indicou falta de memória na etapa de carregamento para as instâncias com 150 nós ou mais. Para estes casos, o *gap* foi calculado utilizando a cobertura máxima possível (coluna *cobmax*), dada pela demanda total. Os resultados para instâncias com 50 e 100 nós são mostrados na Tabela 3.13. Os resultados para instâncias com 150 e 200 nós são mostrados na Tabela 3.14 e os resultados para instâncias com mais de 200 nós são mostrados na Tabela 3.15. Os melhores valores de *gap* (entre nossas heurísticas) foram destacados quando não houve empate.

As Tabelas 3.13, 3.14 e 3.15 mostram que as duas variantes da heurística proposta encontraram o mesmo valor de função objetivo, exceto para a instância com 500 nós e 4 hubs. Os valores de *gap* foram próximos de zero, indicando que a heurística encontra soluções de boa qualidade para este conjunto de dados. O tempo médio de execução da variante ILS-VND-D foi de 2072,11 segundos, e o tempo médio de execução da variante ILS-VND-CP foi de 2059,45 segundos, indicando que esta última variante foi ligeiramente mais rápida para estas instâncias testadas. O tempo máximo de execução foi de 29876,32 segundos e ocorreu para uma instância com 500 nós e 5 hubs. De acordo com o teste de Shapiro, não há indícios de normalidade nos dados de tempo, pois foi encontrado p-valor igual a $2,2 \times 10^{-14}$ para o tempo de ILS-VND-D. Além disso, o teste estatístico Wilcoxon pareado, indicou não haver evidência estatística de diferença entre os tempos de execução (p-valor= 0,15).

Tabela 3.13: Resultados das variantes da heurística proposta aplicada ao USApHMCP para instâncias do conjunto de dados DS3 com 50 e 100 nós.

Inst	CPLEX		cobmax	ILS-VND-D			ILS-VND-CP		
	FO	T[s]		$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
50i.2	56484,3	5,88	56484,3	0	0	0,21	0	0	0,2
50i.3	56484,3	9,05	56484,3	0	0	0,23	0	0	0,21
50i.4	56484,3	6,75	56484,3	0	0	0,6	0	0	0,58
50i.5	56484,3	3,51	56484,3	0	0	1,09	0	0	1,09
50j.2	723731,11	5,68	723731,11	0	0	0,22	0	0	0,21
50j.3	723731,11	9,13	723731,11	0	0	0,23	0	0	0,22
50j.4	723731,11	7,08	723731,11	0	0	0,6	0	0	0,59
50j.5	723731,11	3,64	723731,11	0	0	1,69	0	0	1,69
50k.2	138175,83	5,63	138175,83	0	0	0,42	0	0	0,4
50k.3	138175,83	9,13	138175,83	0	0	1,13	0	0	1,1
50k.4	138175,83	6,95	138175,83	0	0	2,12	0	0	2,1
50k.5	138175,83	3,49	138175,83	0	0	3,36	0	0	3,36
100i.3	574381,86	2474,14	574382,86	0	0	2,84	0	0	2,69
100i.3	574381,86	3767,2	574382,86	0	0	15,87	0	0	15,32
100i.4	574381,86	1133,19	574382,86	0	0	31,39	0	0	30,32
100i.5	574381,86	214,82	574382,86	0	0	25,48	0	0	24,64
100j.2	4110499,34	1680,28	4110502,5	0,0001	0,0001	2,79	0,0001	0,0001	2,65
100j.3	4110499,34	3276,32	4110502,5	0,0001	0,0001	2,87	0,0001	0,0001	2,72
100j.4	4110499,34	1768,85	4110502,5	0,0001	0,0001	8,14	0,0001	0,0001	7,84
100j.5	4110499,34	213,57	4110502,5	0,0001	0,0001	16,15	0,0001	0,0001	15,06
100k.2	562142,73	4473,36	562142,73	0	0	2,82	0	0	2,7
100k.3	562142,73	2584,63	562142,73	0	0	8,39	0	0	7,85
100k.4	562142,73	1193,77	562142,73	0	0	16,21	0	0	15,14
100k.5	562142,73	197,84	562142,73	0	0	25,12	0	0	24,76
Média	-	960,58	-	0,00002	0,00002	7,08	0,00002	0,00002	6,81

3.6.3 Desempenho das variantes da heurística aplicadas ao USAHMCP

As heurísticas implementadas foram aplicadas para resolver o USAHMCP utilizando todas as instâncias do conjunto DS3 (I, II e III). As variantes heurísticas foram executadas 30 vezes para instâncias com até 150 nós e apenas 1 vez para instâncias com 200 nós ou mais. As Tabelas 3.16, 3.17 e 3.18 apresentam os resultados para instâncias com 50 e 100 nós, 150 e 200 nós e 250 e 500 nós, respectivamente.

Observando as Tabelas 3.16, 3.17 e 3.18 percebe-se que as duas versões da heurística encontraram o mesmo valor de função objetivo. Os valores do gap ficaram muito próximos de 0, indicando que a heurística encontra boas soluções para este conjunto de instâncias. O tempo médio de execução da heurística ILS-VND-D foi de 972,55 segundos, enquanto para a heurística ILS-VND-CP foi de 1656,48 segundos. Portanto, a heurística baseada em distância mostrou-se mais rápida para essas instâncias testadas, com uma diferença de aproximadamente 46506,8 segundos. O tempo de execução máximo foi 16086,24 segundos e ocorreu na instância de 450 nós

Tabela 3.14: Resultados das variantes da heurística proposta aplicada ao USA_pHMCP para instâncias do conjunto de dados DS3 com 150 e 200 nós.

Inst	CPLEX		cobmax	ILS-VND-D			ILS-VND-CP		
	FO	T[s]		$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
150i.2	Mem		1231985,12	0	0	13,51	0	0	12,9
150i.3	Mem		1231985,12	0	0	39,54	0	0	37,83
150i.4	Mem		1231985,12	0	0	157,23	0	0	145,66
150i.5	Mem		1231985,12	0	0	257,47	0	0	238,94
150j.2	Mem		9024214,24	0	0	13,68	0	0	12,75
150j.3	Mem		9024214,24	0	0	39,5	0	0	37,32
150j.4	Mem		9024214,24	0	0	152,86	0	0	145,4
150j.5	Mem		9024214,24	0	0	255,72	0	0	237,89
150k.2	Mem		1263009,38	0,0004	0,0004	14,43	0,0004	0,0004	12,69
150k.3	Mem		1263009,38	0,0004	0,0004	39,74	0,0004	0,0004	37,23
150k.4	Mem		1263009,38	0,0004	0,0004	78,71	0,0004	0,0004	73,13
150k.5	Mem		1263009,38	0,0004	0,0004	126,86	0,0004	0,0004	119,9
200i.2	Mem		3111950,16	0	0	36,56	0	0	37,22
200i.3	Mem		3111950,16	0	0	106,57	0	0	109,21
200i.4	Mem		3111950,16	0	0	209,71	0	0	216,07
200i.5	Mem		3111950,16	0	0	347,01	0	0	354,83
200j.2	Mem		18759696	0,0002	0,0002	71,99	0,0002	0,0002	73,32
200j.3	Mem		18759696	0,0002	0,0002	107,7	0,0002	0,0002	109,1
200j.4	Mem		18759696	0,0002	0,0002	417,22	0,0002	0,0002	429,67
200j.5	Mem		18759696	0,0002	0,0002	689,12	0,0002	0,0002	707,6
200k.2	Mem		2255880,5	0,0001	0,0001	72,09	0,0001	0,0001	73,63
200k.3	Mem		2255880,5	0,0001	0,0001	211,72	0,0001	0,0001	217,42
200k.4	Mem		2255880,5	0,0001	0,0001	418,37	0,0001	0,0001	437,97
200k.5	Mem		2255880,5	0,0001	0,0001	690,18	0,0001	0,0001	703,9
Média	-	-	-	0,0001	0,0001	190,31	0,0001	0,0001	190,9

e $p = 5$. Vale ressaltar que o teste estatístico Wilcoxon pareado mostrou evidências de diferença significativa no tempo entre as duas heurísticas com p-valor igual a $7,9 \times 10^{-5}$ e o teste de normalidade de Shapiro obteve p-valor igual a $7,2 \times 10^{-14}$, indicando não haver normalidade no tempo do ILS-VND-D.

Tabela 3.15: Resultados das variantes da heurística proposta aplicada ao USA_pHMCP para instâncias do conjunto de dados DS3 com 250 e 500 nós.

Inst	CPLEX		cobmax	ILS-VND-D			ILS-VND-CP		
	FO	T[s]		$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
250i.2	Mem		2650822,33	0	0	86,5	0	0	87,78
250i.3	Mem		2650822,33	0	0	255,39	0	0	268,39
250i.4	Mem		2650822,33	0	0	503,27	0	0	515,78
250i.5	Mem		2650822,33	0	0	1671,12	0	0	1703,42
350i.2	Mem		3179554,48	0	0	646,68	0	0	654,87
350i.3	Mem		3179554,48	0	0	1928,42	0	0	1947,93
350i.4	Mem		3179554,48	0	0	3920,06	0	0	3986,96
350i.5	Mem		3179554,48	0	0	6713,55	0	0	6529,13
400i.2	Mem		7502944,07	0	0	590,59	0	0	606,27
400i.3	Mem		7502944,07	0	0	3479,26	0	0	3410,22
400i.4	Mem		7502944,07	0	0	6960,91	0	0	6963,47
400i.5	Mem		7502944,07	0	0	11523,01	0	0	11514,44
450i.2	Mem		9300113	0,0007	0,0007	1862,34	0,0007	0,0007	1852,43
450i.3	Mem		9300113	0,0007	0,0007	5582,87	0,0007	0,0007	5526,1
450i.4	Mem		9300113	0,0007	0,0007	11122,44	0,0007	0,0007	11023,72
450i.5	Mem		9300113	0,0007	0,0007	18343,37	0,0007	0,0007	18274,64
500i.2	Mem		12383120	0,0061	0,0061	3159,33	0,0061	0,0061	3128,57
500i.3	Mem		12383120	0,0061	0,0061	9430,15	0,0061	0,0061	9294,54
500i.4	Mem		12383120	0,0061	0,0061	18510,7	0,0067	0,0067	18358,45
500i.5	Mem		12383120	0,0061	0,0061	29876,32	0,0061	0,0061	29650,31
Média	-	-	-	0,0014	0,0014	6808,31	0,0027	0,0027	6764,87

Tabela 3.16: Resultados da heurística proposta aplicada ao USAHMCP para instâncias do conjunto DS3 com 50 e 100 nós.

Inst	CPLEX		cobmax	ILS-VND-D			ILS-VND-CP		
	FO	T[s]		$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
50i.2	56484,3	34,82	56484,3	0	0	0,428	0	0	0,12
50i.3	56484,3	15,34	56484,3	0	0	0,392	0	0	0,398
50i.4	56484,3	14,39	56484,3	0	0	1,542	0	0	1,312
50i.5	56484,3	9,42	56484,3	0	0	2,496	0	0	2,502
50j.2	723731,11	15,65	723731,11	0	0	0,338	0	0	0,294
50j.3	723731,11	11,76	723731,11	0	0	0,554	0	0	0,548
50j.4	723731,11	5,22	723731,11	0	0	1,148	0	0	2,052
50j.5	723731,11	7,71	723731,11	0	0	1,794	0	0	1,924
50k.2	138175,83	11,02	138175,83	0	0	0,18	0	0	0,17
50k.3	138175,83	6,24	138175,83	0	0	0,23	0	0	0,22
50k.4	138175,83	5,62	138175,83	0	0	0,596	0	0	0,592
50k.5	138175,83	4,58	138175,83	0	0	0,6	0	0	0,596
100i.2	574381,86	2375,55	574382,86	0	0	1,946	0	0	1,858
100i.3	574381,86	4241,84	574382,86	0	0	14,724	0	0	14,196
100i.4	574381,86	229,96	574382,86	0	0	17,67	0	0	28,876
100i.5	574381,86	250,06	574382,86	0	0	31,138	0	0	36,52
100j.2	4110499,34	1965,31	4110502,5	0,0001	0,0001	2,32	0,0001	0,0001	2,26
100j.3	4110499,34	1850,45	4110502,5	0,0001	0,0001	2,852	0,0001	0,0001	2,776
100j.4	4110499,34	221,98	4110502,5	0,0001	0,0001	12,78	0,0001	0,0001	12,766
100j.5	4110499,34	232,34	4110502,5	0,0001	0,0001	23,058	0,0001	0,0001	23,078
100k.2	562142,73	1528,2	562142,73	0	0	2,216	0	0	2,142
100k.3	562142,73	1343,8	562142,73	0	0	2,842	0	0	2,702
100k.4	562142,73	220,17	562142,73	0	0	14,732	0	0	14,36
100k.5	562142,73	206,95	562142,73	0	0	15,432	0	0	15,118
Média	-	617,02	-	0,00002	0,00002	6,33	0,00002	0,00002	6,97

Tabela 3.17: Resultados da heurística proposta aplicada ao USAHMCP para instâncias do conjunto DS3 com 150 e 200 nós.

Inst	CPLEX		cobmax	ILS-VND-D			ILS-VND-CP		
	FO	T[s]		$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
150i.2	Mem		1231985,12	0	0	9,874	0	0	9,602
150i.3	Mem		1231985,12	0	0	34,806	0	0	33,612
150i.4	Mem		1231985,12	0	0	138,994	0	0	126,424
150i.5	Mem		1231985,12	0	0	172,448	0	0	163,072
150j.2	Mem		9024214,24	0	0	7,34	0	0	7,728
150j.3	Mem		9024214,24	0	0	26,562	0	0	27,878
150j.4	Mem		9024214,24	0	0	77,718	0	0	125,682
150j.5	Mem		9024214,24	0	0	148,62	0	0	131,814
150k.2	Mem		1263009,38	0,0004	0,0004	17,236	0,0004	0,0004	18,006
150k.3	Mem		1263009,38	0,0004	0,0004	25,718	0,0004	0,0004	30,984
150k.4	Mem		1263009,38	0,0004	0,0004	72,05	0,0004	0,0004	93,958
150k.5	Mem		1263009,38	0,0004	0,0004	137,684	0,0004	0,0004	162,982
200i.2	Mem		3111950,16	0	0	31,54	0	0	117,46
200i.3	Mem		3111950,16	0	0	101,41	0	0	323,76
200i.4	Mem		3111950,16	0	0	187,47	0	0	613,94
200i.5	Mem		3111950,16	0	0	215,23	0	0	684,74
200j.2	Mem		18759696	0,0002	0,0002	31,36	0,0002	0,0002	115,25
200j.3	Mem		18759696	0,0002	0,0002	101,46	0,0002	0,0002	310,28
200j.4	Mem		18759696	0,0002	0,0002	295,89	0,0002	0,0002	759,85
200j.5	Mem		18759696	0,0002	0,0002	241,81	0,0002	0,0002	772,45
200k.2	Mem		2255880,5	0,0001	0,0001	53,78	0,0001	0,0001	194,11
200k.3	Mem		2255880,5	0,0001	0,0001	59,82	0,0001	0,0001	335,09
200k.4	Mem		2255880,5	0,0001	0,0001	276,66	0,0001	0,0001	992,57
200k.5	Mem		2255880,5	0,0001	0,0001	320,23	0,0001	0,0001	564,85
Média	-	-	-	0,0001	0,0001	116,07	0,0001	0,0001	279,84

Tabela 3.18: Resultados da heurística proposta aplicada ao USAHMCP para instâncias do conjunto DS3 com 250 e 500 nós.

