

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

RAFAEL MARANHÃO REGO PRAXEDES

UMA ABORDAGEM EXATA UNIFICADA PARA UMA CLASSE DE
PROBLEMAS DE ROTEAMENTO DE VEÍCULOS COM COLETA E ENTREGA
SIMULTÂNEAS

João Pessoa

2022

RAFAEL MARANHÃO REGO PRAXEDES

UMA ABORDAGEM EXATA UNIFICADA PARA UMA CLASSE DE
PROBLEMAS DE ROTEAMENTO DE VEÍCULOS COM COLETA E ENTREGA
SIMULTÂNEAS

Dissertação submetida ao Programa de Pós-Graduação em Informática do Centro de Informática da Universidade Federal da Paraíba, como requisito parcial para obtenção do Grau de Mestre em Informática.

Orientador: Prof. Dr. Anand Subramanian

Co-orientador: Prof. Dr. Teobaldo Leite Bulhões Júnior

João Pessoa

2022

Catálogo na publicação
Seção de Catalogação e Classificação

P919a Praxedes, Rafael Maranhão Rego.

Uma abordagem exata unificada para uma classe de problemas de roteamento de veículos com coleta e entrega simultâneas / Rafael Maranhão Rego Praxedes. - João Pessoa, 2022.

111 f. : il.

Orientação: Anand Subramanian.

Coorientação: Teobaldo Leite Bulhões Júnior.

Dissertação (Mestrado) - UFPB/CI.

1. Roteamento de veículos. 2. Coleta e entrega simultâneas. 3. Geração de colunas. 4. Planos de corte. I. Subramanian, Anand. II. Bulhões Júnior, Teobaldo Leite. III. Título.

UFPB/BC

CDU 711.7:629.35(043)



UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA



Ata da Sessão Pública de Defesa de Dissertação de Mestrado de Rafael Maranhão Rego Praxedes, candidato ao título de Mestre em Informática na Área de Sistemas de Computação, realizada em 19 de agosto de 2022.

Aos dezenove dias do mês de agosto, do ano de dois mil e vinte e dois, às quinze horas, no Centro de Informática da Universidade Federal da Paraíba, em Mangabeira, reuniram-se os membros da Banca Examinadora constituída para julgar o Trabalho Final do Sr. Rafael Maranhão Rego Praxedes, vinculado a esta Universidade sob a matrícula nº 20201019130, candidato ao grau de Mestre em Informática, na área de “Sistemas de Computação”, na linha de pesquisa “Computação Distribuída”, do Programa de Pós-Graduação em Informática, da Universidade Federal da Paraíba. A comissão examinadora foi composta pelos professores: Anand Subramanian (PPGI-UFPB), Orientador e Presidente da Banca, Bruno Petrato Bruck (PPGI-UFPB), Examinador Interno, Teobaldo Leite Bulhões Junior (PPGI-UFPB), Coorientador, Eduardo Uchoa Barboza (UFF), Examinador Externo à Instituição. Dando início aos trabalhos, o Presidente da Banca cumprimentou os presentes, comunicou a finalidade da reunião e passou a palavra ao candidato para que ele fizesse a exposição oral do trabalho de dissertação intitulado “Uma Abordagem Exata Unificada para uma Classe de Problemas de Roteamento de Veículos com Coleta e Entrega Simultâneas”. Concluída a exposição, o candidato foi arguido pela Banca Examinadora que emitiu o seguinte parecer: “**aprovado**”. Do ocorrido, eu, Fernando Menezes Matos, Coordenador do Programa de Pós-Graduação em Informática, lavrei a presente ata que vai assinada por mim e pelos membros da banca examinadora. João Pessoa, 19 de agosto de 2022.

Prof. Fernando Menezes Matos

Prof. Anand Subramanian
Orientador (PPGI-UFPB)

Prof. Teobaldo Leite Bulhões Junior
Coorientador (PPGI-UFPB)

Prof. Bruno Petrato Bruck
Examinador Interno (PPGI-UFPB)

Prof. Eduardo Uchoa Barboza
Examinador Externo (UFF)

“You don’t have to be great to start, but you have to start to be great.”

Zig Ziglar

AGRADECIMENTOS

Gostaria de agradecer:

- A Deus, junto à intercessão de Nossa Senhora, pela oportunidade concedida de realizar este mestrado, bem como por me acompanharem em todos os instantes da minha vida.
- Aos meus pais, Marcelo Praxedes e Christiane Praxedes, e irmãos, Matheus Praxedes e Miguel Praxedes, por não medirem esforços para me proporcionar todas as condições e recursos necessários que me fizeram alcançar este resultado.
- À minha noiva, Ana Luiza Crispim Araújo, por estar ao meu lado em todos os momentos, sejam fáceis ou difíceis, incentivando-me a prosseguir e sendo sempre a melhor companheira que eu poderia ter.
- Às minhas cunhadas Héllen Uchoa e Mariana Crispim pelo apoio.
- Aos meus familiares, por estarem ao meu lado, bem como aos familiares da minha noiva, os quais também considero como pertencentes à minha família. Dentre eles, destaco sua avó, Marinete Crispim, por ter me colocado sempre em suas orações.
- Aos meus orientadores, Prof. Dr. Anand Subramanian e Prof. Dr. Teobaldo Bulhões, por todos os ensinamentos, paciência, dedicação e confiança no meu trabalho, além de terem acreditado no meu potencial. Certamente são amigos dos quais lembrarei ao longo da minha vida.
- Ao Prof. Dr. Bruno Bruck e ao Prof. Dr. Eduardo Uchoa por terem me concedido a honra de suas participações na banca de defesa desta dissertação, além de todas as contribuições para o desenvolvimento desta pesquisa.

- Aos membros do colegiado e da coordenação do Programa de Pós-Graduação em Informática (PPGI) por todo o apoio concedido a mim durante o período em que pertenci ao programa.
- Aos membros e amigos do *Logistics and Optimization Group* (LOG), grupo de pesquisa do qual tenho a enorme satisfação de fazer parte.
- À CAPES pelo auxílio financeiro através da concessão de uma bolsa de mestrado.
- A todos que contribuíram direta ou indiretamente para que a conclusão deste trabalho fosse possível.

RESUMO

O *Problema de Roteamento de Veículos* (PRV) é um problema de otimização combinatória clássico amplamente estudado na literatura. Por definição, consiste em determinar as rotas de menor custo, que são iniciadas e finalizadas no mesmo depósito, de modo a atender as demandas de um conjunto de clientes. Há uma diversidade de variantes desse problema, as quais podem incluir atributos adicionais, tais como frota de veículos heterogênea, janelas de tempo, entre outros. Dentre esses problemas, há o PRV com Coleta e Entrega Simultâneas (PRVCES), que considera o fato dos clientes possuírem tanto demandas de entrega quanto de coleta a serem satisfeitas em uma única visita. Nesse contexto, este trabalho tem como objetivo propor uma abordagem exata unificada baseada em geração de colunas e de cortes para resolver dez variantes do PRVCES, incluindo a versão clássica do problema. Essa abordagem faz uso do *VRPSolver*, um resolvidor *branch-cut-and-price* estado-da-arte para problemas de roteamento e afins. Os resultados mostram que a abordagem proposta é bastante efetiva na obtenção de soluções ótimas ou aprimoramento dos limites duais para muitas instâncias da literatura em aberto.

Palavras-chave: Roteamento de veículos. Coleta e entrega simultâneas. Geração de colunas. Planos de corte.

ABSTRACT

The Vehicle Routing Problem (VRP) is a classical combinatorial optimization problem widely studied in the literature. By definition, it consists in determining least-cost routes, starting and ending at the depot, to meet the demands of a set of customers. There is a substantial number of variants of the problem, which might include additional attributes such as a heterogeneous fleet of vehicles, time windows, and so on. Among them, there is the VRP with Simultaneous Pickup and Delivery (VRPSPD), where customers have both pickup and delivery demands to be satisfied in a single visit. In this context, this work aims at proposing a unified exact approach based on column generation and cutting planes to solve ten VRPSPD variants including the classic version of the problem. This approach uses the VRPSolver, which is a state-of-the-art branch-cut-and-price solver for routing and similar problems. The results show that the proposed approach is highly effective in obtaining the optimal solutions or improving the dual bounds for many open benchmark instances.

Keywords: Vehicle routing. Simultaneous pickup and delivery. Column generation. Cutting planes.

SUMÁRIO

Abstract	viii
1 Introdução	1
1.1 Definição do tema	1
1.2 Motivação	2
1.3 Objetivos	3
1.3.1 Objetivo geral	3
1.3.2 Objetivos específicos	3
1.4 Estrutura da dissertação	4
2 Trabalhos relacionados	5
2.1 Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas (PRVCES)	5
2.2 PRVCES com Frota Heterogênea (PRVCESFH)	15
2.3 PRVCES com Janelas de Tempo (PRVCESJT)	17
2.4 Problema de Roteamento de Veículos com Coleta e Entrega Mistadas (PRVCEM)	22
2.5 PRVCEM com Múltiplos Depósitos (PRVCEMMD)	27
2.6 Problema de Roteamento e Localização com Coleta e Entrega Simultâneas (PRLCES)	30
2.7 Outras variantes	34
3 Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas e suas variantes	36
3.1 Descrição do problema	36
3.2 Formulação matemática	38

4	Modelo para o VRPSolver	42
4.1	Modelo Genérico	42
4.1.1	Grafos e subproblema	43
4.1.2	Formulação do problema mestre	44
4.1.3	Packing Sets	45
4.2	Aplicação	46
4.2.1	Grafos	46
4.2.2	Recursos	47
4.2.3	Variáveis	48
4.2.4	Modelo	49
5	Resultados computacionais	54
5.1	PRVCES	55
5.2	PRVCEM	58
5.3	PRVCESTL	58
5.4	PRVCEMTL	60
5.5	PRVCESJT	60
5.6	PRVCESJTf	61
5.7	PRVCESA	64
5.8	PRVCESFH	64
5.9	PRVCEMMD	68
5.10	PRLCES	69
6	Considerações Finais	77
	Apêndice A <i>Lazy Cuts</i>	92
	Apêndice B Parâmetros	94