Inst	CPLEX		cobmax	ILS-VND-D			ILS-VND-CP		
	FO	T[s]		$\overline{gap}$	gap^*	T[s]	$\overline{gap}$	gap^*	T[s]
250i.2	Mem		2650822,33	0	0	46,77	0	0	1713,2
250i.3	Mem		2650822,33	0	0	168,46	0	0	3527,06
250i.4	Mem		2650822,33	0	0	367,31	0	0	4942,19
250i.5	Mem		2650822,33	0	0	1644,16	0	0	14115,31
350i.2	Mem		3179554,48	0	0	516,44	0	0	532,14
350i.3	Mem		3179554,48	0	0	1031,12	0	0	1402,36
350i.4	Mem		3179554,48	0	0	3954,27	0	0	3172,43
350i.5	Mem		3179554,48	0	0	3862,41	0	0	3709,5
400i.2	Mem		7502944,07	0	0	601,92	0	0	621,84
400i.3	Mem		7502944,07	0	0	1923,26	0	0	2714,11
400i.4	Mem		7502944,07	0	0	3574,03	0	0	4947,55
400i.5	Mem		7502944,07	0	0	7819,8	0	0	10611,66
450i.2	Mem		9300113	0,0007	0,0007	930,29	0,0007	0,0007	1618,65
450i.3	Mem		9300113	0,0007	0,0007	117,22	0,0007	0,0007	2149,49
450i.4	Mem		9300113	0,0007	0,0007	6650,52	0,0007	0,0007	7308,9
450i.5	Mem		9300113	0,0007	0,0007	10149,76	0,0007	0,0007	16086,24
500i.2	Mem		12383120	0,0061	0,0061	2431,87	0,0061	0,0061	2345,55
500i.3	Mem		12383120	0,0061	0,0061	1603,77	0,0061	0,0061	1551,39
500i.4	Mem		12383120	0,0061	0,0061	7367,64	0,0061	0,0061	8614,69
500i.5	Mem		12383120	0,0061	0,0061	8434,78	0,0061	0,0061	14072,59
Média	-	-	-	0,0014	0,0014	3159,79	0,0014	0,0014	5287,84

3.7 Experimentos computacionais para o USApHMCP usando as instâncias URAND de 1000 nós

O último conjunto de experimentos computacionais deste capítulo foi realizado usando instâncias URAND de 1000 nós. Esses experimentos foram realizados para avaliar as heurísticas ILS-VND-CP e ILS-VND-D para resolver o USApHMCP utilizando instâncias maiores. Foi aplicado apenas para o USApHMCP, pois esta instância não possui custo fixo para instalação de *hubs*. Os resultados dos experimentos computacionais foram comparados com os resultados do Janković et al. (2017). Para os testes foram utilizadas 7 instâncias com o número de nós igual a 1000 e o número de *hubs* sendo 2, 3, 4, 5, 10, 15 e 20. Foram considerados $\chi = 1$, $\delta = 1$ e $\alpha = 0,75$. Os valores de β usados são 75% do valor da função objetivo encontrado no problema de centro de p *hubs* no trabalho de Brimberg et al. (2017) e são os mesmos usados por Janković et al. (2017). Além disso, cada instância foi executada uma vez durante 1800 segundos, que foi o limite de tempo estabelecido por Janković et al. (2017). Não foi possível fazer nenhum ajuste de tempo, pois as informações da máquina disponíveis em Janković et al. (2017) não permitiam comparação de máquinas. Além disso, não foi possível executar o código usado por Janković et al. (2017) em nossa máquina. Os resultados estão disponíveis na Tabela 3.19, na qual foram destacados os maiores valores percentuais de cobertura para cada instância.

Tabela 3.19: Resultados da heurística proposta aplicada ao USApHMCP para instâncias URAND de 1000 nós.

Instância	(Janković et al., 2017)		ILS-VND-CP		ILS-VND-D	
	% Cob	T[s]	%Cob	T[s]	% Cob	T[s]
UR1000.2	91,23	1811,74	92,13	1809,76	91,18	1801,33
UR1000.3	87,87	1801,17	88,21	1800,78	87,9	1800,54
UR1000.4	87,65	1800,19	89,27	1800,88	89,26	1800,42
UR1000.5	85,28	1800,18	88,04	1800,61	87,47	1800,46
UR1000.10	89,94	1801,04	89,06	1802,56	90,04	1800,13
UR1000.15	90,02	1800,13	90,45	1812,3	90,25	1810,52
UR1000.20	90,33	1800,47	92,29	1811,59	91,5	1810,55
Média	88,90	-	89,92	-	89,66	-
Mediana	89,94	-	89,27	-	90,04	-

A Tabela 3.19 mostra que a variante baseada no potencial de cobertura encontra melhores valores percentuais de cobertura em seis das sete instâncias testadas, e a heurística baseada na distância encontra melhores valores percentuais de cobertura em apenas uma das sete instâncias. O maior valor percentual médio de cobertura para essas instâncias foi de 89,92 com a variante ILS-VND-CP. É importante notar que

este trabalho atualiza as melhores soluções conhecidas para essas instâncias URAND de 1000 nós que são as maiores instâncias já testadas para este problema. O teste de Shapiro para os dados de cobertura de Janković et al. (2017), ILS-VND-D e ILS-VND-CP obtiveram p-valor igual a 0,4, 0,57 e 0,24, respectivamente, indicando que estes dados seguem distribuição normal. Além disso, o teste de Bartlett encontrou p-valor igual a 0,78 e indica que há indícios de que as variâncias são iguais, então, neste caso foi aplicado o teste t-Student que encontrou p-valor igual a 0,34 no teste entre as coberturas de Janković et al. (2017) e ILS-VND-CP, encontrou p-valor igual a 0,45 no teste entre as coberturas de Janković et al. (2017) e ILS-VND-D e encontrou p-valor igual a 0,77 no teste entre as coberturas de ILS-VND-D e ILS-VND-CP. Portanto, não há garantias estatísticas de diferenças importantes entre as 3 coberturas.

3.8 Análise e discussão de resultados do USAHMCP

Este capítulo apresentou uma nova versão de HMCP, que difere do USApHMCP por assumir que o número de *hubs* depende de um orçamento para abertura dos *hubs*. Esta versão do problema, denominada USAHMCP, tem o USApHMCP como caso particular.

O capítulo também apresentou um algoritmo heurístico baseado na meta-heurística ILS com busca local utilizando VND para resolver instâncias do problema. Foram apresentadas duas variantes deste algoritmo, diferindo no método de alocação gulosa, que é aplicado em três movimentos. Uma das estratégias de alocação gulosa baseia-se em distâncias mais curtas e a outra baseia-se na alocação com maior potencial de cobertura.

Os experimentos computacionais foram realizados usando quatro conjuntos de instâncias da literatura sobre localização de *hubs*, *Australian Post* (AP), *Civil Aeronautics Board* (CAB), DS3 (proposto em Contreras et al. (2011)) e as instâncias de 1000 nós do URAND.

As variantes do algoritmo proposto, aplicado ao USApHMCP, foram testadas e comparadas com os resultados de Silva & Cunha (2017), para o conjunto de dados AP e CAB; e Janković et al. (2017), para o conjunto de dados URAND. As variantes do algoritmo aplicadas ao USAHMCP tiveram seu desempenho comparado com os resultados do CPLEX.

O capítulo apresentou a avaliação da importância das estruturas de vizinhança e a ordenação dessas estruturas no VND. Os resultados com as instâncias utilizadas mostraram a importância de todas as estruturas de vizinhança ao verificar o fato de que a vizinhança cuja remoção causasse o menor impacto negativo no *gap* implicaria em um aumento significativo no tempo de execução do algoritmo. Os resultados

também indicaram a ordem 1342 como a ordem de exploração mais promissora.

Ao comparar o USA ϕ HMCP e o USAHMCP, percebeu-se que ambos abrem o mesmo número de *hubs* e encontram valores de função objetivo iguais para todas as instâncias do APL, mas com custos de abertura de *hub* diferentes em alguns casos. Entretanto, para instâncias APT, apenas em uma ocasião ambos os problemas abriram o mesmo número de *hubs*. Além disso, para estes casos, o USAHMCP obteve valores de função objetivo mais elevados do que o USA ϕ HMCP, mas gastou mais em custos de abertura de *hubs*. Por outro lado, comparando as heurísticas na resolução das instâncias AP, a média dos *gaps* do ILS-VND-D foi maior que a do ILS-VND-CP e o ILS-VND-CP alcançou *gap* igual a zero mais vezes que o ILS-VND-D, mostrando que as heurísticas encontram boas soluções para estas instâncias.

Para instâncias CAB, considerando USA ϕ HMCP, os resultados dos experimentos computacionais mostraram que as duas variantes do algoritmo proposto apresentam bom desempenho e superam os da literatura em termos de tempo de execução. Vale ressaltar que essa diferença de tempos é mais significativa na resolução de instâncias com maior número de nós. A heurística com alocação baseada no potencial de cobertura mostrou-se superior à heurística com alocação baseada na distância. Os resultados também mostraram que a heurística baseada em potencial de cobertura é pelo menos 6 vezes mais rápida que a heurística da literatura e a heurística baseada em distância é pelo menos 10 vezes mais rápida que a da literatura. A variante heurística baseada no potencial de demanda também apresentou boas soluções, com *gap* zero para 51 das 76 instâncias testadas.

As instâncias DS3 com até 500 nós foram testadas pela primeira vez no problema de cobertura máxima do *hub*. Os testes compararam as duas heurísticas com o CPLEX. Os resultados mostraram bom desempenho da heurística, encontrando *gaps* muito próximos de zero, e indicaram menor tempo de execução para a heurística com alocação baseada em distância para o USAHMCP e menor tempo de execução para a heurística com alocação baseada no potencial de cobertura para o USA ϕ HMCP.

O conjunto de dados URAND de 1000 nós foi aplicado para USA ϕ HMCP. Os testes compararam as duas heurísticas propostas neste capítulo e os resultados do [Janković et al. \(2017\)](#), utilizando um limite de tempo de 1800 segundos. Os resultados atualizaram as melhores soluções conhecidas. Além disso, o ILS-VND-CP encontrou um melhor valor percentual de cobertura em 6 das 7 instâncias testadas, enquanto o ILS-VND-D encontrou um melhor valor percentual de cobertura em 1 das 7 instâncias testadas.

Capítulo 4

Problema de cobertura de *hub* com alocação única e cobertura mínima sob condições de incerteza

Este capítulo apresenta o *Uncapacitated Single Allocation hub Covering Problem with Minimum Coverage* (USAHCPMC) e sua versão robusta, que aborda incerteza sobre o fluxo de demanda por meio de otimização robusta (Seção 4.1). Além disso, apresenta a metodologia heurística utilizada para resolver o problema (Seção 4.2) e expõe os detalhes dos experimentos computacionais (Seção 4.3). Na Seção 4.4, duas instâncias são utilizadas para comparar o USAHCP e o USAHCPMC sem considerar incerteza, enquanto que na Seção 4.5 experimentos foram realizados para analisar as soluções robustas para diferentes valores de nível de incerteza e de robustez. Por fim, a Seção 4.6 expõe o desempenho do algoritmo heurístico implementado para o USAHCPMC e a Seção 4.7 expõe as conclusões deste capítulo.

4.1 Formulação e definição do USAHCPMC

Esta seção apresenta o USAHCPMC (Subseção 4.1.1) e sua versão robusta (Subseção 4.1.2). A formulação matemática é baseada na formulação proposta por [Peker & Kara \(2015\)](#).

4.1.1 Definição do problema e formulação

O USAHCPMC visa determinar em quais nós os *hubs* serão instalados e como os nós não-*hub* serão alocados aos *hubs*, a fim de minimizar o custo de abertura dos mesmos. Assume-se que:

- (i) não há capacidade nos *hubs* ou nos arcos da rede;
- (ii) cada nó deve ser alocado a um único *hub*;
- (iii) há um custo para a abertura de cada *hub*;
- (iv) a demanda total a ser coberta deve ser igual ou superior a um fator limitante;
- (v) há um fator de economia de escala constante, que representa o desconto de transporte nas conexões entre dois *hubs*;
- (vi) as redes de *hubs* estão completas, ou seja, há uma conexão direta entre todos os pares de *hubs*;
- (vii) conexões diretas entre nós não-*hubs* não são permitidas.

A formulação matemática para o USAHCPMC utiliza um fator D que denota um valor mínimo de demanda total que deve ser coberta. Além disso, vários dos outros parâmetros ou notações já foram apresentados no Capítulo 3, a saber,

- (i) $G = (N, A)$ é um grafo com conjunto de nós (N) e arcos (A);
- (ii) W_{ij} é o fluxo do nó de origem até o nó de destino;
- (iii) f_k é o custo de abertura de um *hub* no nó k ;
- (iv) d_{ij} é a distância do nó i até o nó j ;
- (v) α, χ, δ são fatores de desconto, coleta e distribuição, respectivamente;
- (vi) a_{ij}^{km} é o parâmetro de cobertura para cobertura binária;
- (vii) x_{ik} é variável binária de decisão de alocação;
- (viii) z_{ij} é variável não negativa de decisão de cobertura.

Serão considerados dois casos de cobertura: a cobertura binária e a cobertura parcial. O parâmetro β_{ij} será denominado limiar de cobertura total do par origem-destino $i, j \in N$, em que a demanda de i para j é completamente coberta se o custo total da rota que conecta esses dois nós for menor ou igual a β_{ij} . Para o caso de cobertura parcial, para valores de custo total superiores a β_{ij} e inferiores a um limite superior ν_{ij} , haverá cobertura parcial definida com base em uma função de decaimento, denotada por g , cuja imagem está no intervalo $(0, 1)$. A formulação

utiliza o parâmetro de cobertura parcial s_{ij}^{km} proposto por [Peker & Kara \(2015\)](#), que é dado por:

$$s_{ij}^{km} = \begin{cases} 1 & , \text{ se } c_{ij}^{km} \leq \beta_{ij} \\ g(c_{ij}^{km}) & , \text{ se } \beta_{ij} < c_{ij}^{km} \leq \nu_{ij} \\ 0 & , \text{ se } c_{ij}^{km} > \nu_{ij}. \end{cases}$$

Também será usado na formulação o parâmetro auxiliar de limite superior λ_{ij} , dado por:

- $\lambda_{ij} = \max_{km} a_{ij}^{km}, \forall i, j \in N$, no caso de cobertura binária;
- $\lambda_{ij} = \max_{km} s_{ij}^{km}, \forall i, j \in N$, no caso de cobertura parcial.

O modelo para o USAHCPMC, baseado em [Peker & Kara \(2015\)](#), é dado por:

$$\min \sum_{k \in N} f_k x_{kk} \quad (4.1)$$

$$\text{s. a: } \sum_{i \in N} \sum_{j \in N} W_{ij} z_{ij} \geq D \quad (4.2)$$

$$\sum_{k \in N} x_{ik} = 1 \quad \forall i \in N \quad (4.3)$$

$$x_{ik} \leq x_{kk} \quad \forall i, k \in N \quad (4.4)$$

$$z_{ij} \leq \sum_{k \in N} a_{ij}^{km} x_{ik} + \lambda_{ij}(1 - x_{jm}), \quad \forall i, j, m \in N \quad (4.5)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \quad (4.6)$$

$$z_{ij} \geq 0 \quad \forall i, j \in N. \quad (4.7)$$

A função objetivo (4.1), a ser minimizada, representa o custo total com abertura de *hubs*. A restrição (4.2) garante que uma quantidade mínima de demanda seja coberta. As restrições (4.3) impõem que cada nó seja alocado a exatamente um *hub*. As restrições (4.4) garantem que nenhum nó seja alocado a um *hub* que não esteja instalado. As restrições (4.5) determinam o limitante superior para a fração de fluxo entre os nós de origem e destino que é coberta. Finalmente, as expressões (4.6) e (4.7) apresentam o domínio das variáveis.

Vale ressaltar que a mesma formulação pode ser utilizada tanto para cobertura binária quanto para cobertura parcial, em que, para cobertura parcial, basta substituir a_{ij}^{km} por s_{ij}^{km} . Assim, a restrição (4.5) é adaptada para o caso com cobertura

parcial da seguinte forma:

$$z_{ij} \leq \sum_{k \in N} s_{ij}^{km} x_{ik} + \lambda_{ij}(1 - x_{jm}), \forall i, j, m \in N.$$

Além disso, as redes geradas podem variar de acordo com o tipo de cobertura usada, como será mostrado na seção de resultados.

4.1.2 Inserção de incerteza sobre o fluxo de demanda W_{ij}

O modelo robusto a ser apresentado baseia-se na formulação proposta em Bertsimas & Sim (2003), considerando restrições de cardinalidade que assume que nem todos os valores dos parâmetros incertos serão iguais ao pior caso, conforme a Seção 2.2.4. Sejam n a quantidade de nós em N , W_{ij} o valor nominal do fluxo de demanda do nó i até o nó j e $\hat{W}_{ij}$ a maior variação possível do valor real em relação ao valor nominal. A variável incerta $\tilde{W}_{ij}$ denota o fluxo de demanda do nó i até o destino j , de forma que $\tilde{W}_{ij} \in [W_{ij} - \hat{W}_{ij}, W_{ij} + \hat{W}_{ij}]$. Considere também $\Gamma \in [0, n \times n]$ um parâmetro que é responsável pelo número de parâmetros que terão seus valores nominais mudados e controla o nível de robustez do problema. Define-se o conjunto de incertezas

$$\mathcal{U}(\Gamma) = \left\{ \tilde{W} \in \mathbb{R}^{n \times n} : \tilde{W}_{ij} = W_{ij} + s_{ij}\hat{W}_{ij}, s_{ij} \in [-1, 1], \sum_{i \in N} \sum_{j \in N} s_{ij} \leq \Gamma, \forall (i, j) \in N \times N \right\} \quad (4.8)$$

Considerando o modelo matemático nominal proposto, a contraparte robusta pode ser dada por:

$$\min \sum_{k \in N} f_k x_{kk} \quad (4.9)$$

$$\text{s. a: } \sum_{i \in N} \sum_{j \in N} W_{ij} z_{ij} + \min_{S \cup \{t\} \subseteq N \times N: |S| = \lfloor \Gamma \rfloor} \left\{ \sum_{i \in N} \sum_{j \in N} -\hat{W}_{ij} z_{ij} - (\Gamma - \lfloor \Gamma \rfloor) \hat{W}_{t_1 t_2} z_{ij} \right\} \geq D \quad (4.10)$$

$$\sum_{k \in N} x_{ik} = 1 \quad \forall i \in N \quad (4.11)$$

$$x_{ik} \leq x_{kk} \quad \forall i, k \in N \quad (4.12)$$

$$z_{ij} \leq \sum_{k \in N} a_{ij}^{km} x_{ik} + \lambda_{ij}(1 - x_{jm}), \quad \forall i, j, m \in N \quad (4.13)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \quad (4.14)$$

$$z_{ij} \geq 0 \quad \forall i, j \in N, \quad (4.15)$$

em que $\lfloor \Gamma \rfloor$ é a parte inteira de Γ . A minimização interna pode ser dada por

$$\nu(x) = \min_{S \cup \{t\} \subseteq N \times N: |S| = \lfloor \Gamma \rfloor} \left\{ \sum_{i \in N} \sum_{j \in N} -\hat{W}_{ij} z_{ij} - (\Gamma - \lfloor \Gamma \rfloor) \hat{W}_{t_1 t_2} z_{ij} \right\},$$

que pode ser reescrita da seguinte forma:

$$\nu(x) = \min \sum_{i \in N} \sum_{j \in N} -\hat{W}_{ij} z_{ij} s_{ij} \quad (4.16)$$

$$\text{s. a: } \sum_{i \in N} \sum_{j \in N} s_{ij} \leq \Gamma \quad (4.17)$$

$$0 \leq s_{ij} \leq 1 \quad \forall i, j \in N. \quad (4.18)$$

A forma canônica para problemas de minimização relativa ao modelo (4.16)-(4.18) é dada por:

$$\nu(x) = \min \sum_{i \in N} \sum_{j \in N} -\hat{W}_{ij} z_{ij} s_{ij} \quad (4.19)$$

$$\text{s. a: } \sum_{i \in N} \sum_{j \in N} -s_{ij} \geq -\Gamma \quad (4.20)$$

$$-s_{ij} \geq -1 \quad \forall i, j \in N \quad (4.21)$$

$$s_{ij} \geq 0 \quad \forall i, j \in N \quad (4.22)$$

O dual do problema (4.19)-(4.22) é dado por

$$\beta(x) = \max -\Gamma \phi - \sum_{i \in N} \sum_{j \in N} \rho_{ij} \quad (4.23)$$

$$\text{s. a: } -\phi - \rho_{ij} \leq -\hat{W}_{ij} z_{ij} \quad \forall i, j \in N \quad (4.24)$$

$$\phi \geq 0 \text{ e } \rho_{ij} \geq 0 \quad \forall i, j \in N \quad (4.25)$$

Por fim, usando a teoria da dualidade, obtemos o modelo robusto linearizado

$$\min \sum_{k \in N} f_k x_{kk} \quad (4.26)$$

$$\text{s. a: } \sum_{i \in N} \sum_{j \in N} W_{ij} z_{ij} - \Gamma \phi - \sum_{i \in N} \sum_{j \in N} \rho_{ij} \geq D \quad (4.27)$$

$$\sum_{k \in N} x_{ik} = 1 \quad \forall i \in N \quad (4.28)$$

$$x_{ik} \leq x_{kk} \quad \forall i, k \in N \quad (4.29)$$

$$z_{ij} \leq \sum_{k \in N} a_{ij}^{km} x_{ik} + \lambda_{ij}(1 - x_{jm}), \quad \forall i, j, m \in N \quad (4.30)$$

$$\phi + \rho_{ij} \geq \hat{W}_{ij} z_{ij} \quad \forall i, j \in N \quad (4.31)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \quad (4.32)$$

$$z_{ij} \geq 0 \quad \forall i, j \in N \quad (4.33)$$

$$\phi \geq 0 \quad (4.34)$$

$$\rho_{ij} \geq 0 \quad \forall i, j \in N \quad (4.35)$$

Vale ressaltar que o modelo robusto linearizado e o modelo nominal possuem a mesma complexidade computacional e estão na mesma classe de problema de otimização linear inteira mista.

4.2 Algoritmo proposto para o USAHCPMC

O algoritmo proposto para resolver instâncias do USAHCPMC baseia-se na metaheurística ILS combinada com VND. A descrição do método contempla principalmente a representação das soluções, a função de avaliação, o processo de construção da solução inicial, as estruturas de vizinhança, o procedimento guloso de alocação e a busca local baseada em VND e finaliza com os detalhes do algoritmo ILS utilizado neste capítulo.

4.2.1 Representação da solução e procedimento de alocação gulosa

Uma solução s é representada pelo triplo $s = (\bar{p}, y, h)$ da mesma forma que na Seção 3.2.1 e o critério de alocação gulosa dos *hubs* é realizado com base no potencial de cobertura de forma que o nó não *hub* é alocado ao *hub* com maior potencial de cobertura. O procedimento de alocação está detalhado na Seção 3.2.5.

4.2.2 Função de avaliação

A avaliação da solução leva em consideração se a solução é viável ou não. Se não for viável, um valor suficientemente grande, denotado por M , é retornado. Além disso, para verificar a viabilidade de uma solução, deve-se levar em conta que o valor do fluxo de demanda de Γ pares de origem e destino será igual ao seu pior caso e haverá uma variação de uma fração igual a $\Gamma - \lfloor \Gamma \rfloor$ para a cobertura de um único par de origem e destino, sempre considerando o pior caso. No caso da restrição de

cobertura mínima, os piores casos correspondem a valores de demanda coberta mais baixos.

Portanto, seja w um vetor que ordena os elementos da matriz $\hat{W}_{ij}a_{ij}^{y_i y_j}$ em ordem decrescente. Considere também $\bar{\nu}(s)$ como o valor da minimização interna nas restrições (4.10) para uma solução s . Seu valor pode ser dado por:

$$\bar{\nu}(s) = -\left(\sum_{t=1}^{\lfloor \Gamma \rfloor} w_t + w_{\bar{t}}(\Gamma - \lfloor \Gamma \rfloor)\right), \quad (4.36)$$

onde $\bar{t} = \lfloor \Gamma \rfloor + 1$.

Assim, dada uma solução $s = (\bar{p}, h, y)$, a função de avaliação é:

$$f(s) = \begin{cases} \sum_{i=1}^{\bar{p}} f_{h_i}, & \text{se } \sum_{k \in N} \sum_{j \in N} W_{ij}a_{ij}^{y_i y_j} + \bar{\nu}(s) \geq D \\ M, & \text{caso contrário} \end{cases}$$

Note que a ordem de complexidade da função de avaliação é $\mathcal{O}(n^2)$.

4.2.3 Construção da solução inicial

Da mesma forma que no capítulo anterior, a construção da solução inicial utiliza um vetor que ordena em ordem decrescente a razão entre o fluxo e o custo de instalação do *hub*, onde o fluxo associado a um nó é dado pela soma das demandas originadas e das demandas destinadas a esse nó. Assim, para cada nó $i \in N$, a razão r_i dada por

$$r_i = \frac{\sum_{j \in N} (W_{ij} + W_{ji})}{f_i}$$

é calculada.

Seguindo a ordem do vetor ordenado contendo a razão entre o fluxo e o custo de instalação do *hub* e iniciando com um único *hub*, um laço de repetição é iniciado. Nele, é verificado se após a alocação gulosa, aquela solução é factível. Caso seja factível, ela se torna a solução inicial. Caso não seja factível, um novo *hub* é instalado e a nova solução é completada com alocação gulosa. Esse processo se encerra quando uma solução factível é encontrada ou em caso de comprovação da não existência de solução factível. Vale ressaltar que, em último caso, esse processo vai instalar todos os *hubs* da rede e completar a solução com alocação gulosa e se essa última solução não for factível, então não existe solução.

4.2.4 Estruturas de vizinhança

O espaço de soluções é explorado aplicando-se cinco tipos de movimentos, que dão origem a cinco estruturas de vizinhança. Cada estrutura de vizinhança utilizada na heurística é detalhada abaixo:

- (i) vizinhança de realocação ($\hat{\mathcal{N}}$): O conjunto de soluções que podem ser obtidas modificando a alocação de um nó não-*hub*. Assim, um nó não-*hub* alocado a um *hub* é alocado a outro *hub*. O número de vizinhos gerados a partir de uma solução com $\bar{p}$ *hubs* usando essa vizinhança é igual a $(n - \bar{p})(\bar{p} - 1)$. Essa estrutura de vizinhança é denotada distintamente porque é usada apenas como parte de outras estruturas de vizinhança.
- (ii) Vizinhança de fechamento de *hub* ($\mathcal{N}_1$): O conjunto de soluções que podem ser obtidas fechando um *hub*. Portanto, os vizinhos de uma solução com $\bar{p}$ *hubs* têm $\bar{p} - 1$ *hubs*. Além disso, essa estrutura de vizinhança assume que uma alocação gulosa é feita após o fechamento do *hub*, resultando em uma nova solução. Observe que uma solução com $\bar{p}$ *hubs* tem $\bar{p}$ vizinhos nesta vizinhança.
- (iii) Vizinhança de fechamento de *hub* com busca local de realocação ($\mathcal{N}_2$): Conjunto de soluções que podem ser obtidas fechando um *hub* aberto anteriormente combinado com a estrutura de vizinhança $\hat{\mathcal{N}}$, isto é, esta estrutura de vizinhança é uma combinação das estruturas $\mathcal{N}_1$ e $\hat{\mathcal{N}}$. Inicialmente, após o fechamento do *hub*, uma alocação gulosa é realizada para completar a solução e após isso, a vizinhança $\hat{\mathcal{N}}$ é acionada. O número de vizinhos gerados a partir desta vizinhança é igual a $\bar{p}$. No entanto, como uma busca local é realizada na vizinhança $\hat{\mathcal{N}}$ para cada vizinho, a complexidade desta vizinhança é, na verdade, muito maior do que a de $\mathcal{N}_1$.
- (iv) Vizinhança de substituição de *hub* ($\mathcal{N}_3$): Conjunto de soluções que podem ser obtidas pelo fechamento de um *hub* aberto e abertura de um *hub* fechado. Após a substituição, uma alocação gulosa de todos os nós não *hub* para os *hubs* abertos é realizada. O número de vizinhos gerados a partir dessa vizinhança é igual a $\bar{p}(n - \bar{p})$.
- (v) Vizinhança de substituição de *hub* com uma busca local de realocação ($\mathcal{N}_4$): Essa estrutura de vizinhança é uma combinação das estruturas $\mathcal{N}_3$ e $\hat{\mathcal{N}}$. Inicialmente, após a substituição do nó *hub*, uma alocação gulosa é realizada para completar a solução e, em seguida, uma busca local na vizinhança $\hat{\mathcal{N}}$ é ativada. O número de vizinhos gerados a partir dessa vizinhança é igual a $\bar{p}(n - \bar{p})$. No entanto, como

uma busca local é realizada na vizinhança $\hat{\mathcal{N}}$ para cada vizinho, a complexidade dessa vizinhança é, na verdade, muito maior do que a de $\mathcal{N}_3$.

Neste capítulo, a ordem de exploração das vizinhanças é sempre $\mathcal{N}_1$, seguida de $\mathcal{N}_2$, posteriormente $\mathcal{N}_3$ e finalizando com $\mathcal{N}_4$.

4.2.5 Procedimento de busca local para o USAHCPMC

Como este método já foi explorado e não há tantas diferenças com relação ao método utilizado no capítulo anterior, esta seção será breve e tem o objetivo de relembrar os principais tópicos. O procedimento de busca local adotado utiliza o método *Variable Neighborhood Descent* (VND) na versão cíclica descrita por Hansen et al. (2019). O algoritmo inicia com uma solução inicial e uma estrutura de vizinhança, explorando sucessivamente diferentes vizinhanças. Em cada iteração, identifica-se o melhor vizinho da solução corrente: se houver melhoria, a solução é atualizada e o contador de vizinhanças sem melhora é reiniciado; caso contrário, o algoritmo apenas incrementa o contador. O algoritmo passa para a vizinhança seguinte independentemente de melhoria. O processo continua até que todas as quatro estruturas de vizinhança sejam exploradas sem melhorias, resultando em uma solução localmente ótima. O pseudocódigo é apresentado no Algoritmo 6.