LISTA DE TABELAS

2.1	<i>Surveys</i> relacionados ao PRVCES	7
2.2	Trabalhos relacionados ao PRVCES	15
2.3	Trabalhos relacionados ao PRVCESFH	17
2.4	Trabalhos relacionados ao PRVCESJT	22
2.5	Trabalhos relacionados ao PRVCEM	27
2.6	Trabalhos relacionados ao PRVCEMMD	30
2.7	Trabalhos relacionados ao PRLCES	34
2.8	Outras variantes do PRVCES	35
3.1	Diferenças entre as variantes consideradas.	41
5.1	Descrição dos conjuntos de instâncias considerados.	55
5.2	Resultados computacionais das instâncias do PRVCES — SN — Com- paração com o B&C com <i>lazy separations</i>	56
5.3	Resultados computacionais das instâncias do PRVCES — SN — Com- paração com o BCP.	57
5.4	Resultados computacionais das instâncias do PRVCES — MG — Com- paração com o B&C com <i>lazy separations</i>	57
5.5	Resultados computacionais das instâncias do PRVCES — MG — Com- paração com o BCP.	58
5.6	Resultados computacionais das instâncias do PRVCEM — Comparação com o B&C com <i>lazy separations</i>	59
5.7	Resultados computacionais das instâncias do PRVCEM — Comparação com o BCP.	59
5.8	Resultados computacionais das instâncias do PRVCESTL.	60
5.9	Resultados computacionais das instâncias do PRVCEMTL.	61

5.10	Resultados computacionais das instâncias do PRVCESJT — W12_C. . .	62
5.11	Resultados computacionais das instâncias do PRVCESJT — W12_RC. . .	62
5.12	Resultados computacionais das instâncias do PRVCESJT — W12_R. . .	63
5.13	Resultados computacionais das instâncias do PRVCESJTF — W13_C. . .	64
5.14	Resultados computacionais das instâncias do PRVCESJTF — W13_RC. . .	65
5.15	Resultados computacionais das instâncias do PRVCESJTF — W13_R. . .	66
5.16	Resultados computacionais das instâncias do PRVCESA — RZC6.	67
5.17	Resultados computacionais das instâncias do PRVCESA — RZC7.	67
5.18	Resultados computacionais das instâncias do PRVCESFH.	68
5.19	Resultados computacionais das instâncias do PRVCEMMD.	69
5.20	Resultados computacionais das instâncias do PRLCES — Bar11.	71
5.21	Resultados computacionais das instâncias do PRLCES — Prod11.	72
5.22	Resultados computacionais das instâncias do PRLCES — Prod12.	74
A.1	Comparação entre os tipos de <i>Lazy Cuts</i> — PRVCES.	93
A.2	Comparação entre os tipos de <i>Lazy Cuts</i> — PRVCEM.	93

GLOSSÁRIO

ACS	<i>Ant Colony System</i>
ADMM	<i>Alternating Direction Multiplier Method</i>
ALNS	<i>Adaptive Large Neighborhood Search</i>
ALR	<i>Relaxação Lagrangiana Aumentada</i>
ALS	<i>Adaptive Local Search</i>
AM	<i>Adaptive Memory</i>
ANS	<i>Adaptive Neighborhood Selection</i>
AP	<i>Australian Post</i>
B&B	<i>Branch-and-Bound</i>
B&C	<i>Branch-and-Cut</i>
B&P	<i>Branch-and-Price</i>
BCP	<i>Branch-Cut-and-Price</i>
BKS	<i>Best Known Solutions</i>
BSTS	<i>Bi-Structure Tabu Search</i>
CEA	<i>Algoritmo co-evolucionário</i>
CI	<i>Cheapset Insertion</i>
CW	<i>Método de Clark-Wright</i>
DARP	<i>Dial-a-Ride-Problem</i>
DE	<i>Differential Evolution</i>
DP	<i>Programação Dinâmica</i>
EHH	<i>Evolution-based hyperheuristic</i>
ETSP	<i>Efficient Tabu Search based Procedure</i>
FSMPDPTW	<i>Fleet Size and Mix Pickup and Delivery Problem with Time Windows</i>
FO	<i>Função Objetivo</i>
GA	<i>Genetic Algorithm</i>

GCM	<i>Greedy Clustering Method</i>
GRAMPS	<i>Greedy Randomised Adaptive Memory Programming Search</i>
GrSCM	<i>Green Supply Chain Managementem</i>
HLS	<i>Hybrid Local Search</i>
ICA	<i>Imperialist Competitive Algorithm</i>
IDE	<i>Improved Differential Evolution</i>
IG	<i>Método Iterativo Guloso</i>
ILS	<i>Iterated Local Search</i>
LNS	<i>Large Neighborhood Search</i>
LB	<i>Lower Bound</i>
MA	<i>Memetic Algorithm</i>
MATE	<i>Memetic Algorithm with efficienT local Search and Extended neighborhood</i>
MAVRPs	<i>Multi-Attribute Vehicle Routing Problems</i>
MSA	<i>Multi-Start Simulated Annealing</i>
MSALNS	<i>Multi-Start Adaptive Large Neighborhood Search</i>
MRLRP	<i>Multicommodity-Ring Location Routing Problem</i>
NSGA	<i>Nondominated Sorting Genetic Algorithm</i>
PACE	<i>Program of All-Inclusive Care for the Elderly</i>
PDPTW	<i>Pickup and Delivery Problem with Time Windows</i>
PCE	<i>Problema de Coleta e Entrega</i>
PLI	<i>Programação Linear Inteira</i>
PR	<i>Path Relinking</i>
PRV	<i>Problema de Roteamento de Veículos</i>
PRVCED	<i>PRV com demandas de coleta e entrega divisíveis</i>
PRVCEM	<i>PRV com Coleta e Entrega Mistas</i>
PRVCEMFHJT	<i>PRVCEM com Frota Heterogênea e Janelas de Tempo</i>
PRVCEMJT	<i>PRVCEM com Janelas de Tempo</i>
PRVCEMJTF	<i>PRVCEM com Janelas de Tempo Flexíveis</i>
PRVCEMMD	<i>PRV com Coleta e Entrega Mistas e Múltiplos Depósitos</i>
PRVCEMTL	<i>PRV com Coleta e Entrega Mistas e Tempo Limite</i>
PRVCES	<i>PRV com Coleta e Entrega Simultâneas</i>
PRVCESA	<i>PRVCES Assimétrico</i>

PRVCESFH	<i>PRV com Coleta e Entrega Simultâneas e Frota Heterogênea</i>
PRVCESJT	<i>PRV com Coleta e Entrega Simultâneas e Janelas de Tempo</i>
PRVCESTL	<i>PRV com Coleta e Entrega Simultâneas e Tempo Limite</i>
PRVFH	<i>Problema de Roteamento de Veículos com Frota Heterogênea</i>
PRVJT	<i>PRV com Janelas de Tempo</i>
PRVMD	<i>Problema de Roteamento de Veículos com Múltiplos Depósitos</i>
PRLCES	<i>Problema de Roteamento e Locação com Coleta e Entrega Simultâneas</i>
p-SA	<i>Parallel Simulated Annealing</i>
PSO	<i>Particle Swarm Optimization</i>
RCRS	<i>Residual Capacity and Radial Surcharge</i>
R-INS	<i>Algoritmo de inserção restrita</i>
RVND	<i>Randomized Variable Neighborhood Descent</i>
SA	<i>Simulated Annealing</i>
SMD	<i>Static Move Descriptor</i>
SP	<i>Set Partitioning</i>
TN	<i>Turkish Network</i>
TS	<i>Tabu Search</i>
UB	<i>Upper Bound</i>
UDCs	<i>Urban Distribution Centers</i>
UHGS	<i>Unified Hybrid Genetic Search</i>
VND	<i>Variable Neighborhood Descent</i>
VNS	<i>Variable Neighborhood Search</i>
WS	<i>Waiting Strategy</i>

Capítulo 1

Introdução

1.1 Definição do tema

O *Problema de Roteamento de Veículos* (PRV) é um problema de otimização combinatória clássico. Proposto por Dantzig e Ramser (1959), trata-se de um problema bastante estudado ao longo de mais de 60 anos. Como definição para o PRV, tem-se: “Dados um conjunto de requisições de transporte e uma frota de veículos, o problema consiste em determinar um conjunto de rotas de veículos a fim de que todas as requisições (ou algumas delas) sejam atendidas com a frota dada, a um custo mínimo” (Toth e Vigo, 2014). Em outras palavras, dados um conjunto de clientes com suas respectivas demandas e uma frota de veículos, determinar quais veículos atenderão quais clientes, bem como a sequência de atendimento, de modo que todos eles sejam atendidos uma única vez e que o custo total de transporte seja mínimo.

Há diferentes variações para o PRV, cada uma delas possuindo peculiaridades que as diferenciam entre si (Braekers et al., 2016). Dentre eles, pode-se destacar os Problemas de Coleta e Entrega (PCE) que, para Berbeglia et al. (2007), são uma importante classe do PRV, a qual lida com o transporte de pessoas e objetos, seja coletando-os e/ou distribuindo-os. Dentre esses problemas, está o Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas (PRVCES), proposto por Min (1989), no qual as tarefas de coleta e entrega são realizadas simultaneamente.

Nesse sentido, a presente pesquisa tem como objetivo propor uma abordagem exata, baseada em um método geração de colunas e de cortes utilizando o *VRPSolver* (Pessoa et al., 2020; Sadykov e Vanderbeck, 2021), para que possam ser encontradas soluções ótimas ou melhorados os *lower bounds* (LBs) já conhecidos de diferentes conjuntos de instâncias que compõem o estado da arte do PRVCES e algumas de suas variantes, a saber: PRVCES com Frota Heterogênea (PRVCESFH), PRVCES com Janelas de Tempo (PRVCESJT), PRV com Coleta e Entrega Mista (PRVCEM), PRVCEM com múltiplos depósitos (PRVCEMMD, Problema de Roteamento e Localização com Coleta e Entrega Simultâneas (PRLCES), PRVCES com Tempo Limite (PRVCESTL), PRVCEM com Tempo Limite (PRVCEMTL), PRVCES Assimétrico (PRVCESA) e PRVCES com Janelas de Tempo Flexíveis (PRVCESJTF). Conforme Pecin et al. (2017a), os melhores desempenhos de algoritmos exatos para a resolução de PRV Capacitado nos últimos anos estão baseados na utilização de técnicas que combinam geração de colunas e cortes, justificando-se, assim, a escolha do método utilizado.

1.2 Motivação

De um modo geral, o PRV é fundamental na gestão da cadeia de suprimentos de uma empresa, sendo utilizado no cotidiano de diversas organizações que, de alguma maneira, lidam com serviços de distribuição (Subramanian, 2012). Além disso, deve-se destacar o impacto econômico advindo do uso da computação na resolução desse tipo de problema. Segundo Toth e Vigo (2014), o grande número de aplicações reais mostra que o uso de métodos computacionais para a obtenção de soluções do PRV impacta substancialmente na redução dos custos de transporte. Ressaltam que isso se deve não apenas ao avanço computacional, mas também ao desenvolvimento de modelos matemáticos cada vez mais rigorosos, que estão aptos a incorporar quase todas as características das aplicações de PRV do mundo real.

Em se tratando do PRVCES, deve ser levada em consideração não apenas a entrega ou a coleta, mas ambas as atividades simultaneamente. Nesse sentido, percebe-se a combinação de dois tipos de logística, a saber: logística direta e reversa. Esta, por sua vez, pode ser definida como: “Processo de planejamento, implementação e controle

do fluxo eficiente e econômico de matérias-primas, inventário de processo, produtos acabados e informações relacionadas do ponto de consumo ao ponto de origem com o objetivo de recuperação do valor ou o descarte adequado” (Rogers, 1999). Em outras palavras, pode-se afirmar que se trata do processo de recolhimento de produtos após sua comercialização ou consumo, a fim de que seja conseguido algum lucro com a sua recuperação ou mesmo para que seus resíduos sejam descartados adequadamente. Particularmente para esse segundo objetivo, Srivastava (2008) aponta que o chamado Gerenciamento Verde da Cadeia de Suprimentos (*Green Supply Chain Management*, GrSCM) é alvo do interesse de pesquisadores e profissionais que atuam no gerenciamento de cadeia de suprimentos, seja por fatores de natureza econômica, regulatória e de consumo. Investimentos e esforços em logística reversa são fundamentais nesse tipo de gerenciamento, uma vez que o reaproveitamento dos produtos coletados ou mesmo o descarte apropriado contribui para o desenvolvimento sustentável da sociedade.

Em face do exposto, percebe-se a importância de estudos acerca do PRV, em particular do PRVCES, tendo em vista os impactos econômicos que o uso de aplicações de otimização voltadas a esse problema podem proporcionar, bem como as contribuições no contexto da sustentabilidade ambiental, tema este de muito interesse e bastante discutido nos últimos anos.

1.3 Objetivos

1.3.1 Objetivo geral

Desenvolver um método de resolução exato, utilizando o *VRPSolver*, para o PRVCES e algumas de suas variantes (PRVCESFH, PRVCESJT, PRVCEM, PRVCEMMD, PRLCES, PRVCESTL, PRVCEMTL, PRVCESA, PRVCESJTF), a fim de que sejam encontradas soluções ótimas ou melhorados os LBs já conhecidos de importantes instâncias do problema.

1.3.2 Objetivos específicos

Os objetivos específicos referentes ao presente estudo estão listados adiante.

- Fazer um levantamento sobre o estado da arte do tema estudado.
- Obter importantes conjuntos de instâncias para o problema e suas variantes.
- Testar o método proposto com as instâncias obtidas.
- Comparar os resultados obtidos com os de outros algoritmos que constituem o estado da arte.
- Propor uma abordagem exata baseada em *Branch-Cut-and-Price* (BCP), colaborando com o desenvolvimento de métodos exatos para a resolução do problema em questão.
- Propor uma formulação matemática unificada que englobe as dez variantes do PRVCES consideradas nesta pesquisa.

1.4 Estrutura da dissertação

O restante desta dissertação se encontra estruturado da seguinte maneira: no Capítulo 2, é realizado um levantamento bibliográfico do PRVCES e suas variantes consideradas. Por sua vez, no Capítulo 3, é apresentada uma formulação unificada que engloba todos os dez problemas estudados. Já no Capítulo 4, são realizadas explicações acerca do *VRPSolver* e, principalmente, do modelo voltado à utilização desse *solver* para o problema unificado. Por fim, no Capítulo 5, são expostos os resultados obtidos com a abordagem de resolução exata proposta, bem como as considerações finais e perspectivas para pesquisas futuras, no Capítulo 6.

Capítulo 2

Trabalhos relacionados

Neste capítulo é feita uma análise de alguns artigos que compõem o estado da arte dos Problemas de Coleta e Entrega Simultâneas, bem como de suas variantes consideradas nesta dissertação. Mais precisamente, na Seção 2.1 é feito um levantamento bibliográfico do problema clássico (PRVCES). Por sua vez, nas Seções 2.2 e 2.3 são abordadas as variantes com frota heterogênea (PRVCESFH) e janelas de tempo (PRVCESJT), respectivamente. Na Seção 2.4, a variante com coleta e entrega mistas (PRVCEM) é tratada, bem como sua generalização com múltiplos depósitos (PRVCEMMD) na Seção 2.5. Por fim, a variante que considera a escolha dos depósitos, além do roteamento, como parte do objetivo do problema (PRLCES) é explanada na Seção 2.6, enquanto que as demais variantes consideradas são abordadas na Seção 2.7.

2.1 Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas (PRVCES)

O primeiro trabalho a lidar com o PRVCES foi proposto por Min (1989), no qual foi permitido, pela primeira vez, a possibilidade dos serviços de coleta e entrega serem realizados em um mesmo cliente simultaneamente. Para resolver o problema, desenvolveu uma heurística de três fases – clusterização dos clientes, atribuição dos veículos aos

clusters e estruturação das rotas – considerando um estudo de caso real. Uma década depois, Salhi e Nagy (1999) abordaram o PRVCES e o PRVCEM conjuntamente e, por essa razão, tal trabalho encontra-se descrito com mais detalhes na Seção 2.4.

Desde sua criação (Min, 1989), diversos trabalhos relacionados ao PRVCES foram desenvolvidos, os quais são analisados na tese de Subramanian (2012) e nos *surveys* de Bouanane et al. (2020) e Koç et al. (2020). No primeiro, são propostas abordagens exatas, heurísticas e híbridas para a resolução de diversas classes do PRV, dentre as quais está incluso o PRVCES. Quanto aos *surveys*, Bouanane et al. (2020) propuseram uma classificação taxonômica do PRVCES, com o objetivo de facilitar o desenvolvimento de futuros trabalhos acerca do tema. Tal classificação utiliza quatro classes principais, “Características do estudo”, “Características dos cenário”, “Características do problema físico” e “Características dos dados e informações”, as quais possuem suas próprias subclasses. Como resultado, os autores foram capazes de classificar 191 artigos da literatura, com base nos critérios adotados. Por outro lado, no trabalho de Koç et al. (2020) são abordados formulações matemáticas sobre o problema; métodos de resolução, exatos e heurísticos, utilizados por diversos pesquisadores; variações do PRVCES; estudos de caso e os principais resultados que compõem o estado da arte sobre o tema. Os próprios autores destacam outros *surveys* que, de alguma maneira, lidam com o PRVCES.

Dentre esses *surveys*, Berbeglia et al. (2007) fizeram um levantamento bibliográfico acerca do PCE, classificando-os em três categorias principais, a saber: *many-to-many*, *one-to-many-to-one* e *one-to-one problems*. Na primeira, cada produto transportado (*commodity*) pode ter múltiplos pontos (nós) de partida e de destino, bem como qualquer um dos nós pode desempenhar essa função. Por sua vez, a segunda classificação se refere aos problemas nos quais alguns produtos são transportados de um depósito para diversos clientes, enquanto outros são coletados de tais clientes e levados ao depósito. É nessa categoria que se encaixa o PRVCES, tema de interesse desta pesquisa. Por fim, para os problemas da terceira categoria, cada produto possui pontos de partida e de destino únicos. Posteriormente, esse levantamento foi complementado pela inclusão da análise do PCE dinâmico, (Berbeglia et al., 2010), no qual os itens devem ser coletados e distribuídos em tempo real.

Outra revisão bibliográfica acerca do PCE foi realizada, a qual se encontrada dividida em duas partes, a saber: (Parragh et al., 2008a) e (Parragh et al., 2008b). Na primeira, foram abordados os problemas em que os bens são transportados do depósito para os clientes *linehauls* (com demandas de entrega) e dos clientes *backhauls* (com demandas de coleta) para o depósito. Por sua vez, na segunda parte, são considerados os problemas em que os bens são transportados entre os locais de coleta e entrega, categoria na qual está incluso o PRVCES.

Por fim, Battarra et al. (2014) fizeram um levantamento bibliográfico semelhante a Berbeglia et al. (2007) no que se refere à classificação dos problemas como *many-to-many*, *one-to-many-to-one* e *one-to-one*. Além disso, consideraram os problemas de coleta e entrega que possuem restrições relacionadas ao carregamento dos veículos (*loading constraints*). Todos os *surveys* apresentados nesta seção estão destacados na Tabela 2.1.

Tabela 2.1: *Surveys* relacionados ao PRVCES

<i>Survey</i>	<i>Tópico do survey</i>
Berbeglia et al. (2007)	PCE estáticos
Parragh et al. (2008a)	PCE (linehauls e backhauls)
Parragh et al. (2008b)	PCE (demais problemas)
Berbeglia et al. (2010)	PCE dinâmicos
Battarra et al. (2014)	PCE para transporte de bens
Bouanane et al. (2020)	PRV com coleta e entrega simultâneas
Koç et al. (2020)	PRV com coleta e entrega simultâneas

Ao longo dos anos, abordagens exatas, heurísticas e híbridas foram propostas para o PRVCES. Especificamente sobre os métodos exatos, Dethloff (2001) investigou a relação entre o PRVCES e outras variantes do PRV, propondo uma formulação matemática de três índices para o problema. Similarmente, Montané e Galvão (2006) também desenvolveram uma formulação de três índices. Por outro lado, Dell’Amico et al. (2006) simplificaram os modelos existentes para o PRVCES, propondo uma formulação baseada em apenas dois índices. Estes desenvolveram um método de resolução exato, baseado no algoritmo *Branch-and-Price* (B&P), considerando tanto programação dinâmica quanto relaxação de espaço de estados como estratégias de resolução do subproblema de *pricing*. Esse trabalho representou a primeira tentativa de utilização

de algoritmos de B&P na busca por soluções para o PRVCES, cujos resultados permitiram afirmar que a aplicação de tais algoritmos é viável com instâncias de pequeno e médio porte.