Algoritmo 6 VND Cíclico para o USAHCPMC

```

1: Entrada: Solução  $s$ 
2: saída: Solução  $s$ 
3:  $k \leftarrow 1$ 
4:  $count \leftarrow 0$ 
5: while  $count < |\mathcal{N}|$  do
6:    $\bar{s} \leftarrow$  melhor vizinho de  $s$  na vizinhança  $\mathcal{N}_k$ 
7:   if  $f(\bar{s}) < f(s)$  then
8:      $s \leftarrow \bar{s}$ 
9:      $count \leftarrow 0$ 
10:  else
11:     $count \leftarrow count + 1$ 
12:  end if
13:   $k \leftarrow (k \bmod 4) + 1$ 
14: end while
15: Retorne  $s$ .
```

4.2.6 Algoritmo ILS-VND para solução do USAHCPMC

Como já mencionado, o algoritmo ILS é uma meta-heurística composta por duas fases: uma fase de perturbação que tenta evitar que a busca fique presa em ótimos

Algoritmo 7 Procedimento de perturbação para o USAHCPMC

```

1: Entrada: Solução  $s$ 
2: Saída: Solução perturbada  $s'$ 
3:  $s_1 \leftarrow s$ ;
4:  $\bar{p}_{novo} \leftarrow \bar{p} - 1$ ;
5: Escolha aleatoriamente dois hubs distintos:  $h_1, h_2$ ;
6: Escolha aleatoriamente um nó não-hub:  $u$ ;
7: Remova  $h_1$  e  $h_2$  do vetor de hubs de  $s_1$ ;
8: Insira o nó  $u$  como novo hub em  $s_1$ ;
9:  $s' \leftarrow \text{alocação gulosa}(s_1)$ ;
10: Retorne  $s'$ 

```

locais e uma fase de busca local que visa intensificar a busca por soluções melhores. Neste capítulo, a perturbação é realizada trocando aleatoriamente dois nós *hubs* por um nó não *hub*. Observe que o número atual de nós *hubs* diminui uma unidade em relação ao número anterior à perturbação. A metodologia heurística utilizada neste capítulo também emprega o método de *Variable Neighborhood Descent* (VND) como busca local.

O Algoritmo 7 descreve o procedimento de perturbação proposto. A solução corrente é copiada na linha 3, a variável responsável pela quantidade de *hubs* na solução perturbada é uma unidade menor que a quantidade de *hubs* corrente (linha 4). Dois *hubs* escolhidos aleatoriamente (linha 5) são retirados da lista de *hubs* (linha 7), enquanto que um nó não *hub* (escolhido aleatoriamente na linha 6) é inserido na lista de *hubs* na linha 8. Por fim, a solução perturbada é completada com uma alocação gulosa (linha 9).

O Algoritmo 8 descreve o procedimento ILS-VND proposto para o USAHCPMC e sua versão robusta. Ele funciona aplicando uma busca local à solução inicial. A solução obtida pela busca local é perturbada e, em seguida, a solução perturbada é refinada por meio de um procedimento de busca local. Se a nova solução for melhor, ela se torna a solução atual e o valor da função objetivo encontrado é atualizado como o melhor. Esse processo é repetido até que o número máximo de iterações definido pelo parâmetro ILSMAX seja atingido. O algoritmo proposto não reinicia as iterações em caso de melhora.

Algoritmo 8 ILS-VND para o USAHCPMC

```

1: Entrada: ILSMAX
2: Saída:  $s$ 
3:  $s_0 \leftarrow$  Solução inicial; ▷ Seção 4.2.3
4:  $s \leftarrow VND(s_0)$ ; ▷ Algoritmo 6
5:  $iter \leftarrow 1$ ;
6: while  $iter \leq ILSMAX$  do
7:    $s_1 \leftarrow$  perturbação( $s$ ); ▷ Algoritmo 7
8:    $s_2 \leftarrow VND(s_1)$ ; ▷ Algoritmo 6
9:   if  $f(s_2) < f(s)$  then
10:      $f(s) \leftarrow f(s_2)$ 
11:      $s \leftarrow s_2$ 
12:   end if
13:    $iter \leftarrow iter+1$ 
14: end while
15: Retorne  $s$ 

```

4.3 Detalhes dos experimentos computacionais aplicados ao USAHCPMC

Esta seção expõe as instâncias utilizadas nos testes computacionais deste capítulo, parâmetro do algoritmo e o procedimento de cálculo de *gap*.

Os algoritmos foram implementados em linguagem C++ e executados em um *notebook* equipado com processador Intel Core i7-10750H (2.60 GHz), 16 GB de memória RAM e sistema operacional Windows. Os modelos matemáticos foram desenvolvidos com a biblioteca *Concert Technology* e resolvidos utilizando a versão acadêmica 20.1.0.0 do *software* IBM ILOG CPLEX. Os experimentos computacionais utilizaram instâncias AP e CAB (Seção 3.3.1).

Com base em Peker & Kara (2015), o parâmetro de cobertura parcial s_{ij}^{km} é dado por:

$$s_{ij}^{km} = \begin{cases} 1 & , \text{ se } c_{ij}^{km} \leq \beta_{ij} \\ 0,66 & , \text{ se } \beta_{ij} < c_{ij}^{km} \leq 1,05\beta_{ij} \\ 0,33 & , \text{ se } 1,05\beta_{ij} < c_{ij}^{km} \leq 1,10\beta_{ij} \\ 0 & , \text{ se } c_{ij}^{km} > 1,10\beta_{ij}. \end{cases}$$

Para problemas de minimização, como é o problema tratado neste capítulo, o valor de *gap* foi calculado por:

$$gap = 100 \left(\frac{FO_{Heur} - FO_{CPLEX}}{FO_{Heur}} \right)$$

em que FO_{CPLEX} é o valor de função objetivo obtido pelo *solver* CPLEX e FO_{Heur}

é o valor de função objetivo obtido pela heurística.

4.4 Comparação entre o USAHCPMC e o USAHCP

Esta seção compara o já conhecido USAHCP (caso particular do USAHCPMC que exige a cobertura total da rede) com o USAHCPMC, proposto neste capítulo. A comparação consiste em variar o valor da demanda mínima a ser coberta usando o modelo determinístico e analisar os resultados obtidos de tempo e valor de função objetivo. O experimento visa ilustrar e analisar a diferença entre os dois problemas (USAHCP e USAHCPMC) e utiliza duas instâncias: a instância AP40.4L, com 40 nós e $p = 4$, (Subseção 4.4.1) e a instância CAB25.5.8, com 25 nós, $p = 5$ e $\alpha = 0,8$ (Subseção 4.4.2). Seja d a porcentagem da demanda total correspondente a D . Estes testes utilizaram apenas o *solver* CPLEX para resolver o problema nominal, que é o problema sem incertezas nos fluxos de demanda, e consideraram apenas cobertura binária.

4.4.1 Comparação entre USAHCPMC e USAHCP usando a instância APL com 40 nós e $p = 4$

Inicialmente, os testes foram feitos considerando incrementos de 5% sobre d . Foram coletados o valor de função objetivo (FO), o tempo computacional (T) em segundos e os *hubs* instalados. Os resultados estão disponíveis na Tabela 4.1.

Tabela 4.1: Resultados para variações na cobertura mínima em incrementos de 5% da demanda total para o instância AP40.4L usando o CPLEX

$d\%$	5	10	15	20	25	30	35	40	45	50
T(s)	2,5	2,2	2,27	2,53	2,55	2,5	2,83	2,72	2,69	2,55
FO	335,88	335,88	335,88	335,88	335,88	335,88	335,88	335,88	335,88	335,88
<i>hubs</i>	24	24	24	24	24	24	24	24	24	24
$d\%$	55	60	65	70	75	80	85	90	95	100
T(s)	2,61	2,56	2,61	2,59	17,92	31,25	14,85	7,64	44,3	0,89
FO	335,88	335,88	335,88	335,88	411,82	411,82	411,82	411,82	1098,48	6467,47
<i>hubs</i>	24	24	24	24	28	28	28	28	2, 24, 28	3, 8, 11, 24, 33

Note que para cobrir 100% da cobertura total, o custo mínimo com abertura de *hub* é de 6467,47, enquanto que a permissão de que 5% da demanda total não seja necessariamente coberta resulta em uma economia de custos de 83%, já que o custo com abertura de *hub* neste caso seria igual a 1098,48. Além disso, não cobrir 10% da demanda total gera uma economia de 93% sobre o custo de cobrir totalmente a demanda. Isto é, pode ser interessante para o investidor deixar de cobrir 10% da demanda total e economizar 93% com custo com abertura de *hub*. É importante

notar também que, para esta instância, é necessário que se instale apenas um único *hub* para cobrir 90% da demanda total (custo de abertura de *hubs* igual a 411,82). Com relação ao tempo, o CPLEX atingiu os três maiores valores quando d foi 75%, 80% e 95%.

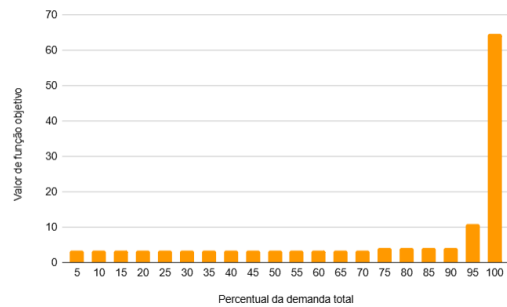
Os próximos resultados foram extraídos de testes considerando incrementos de 1% a partir de $d = 90\%$ e estão disponíveis na Tabela 4.2.

Tabela 4.2: Resultados para variações na cobertura mínima em incrementos de 1% da demanda total para o Instância AP40.4L usando CPLEX

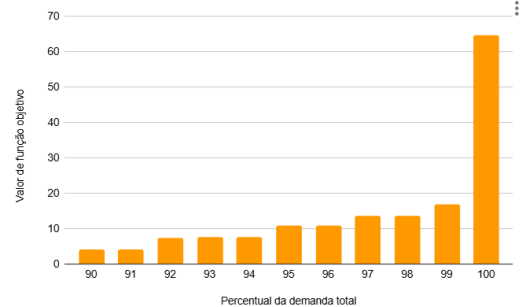
$d\%$	90	91	92	93	94	95	96	97	98	99	100
T(s)	7,64	7,44	25,56	34,98	21,73	44,3	89,5	64,22	34,55	15,28	0,89
FO	411,82	411,82	747,7	762,61	762,61	1098,48	1098,48	1352,68	1352,68	1692,47	6467,47
<i>hubs</i>	28	28	24,28	2,26	2,28	2,28	2,24,28	13	13	13,33	3,8,11,24,33

Note que, permitir que 1% da demanda total não seja coberta, implica em economia de 4775, que equivale a 74% em relação ao custo de se exigir que toda demanda seja coberta. Além disso, pode-se perceber que a configuração de *hubs* da rede se mostra muito sensível à variação de d . Por exemplo, dos 5 *hubs* selecionados pelo CPLEX quando $d = 100\%$, apenas um deles é escolhido pelo CPLEX quando $d = 99\%$. Os três maiores valores de tempo ocorreram para d sendo 96%, 97% e 95%.

Os gráficos das Figuras 4.1 e 4.2 mostram a variação de valor de função objetivo e de tempo para os dois casos considerados nas Tabelas 4.1 e 4.2. Por questões de visualização, os valores de função objetivo foram divididos por 100. A cor laranja representa valores de função objetivo e a cor verde representa valores de tempo.



(a) D variando em incrementos de 5% da demanda total.



(b) D variando em incrementos de 1% da demanda total.

Figura 4.1: Valores de custo mínimo usando CPLEX com a instância AP40.4L.

Observando a Figura 4.1a, pode-se perceber uma grande variação nos valores de função objetivo de $d = 90\%$ para $d = 95\%$ e, de forma mais expressiva, de $d = 95\%$ para $d = 100\%$. Além disso, é notável a variação de valor de função objetivo observada na Figura 4.1b quando d passa de 99% para 100%, demonstrando que exigir cobertura de 100% da demanda pode ser excessivamente custoso. As Figuras

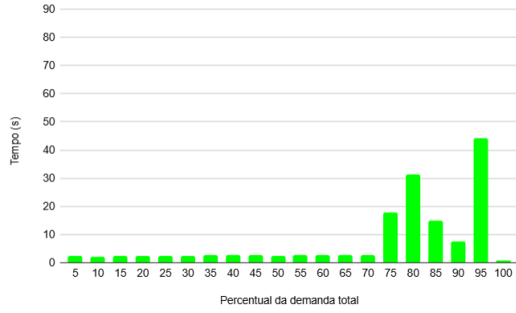
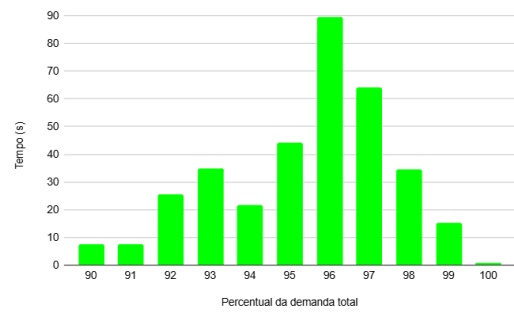
(a) D variando em incrementos de 5% da demanda total.(b) D variando em incrementos de 1% da demanda total.

Figura 4.2: Tempo de execução usando CPLEX para a instância AP40.4L.

4.2a e 4.2b mostram que o tempo de execução do CPLEX quando $d = 100\%$ é o menor de todos, indicando que este novo problema, considerando não ser necessário cobrir toda a demanda, parece ser mais desafiador.

4.4.2 Comparação entre o USAHCPMC e o USAHCP usando a instância CAB com 25 nós, $p = 5$ e $\alpha = 0,8$

Foram feitos testes usando a instância CAB com 25 nós, $p = 5$ e $\alpha = 0,8$, considerando os mesmos valores de d da Subseção 4.4.1. Os resultados para incrementos de 5% de d estão disponíveis na Tabela 4.3.

Tabela 4.3: Resultados para variações na cobertura mínima em incrementos de 5% da demanda total para o Instância CAB25.5.8 usando o CPLEX

$d\%$	5	10	15	20	25	30	35	40	45	50
T(s)	0,38	0,38	0,39	0,39	0,41	0,41	0,38	0,39	0,41	0,39
FO	32554	32554	32554	32554	32554	32554	32554	32554	32554	32554
hubs	25	25	25	25	25	25	25	25	25	25
$d\%$	55	60	65	70	75	80	85	90	95	100
T(s)	0,39	0,41	0,39	0,42	0,72	0,7	1,64	0,98	1,16	0,08
FO	32554	32554	32554	32554	138525	138525	292713	292713	325267	1000202
hubs	25	25	25	25	13	13	13, 19	13, 19	13, 19, 25	13, 19, 22, 23, 25

Note que, para cobrir 100% da cobertura total, o custo mínimo com abertura de *hub* é de 1000202, enquanto que, se for permitido que não seja coberto 5% da demanda total, o custo com abertura de *hub* se torna 325267. Isto gera uma economia de 67% em custos com abertura de *hub*. No caso de $D = 90\%$ da demanda total, seria possível uma economia de 71% sobre o custo de exigir cobrir 100% da demanda. É importante notar também que abrindo os *hubs* 13 e 19 é possível cobrir pelo menos 90% da demanda e é possível cobrir o mínimo de 95% da demanda acrescentando o *hub* 25 dessa instância. Além disso, é necessário que se instale apenas um único *hub* para cobrir 80% da demanda total (custo de abertura de *hubs* igual a 138525). Com

relação ao tempo, o CPLEX atingiu os três maiores valores quando d foi 85%, 95% e 90%.