Subramanian (2008) propôs uma formulação baseada em fluxo de duas *commodities* indireta para o PRVCES, contudo não realizou experimentos computacionais com a mesma. Dois anos depois, Subramanian et al. (2010b) apresentaram duas formulações de fluxo de duas *commodities*, uma direta e outra indireta, para o problema em questão. Ambas as formulações foram comparadas ao modelo proposto por Dell’Amico et al. (2006). A fim de testá-las, foram utilizados um esquema *Branch-and-Cut* (B&C) e os mesmos conjuntos de instâncias do trabalho de Subramanian et al. (2010a). Da análise dos resultados, inferiu-se que a formulação indireta de fluxo de duas *commodities* obteve um melhor desempenho do que as demais. Quanto ao trabalho de Subramanian et al. (2011), foi utilizada uma variação do *Branch-and-Cut* (B&C), ao acrescentar *lazy separations*, para lidar com o PRVCES e com o PRVCEM. Além disso, propuseram uma formulação reduzida para os problemas, baseada em fluxo, a qual possui apenas as variáveis correspondentes às arestas, ignorando aquelas correspondentes ao controle das cargas contidas nos veículo, presentes em formulações clássicas. Em relação aos experimentos computacionais, foram utilizadas as instâncias propostas por Dethloff (2001), Montané e Galvão (2006) e Salhi e Nagy (1999) para o PRVCES, bem como as instâncias destes para o PRVCEM. Como resultados, obtiveram novas soluções ótimas e limites inferiores melhorados, considerando as 87 instâncias testadas com 50-200 clientes. Por fim, atribuíram o sucesso do método B&C proposto, quando comparado abordagens anteriores, ao modelo mais simplificado, evitando-se o uso de formulações extensas e mais complicadas.

No ano seguinte, Subramanian (2012) desenvolveu um estudo completo sobre diversas variantes do PRV, dentre as quais estão presentes o PRVCES, o PRVCEM e o PRVCEMMD. Para lidarem com esses problemas, foram implementados métodos exatos (B&C com *lazy separations* e BCP), heurísticos (ILS-RVND) e híbridos (ILS-RVND-SP). Especificamente para o BCP, este foi desenvolvido com base no algoritmo proposto por Fukasawa et al. (2006) para o CVRP, o qual, por sua vez, foi a primeira utilização de algoritmos dessa natureza no contexto de roteamento de veículos. A prin-

principal diferença entre os dois algoritmos BCP citados está relacionada ao procedimento de geração de colunas, adaptado para viabilizar a simultaneidade entre as demandas de coleta e entrega. Durante os experimentos computacionais, foram obtidos resultados bastante expressivos, considerando as instâncias propostas por Dethloff (2001), Salhi e Nagy (1999) e Montané e Galvão (2006).

Rieck e Zimmermann (2013) contribuíram para a literatura do PRVCES ao proporem duas formulações para o problema, uma baseada em fluxo de veículos e outra baseada em fluxo de *commodities*. Visando analisar as contribuições advindas das formulações propostas, experimentos computacionais foram realizados com instâncias assimétricas derivadas de problemas reais, instâncias simétricas advindas de importantes conjuntos da literatura e, por fim, com modificações dos conjuntos propostos por Dethloff (2001) e Salhi e Nagy (1999), para o acréscimo da assimetria. Para tanto, foi utilizado o CPLEX 12.1 e os resultados obtidos indicaram que a maioria das instâncias testadas foi resolvida, com suas soluções ótimas encontradas. No mesmo ano, Subramanian et al. (2013b) aplicaram o método *Branch-Cut-and-Price* (BCP) à resolução do PRVCES, bem como do PRVCEM. Para os autores, o que eles propuseram consiste na primeira tentativa de resolução dos dois problemas citados por meio desse método. Além disso, utilizaram a mesma formulação de fluxo proposta por Subramanian et al. (2011), bem como os mesmos conjuntos de instâncias considerados pelos mesmos. Como resultados, quatro instâncias foram resolvidas pela primeira vez, além de alguns limites inferiores melhorados. Já Agarwal e Venkateshan (2021) abordaram o PRVCES generalizando os cortes *no-good*, bem como analisando maneiras de gerá-los mais cedo na árvore de B&B, a fim de que houvessem melhorias relacionadas aos tempos de execução do algoritmo. Experimentos computacionais foram realizados com instâncias consideradas difíceis para o problema e pertencentes a conjuntos conhecidos da literatura, sendo o modelo do trabalho de Agarwal e Venkateshan (2020) utilizado na comparação dos resultados. Este, por sua vez, como também mais detalhes acerca da generalização dos cortes, serão abordados na Seção 2.7.

Quanto aos métodos heurísticos e híbridos, Dethloff (2001) desenvolveu uma heurística construtiva aplicada a instâncias já conhecidas, como também criou novas instâncias, com 50 clientes, geradas randomicamente. Já Montané e Galvão (2006) im-

plementaram a meta-heurística *Tabu Search* (TS) aplicada a 87 instâncias contendo 50-400 clientes, advindas dos trabalhos de Dethloff (2001), Salhi e Nagy (1999) e derivadas de outros *benchmarks* já conhecidos da literatura. Outro método heurístico para o PRVCES foi desenvolvido por Subramanian (2008), quem implementou um método que combina *Iterated Local Search* (ILS) e a técnica de descida em vizinhança variável (*Variable Neighborhood Descent*, VND). Estes fazem parte de uma extensa quantidade de trabalhos envolvendo heurísticas para o problema e, portanto, a análise na presente seção será limitada aos trabalhos desenvolvidos desde o início da década passada até o momento de escrita deste texto, cabendo ao leitor a consulta aos *surveys* referenciados, caso deseje conhecer as demais pesquisas realizadas acerca do tema. Desse modo, Gajpal e Abad (2010) desenvolveram uma heurística *parallel saving* para o PRVCES, a qual originava novas rotas a partir da combinação de duas rotas já existentes e cuja viabilidade desse procedimento era verificada por uma abordagem *cumulative net-pickup*. No mesmo ano, Zachariadis et al. (2010) implementaram um algoritmo de memória adaptativa (*Adaptive Memory*, AM), responsável por extrair características das soluções de boa qualidade encontradas por métodos de busca, originando novas soluções que são melhoradas pelo método TS. Experimentos computacionais foram realizados com as instâncias de Dethloff (2001), Salhi e Nagy (1999) e Montané e Galvão (2006), provando o método desenvolvido ser capaz de encontrar soluções de boa qualidade a um esforço computacional baixo.

Ainda no mesmo ano, Subramanian et al. (2010a) aplicaram uma abordagem heurística *multi-start* paralela para a resolução do PRVCES, que combina os métodos ILS e *Randomized Variable Neighborhood Descent* (RVND). Testes computacionais foram executados em uma máquina com arquitetura *multi-core*, chegando a até 256 núcleos. Foram utilizados os conjuntos de instâncias de Dethloff (2001), Salhi e Nagy (1999) e Montané e Galvão (2006), sendo melhoradas as soluções de 3 instâncias do segundo conjunto e de 12 instâncias referentes ao terceiro conjunto. Já Souza et al. (2011) propuseram um algoritmo heurístico híbrido, denominado GENILS, o qual combina os métodos ILS, VND e GENIUS. Novamente, durante os experimentos computacionais, foram utilizados os conjuntos de instâncias de Dethloff (2001), Salhi e Nagy (1999) e Montané e Galvão (2006), com o GENILS tendo sido capaz de melhorar as soluções

já conhecidas para 9 dentre as instâncias testadas. Zachariadis e Kiranoudis (2011) desenvolveram uma heurística de busca local para o PRVCES, envolvendo os conceitos algorítmicos de *Static Move Descriptor* (SMD) e de mecanismos para evitar ciclagem nas buscas, com diversificação. No testes computacionais, foram resolvidas 18 instâncias de grande porte contidas no conjunto proposto por Montané e Galvão (2006) e melhoradas soluções já conhecidas.

Cruz et al. (2012) propuseram um algoritmo, denominado GENVNS-TS-CL-PR, que combina múltiplas heurísticas, a saber: *Cheapest Insertion*, *Cheapest Insertion* com múltiplas rotas, GENIUS, VND, *Variable Neighborhood Search* (VNS), TS e *Path Relinking* (PR). Enquanto as três primeiras heurísticas são utilizadas na obtenção de soluções iniciais de boa qualidade, o VND e o TS são utilizados com métodos de busca local para o VNS. Quanto ao PR, este é utilizado na conexão de soluções ótimas locais com as melhores soluções já conhecidas. Novamente foram utilizadas as instâncias de Dethloff (2001), Salhi e Nagy (1999) e Montané e Galvão (2006) nos testes computacionais, sendo obtidos resultados de boa qualidade, com desempenho próximo aos melhores métodos já conhecidos. No mesmo ano, Jun e Kim (2012) implementaram um algoritmo heurístico envolvendo um método construtivo *sweep-based*, operadores (op.) intra-rotas e inter-rota, mecanismos de perturbação e técnicas de melhoramento. Experimentos computacionais foram realizados com as mesmas instâncias utilizadas por Nagy e Salhi (2005), a fim de validar a heurística construtiva (HC) desenvolvida, e soluções foram melhoradas para parte das instâncias testadas. Já Tasan e Gen (2012) propuseram uma abordagem baseada em algoritmos genéticos (*Genetic Algorithm*, GA) para resolver o PRVCES, com a justificativa desse problema ter natureza combinatória, sendo o GA um método eficiente na resolução de problemas combinatórios. Para a validação do método, foram utilizadas 24 instâncias consideradas de porte médio, derivadas do trabalho de Augerat et al. (1995), ao serem acrescentadas as demandas de coleta a partir de uma distribuição uniforme compreendida no intervalo $[0, 26]$.

Dando continuidade, Goksal et al. (2013) lidaram com o PRVCES propondo uma abordagem heurística baseada no método de enxame de partículas (*Particle Swarm Optimization*, PSO) em conjunto com o método VND para a realização do procedimento de busca local e uma estratégia *annealing-like* para a diversificação do enxame. Testes

computacionais foram realizados, a fim de validar o método desenvolvido. Para tanto, foram utilizadas as instâncias dos conjuntos de Dethloff (2001) e de Salhi e Nagy (1999), sendo obtidos resultados bastante significativos, tais como a descoberta de novas 104 melhores soluções dentre o conjunto de 141 instâncias testadas, com tal melhoria em torno de 2% na média. Por sua vez, Subramanian et al. (2013a) propuseram um algoritmo híbrido para uma classe de variantes do PRV com frota homogênea, denominado ILS-RVND-SP, o qual combina modelos de *Set Partitioning* (SP) com uma meta-heurística baseada em ILS. Outra contribuição consistiu no desenvolvimento de um mecanismo reativo para o controle do tamanho dos modelos SP, bastante útil para instâncias de grande porte, em particular. Quanto aos testes computacionais, especificamente para o PRVCES, foram utilizados os conjuntos de instâncias de Salhi e Nagy (1999) e Montané e Galvão (2006). Vale salientar que o PRVCEM faz parte do conjunto de variantes do PRV analisadas e, portanto, experimentos foram realizados com as instâncias de Salhi e Nagy (1999) para essa variante. Dentre os resultados obtidos, o método híbrido analisado foi capaz de encontrar soluções já conhecidas e melhorar algumas delas para todos os conjuntos analisados e, conseqüentemente, para ambas as variantes analisadas.

No ano seguinte, Yousefikhoshbakht et al. (2014) implementaram um algoritmo híbrido baseado nos métodos TS e de sistemas de colônia de formigas (*Ant Colony System*, ACS), mais precisamente, *elite ant system* (MTSEAS). Nele, soluções iniciais são geradas a partir de uma heurística de inserção do vizinho mais próximo e então melhoradas pelo método TS. São aplicados movimentos de vizinhanças *2-opt* e *exchanges* 0-1 e 1-1, além de operações presentes em algoritmos ACS. Durante os testes computacionais, foram utilizadas as instâncias de Dethloff (2001) e Salhi e Nagy (1999), cujos resultados demonstraram que o método proposto é competitivo perante outras meta-heurísticas para o PRVCES.

Vidal et al. (2014) realizaram um extenso trabalho ao propor um algoritmo genético unificado híbrido (*Unified Hybrid Genetic Search*, UHGS) para resolver diversas variantes do PRV (*Multi-Attribute Vehicle Routing Problems*, MAVRPs), incluindo o PRVCES. A ideia do método proposto é ser uma abordagem genérica, envolvendo os conceitos de hibridização entre algoritmos genéticos e uma busca local eficiente, repre-

sentação de soluções sem *trip delimiters*, penalização de soluções inviáveis e técnicas de gerenciamento populacional *diversity-and-cost objective* para a avaliação das soluções encontradas. Restringindo-se aos testes realizados apenas com as instâncias do PRVCES, foram utilizadas aquelas propostas por Salhi e Nagy (1999) e Montané e Galvão (2006). Dentre os resultados obtidos, a heurística proposta demonstrou ser bastante robusta, apresentando um bom desempenho para os diversos problemas estudados. Anos mais tarde, Kalayci e Kaya (2016) implementaram um método híbrido envolvendo ACS com VNS para resolverem o PRVCES. A justificativa para escolha de ambos os métodos está relacionada ao fato do VNS ser uma boa meta-heurística para problemas de otimização, a qual se torna ainda melhor com seus problemas relacionados à memória minimizados pela técnica de memória de longo prazo do ACS. Para a validação do método, foram utilizadas as instâncias de Dethloff (2001) e Salhi e Nagy (1999).

Hornstra et al. (2020) abordaram uma versão diferente do PRVCES, a qual leva em consideração *handling costs* (PRVCES-H), ou seja, as operações de manuseio dos itens dentro dos veículos também são otimizadas. Para resolverem o problema, foi desenvolvido um algoritmo baseado na meta-heurística *Adaptive Large Neighborhood Search* (ALNS), ao qual foram incorporadas as políticas de manuseio (*handling*) criadas. Para os experimentos computacionais, foram consideradas instâncias do PRVCES, PRVCEM, PRV com Coleta e Entrega Divisíveis (PRVCED) e do Problema do Caixeiro Viajante com Coleta e Entrega e *handling costs* (PCVCE-H), que são casos especiais do PRVCES-H. Além disso, implementaram as formulações matemáticas para o PRVCES-H, PRVCEM-H e PRVCED-H apresentadas, utilizando-se o CPLEX para resolvê-las. Os resultados obtidos dos testes com esses modelos foram comparados com as soluções advindas da meta-heurística desenvolvida, para as instâncias pequenas, isto é, com poucos clientes. Da análise desses resultados, percebeu-se que todas as instâncias com até 15 clientes tiveram suas soluções ótimas encontradas pela meta-heurística. Além disso, BKS foram melhorados para instâncias dos casos particulares analisados.

Por fim, Park et al. (2021) desenvolveram um método baseado em GA e implementaram uma estratégia de espera (*Waiting Strategy*, WS) para lidar com as demandas que surgem dinamicamente, considerando um indicador *rerouting* (RI). Tal indicador

atua na determinação dos pontos da rota nos quais os clientes com as novas demandas serão acomodados. Experimentos computacionais foram realizados com instâncias de até 20 clientes geradas pelos próprios autores, sendo o método proposto comparado às soluções obtidas utilizando-se o CPLEX. Além disso, comparações foram realizadas entre o algoritmo genético proposto e métodos de busca em vizinhança, são eles: *general*, *adjacent and 2-Segment swaps*. Vale ressaltar também que foi realizada uma análise de sensibilidade do RI sobre os resultados obtidos.

Em face do exposto, é possível perceber a elevada quantidade de trabalhos que compõem a literatura acerca do PRVCES. A fim de resolvê-lo, métodos exatos, heurísticos e híbridos foram propostos. Restringindo-se a análise aos primeiros, uma vez que são o foco de interesse da pesquisa descrita nesta dissertação, os melhores resultados foram apresentados por Subramanian (2012) — e referências associadas (Subramanian et al., 2010b), (Subramanian et al., 2011) e (Subramanian et al., 2013b) — para os conjuntos de instâncias de Dethloff (2001), Salhi e Nagy (1999) e Montané e Galvão (2006), os quais, por sua vez, foram considerados em grande parte dos trabalhos citados. Por essa razão, a abordagem exata proposta na pesquisa em questão será comparada aos métodos BCP e B&C com *lazy constraints* de Subramanian (2012). Note também que todos os trabalhos citados até então estão relacionados ao problema clássico, contudo, variantes dele também já foram estudadas, cujos levantamentos bibliográficos estão descritos nas Seções 2.2 – 2.6.

A Tabela 2.2 reúne todos os trabalhos citados nesta seção, dando destaque às abordagens propostas em cada um deles. Vale lembrar que, para mais informações acerca de outros trabalhos relacionados ao PRVCES cujas datas de publicação não pertencem ao período considerado, recomenda-se a consulta aos *surveys* referenciados na Tabela 2.1.

Tabela 2.2: Trabalhos relacionados ao PRVCES

Referência	Abordagem
Min (1989)	Heurística de três fases
Salhi e Nagy (1999)	Heurística de inserção em <i>cluster</i>
Dethloff (2001)	Heurística construtiva
Montané e Galvão (2006)	TS
Dell’Amico et al. (2006)	B&P
Subramanian (2008)	ILS + VND
Gajpal e Abad (2010)	Heurística <i>parallel saving</i>
Zachariadis et al. (2010)	AM + TS
Subramanian et al. (2010a)	ILS + RVND paralelo
Subramanian et al. (2010b)	Formulações de fluxo de duas <i>commodities</i> + B&C
Souza et al. (2011)	GENILS (ILS + VND + GENIUS)
Zachariadis e Kiranoudis (2011)	Heurística de busca local
Subramanian et al. (2011)	B&C com <i>lazy separations</i>
Subramanian (2012)	Abordagens exatas, heurísticas e híbridas
Cruz et al. (2012)	GENVNS-TS-CL-PR
Jun e Kim (2012)	HC <i>sweep-based</i> com op. intra-rotas e inter-rota
Tasan e Gen (2012)	GA
Goksal et al. (2013)	PSO + VND
Rieck e Zimmermann (2013)	Formulações (fluxo de veículos e de commodities)
Subramanian et al. (2013a)	ILS-RVND-SP
Subramanian et al. (2013b)	BCP
Yousefikhoshbakht et al. (2014)	TS + MTSEAS
Vidal et al. (2014)	UHGS
Kalayci e Kaya (2016)	ACS + VNS
Hornstra et al. (2020)	ALNS + <i>handling polices</i>
Agarwal e Venkateshan (2021)	Generalização dos <i>no-good cuts</i>
Park et al. (2021)	GA + WS + RI

2.2 PRVCES com Frota Heterogênea (PRVCESFH)

Em relação ao PRVCESFH, Qu e Bard (2013) lidaram com uma variante que, além da frota heterogênea, envolve a opção de capacidade dos veículos configurável, a fim de atender as necessidades inerentes ao planejamento diário de rotas do *Program of All-Inclusive Care for the Elderly* (PACE), que consiste em um programa estabelecido pela Medicare para atendimento médico domiciliar voltado aos idosos, sendo uma opção economicamente mais viável para essas pessoas que necessitam de mais cuidados. Para tanto, os autores desenvolveram um algoritmo *Multi-Start Adaptive Large Neighborhood Search* (MSALNS), além de proporem um modelo de programação inteira para

o problema. Os experimentos computacionais foram realizados com as instâncias de Parragh (2011) para o *Dial-a-Ride-Problem* (DARP), como também com instâncias reais providas pela própria PACE, em Wichita, Kansas. Os autores também propuseram novas instâncias para o problema e alcançaram resultados interessantes, com uma redução de custos de até 40%.

Por sua vez, Qu e Bard (2015) lidaram com um problema bastante similar ao Qu e Bard (2013), também relacionado às atividades do PACE que necessitam do uso de veículos com capacidade configurável, contudo, consideram o uso de janelas de tempo mais restritivas. Propuseram um modelo de programação inteira para o problema e um método BCP para resolvê-lo, o qual utiliza *subset row inequalities*, (Jepsen et al., 2008), para o fortalecimento dos LBs encontrados pela geração de colunas e condições de dominância, a fim de acelerar os tempos de execução. Para a validação, foram utilizadas instâncias reais e outras com até 50 clientes, geradas randomicamente, sendo encontradas as soluções ótimas para a maioria delas.