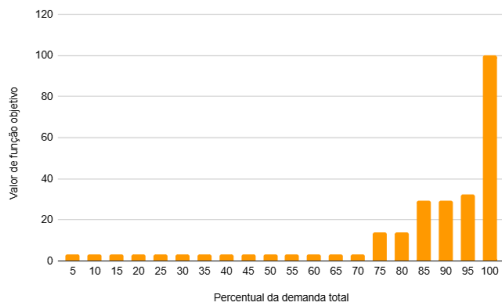
Da mesma forma que na Subseção 4.4.1, foram feitos testes considerando uma variação de d de 1% a partir de $d = 90\%$. Os resultados estão disponíveis na Tabela 4.4.

Tabela 4.4: Resultados para variações na cobertura mínima em incrementos de 1% da demanda total para o Instância CAB25.5.8 usando o CPLEX

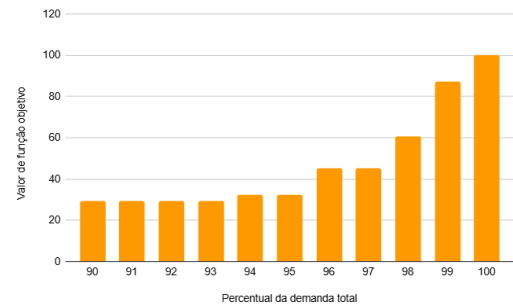
$d\%$	90	91	92	93	94	95	96	97	98	99	100
T(s)	0,98	0,97	1,05	1,44	1,2	1,16	0,95	0,77	0,98	0,84	0,08
FO	292713	292713	292713	292713	325267	325267	453184	453184	607372	871436	1000202
hubs	13,19	13,19	13,19	13,19	13,19,25	13,19,25	8,13,25	8,13,25	8,13,19,25	8,13,19,24,25	13,19,22,23,25

Note que a redução de apenas 1% no nível de cobertura total resulta em uma economia de 13% (abrindo 5 *hubs*), enquanto que reduzir 2% no nível de cobertura total gera uma economia de 39% (abrindo 4 *hubs*). Além disso, dos cinco *hubs* selecionados quando $d = 100\%$, três permanecem quando $d = 99\%$. Observou-se também que os maiores tempos de execução ocorreram para $d = 93\%$, 94% e 95%. Esses resultados demonstram que pequenas flexibilizações no nível de cobertura podem proporcionar economias expressivas, embora possam exigir maior esforço computacional em determinados cenários.

Os gráficos das Figuras 4.3 e 4.4 mostram a variação de valor de função objetivo e de tempo para os dois casos considerados nas Tabelas 4.3 e 4.4. Por questões de visualização, os valores de função objetivo foram divididos por 10000.



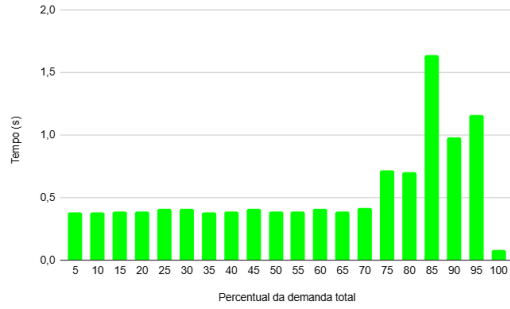
(a) D variando em incrementos de 5% da demanda total.



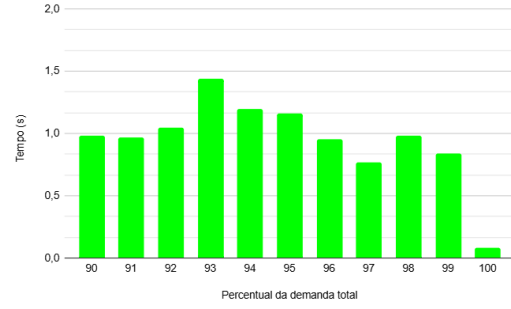
(b) D variando em incrementos de 1% da demanda total.

Figura 4.3: Valores de custo mínimo usando CPLEX com a instância CAB25.5.8.

Observando a Figura 4.3a, nota-se um acréscimo significativo nos valores da função objetivo ao se alterar d de 70% para 75%, de 80% para 85% e de 95% para 100%. Também é possível observar variações expressivas na Figura 4.3b quando d passa de 97% para 98%, de 98% para 99% e de 99% para 100%, essa última de forma menos acentuada do que nos testes com a instância AP40.4L. As Figuras 4.4a e 4.4b



(a) D variando em incrementos de 5% da demanda total.



(b) D variando em incrementos de 1% da demanda total.

Figura 4.4: Tempo de execução usando CPLEX para a instância CAB25.5.8.

evidenciam que o tempo de execução do CPLEX para $d = 100\%$ é o menor entre todos os valores analisados, reforçando que o problema permitindo cobertura inferior a 100% se mostra mais desafiador do ponto de vista computacional. A partir deste ponto, os experimentos considerarão $d = 90\%$.

4.5 Comparação das soluções robustas e nominais para diferentes níveis de robustez e de incerteza

Esta seção tem como objetivo analisar a variação no valor da função objetivo, o tempo de resolução do modelo pelo solver CPLEX e os *hubs* instalados das soluções ao variar o nível de robustez (Γ) e o nível de incerteza ($\hat{W}_{ij}$). Os experimentos foram realizados utilizando as instâncias CAB25.2.6 e AP40.4L. Foram considerados valores de Γ variando de 0 a n^2 e valores de $\hat{W}_{ij}$ variando em 5%, 10% e 15% do valor nominal de W_{ij} . Para representar a variação de Γ , foi utilizado o parâmetro $\gamma \in [0, 1]$, de forma que $\gamma = \Gamma/n^2$. Estes experimentos utilizaram apenas cobertura binária e um valor de D igual a 90% do fluxo de demanda total. Os valores da função objetivo foram destacados em negrito quando um valor diferente do obtido para o Γ anterior foi encontrado.

4.5.1 Resultados para a instância CAB25.2.6

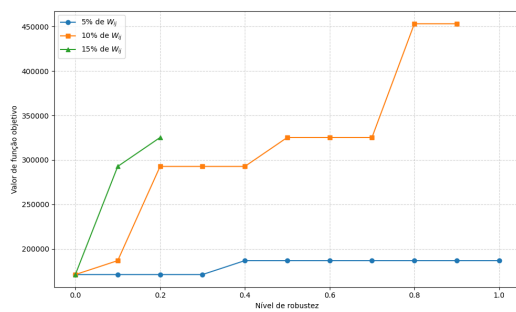
O *solver* CPLEX foi aplicado à instância CAB de 25 nós, $p = 2$ e $\alpha = 0,6$ e foram coletados o valor de função objetivo (FO), tempo (T(s)) e *hubs* instalados. Os resultados estão disponíveis na Tabela 4.5.

De acordo com a Tabela 4.5, para todos os três cenários, com o valor de $\gamma = 0$, o CPLEX encontra o mesmo valor da função objetivo com tempos computacionais semelhantes, como esperado. Para variações de 10% e 15% no fluxo de demanda

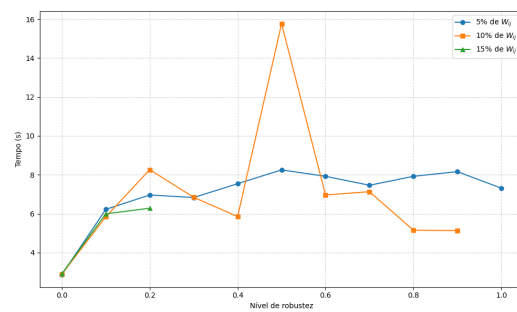
Tabela 4.5: Resultados para diferentes níveis de incerteza (γ) e variações percentuais do fluxo de demanda aplicados à instância CAB 25.2.6

γ	$\hat{W}_{ij}=5\%$ de W_{ij}			$\hat{W}_{ij}=10\%$ de W_{ij}			$\hat{W}_{ij}=15\%$ de W_{ij}		
	FO	T(s)	hubs	FO	T(s)	hubs	FO	T(s)	hubs
0,0	171079	2,87	13,25	171079	2,89	13,25	171079	2,91	13,25
0,1	171079	6,22	13,25	186742	5,84	19,25	292713	5,99	13,19
0,2	171079	6,96	13,25	292713	8,26	13,19	325267	6,28	13,19,25
0,3	171079	6,83	13,25	292713	6,84	13,19	Infactível	X	X
0,4	186742	7,54	19,25	292713	5,85	13,19	Infactível	X	X
0,5	186742	8,25	19,25	325267	15,76	13,19,25	Infactível	X	X
0,6	186742	7,92	19,25	325267	6,96	13,19,25	Infactível	X	X
0,7	186742	7,46	19,25	325267	7,13	13,19,25	Infactível	X	X
0,8	186742	7,92	19,25	453184	5,15	8,13,25	Infactível	X	X
0,9	186742	8,16	19,25	453184	5,13	8,13,25	Infactível	X	X
1,0	186742	7,31	19,25	Infactível	X	X	Infactível	X	X

nominal, os valores da função objetivo alteram de γ igual a 0,0 para 0,1 e de 0,1 para 0,2. O maior custo de abertura do *hub* ocorreu para uma variação de 10% no fluxo de demanda nominal. Para uma variação de 5% no fluxo de demanda nominal, houve apenas uma mudança no valor da função objetivo, e para uma variação de 15% no fluxo de demanda nominal, o problema torna-se inviável a partir de $\gamma = 0,3$. A Figura 4.5 apresenta os valores da função objetivo e o tempo computacional para diferentes valores de nível de robustez e variações no fluxo de demanda.



(a) Valores de função objetivo.



(b) Valores de tempo.

Figura 4.5: Valores da função objetivo e do tempo computacional do *solver* CPLEX para diferentes valores de variação de $\hat{W}_{ij}$ e de γ para uma instância CAB com 25 nós, $p = 2$ e $\alpha = 0,6$.

Analisando a Figura 4.5, observa-se pouca variação para o cenário de 5%, indicando que o modelo não é muito sensível à incerteza neste cenário mais estável. Para o cenário de 10%, o valor da função objetivo cresce mais significativamente, mais que dobrando de $\gamma = 0,0$ para $\gamma = 0,9$, indicando que a robustez tem um impacto maior

neste cenário. Vale ressaltar também que, neste cenário, o tempo computacional é mais imprevisível, com um pico em $\gamma = 0,5$. Para a variação de 15% no fluxo de demanda, há um rápido crescimento no valor da função objetivo, mas o problema torna-se inviável em $\gamma = 0,3$, e os tempos computacionais indicam um crescimento moderado antes da inviabilidade.

A Figura 4.6 mostra as redes que representam a solução ótima para a instância CAB25.2.6 com γ variando de 0,0 a 0,9 e o cenário de variação de 10% no fluxo de demanda. Nesta figura, os nós *hubs* são representados por quadrados e os nós não *hubs* por círculos. Os nós são formatados de forma que quanto maior o tamanho do nó, maior o seu custo de instalação, e quanto mais escuro o nó, maior o seu fluxo de demanda.

De acordo com a Figura 4.6, a solução ótima contém dois *hubs* para γ de 0,0 a 0,4 e 3 *hubs* para γ de 0,5 a 0,9. O problema determinístico instala os *hubs* 13 (com 13 nós alocados a ele) e 25 (com 12 nós alocados a ele), sendo que o nó 25 tem o menor custo de instalação. Quando $\gamma = 0,1$, o nó 19 torna-se um *hub* em vez do nó 13, com a maioria dos nós alocados ao *hub* 25. Para $\gamma = 0,2$, o nó 13 retorna como *hub*, juntamente com o nó 19. De $\gamma = 0,5$ a $\gamma = 0,7$, os mesmos três *hubs* permanecem: nós 13, 19 e 25. Para $\gamma = 0,8$, o nó 8 torna-se um *hub* pela primeira vez, aumentando o custo da rede, o que indica que o nó 8 tende a ser instalado em cenários que exigem alta robustez. Os *hubs* com o maior número de nós alocados são os nós 13 e 25. Vale ressaltar que o nó 17, que possui a maior demanda, não aparece como *hub* em nenhum dos cenários apresentados. Apesar da maior demanda, este nó não pode não ser muito atrativo como *hub* devido ao alto custo de sua ativação. Assim, parece mais conveniente abrir um *hub* mais próximo, que tenha um custo de abertura menor (nó 25).

4.5.2 Resultados para a instância AP40.4L

Os resultados dos experimentos usando a instância APL com 40 nós e $p = 4$ estão disponíveis na Tabela 4.6. Para uma variação de 5% nos fluxos de demanda, o valor da função objetivo sobe de 411,82 ($\gamma = 0,0$) para 747,70 ($\gamma = 0,1$) e então se estabiliza em 762,61 de $\gamma = 0,2$ para $\gamma = 1,0$, com uma variação total de 85%. O tempo computacional não é estável, mesmo neste cenário com baixa variação percentual de $\hat{W}_{ij}$. Para uma variação de 10% nos fluxos de demanda, há um aumento contínuo e acentuado, visto que a variação no valor da função objetivo é superior a 1270% de $\gamma = 0,0$ para $\gamma = 0,9$. Para o cenário de variação de 15%, o problema torna-se inviável para $\gamma = 0,3$, novamente indicando que o problema não suporta alta robustez com alta incerteza. Em relação aos *hubs* instalados, o nó 28 está instalado em todos

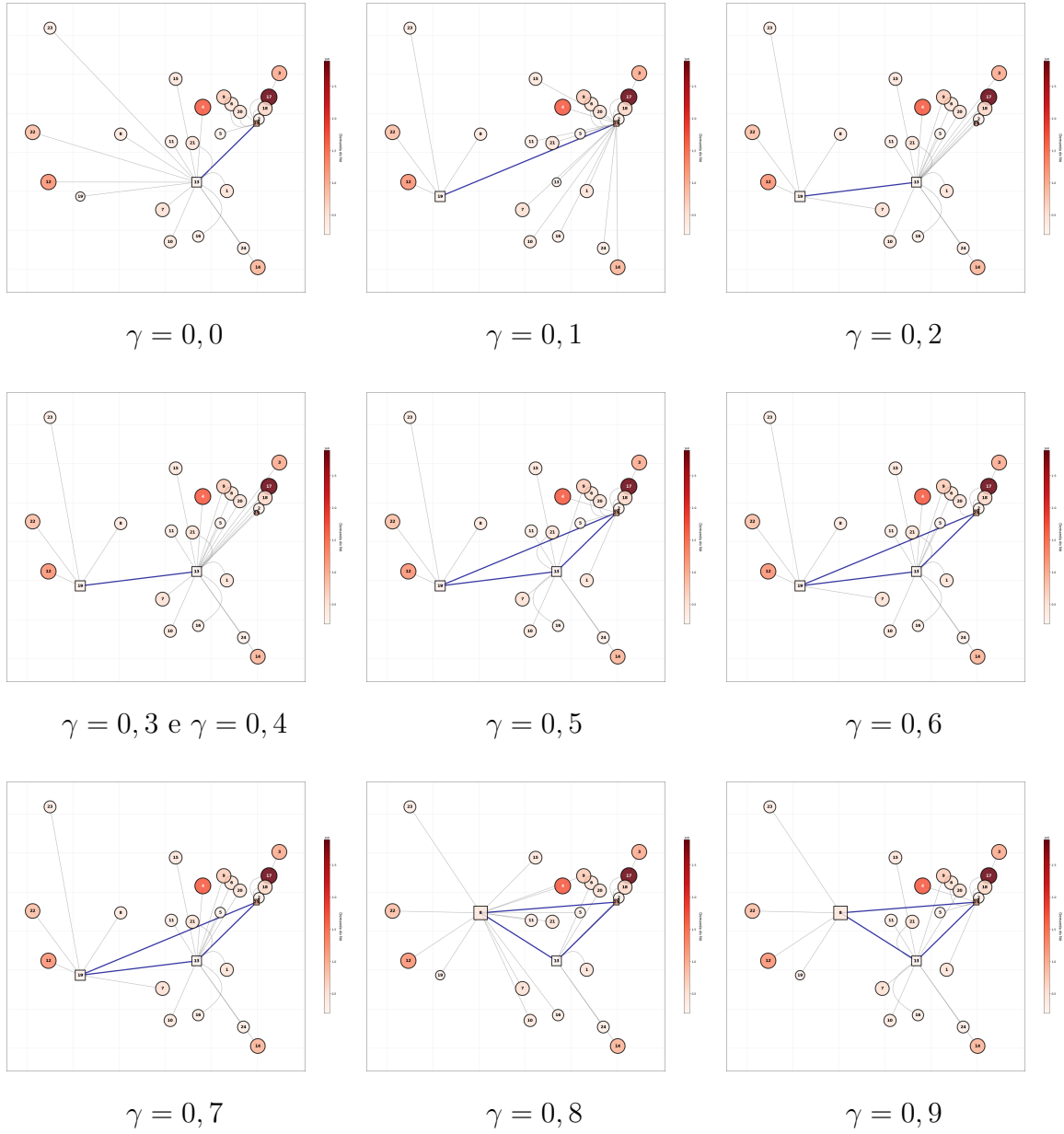


Figura 4.6: Redes relacionadas à solução ótima da instância CAB25.2.6, com γ variando de 0,0 a 0,9, onde o tamanho dos nós representa a amplitude dos custos de abertura e a intensidade da cor representa a amplitude do fluxo de demanda no nó.

os cenários de variação de 5%. Nesse caso, o nó 2 é escolhido como *hub* em quase todos os cenários. No caso de uma variação de 10%, o nó 13 é selecionado como *hub* para $\gamma = 0,3$ a $\gamma = 0,8$, e vale ressaltar que, dos quatro *hubs* selecionados quando $\gamma = 0,9$, apenas o *hub* 33 havia sido selecionado para outros níveis de robustez até então.