Dando continuidade à descrição das referências que compõem a literatura do PRV-CESFH, Avci e Topaloglu (2016) propuseram um algoritmo auto-ajustável de busca local híbrido (*Hybrid Local Search*, HLS), envolvendo um mecanismo de ajuste de limiar adaptativo (*Threshold adjusting mechanism*) com a meta-heurística Busca Tabu (TS). Em seus experimentos, foi considerada uma frota ilimitada de veículos heterogêneos e utilizados dois conjuntos de instâncias gerados randomicamente, as quais podem chegar até 550 clientes. Em relação aos resultados obtidos, o HLS obteve um melhor desempenho, quando comparado a outros métodos propostos até então, encontrando soluções de boa qualidade a um baixo custo computacional no que se refere ao tempo.

Finalizando o levantamento bibliográfico acerca do PRV-CESFH, Tchoupo et al. (2018) lidaram com uma variante do problema que envolve coleta e entrega, frota heterogênea e janelas de tempo, em inglês denominada *Fleet Size and Mix Pickup and Delivery Problem with Time Windows* (FSMPDPTW), cujo objetivo incluía a minimização do custo total de roteamento, incluindo os custos fixos associados aos diferentes veículos. Para resolver o problema, desenvolveram um algoritmo de geração de colunas, cujos subproblemas de *pricing* foram estabelecidos de tal maneira que cada tipo de veículo possuía o seu próprio. Além disso, foram propostas heurísticas a fim de

acelerar a resolução desses subproblemas. Instâncias próprias, derivadas de *benchmarks* existentes na literatura. Nos testes, 56 dessas instâncias (caracterizadas por conterem até 100 clientes) foram testadas, sendo encontrados LBs para 45% delas e soluções ótimas para 11% desse conjunto.

Por sua vez, Nepomuceno et al. (2019) desenvolveram um algoritmo aleatório de busca pelo vizinho mais próximo denominado *Nearest-Neighbor-Based Randomized Algorithm*, o qual foi validado com as mesmas instâncias propostas por Avci e Topaloglu (2016), obtendo resultados competitivos ao HLS, especialmente para as instâncias com mais de 100 clientes.

Analisando-se o que foi apresentado nesta seção, percebe-se a existência de uma quantidade reduzida de trabalhos envolvendo o PRVCESFH. Especificamente para as instâncias propostas por Avci e Topaloglu (2016), consideradas nos experimentos computacionais referentes a presente pesquisa (v. Seção 5.8), o trabalho em questão neste texto se caracteriza como a primeira aplicação de uma abordagem exata na tentativa de solucioná-las. A Tabela 2.3 sintetiza as informações das obras citadas.

Tabela 2.3: Trabalhos relacionados ao PRVCESFH

Referência	Abordagem
Qu e Bard (2013)	MSALNS
Qu e Bard (2015)	BCP
Avci e Topaloglu (2016)	HLS
Tchoupo et al. (2018)	Geração de colunas
Nepomuceno et al. (2019)	NNRA

2.3 PRVCEs com Janelas de Tempo (PRVCEsJT)

O PRVCEsJT foi tratado pela primeira vez por Angelelli e Mansini (2002), os quais implementaram um método de *Branch-and-Price* (B&P) para a resolução desse problema, baseado em uma formulação de *set covering* para o problema mestre e em uma relaxação do problema de caminho elementar mais curto com janelas de tempo e restrições de capacidade para o problema de *pricing*, com o algoritmo de *Branch-and-Bound* (B&B) sendo utilizado na busca por soluções inteiras. Como não haviam conjuntos de instâncias para esse problema, pode-se ressaltar uma das mais impor-

tantes contribuições desse trabalho, a qual consiste em uma estratégia de separação de demandas, a fim de que instâncias viáveis para o problema pudessem ser produzidas. Mais precisamente, trata-se de um procedimento que, a partir das instâncias de Solomon (1987) para o PRV com Janelas de Tempo (PRVJT), transforma as únicas demandas dos clientes, consideradas originalmente como de entrega, em duas (coleta e entrega), gerando, portanto, instâncias para o PRVCESJT. Tal estratégia de separação foi bastante utilizada em outras pesquisas, conforme poderá ser percebido ao longo desta dissertação.

Anos mais tarde, Mingyong e Erbao (2010) desenvolveram um algoritmo *Improved Differential Evolution* (IDE) para resolver o PRVCESJT, para o qual também propuseram um modelo de programação inteira. Esse método se caracteriza por combinar operadores aritméticos com operadores de *crossover*, seleção e mutação, aplicando-os em soluções iniciais geradas randomicamente. Originalmente, o *Differential Evolution* (DE) não pode ser aplicado a problemas de otimização combinatória, devido a possibilidade de não-linearidade nos valores de alguns parâmetros. No entanto, a fim de torná-lo viável ao PRVCES, mecanismos de adaptação específicos ao problema foram implementados. Vale ressaltar que, segundo os autores, esse trabalho se caracteriza como a primeira tentativa de se utilizar uma meta-heurística para solucionar o problema em questão.

No ano seguinte, Fan (2011) lida com uma versão especial do PRVCES, a qual inclui, como função objetivo, a satisfação do cliente. Mais precisamente, tal variante consiste em um caso particular do PRVCESJT, uma vez que engloba a satisfação do cliente que, por sua vez, é uma medida inversamente proporcional ao tempo de espera, ou seja, quanto menor o tempo de espera dos clientes, maior será a satisfação dos mesmos. Para lidarem com o problema, desenvolveram um algoritmo de Busca Tabu (TS), responsável por melhorar as soluções iniciais obtidas por meio de um método de *Cheap-set Insertion* (CI). Por outro lado, Wang e Chen (2012) desenvolveram um algoritmo genético (GA) com variantes do método CI, além de uma formulação de programação inteira para o problema. Durante os experimentos computacionais, foram geradas instâncias próprias, tendo como base o *benchmark* de Solomon (1987) para o PRVJT, cuja quantidade de clientes varia entre 10 e 100. Resultados computacionais indica-

ram que o método desenvolvido é capaz de obter melhores soluções quando comparado ao CPLEX ou ao GA básico, considerando baixos tempos de execução. Já Kassem e Chen (2013) propuseram uma formulação matemática para o PRVCESJT, além de terem desenvolvido um método heurístico baseado no algoritmo do *Simulated Annealing* (SA) com o intuito de tentar resolvê-lo. Novamente foram geradas instâncias próprias a partir do *benchmark* de Solomon (1987).

Dando continuidade, Liu et al. (2013) lidaram com um caso especial do PRVCESJT, aplicado à área de *home health care*. Para esse problemas foram consideradas quatro tipos de demandas: entregas do depósito aos pacientes, do hospital aos pacientes, coletas dos pacientes aos depósitos e dos pacientes aos laboratórios médicos (Liu et al., 2013). Além de duas formulações de programação inteira, desenvolveram um método que combina as meta-heurísticas GA e TS. Para os experimentos computacionais, derivaram instâncias dos *benchmarks* de Solomon (1987) e Gehring e Homberger (1999) para o PRVJT. Os resultados obtidos apontaram a capacidade do método de encontrar boas soluções em intervalos de tempo razoáveis. Vale ressaltar também que testes foram realizados com instâncias do PRVCEM com Janelas de Tempo (PRVCEMJT). No mesmo ano, Wang et al. (2013) desenvolveram uma meta-heurística SA para o PRVCESJT, sendo obtidos resultados com uma melhora em torno de 0,22%, na média para as instâncias de Wang e Chen (2012), quando comparado aos resultados dos criadores do *benchmark*.

Ainda na descrição dos trabalhos envolvendo o PRVCESJT, Wang et al. (2015) consideraram a utilização de um método denominado *parallel Simulated Annealing* (p-SA), envolvendo também a heurística baseada em inserção *Residual Capacity and Radial Surcharge* (RCRS), para resolverem o problema. Uma formulação matemática também foi proposta, cujo objetivo consiste em minimizar o custo total, no qual estavam inseridos os custos associados aos veículos e ao transporte (custo de viagem). Para os testes computacionais, foram utilizadas, novamente, as instâncias de Wang e Chen (2012), classificadas como de pequeno e médio porte, contudo, também foram propostas instâncias de grande porte, isto é, contendo de 200 a até 1000 clientes, derivadas do *benchmark* de Gehring e Homberger (1999). Dentre os resultados obtidos para as instâncias de pequena e média escala, o *p-SA* foi capaz de reduzir o número de veículos

utilizados de 12 instâncias e, dentre as restantes, 16 delas tiveram seus custos de viagem melhorados, quando comparado ao GA de Wang e Chen (2012).

Além disso, Shi et al. (2018) implementaram um *Efficient Tabu Search based Procedure* (ETSP) para o PRVCESJT, tendo como objetivo a minimização do número de veículos utilizados e dos custos totais de viagem. Esse método é composto por dois tipos de estratégias de TS, denominadas TS I e TS II as quais, de um modo simplificado, diferenciam-se em relação à dimensão da lista tabu utilizada, bidimensional e tridimensional, respectivamente. O método ETSP, portanto, alterna entre as aplicações dessas duas estratégias. Quanto aos experimentos computacionais, estes foram realizados com as instâncias de Wang e Chen (2012). Os resultados desses experimentos foram comparados aos reportados por Wang e Chen (2012) e Wang et al. (2015), sendo obtidas melhorias em algumas soluções já conhecidas. Já Hof e Schneider (2019), por sua vez, abrangeram uma classe de problemas, a saber: PRVCES com Tempo Limite (PRVCESTL), no qual é estabelecida uma duração máxima para cada rota; o próprio PRVCESJT e uma variante do PRV com demandas de coleta e entrega divisíveis (PRVCED). Para esta última, também analisaram os casos em que há restrições de manutenção de uma porcentagem da capacidade dos veículos desocupada, quando são requisitados os serviços de coleta e entrega simultaneamente, PRVCEMRD, bem como com a inserção de janelas de tempo, PRVCEDJT, totalizando, portanto, cinco variantes de problemas abordadas. A fim de lidarem com todos os problemas citados, desenvolveram um método híbrido que combina as heurísticas ALNS com PR, denominado ALNS-PR. Particularmente para o PRVCESJT, variante de interesse nesta seção, foram realizados experimentos computacionais com as instâncias de Wang e Chen (2012) e de Wang et al. (2015), a partir dos quais foram obtidos resultados bastante competitivos como, por exemplo, a redução do número de veículos utilizados para 17 dentre as 56 instâncias de pequeno e médio porte e para 22 dentre as 30 instâncias de grande porte.