Tabela 4.6: Resultados para diferentes níveis de incerteza (γ) e variações percentuais do fluxo de demanda aplicados à instância AP40.4L

γ	$\hat{W}_{ij} = 5\%$ de W_{ij}			$\hat{W}_{ij} = 10\%$ de W_{ij}			$\hat{W}_{ij} = 15\%$ de W_{ij}		
	FO	T(s)	<i>hubs</i>	FO	T(s)	<i>hubs</i>	FO	T(s)	<i>hubs</i>
0,0	411,82	54,80	28	411,82	51,85	28	411,82	62,86	28
0,1	747,70	145,87	24,28	762,61	184,52	2,28	1352,68	213,45	13
0,2	762,61	214,52	2,28	1162,24	287,62	22	2043,26	274,17	2,13,33
0,3	762,61	1298,69	2,28	1352,68	312,51	13	Infactível	X	X
0,4	762,61	1105,74	2,28	1352,68	105,83	13	Infactível	X	X
0,5	762,61	215,84	2,28	1352,68	89,47	13	Infactível	X	X
0,6	762,61	532,21	2,28	1692,47	179,52	13,33	Infactível	X	X
0,7	762,61	1026,87	2,28	2043,26	396,59	2,13,33	Infactível	X	X
0,8	762,61	514,23	2,28	3243,91	248,37	2,13,25,28	Infactível	X	X
0,9	762,61	951,02	2,28	5648,36	384,52	1,8,12,33	Infactível	X	X
1,0	762,61	818,72	2,28	Infactível	X	X	Infactível	X	X

A Figura 4.7 apresenta os valores da função objetivo e o tempo computacional para diferentes níveis de robustez e níveis de incerteza. De acordo com esta figura, também há pouca variação no valor da função objetivo para o cenário de variação de 5% do valor nominal de W_{ij} , mas, com relação ao tempo computacional, observa-se um comportamento mais instável do que o apresentado na seção anterior. Para o cenário de variação de 10% no fluxo de demanda, o valor da função objetivo também cresce acentuadamente, indicando o impacto da robustez. Diferentemente do teste com a instância CAB, não houve um grande pico nos valores de tempo computacional para este cenário. Da mesma forma que no teste anterior, para o cenário de variação de 15% no fluxo de demanda, há um crescimento contínuo no valor da função objetivo até que se atinja a inviabilidade. O mesmo é observado para os valores de tempo de execução. A partir de agora, os testes utilizarão uma variação do fluxo de demanda em 10% do seu valor central.

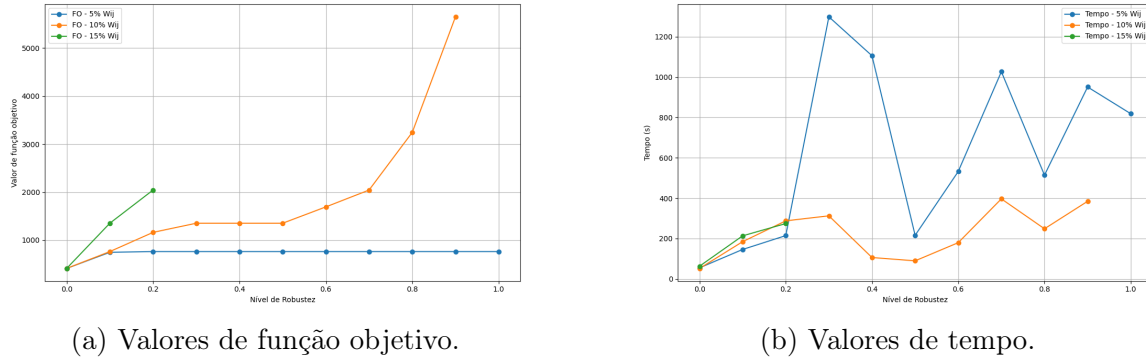


Figura 4.7: Valores da função objetivo e do tempo de execução do *solver* CPLEX para diferentes valores de variação do fluxo de demanda e de γ para a instância APL com 40 nós e $p = 4$.

4.6 Performance do algoritmo heurístico proposto para o USAHCPMC

Esta seção tem como objetivo analisar o desempenho do algoritmo proposto para resolver a versão robusta do problema com cobertura binária e cobertura parcial. Experimentos computacionais foram realizados utilizando as instâncias APL, APT e CAB. A heurística foi executada 30 vezes para instâncias com quantidade de nós inferior a 100. Na apresentação dos resultados, gap^* e $\overline{gap}$ representam, respectivamente, o melhor gap e o gap médio das 30 execuções da heurística. Esses experimentos utilizaram $d = 90\%$, $\hat{W}_{ij} = 10\%$ de W_{ij} e $\gamma \in \{0, 0, 0, 3, 0, 6\}$. Um limite de tempo de 24 horas foi definido para o solver CPLEX.

4.6.1 Resultados para as instâncias AP

Utilizando instâncias AP com até 200 nós, o CPLEX e a heurística ILS-VND foram executados para cobertura parcial e binária. Em 5 casos, o CPLEX não concluiu a resolução em 24 horas. Além disso, para as instâncias de 200 nós, ele interrompeu a execução prematuramente devido à falta de memória. Os testes consideraram p igual a 2, 3, 4 e 5. Um resumo dos resultados separados por número de nó e valor de γ está disponível nas Tabelas 4.7-4.10 para as instâncias APL e APT. Nas Tabelas 4.8 e 4.10 constam informações de instâncias em que não foi possível obter valores de *Lower Bound*. Nesses casos, FO^* indica o melhor dos valores de função objetivo e $\overline{FO}$ denota o valor médio de função objetivo. Nas tabelas, as informações do conjunto de instâncias estão no formato $APLn-\gamma$ e $APLn-p-\gamma$ ou $APLn-p-\gamma$. Os resultados de média das Tabelas 4.7 e 4.9 levam em consideração apenas instâncias em que foi possível obter valores de gap , isto é, desconsideram instâncias sem dados

completos.

Tabela 4.7: Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ usando Instâncias APL.

Instância	Binária					Parcial				
	CPLEX		ILS-VND			CPLEX		ILS-VND		
	FO	T(s)	T(s)	gap*	$\overline{gap}$	FO	T(s)	T(s)	gap*	$\overline{gap}$
APL10-0,0	7781,06	0,11	0,04	0	0	5523,32	0,18	0,03	0	0
APL10-0,3	10008,84	0,12	0,07	0	0	8909,93	0,17	0,05	0	0
APL10-0,6	13467,02	0,13	0,05	0	1,22	11963,29	0,19	0,08	0,61	1,63
APL20-0,0	2794,52	1,51	0,73	0	0	2327,09	1,33	1,11	0	0
APL20-0,3	5302,37	4,14	1,17	0,66	1,22	4510,77	5,34	3,06	0,59	1,22
APL20-0,6	7753,69	4,06	1,76	0,61	1,17	6926,31	7,67	4,13	0	0
APL25-0,0	1470,08	6,60	1,03	0	0,45	1155,04	8,78	1,61	0,61	1,24
APL25-0,3	2002,49	6,87	3,77	0	0	1676,43	7,21	3,55	0	0
APL25-0,6	3792,18	16,95	2,76	0	0	2722,69	17,50	5,8	0,64	1,25
APL40-0,0	396,56	87,75	1,59	0	0	377,58	34,55	1,03	0	0
APL40-0,3	1057,65	325,73	7,62	0,78	1,15	969,95	371,05	25,57	0	1,29
APL40-0,6	1649,68	348,32	57,1	0,94	1,75	1310,53	362,23	24,4	1,54	3,05
APL50-0,0	366,57	11,69	1,24	0	0	366,57	9,61	0,55	0	0
APL50-0,3	400,69	47,49	1,25	0	0	389,32	53,31	0,73	0	0
APL50-0,6	902,59	251,56	24,66	0,38	0,38	717,33	2449,59	25,17	0	0
APL100-0,0	2198,53	419,05	5,17	0	0	2198,53	485,30	5,04	0	0
APL100-0,3	2198,53	4865,17	286,01	0	0	2198,53	4247,40	107,12	0	0,40
APL100-0,6	-	-	-	-	-	-	-	-	-	-
Média	3737,82	376,31	23,3	0,2	0,43	3190,78	474,2	12,3	0,23	0,59

Tabela 4.8: Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ aplicados à instância APL para instâncias em que não foi possível obter o *Lower Bound*.

Instância	Binária					Parcial				
	CPLEX		ILS-VND			CPLEX		ILS-VND		
	FO	T(s)	T(s)	FO*	$\overline{FO}$	FO	T(s)	T(s)	FO*	$\overline{FO}$
APL100_4-0,6	> 24h	> 24h	515,37	4397,06	4397,06	-	-	-	-	-
APL100_5-0,6	> 24h	> 24h	518,03	6623,65	6623,65	> 24h	> 24h	263,29	4397,06	4397,06
APL200-0,0	MEM	MEM	62,35	319,34	319,34	MEM	MEM	56,04	314,90	314,90
APL200-0,3	MEM	MEM	2393,9	496,09	496,09	MEM	MEM	54,12	332,64	332,64
APL200-0,6	MEM	MEM	3437,85	694,21	694,21	MEM	MEM	6718,26	332,64	332,64

Observando as tabelas, nota-se que, em geral, os valores da função objetivo para cobertura parcial são menores do que para cobertura binária, já que isto ocorreu em 26 das 34 observações. Em alguns casos, a diferença percentual entre os valores da função objetivo (parcial e binária) ultrapassou 25%. Para APL25-0,6, a diferença percentual foi de 28,2%. Isso implica que considerar a cobertura parcial pode resultar em um

Tabela 4.9: Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ usando Instâncias APT.

Instância	Binária					Parcial				
	CPLEX		ILS-VND			CPLEX		ILS-VND		
	FO	T(s)	T(s)	gap*	$\overline{gap}$	FO	T(s)	T(s)	gap*	$\overline{gap}$
APT10-0,0	1741,30	0,20	0,04	0	0	1452,96	0,14	0,04	0	0
APT10-0,3	3592,99	0,22	0,09	0	0	2625,86	0,17	0,08	0	0,60
APT10-0,6	4178,37	0,18	0,07	0	1,16	3515,45	0,15	0,14	0	0
APT20-0,0	138,92	0,93	0,21	0	0	78,57	0,86	0,14	0,23	0,85
APT20-0,3	737,18	3,62	0,62	0,11	1,60	571,62	2,30	1,56	0,30	0,79
APT20-0,6	1473,91	3,02	2,26	0,66	2,58	1248,90	2,41	2,13	1,16	1,67
APT25-0,0	59,96	3,36	0,26	0	0	55,46	4,45	0,12	0	0
APT25-0,3	229,59	4,95	1,18	0,94	2,76	176,66	8,71	0,55	0	0
APT25-0,6	793,26	8,53	3,21	1,24	3,47	635,17	13,12	3,68	0,55	0,86
APT40-0,0	14,49	4,08	0,16	0	0	14,49	4,23	0,14	0	0
APT40-0,3	14,49	7,28	0,22	0	0	14,49	7,50	0,14	0	0
APT40-0,6	91,49	32,65	18,59	0	0,44	57,68	74,61	9,88	0	0
APT50-0,0	26,80	23,15	0,38	0	0	26,80	920,24	0,35	0	0
APT50-0,3	51,79	783,72	17,63	0	0,33	39,30	1090,95	5,99	0	0
APT50-0,6	84,50	157,35	29,43	0,77	1,83	67,58	1319,05	14,72	0	0,58
APT100-0,0	-	-	-	-	-	-	-	-	-	-
APT100-0,3	33782,82	2441,56	7,27	0	0	33782,82	3863,16	5,81	0	0
APT100-0,6	33782,82	2247,00	7,41	0	0	33782,82	3484,95	15,20	0	0
Média	4752,63	336,58	5,24	0,22	0,83	4596,86	635,12	3,57	0,13	0,31

Tabela 4.10: Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ aplicados a Instâncias APT para instâncias em que não foi possível obter o *Lower Bound*.

Instância	Binária					Parcial				
	CPLEX		ILS-VND			CPLEX		ILS-VND		
	FO	T(s)	T(s)	FO*	$\overline{FO}$	FO	T(s)	T(s)	FO*	$\overline{FO}$
APT100-0,0	> 24h	> 24h	6,79	33782,82	33782,82	> 24h	> 24h	4,82	33782,82	33782,82
APT200-0,0	MEM	MEM	42,54	22180,09	22180,09	MEM	MEM	63,56	22180,09	22180,09
APT200-0,3	MEM	MEM	81,42	24148,84	24148,84	MEM	MEM	59,97	22578,61	22578,61
APT200-0,6	MEM	MEM	8142,04	41467,22	41467,22	MEM	MEM	7581,44	31440,62	31440,62

custo de abertura de *hubs* muito menor do que ao considerar apenas a cobertura binária.

De acordo com as tabelas, a maioria dos *gaps* é igual a zero. As médias para instâncias APL são de 0,2% e 0,43% para *gap** e $\overline{gap}$, respectivamente, considerando a cobertura binária, e de 0,23% e 0,59%, respectivamente, para cobertura parcial. Para instâncias APT, as médias são de 0,22% e 0,83% para *gap** e $\overline{gap}$, respectivamente, considerando a cobertura binária, e de 0,13% e 0,31%, respectivamente, para

cobertura parcial.

Em relação ao tempo, para as instâncias APL, o tempo médio para resolver o problema com cobertura binária é de 376,31 segundos para o solver CPLEX e 23,3 segundos para a heurística, enquanto para o problema com cobertura parcial, é de 474,2 segundos para o solver CPLEX e 12,3 segundos para a heurística. Os baixos tempos de execução da heurística nas instâncias APL100-0,0 e APL200-0,0 devem-se ao fato de a solução inicial possuir apenas 1 hub e a busca local e o ILS-VND estarem praticamente inativos. Para instâncias APT, o tempo médio para resolver o problema com cobertura binária é de 336,58 segundos para o solver CPLEX e 5,24 segundos para a heurística, enquanto para o problema com cobertura parcial é de 635,12 segundos para o solver CPLEX e 3,57 segundos para a heurística.

Ao comparar os valores da função objetivo e do tempo de execução para diferentes valores de γ , observa-se que o valor da função objetivo aumenta com o aumento de γ . De APT10-0,0 para APT10-0,3, há um aumento no custo de abertura de um *hub* superior a 100%. Tanto para instâncias APL quanto APT, o tempo médio de execução do solver CPLEX, em geral, aumenta quando γ varia de 0,0 a 0,3. Esse mesmo comportamento não se repete quando o valor de γ varia de 0,3 a 0,6. Além disso, o tempo de execução do ILS-VND, em geral, aumenta com o aumento do valor de γ .

4.6.2 Resultados dos testes usando as instancias CAB

Usando as instâncias CAB de até 25 nós, o valor de função objetivo e tempo do CPLEX e da heurística foram coletados tanto para cobertura parcial quanto para cobertura binária. Os testes consideraram 10, 15, 20 e 25 nós com p sendo 2, 3, 4 ou 5 e $\alpha \in \{0,4, 0,6, 0,8\}$. Os resultados estão disponíveis nas Tabelas 4.11 e 4.12 e foram resumidos para a mesma quantidade de nós e de Γ . Os valores de média da Tabela 4.11 desconsideram as instâncias com 15 nós e $\gamma = 0,6$, pois alguns testes com essas instâncias tiveram resultados infactíveis.

Em todos os testes, o valor da função objetivo do *solver* CPLEX para cobertura parcial foi menor do que o valor da função objetivo do *solver* para cobertura binária. A maior diferença percentual entre os valores da função objetivo do CPLEX foi de aproximadamente 20% e ocorreu na instância CAB20-0,6, e a menor diferença percentual entre os valores da função objetivo do CPLEX foi de aproximadamente 2,3% e ocorreu na instância CAB10-0,3.

Em geral, os valores de gap^* são inferiores a 1%, sendo iguais a zero em 5 observações para cobertura binária e em 7 observações para cobertura parcial. Além disso, a heurística ILS-VND é ligeiramente mais rápida, em média, do que o CPLEX para

Tabela 4.11: Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ usando Instâncias CAB

Instância	Binária					Parcial				
	CPLEX		ILS-VND			CPLEX		ILS-VND		
	FO	T(s)	T(s)	gap*	$\overline{gap}$	FO	T(s)	T(s)	gap*	$\overline{gap}$
CAB10-0,0	612378,75	0,17	0,03	0	0,08	519228,92	0,16	0,04	0	0,10
CAB10-0,3	790692,25	0,20	0,05	0,15	0,52	772406,67	0,21	0,05	0	0
CAB10-0,6	1063910,83	0,19	0,04	0	0,37	977362,08	0,19	0,07	0,07	1,15
CAB15-0,0	513904,33	0,61	0,35	0,11	0,39	440968,08	0,57	0,33	0,19	0,40
CAB15-0,3	895100,17	0,83	0,49	0,21	0,93	742260,67	0,77	0,59	0,24	1,15
CAB15-0,6	-	-	-	-	-	-	-	-	-	-
CAB20-0,0	321616,50	2,37	1,07	0	0	303059,92	2,22	0,86	0	0
CAB20-0,3	616519,00	4,26	2,74	0	0,07	554231,08	4,13	2,86	0	0,17
CAB20-0,6	938261,33	4,09	2,95	0,18	1,28	749517,92	3,49	3,46	0	0,10
CAB25-0,0	238973,42	4,00	1,34	0	0,07	210637,33	4,25	1,35	0,20	0,38
CAB25-0,3	452842,67	11,17	5,54	0,47	1,58	374777,58	10,52	6,11	0	0,16
CAB25-0,6	630512,42	10,12	7,15	0,14	0,54	566458,92	10,48	8,81	0,19	1,17
Média	643155,61	3,46	1,98	0,11	0,53	564628,11	3,36	2,23	0,08	0,43

Tabela 4.12: Resultados para o problema com cobertura binária e parcial considerando diferentes valores de γ para instâncias CAB em casos em que não foi possível obter um *Lower Bound*.

Instância	Binária					Parcial				
	CPLEX		ILS-VND			CPLEX		ILS-VND		
	FO	T(s)	T(s)	FO*	$\overline{FO}$	FO	T(s)	T(s)	FO*	$\overline{FO}$
CAB15.5.6-0.6	Infactível	X	X	X	X	-	-	-	-	-
CAB15.5.8-0.6	Infactível	X	X	X	X	-	-	-	-	-

todos os grupos de instâncias, tanto para cobertura parcial quanto para cobertura binária.

Em relação ao tempo, o CPLEX aplicado ao problema com cobertura parcial foi mais rápido em 7 dos 11 casos. O *solver* obteve uma média de 3,46 segundos para cobertura binária e 3,36 segundos para cobertura parcial, porém, o mesmo comportamento não foi observado na heurística, já que a heurística aplicada ao problema com cobertura parcial foi mais rápida que no caso de cobertura binária apenas em 2 dos 11 casos analisados. Além disso, a heurística foi em média mais rápida que o *solver* para todos os grupos de instância CAB. Em média, a heurística executou o algoritmo em 1,98 segundos para cobertura binária e 2,23 segundos para cobertura parcial.

Ainda de acordo com a tabela, observa-se um aumento no custo mínimo de abertura do *hub* à medida que o nível de robustez aumenta. Em relação ao tempo, em

média, há um aumento no tempo de execução do *solver* e da heurística ILS ao comparar a solução do problema nominal ($\gamma = 0$) com a solução do problema robusto ($\gamma > 0$).

4.7 Análise e discussão dos resultados do USAHCPMC

Este capítulo apresentou uma nova versão de problema de cobertura de *hub*, que difere dos problemas HCPs tradicionais da literatura por não exigir que 100% da demanda seja coberta. O novo problema USAHCPMC tem o USAHCP como caso particular. Além disso, o capítulo analisou o impacto de se considerar incerteza nos valores nominais de fluxo do nó de origem até o nó de destino no problema USAHCPMC e tratou tanto cobertura binária, como cobertura parcial. Um algoritmo heurístico ILS-VND foi proposto para tratar principalmente instâncias com grande quantidade de nós.

Os experimentos computacionais realizados utilizaram dois conjuntos de instâncias da literatura sobre localização de *hubs*, *Australian Post* (AP versões L e T) com até 200 nós e *Civil Aeronautics Board* (CAB) com até 25 nós.

Ao comparar o USAHCPMC com o USAHCP, utilizando duas instâncias de teste, observou-se que não exigir cobertura total da demanda ao nível de 5% proporciona uma economia de pelo menos 67% nos custos de instalação de *hubs*. Quando se considera não exigir cobertura total da demanda ao nível de 10%, essa economia aumenta para pelo menos 71%. Além disso, a solução encontrada pelo *solver* CPLEX para o caso em que se cobre no mínimo 90% da demanda resulta em uma rede com no máximo dois *hubs*, o que é uma vantagem em caso de necessidade de modificação futura da rede. Por fim, os resultados de tempo de execução indicam que o CPLEX apresenta maior dificuldade computacional quando a cobertura total da demanda não é imposta, evidenciando maior complexidade na resolução desses cenários.

Foram realizados experimentos para analisar a sensibilidade da solução às faixas de variação dos parâmetros incertos e ao nível de robustez e para isso, duas instâncias foram utilizadas. Supondo variação de 5% do valor nominal de W_{ij} , houve pouca alteração nas soluções robustas geradas pela variação de Γ . Por outro lado, ao considerar variação de 15% do valor nominal de W_{ij} , as soluções já se tornaram infactíveis em $\gamma = 0, 3$, porém os valores de função objetivo não permaneceram iguais em nenhuma variação de Γ .

Em variações de 10% do valor nominal de W_{ij} , as soluções foram factíveis até $\gamma = 0, 9$ e ocorreram resultados inesperados com inserção de *hubs* em cenários de alta incerteza que não aparecem em cenários com γ até 0,7. Ainda na mesma variação, as redes (ao variar os valores de γ) começam com dois *hubs* ($\gamma = 0, 0$ até $\gamma = 0, 4$) e

terminam com três *hubs* ($\gamma = 0,5$ até $\gamma = 0,9$) não sendo necessariamente os mesmos *hubs*. Verificou-se também que, ao variar a robustez, o custo de instalação dos *hubs* tende a aumentar, enquanto que o número de *hubs* abertos geralmente aumenta ou permanece o mesmo que no caso de robustez anterior. Nas duas instâncias testadas, no cenário de $\hat{W}_{ij} = 10\%$ de W_{ij} foi encontrado o maior valor de função objetivo.

Os últimos experimentos computacionais foram realizados para avaliar os resultados da heurística ILS-VND proposta na resolução de instâncias de problemas de cobertura binária e cobertura parcial, considerando diferentes níveis de robustez. Em todos os casos, as soluções com cobertura parcial geraram valores de função objetivo inferiores (ou pelo menos iguais) aos valores de função objetivo das soluções do problema com cobertura binária. Em todos os testes, houve aumento de custo com instalação de *hub* ao aumentar o valor de Γ . Por fim, a heurística ILS-VND implementada encontrou *gap* máximo de 3,47% comprovando conseguir obter boas soluções. Com relação ao tempo, a heurística gastou o tempo máximo de 6263,08 na instância APT200-0,6. Vale destacar o tempo de execução de no máximo sete segundos da heurística para instância APT100-0,0, enquanto o CPLEX não finalizou sua execução em até 24 horas.

Capítulo 5

Considerações finais e direções futuras

Neste capítulo, são discutidas as considerações finais (Seção 5.1) e as possíveis pesquisas que podem surgir como continuidade desta tese (Seção 5.2).

5.1 Considerações finais

Este trabalho apresentou duas novas abordagens para problemas de cobertura de *hub*, ampliando os modelos tradicionais da literatura. Primeiramente, foi introduzido o problema denominado *Uncapacitated Single Allocation Hub Maximal Covering Problem* com orçamento (USAHMCP), que difere do USA p HMCP por assumir que o número de *hubs* a serem abertos depende de um orçamento disponível para instalação, e não de um número pré-fixado de *hubs*. O problema USAHMCP possui o USA p HMCP como um caso particular.

Para resolver o USAHMCP, foi proposto um algoritmo heurístico baseado na meta-heurística *Iterated Local Search* (ILS) com busca local utilizando *Variable Neighborhood Descent* (VND). Foram desenvolvidas duas variantes do algoritmo, que se diferenciam pelo método de alocação gulosa, aplicado em três movimentos de vizinhança. Uma das estratégias de alocação baseia-se nas menores distâncias, enquanto a outra utiliza o potencial de cobertura como critério de decisão. A alocação baseada em potencial de cobertura foi proposta neste trabalho.

Os experimentos computacionais foram realizados utilizando quatro conjuntos clássicos de instâncias da literatura sobre problemas de localização de *hubs*: *Australian Post* (AP), *Civil Aeronautics Board* (CAB), DS3 (proposto em Contreras et al. (2011)) e URAND de 1000 nós. As variantes do algoritmo aplicadas ao USA p HMCP foram comparadas com os resultados do trabalho de Silva & Cunha (2017) (para os

conjuntos AP e CAB) e de Janković et al. (2017) (para o conjunto URAND). Já as variantes aplicadas ao USAHMCP foram comparadas com os resultados do *solver* CPLEX.

O USApHMCP e o USAHMCP apresentaram o mesmo número de *hubs* e valores idênticos de função objetivo nas instâncias APL, diferindo apenas nos custos de instalação. Nas instâncias APT, o USAHMCP obteve valor de função objetivo superior, porém com custos mais altos de instalação de *hubs*. As heurísticas propostas mostraram bom desempenho: o ILS-VND-CP (associado à alocação baseada em potencial de cobertura) superou o ILS-VND-D (associado à alocação baseada em distância), alcançando menores *gaps* e melhor cobertura. Nos conjuntos URAND de 1000 nós, as heurísticas propostas superaram as da literatura em qualidade das soluções, atualizando as melhores soluções conhecidas.

Na segunda etapa deste trabalho, foi proposta uma extensão do problema *Uncapacitated Single Allocation Hub Covering Problem* (USAHCP), o qual foi denominado *Uncapacitated Single Allocation Hub Covering Problem with Minimum Coverage* (USAHCPMC), que difere dos modelos tradicionais por não exigir que 100% da demanda seja necessariamente coberta, tendo o USAHCP como caso particular. O modelo considera cobertura binária e cobertura parcial, além de analisar o impacto de condições de incertezas no parâmetro de fluxo do nó de origem para o nó de destino usando a abordagem robusta de Bertsimas & Sim (2003). Um algoritmo heurístico ILS-VND foi desenvolvido para resolver principalmente instâncias de grande porte.

Os experimentos computacionais para o USAHCPMC utilizaram dois conjuntos de instâncias da literatura, *Australian Post* (AP) e *Civil Aeronautics Board* (CAB). Ao comparar o USAHCPMC com o USAHCP, observou-se que permitir a não cobertura de uma pequena parcela da demanda total pode gerar economias significativas nos custos de instalação de *hubs*, visto que permitir não cobrir 10% da demanda total gerou economia mínima de 71%. Em relação ao tempo computacional, o CPLEX apresentou maior dificuldade para resolver cenários em que a cobertura total não era necessariamente exigida.

Nos testes sob condições de incerteza nos valores de fluxo do nó de origem até o nó de destino, observou-se que a variação de 5% no fluxo de demanda pouco impactou as soluções robustas, enquanto variações de 15% tornaram o problema inviável para um nível de robustez relativamente baixo. Para variações de 10% obtém-se um comportamento mais variado, além de ser possível encontrar soluções viáveis para níveis de robustez relativamente altos.

A heurística ILS-VND implementada encontrou *gaps* geralmente menores que 1%, comprovando que pode obter boas soluções com menos tempo computacional do que o solucionador CPLEX. Vale destacar o tempo máximo de execução de sete segundos

para a heurística em uma instância com 100 nós, para a qual o solucionador CPLEX não conseguiu concluir sua execução em 24 horas.

5.2 Direções futuras

As principais limitações deste trabalho incluem não considerar: (i) restrições de capacidade; (ii) possibilidade de alocação múltipla; (iii) outros custos além do custo de abertura de *hub*.

Por isso, estudos futuros com base nesta tese podem incluir:

- (i) incorporar restrições de capacidade nos *hubs* e/ou nos arcos que os conectam, permitindo analisar situações de congestionamento, por exemplo;
- (ii) desenvolver modelo e algoritmos que permitam alocação múltipla, ampliando a flexibilidade da rede;
- (iii) adotar uma função objetivo que minimize o custo total do sistema, incluindo, por exemplo, custos com transporte.

Outra possibilidade de pesquisa é investigar versões multiobjetivo do problema de cobertura de *hubs*, visando minimizar simultaneamente o custo de abertura e o custo total de transporte, ao mesmo tempo que buscam maximizar a cobertura da rede.