Shi et al. (2020) propuseram um algoritmo de dois estágios baseados em aprendizagem, que combina os métodos de busca em vizinhança variável (VNS) modificado com um método TS bi-estruturado (*Bi-Structure Tabu Search*, BSTS) para lidar com o PRVCESJT, mantendo-se o duplo objetivo de minimização do número de veículos

utilizados e do custo total de viagem. Mais precisamente, o VNS é utilizado no primeiro estágio do algoritmo, a fim de tentar cumprir o primeiro objetivo, enquanto que o BSTS está presente na segunda etapa, utilizado na tentativa de cumprimento de ambas as metas. Durante os testes realizados, foram utilizadas as instâncias de Wang e Chen (2012), cujos resultados foram comparados com o algoritmo genético dos mesmos autores e com o p-SA de Wang et al. (2015). A análise dos resultados obtidos demonstrou que o VNS-BSTS é capaz de encontrar soluções de boa qualidade e melhorar soluções já conhecidas.

No ano seguinte, Liu et al. (2021) desenvolveram um *Memetic Algorithm* (MA) *with efficient local Search and Extended neighborhood* (MATE) para o PRVCESJT. O método proposto introduz novos componentes, tais como combinação entre heurísticas de construção e métodos de busca populacional, novo operador de *crossover* e modificações no método de busca local, sendo todos esses componentes incorporados ao MA. Para os experimentos computacionais, foram novamente utilizadas as instâncias de Wang e Chen (2012) e comparado o MATE com os métodos propostos por Wang e Chen (2012), Wang et al. (2015), Hof e Schneider (2019) e Shi et al. (2020). Da análise dos resultados, pôde ser inferido que o MATE foi capaz de encontrar novas melhores soluções para 12 dentre as 65 instâncias testadas. Outra contribuição dos autores está relacionada à proposição de um novo conjunto de instâncias baseado em um estudo de caso real.

Por fim, Agius et al. (2022) propuseram um novo problema denominado *Multi-Trip Vehicle Routing Problem with Mixed Pickup and Delivery, and Release and Due dates* (MTMPD-RD), no contexto de transporte de pacientes em casos não emergenciais. Tais pacientes precisam ser transportados de suas casas ao hospital, para que sejam realizadas as consultas médicas, e, posteriormente, devem ser transportados de volta às suas casas, respeitando-se os tempos associados a cada uma das consultas (*releases and due dates*). Para resolverem o problema, desenvolveram um algoritmo B&P, validando-o com instâncias reais. Além disso, a fim de serem testadas instâncias da literatura, desenvolveram um método de simplificação do problema para o PRVCESJT e realizaram testes com as instâncias propostas por Wang e Chen (2012), provando a otimalidade para 11 instâncias do conjunto.

Dentre as referências citadas nesta seção, observa-se uma predominância da aplicação de métodos heurísticos. Considerando os conjuntos de instâncias criados por Wang e Chen (2012), utilizados nos experimentos computacionais envolvendo o PRVCESJT (v. Seção 5.5), os trabalhos que mais se destacam são o do próprios criadores de tais instâncias (Wang e Chen, 2012), de Wang et al. (2015), Shi et al. (2018), Hof e Schneider (2019), Shi et al. (2020) e, por fim, de Liu et al. (2021). Da análise dos mesmos, são obtidos os BKS para essas instâncias que, por sua vez, são utilizados como *upper bounds* (UBs) iniciais para o método exato proposto nesta dissertação. Na Tabela 2.4, estão indicados todos os trabalhos citados nesta seção, com ênfase nas abordagens utilizadas em cada um deles.

Tabela 2.4: Trabalhos relacionados ao PRVCESJT

Referência	Abordagem
Angelelli e Mansini (2002)	B&P
Mingyong e Erbao (2010)	IDE
Fan (2011)	TS + CI
Wang e Chen (2012)	GA + CI
Kassem e Chen (2013)	SA
Liu et al. (2013)	GA + TS
Wang et al. (2013)	SA
Wang et al. (2015)	p-SA
Shi et al. (2018)	ETSP
Hof e Schneider (2019)	ALNS-PR
Shi et al. (2020)	VNS-BSTS
Liu et al. (2021)	MATE
Agius et al. (2022)	B&P

2.4 Problema de Roteamento de Veículos com Coleta e Entrega Mistas (PRVCEM)

Os primeiros trabalhos envolvendo o PRVCEM surgiram quando Golden et al. (1985) propuseram um método de inserção de *backhauls* (*stop-based*), no qual são construídas, inicialmente, rotas com apenas os clientes cujas demandas são de entrega e, então, analisa-se as possibilidades de inserção dos clientes de coleta. Similarmente, Casco et al. (1988) também utilizaram um procedimento de inserção, contudo, consideraram uma penalidade associada à carga de entrega após a coleta, aplicada aos

custos inerentes à inserção dos clientes que demandam serviços de coleta (*load-based*). Dez anos depois, Mosheiov (1998) propôs uma heurística de particionamento, responsável por particionar as rotas em segmentos disjuntos atendidos por diferentes veículos. Inicialmente utilizada para resolver o PRV com capacidade, foi adaptada ao problema com coletas e entregas.

Dando continuidade, Salhi e Nagy (1999), Nagy e Salhi (2005) e Ropke e Pisinger (2006) lidaram em suas pesquisas com situações envolvendo um e múltiplos depósitos, isto é, PRVCEM e PRVCEMMD. Desse modo, por introduzirem a generalização com mais de um depósito do PRVCEM, os métodos por eles desenvolvidos estão descritos na Seção 2.5. Já Dethloff (2002) aplicou a mesma heurística de inserção desenvolvida para o PRVCEM ao PRVCEMMD, tendo em vista as similaridades existente entre ambos os problemas. No mesmo ano, Wade e Salhi (2002) lidaram com um caso particular, no qual não há a necessidade de servir os clientes *backhauls*, apenas após todos os clientes *linehauls* terem sido atendidos, contudo, a ordem entre eles não é completamente livre, caracterizando, assim, um PRVCEM restrito. Para resolverem o problema, propuseram um algoritmo baseado em inserção restrita (R-INS). Por sua vez, Wade e Salhi (2004) desenvolveram um método de colônia de formigas (ACS) para lidarem com o PRVCEM, propondo algumas melhorias em relação ao método tradicional, especialmente no que se refere a construção da lista de candidatos a comporem as rotas, um mecanismo de antecipação aplicado à análise de viabilidade e regras eficientes relacionadas às atualizações locais e globais. Já Crispim e Brandão (2005) combinaram duas meta-heurísticas, TS e VND, para resolverem tanto o PRVCEM quanto o PRVCEMMD.

Wassan et al. (2008) fizeram uma análise da relação entre o PRVCEM e o PRVCEMMD, investigando se abordagens desenvolvidas para aquele podem ser aplicadas na resolução deste. Além disso, desenvolveram uma meta-heurística TS, capaz de encontrar boas soluções para os problemas. No ano seguinte, Gajpal e Abad (2009) utilizaram um algoritmo ACS, com regras de construção e esquemas de busca local multi-rota, para resolver o PRVCEMMD. Apesar disso, o método também foi aplicado em instâncias do PRVCEM. No mesmo ano, Tütüncü et al. (2009) propuseram um sistema de suporte a decisão baseado no *Greedy Randomised Adaptive Memory Programming Search* (GRAMPS) para resolver o PRV com *backhauls* clássico, além do PRVCEM e do PRV-

CEM restrito.

Alguns anos depois, Belmecheri et al. (2013) desenvolveram um método heurístico baseado no algoritmo PSO para lidar com uma variante mais complexa do PRV-CEM, na qual estão inclusos o uso de frota heterogênea e de janelas de tempo (PRV-CEMFHJT). Para os experimentos computacionais, foram adaptadas as instâncias de Solomon (1987) para o PRV com Janelas de Tempo, com o PSO superando outros métodos já existentes em 34 das 56 instâncias testadas. Por outro lado, conforme descrito na Seção 2.1, Subramanian et al. (2013a) desenvolveram um método híbrido ILS-RVND-SP para diversas classes de PRV, dentre as quais estão inclusos o PRVCES e o PRVCEM. Além disso, Avci e Topaloglu (2015) desenvolveram uma abordagem de busca local adaptativa auto-ajustável (*Adaptive Local Search*, ALS), que combina a meta-heurística VND com um algoritmo inspirado no SA. Para os experimentos computacionais, foram utilizadas as instâncias de Dethloff (2001) e Salhi e Nagy (1999) voltadas ao PRVCES, bem como as instâncias de Wassan et al. (2008) inerentes ao PRVCEM. Dentre os resultados obtidos, o ALS demonstrou ser um método livre de parâmetros capaz de encontrar soluções de qualidade equiparável a outros métodos mais sofisticados existentes na literatura, o que representa vantagens de aplicabilidade prática.

No ano seguinte, Berghida e Boukra (2016) também atuaram na proposição de um método de resolução para o PRVCEMFHJT, assim como feito por Belmecheri et al. (2013). Neste caso, desenvolveram um algoritmo inspirado em princípios quânticos e que utiliza tamanho de população variável, a fim de diminuir o número de avaliações de solução. Por sua vez, Belloso et al. (2017) propuseram um algoritmo que combina um *biased-randomized saving heuristic* com um *framework* meta-heurístico, a fim de resolverem, inicialmente, instâncias do PRV com *backhauls* “clusterizado”, contudo, a aplicação do método foi estendida para englobar o PRVCEM. Focando especificamente neste, foram testadas as instâncias propostas por Salhi e Nagy (1999) e obtidos resultados, após rápida adaptação do método, que implicaram gaps inferiores, na média, a 0,4%. Já Reil et al. (2018) consideraram uma variante do PRV com *backhauls*, a qual engloba o uso de janelas de tempo, bem como de restrições de carregamento tridimensional, tendo em vista os custos inerentes ao processo de carga/descarga do veículo a medida que os produtos são entregues ou coletados. Uma abordagem de duas fases

foi proposta, com as questões de empacotamento resolvidas na primeira, enquanto a segunda era responsável pelo roteamento, aplicando-se uma estratégia *multi-start* evolutiva para tal. Os autores demonstram que diferentes variantes do PRV com *backhauls* podem ser analisadas por meio desse *framework*, incluindo o PRVCEM. Experimentos computacionais demonstraram que soluções de boa qualidade puderam ser encontradas em baixos tempos de execução. Ainda como contribuição, foram geradas novas instâncias, randomicamente, para o problema.