Referências

Alumur, S., & Kara, B. Y. (2009). A hub covering network design problem for cargo applications in turkey. *Journal of the Operational Research Society*, 60, 1349–1359. doi: <https://doi.org/10.1057/jors.2008.92>.

Alumur, S. A., Yaman, H., & Kara, B. Y. (2012). Hierarchical multimodal hub location problem with time-definite deliveries. *Transportation Research Part E: Logistics and Transportation Review*, 48, 1107–1120. URL: <https://www.sciencedirect.com/science/article/pii/S136655451200035X>. doi: <https://doi.org/10.1016/j.tre.2012.04.001>.

Bartlett, M. S. (1937). Properties of sufficiency and statistical tests. *Proceedings of the royal society of london. series a-mathematical and physical sciences*, 160, 268–282.

Ben-Tal, A., & Nemirovski, A. (2000). Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical programming*, 88, 411–424. doi: <https://doi.org/10.1007/PL00011380>.

Bertsimas, D., & Sim, M. (2003). Robust discrete optimization and network flows. *Mathematical programming*, 98, 49–71. doi: <https://doi.org/10.1007/s10107-003-0396-4>.

Bidoki, M. Z., Galankashi, M. R., & Setak, M. (2021). Pricing decision in a multi-period hub location problem under uncertainty: a mathematical model. *International Journal of Value Chain Management*, 12, 370–399. doi: <https://doi.org/10.1504/IJVC.2021.119400>.

Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM computing surveys (CSUR)*, 35, 268–308.

Brimberg, J., Mladenovi, N., Todosijevi, R., & UroÅjevi, D. (2017). A basic variable neighborhood search heuristic for the uncapacitated multiple allocation p-hub center problem. *European Journal of Operational Research*, (p. 313–327). doi: <https://doi.org/10.1007/s11590-015-0973-5>.

- Bryan, D., & O'Kelly, M. (1999). Hub-and-spoke networks in air transportation: An analytical review. *Journal of Regional Science*, *39*, 275–295. doi: <https://doi.org/10.1111/1467-9787.00134>.
- Calik, H., Alumur, S. A., Kara, B. Y., & Karasan, O. E. (2009). A tabu-search based heuristic for the hub covering problem over incomplete hub networks. *Computers & Operations Research*, *36*, 3088–3096. doi: <https://doi.org/10.1016/j.cor.2008.11.023>.
- Campbell, J. F. (1994). Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, *72*, 387–405. URL: <https://www.sciencedirect.com/science/article/abs/pii/0377221794903182>. doi: [https://doi.org/10.1016/0377-2217\(94\)90318-2](https://doi.org/10.1016/0377-2217(94)90318-2).
- Chanta, S., & Sangsawang, O. (2019). A single allocation p-hub maximal covering model for optimizing railway station location. In *Intelligent Computing & Optimization 1* (pp. 522–530). Springer. doi: https://doi.org/10.1007/978-3-030-00979-3_55.
- Chung, S., Myung, Y., & Tcha, D. (1992). Optimal design of a distributed network with a two-level hierarchical structure. *European Journal of Operational Research*, *62*, 105–115. doi: [https://doi.org/10.1016/0377-2217\(92\)90181-8](https://doi.org/10.1016/0377-2217(92)90181-8).
- Contreras, I. A., Cordeau, J.-F., & Laporte, G. (2011). Benders decomposition for large-scale uncapacitated hub location. *Oper. Res.*, *59*, 1477–1490. URL: <https://api.semanticscholar.org/CorpusID:15811273>. doi: <https://doi.org/10.1287/opre.1110.0965>.
- Dantzig, G. B. (1955). Linear programming under uncertainty. *Management science*, *1*, 197–206. doi: <https://doi.org/10.1287/mnsc.1.3-4.197>.
- Daskin, M. S., & Owen, S. H. (1999). Two new location covering problems: The partial p-center problem and the partial set covering problem. *Geographical Analysis*, *31*, 217–235. doi: <https://doi.org/10.1111/j.1538-4632.1999.tb00979.x>.
- Davari, S., Fazel Zarandi, M., & Turksen, I. (2013). The incomplete hub-covering location problem considering imprecise location of demands. *Scientia Iranica*, *20*, 983–991. URL: <https://www.sciencedirect.com/science/article/pii/S1026309813000813>. doi: <https://doi.org/10.1016/j.scient.2013.04.010>.
- de Camargo, R., Miranda, G., & Luna, H. (2008). Benders decomposition for the uncapacitated multiple allocation hub location problem. *Computers & Operations Research*, *35*, 1047–1064. URL: <https://www.sciencedirect.com/science/article/pii/S0305054806001493>. doi: <https://doi.org/10.1016/j.cor.2006.07.002>.

- Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE transactions on systems, man, and cybernetics, part b (cybernetics)*, 26, 29–41. doi: DOI:10.1109/3477.484436.
- Ebrahimi-zade, A., Hosseini-Nasab, H., zare mehrjerdi, Y., & Zahmatkesh, A. (2016). Multi-period hub set covering problems with flexible radius: A modified genetic solution. *Applied Mathematical Modelling*, 40, 2968–2982. URL: <https://www.sciencedirect.com/science/article/pii/S0307904X15005983>. doi: <https://doi.org/10.1016/j.apm.2015.09.064>.
- Ernst, A. T., Jiang, H., Krishnamoorthy, M., & Baatar, D. (2011). Reformulations and computational results for the uncapacitated single allocation hub covering problem. In *Data and Decision Sciences in Action: Proceedings of the Australian Society for Operations Research Conference 2011* (pp. 133–148). Springer. doi: https://doi.org/10.1007/978-3-319-55914-8_10.
- Ernst, A. T., & Krishnamoorthy, M. (1996). Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science*, 4, 139–154. doi: [https://doi.org/10.1016/S0966-8349\(96\)00011-3](https://doi.org/10.1016/S0966-8349(96)00011-3). Hub Location.
- Farahani, R. Z., Hekmatfar, M., Arabani, A. B., & Nikbakhsh, E. (2013). Hub location problems: A review of models, classification, solution techniques, and applications. *Computers & industrial engineering*, 64, 1096–1109. doi: <https://doi.org/10.1016/j.cie.2013.01.012>.
- Gendreau, M., Potvin, J.-Y. et al. (2010). *Handbook of metaheuristics* volume 2. Springer. doi: 10.1007/978-1-4419-1665-5.
- Ghaffari-Nasab, N., Ghazanfari, M., Saboury, A., & Fathollah, M. (2015). The single allocation hub location problem: a robust optimisation approach. *European Journal of Industrial Engineering*, 9, 147–170. doi: <https://doi.org/10.1504/EJIE.2015.068648>.
- Glover, F. (1989). Tabu search—part i. *ORSA Journal on computing*, 1, 190–206.
- Glover, F. W., & Kochenberger, G. A. (2003). *Handbook of metaheuristics* volume 57. Springer Science & Business Media.
- Guo, H., Zhang, L., Ren, Y., Li, Y., Zhou, Z., & Wu, J. (2023). Optimizing a stochastic disassembly line balancing problem with task failure via a hybrid variable neighborhood descent-artificial bee colony algorithm. *International Journal of Production Research*, 61, 2307–2321. doi: <https://doi.org/10.1080/00207543.2022.2069524>.

- Hamacher, H. W., & Meyer, T. (2006). Hub cover and hub center problems. *Technical paper Nr. 98. Department of Mathematics, University of Kaiserslautern, Germany.*, . URL: <https://kluedo.ub.rptu.de/frontdoor/deliver/index/docId/1748/file/Nr.98.pdf>.
- Hansen, P., Mladenović, N., Brimberg, J., & Pérez, J. A. M. (2019). Variable neighborhood search. In M. Gendreau, & J.-Y. Potvin (Eds.), *Handbook of Metaheuristics* (pp. 57–97). Cham: Springer International Publishing. doi: 10.1007/978-3-319-91086-4_3.
- Holland, J. (1975). adaptation in natural and artificial systems, university of michigan press, ann arbor,”. *Cité page, 100*, 33.
- Hwang, Y. H., & Lee, Y. H. (2012). Uncapacitated single allocation p-hub maximal covering problem. *Computers & Industrial Engineering*, 63, 382–389. doi: <http://dx.doi.org/10.1016/j.cie.2012.03.014>.
- Ilić, A., Urošević, D., Brimberg, J., & Mladenović, N. (2010). A general variable neighborhood search for solving the uncapacitated single allocation p-hub median problem. *European Journal of Operational Research*, 206, 289–300. doi: <https://doi.org/10.1016/j.ejor.2010.02.022>.
- Janković, O., Mišković, S., Stanimirović, Z., & Todosijević, R. (2017). Novel formulations and vns-based heuristics for single and multiple allocation p-hub maximal covering problems. *Annals of Operations Research*, 259, 191–216. doi: <https://doi.org/10.1007/s10479-017-2508-1>.
- Janković, O., & Stanimirović, Z. (2017). A general variable neighborhood search for solving the uncapacitated r-allocation p-hub maximal covering problem. *Electronic Notes in Discrete Mathematics*, 58, 23–30. doi: <https://doi.org/10.1016/j.endm.2017.03.004>.
- Kara, B., & Tansel, B. (2003). The single-assignment hub covering problem: Models and linearizations. *Journal of the Operational Research Society*, 54, 59–64. doi: 10.1057/palgrave.jors.2601473.
- Karimi, H., & Bashiri, M. (2011). Hub covering location problems with different coverage types. *Scientia Iranica*, 18, 1571–1578. doi: <https://doi.org/10.1016/j.scient.2011.09.018>.
- Kirkpatrick, S., Gelatt Jr, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science*, 220, 671–680. doi: <https://doi.org/10.1126/science.220.4598.6>.

- Lourenço, H., Martin, O., & Stutzle, T. (2003). Iterated local search. *Handbook of Metaheuristics, Pp.*, (pp. 321–353). doi: 10.1007/0-306-48056-5_11.
- Madani, S. R., Nookabadi, A. S., & Hejazi, S. R. (2018). A bi-objective, reliable single allocation p-hub maximal covering location problem: Mathematical formulation and solution approach. *Journal of Air Transport Management*, 68, 118–136. doi: <https://doi.org/10.1016/j.jairtraman.2017.09.001>.
- Maleki, M., Majlesinasab, N., & Sinha, A. K. (2023). An efficient model for the multiple allocation hub maximal covering problem. *Optimization Methods and Software*, (pp. 1–22). doi: <https://doi.org/10.1080/10556788.2023.2196726>.
- Marianov, V., Serra, D., & ReVelle, C. (1999). Location of hubs in a competitive environment. *European journal of operational research*, 114, 363–371. doi: [https://doi.org/10.1016/S0377-2217\(98\)00195-7](https://doi.org/10.1016/S0377-2217(98)00195-7).
- Meyer, T., Ernst, A. T., & Krishnamoorthy, M. (2009). A 2-phase algorithm for solving the single allocation p-hub center problem. *Computers & Operations Research*, 36, 3143–3151. doi: <https://doi.org/10.1016/j.cor.2008.07.011>.
- Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24, 1097–1100. doi: https://doi.org/10.1007/978-3-319-91086-4_3.
- Mohammadi, M., Torabi, S., & Tavakkoli-Moghaddam, R. (2014). Sustainable hub location under mixed uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, 62, 89–115. URL: <https://www.sciencedirect.com/science/article/pii/S1366554513001993>. doi: <https://doi.org/10.1016/j.tre.2013.12.005>.
- O’Kelly, M. E. (1987). A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32, 393–404. doi: [https://doi.org/10.1016/S0377-2217\(87\)80007-3](https://doi.org/10.1016/S0377-2217(87)80007-3).
- O’Kelly, M. E. (1992). Hub facility location with fixed costs. *Papers in Regional Science*, 71, 293–306. doi: <https://doi.org/10.1007/BF01434269>.
- Peker, M., & Kara, B. Y. (2015). The p-hub maximal covering problem and extensions for gradual decay functions. *Omega*, 54, 158–172. doi: <https://doi.org/10.1016/j.omega.2015.01.009>.
- Qu, B., & Weng, K. (2009). Path relinking approach for multiple allocation hub maximal covering problem. *Computers & Mathematics with Applications*, 57, 1890–1894. doi: <https://doi.org/10.1016/j.camwa.2008.10.004>.

- Sadeghi, M., Jolai, F., Tavakkoli-Moghaddam, R., & Rahimi, Y. (2015). A new stochastic approach for a reliable p-hub covering location problem. *Computers & Industrial Engineering*, 90, 371–380. URL: <https://www.sciencedirect.com/science/article/pii/S0360835215003964>. doi: <https://doi.org/10.1016/j.cie.2015.10.003>.
- Shang, X., Yang, K., Luo, Q., & and, B. M. (2025). Robust optimization models for the hierarchical cost-minimizing hub covering problem in the urban agglomeration comprehensive transport system. *International Journal of General Systems*, 0, 1–35. URL: <https://doi.org/10.1080/03081079.2025.2484079>. doi: 10.1080/03081079.2025.2484079. arXiv:<https://doi.org/10.1080/03081079.2025.2484079>.
- Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52, 591–611.
- Silva, M. R., & Cunha, C. B. (2017). A tabu search heuristic for the uncapacitated single allocation p -hub maximal covering problem. *European Journal of Operational Research*, 262, 954–965. doi: <https://doi.org/10.1016/j.ejor.2017.03.066>.
- Soyster, A. L. (1973). Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations research*, 21, 1154–1157. doi: <https://doi.org/10.1287/opre.21.5.1154>.
- Storn, R., & Price, K. (1997). Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11, 341–359. doi: 10.1023/A:1008202821328.
- Vincent, F. Y., Anh, P. T., Gunawan, A., & Han, H. (2024). A simulated annealing with variable neighborhood descent approach for the heterogeneous fleet vehicle routing problem with multiple forward/reverse cross-docks. *Expert Systems with Applications*, 237, 121631. doi: <https://doi.org/10.1016/j.eswa.2023.121631>.
- Vukićević, M., Ratli, M., Rivenq, A., & Zrikem, M. (2023). Covering delivery problem with electric vehicle and parcel lockers: Variable neighborhood search approach. *Computers & Operations Research*, 157, 106263. doi: <https://doi.org/10.1016/j.cor.2023.106263>.
- Wagner, B. (2008). Model formulations for hub covering problems. *Journal of the Operational Research Society*, 59, 932–938. doi: <https://doi.org/10.1057/palgrave.jors.2602424>.
- Wang, J., & Qin, Z. (2020). Chance constrained programming models for uncertain hub covering location problems. *Soft Computing*, 24, 2781–2791. URL: <https://doi.org/10.1007/s00182-020-00900-0>.

//api.semanticscholar.org/CorpusID:209096632. doi: <https://doi.org/10.1007/s00500-019-04476-4>.

Welch, B. L. (1947). The generalization of ‘student’s’ problem when several different population variances are involved. *Biometrika*, *34*, 28–35.

Weng, K.-r., Yang, C., & Ma, Y.-f. (2006). Two artificial intelligence heuristics in solving multiple allocation hub maximal covering problem. In *Intelligent Computing: International Conference on Intelligent Computing, ICIC 2006, Kunming, China, August 16-19, 2006. Proceedings, Part I 2* (pp. 737–744). Springer. doi: https://doi.org/10.1007/11816157_90.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin*, *1*, 80–83.

Yaman, H. (2011). Allocation strategies in hub networks. *European Journal of Operational Research*, *211*, 442–451. doi: <https://doi.org/10.1016/j.ejor.2011.01.014>.