Pinto et al. (2020) retomaram a consideração de restrições de carregamento para o PRVCEM, contudo em uma análise bidimensional. Para essa variante, já haviam sido propostas uma heurística de inserção por Pinto et al. (2015), bem como um algoritmo VNS por Pinto et al. (2017), sendo a referência em questão uma extensão deste. Assim como Reil et al. (2018), Koch et al. (2020) lidaram com o PRVCEM envolvendo restrições de carregamento tridimensionais. Para resolverem o problema, desenvolveram uma meta-heurística híbrida, combinando o método TS, para o roteamento, com heurísticas de *packing*. No mesmo ano, Santos et al. (2020) realizaram uma análise da literatura do PRV com *backhauls*, sob uma perspectiva sustentável, cobrindo questões ambientais, objetivos sociais, redes colaborativas e logística reversa. Dessa análise, os autores observaram que, considerando os princípios de sustentabilidade, muitos esforços estão sendo despendidos em pesquisas envolvendo consumo de combustível e emissão de CO₂.

Métodos exatos também foram propostos. Dentre as referências nas quais abordagens exatas foram aplicadas ao PRVCEM, encontram-se os trabalhos de Subramanian et al. (2011), Subramanian (2012) e Subramanian et al. (2013b) que, por também englobarem o PRVCEM, estão descritos com mais detalhes na Seção 2.1. Por sua vez, Yang et al. (2020) reformularam o PRVCEM com Janelas de Tempo (PRVCEMJT) como um modelo de otimização de fluxo em redes de *multi-commodity* de tempo discretizado com coletas e entregas baseadas em um *framework* de estado-espaco-tempo. Essa abordagem engloba o uso de técnicas como Relaxação Lagrangiana Aumentada (ALR), Programação Dinâmica (DP) e o método de bloco não-linear de Gauss-Seidel. Para os experimentos computacionais, foram utilizadas instâncias baseadas em redes com 9 nós e 28 arcos (9-*node*), como também instâncias reais. Os resultados obtidos

demonstram que o ALR converge mais rapidamente do que os métodos LR puros.

Similarmente ao trabalho de Santos et al. (2020) no que se refere à consideração de uma perspectiva sustentável, Yang et al. (2021) propuseram uma nova variante para o PRVCEM, considerando janelas de tempo e estratégias de recarregamento de baterias no contexto de veículos elétricos. Assim como Yang et al. (2020), um modelo de otimização de fluxo em redes de *multi-commodity* de tempo discretizado é utilizado na resolução do problema, aplicando-se a técnica ALR e o método *Alternating Direction Multiplier Method* (ADMM). Outra semelhança pode ser observada em relação às instâncias utilizadas, uma vez que estas foram baseadas em redes com 7 nós (*7-node*) e outras construídas com dados reais.

Devido à similaridade entre o PRVCEM e o PRVCEM, novamente os métodos B&C com *lazy separations* e BCP de Subramanian (2012) se destacaram dentre as abordagens propostas, apresentando os melhores resultados no que se refere aos testes com as instâncias propostas por Dethloff (2001), Salhi e Nagy (1999) e Montané e Galvão (2006). Desse modo, comparações entre a abordagem proposta nesta dissertação e os métodos citados são realizadas (v. Seção 5.2). Na Tabela 2.5 são lembradas as abordagens utilizadas em cada um dos artigos considerados nesta seção.

Tabela 2.5: Trabalhos relacionados ao PRVCEM

Referência	Abordagem
Golden et al. (1985)	Heurística de inserção <i>stop-based</i>
Casco et al. (1988)	Heurística de inserção <i>load-based</i>
Mosheiov (1998)	Heurística de particionamento
Salhi e Nagy (1999)	Heurística de inserção em <i>cluster</i>
Dethloff (2002)	Heurística de inserção
Wade e Salhi (2002)	R-INS
Wade e Salhi (2004)	ACS
Nagy e Salhi (2005)	Abordagem heurística composta
Crispim e Brandão (2005)	TS + VND
Ropke e Pisinger (2006)	<i>Framework</i> com LNS
Wassan et al. (2008)	TS
Gajpal e Abad (2009)	ACS
Tütüncü et al. (2009)	GRAMPS
Subramanian et al. (2011)	B&C com <i>lazy separations</i>
Subramanian (2012)	Abordagens exatas, heurísticas e híbridas
Subramanian et al. (2013a)	ILS-RVND-SP
Subramanian et al. (2013b)	BCP
Belmecheri et al. (2013)	Heurística baseada no PSO
Avci e Topaloglu (2015)	ALS
Pinto et al. (2015)	Heurística de inserção
Berghida e Boukra (2016)	Algoritmo inspirado em princípios quânticos
Belloso et al. (2017)	<i>biased-randomized saving heuristic</i>
Pinto et al. (2017)	VNS
Reil et al. (2018)	Estratégia de duas fases
Yang et al. (2020)	ALR + DP + método de Gauss-Seidel
Pinto et al. (2020)	<i>General VNS</i>
Koch et al. (2020)	TS + heurísticas de <i>packing</i>
Santos et al. (2020)	Algoritmos exatos e meta-heurísticos (<i>review</i>)
Yang et al. (2021)	ALR + ADMM

2.5 PRVCEM com Múltiplos Depósitos (PRVCEMMD)

O PRVCEMMD foi introduzido por Salhi e Nagy (1999), responsáveis também por conceituarem as diferenças entre o PRVCES e o PRVCEM. Os autores desenvolveram uma heurística de inserção em *cluster* aplicada a ambos os problemas, bem como à generalização destes com múltiplos depósitos, dando origem, portanto, ao PRVCEMMD e o PRVCESMD. Outra contribuição dos mesmos está relacionada a criação de um conjunto de 14 instâncias contendo 1 depósito e variando a quantidade de clientes entre 50

e 199, bem como 11 instâncias contendo de 2 a 5 depósitos e de 50 a 249 clientes. Além disso, assim como Angelelli e Mansini (2002), também desenvolveram uma estratégia de separação de demandas para a geração de instâncias para o problema, a qual também obteve um papel importante para outras pesquisas. Dentre resultados, não houve melhorias substancialmente significativas, contudo, foram alcançados pequenos avanços a custos computacionais baixos. Anos mais tarde, Nagy e Salhi (2005) desenvolveram uma abordagem heurística para os mesmos problemas, incluindo as generalizações com múltiplos depósitos. Essa abordagem tinha como característica uma permissão inicial de soluções viáveis apenas ao PRV e, depois de encontradas, eram aplicadas técnicas de melhoramento para torná-las viáveis também ao PRVCES (ou PRVCEM). Para a validação da heurística desenvolvida, foram realizados experimentos computacionais com as mesmas instâncias de Salhi e Nagy (1999), sendo obtidos, em geral, resultados melhores do que aqueles já conhecidos na época.

Por sua vez, Ropke e Pisinger (2006) construíram um *framework* capaz de resolver diversas variantes do PRV com *Backhauls*, incluindo o PRVCEMMD. Todos esses problemas são transformados em casos particulares do *Rich Pickup and Delivery Problem with Time Windows* (*Rich PDPTW*), com a justificativa de serem extensões do PDPTW. Feito isso, aplica-se a heurística *Large Neighborhood Search* (LNS) para resolvê-los e, então, as soluções obtidas são adaptadas ao correspondente problema original. Quanto aos experimentos computacionais, foram realizados testes com nove conjuntos de instâncias presentes na literatura, sendo obtidas soluções melhores para 67% dos problemas testados.

Alguns anos depois, Subramanian (2012) e Subramanian et al. (2013a) lidaram com o PRVCEMMD, aplicando métodos exato (B&C com *lazy separations*) e híbrido (ILS-RVND-SP) para resolverem o problema, os quais estão descritos na Seção 2.1. Por sua vez, Li et al. (2015) propuseram uma heurística que combina o ILS com um mecanismo de seleção de vizinhança adaptativo, denominado *Adaptive Neighborhood Selection* (ILS_ANGS). Além disso, foram propostos novos métodos de perturbação de vizinhança. Para a validação da heurística desenvolvida, foram utilizadas as mesmas instâncias propostas por Salhi e Nagy (1999) e obtido um desempenho superior a outros métodos utilizados para o problema. Já Koulaeian et al. (2015) desenvol-

veram um modelo matemático para o PRVCESMD, considerando a heterogeneidade da frota. O objetivo deles era a minimização do custo total de roteamento, incluindo custos fixos associados aos motoristas e penalidades em caso de descumprimento dos tempos limite associados às rotas. Para lidar com o problema, foram desenvolvidos uma meta-heurística *Imperialist Competitive Algorithm* (ICA) e um algoritmo genético (GA). Testando-se instâncias geradas pelos próprios autores, percebeu-se que o ICA se sobrepôs ao GA no que se refere a desempenho, especialmente para as instâncias maiores.

Por fim, Wang et al. (2021) lidam com uma versão particular do PRVCEM envolvendo compartilhamento de recursos, seja por compartilhamento de informações dos clientes, considerando o agrupamento dos mesmos, ou de recursos relacionados ao transporte, tendo em vista que um mesmo veículo pode ser utilizado múltiplas vezes por uma ou múltiplas *logistics facilities*. Para resolverem esse problema, foi combinado um modelo de programação matemática de duplo objetivo, responsável pela minimização dos custos e do total de veículos utilizados, com um método híbrido de duas fases, composto pelos algoritmos de k-médias, de Clark-Wright (CW), (Clarke e Wright, 1964), e o *Nondominated Sorting Genetic Algorithm II* (NSGA-II). Para a avaliação do método proposto, foram utilizadas instâncias reais obtidas em Chongqing, China, bem como instâncias conhecidas referentes ao PRVCEMMDJT. Após os testes, foi ratificada a aplicabilidade prática do algoritmo proposto.

Dá análise das referências citadas, percebe-se a dominância de métodos heurísticos sobre os exatos no que se refere ao PRVCEMMD. No entanto, resultados interessantes foram propostos por Subramanian (2012) e, por essa razão, o método por ele proposto (B&C com *lazy separations*) é comparado com a abordagem exata desenvolvida na presente pesquisa (v. Seção 5.9). Um resumo de algumas informações relevantes desses artigos estão dispostas na Tabela 2.6.

Tabela 2.6: Trabalhos relacionados ao PRVCEMMD

Referência	Abordagem
Salhi e Nagy (1999)	Heurística de inserção em <i>cluster</i>
Nagy e Salhi (2005)	Abordagem heurística composta
Ropke e Pisinger (2006)	<i>Framework</i> com LNS
Subramanian (2012)	Abordagens exatas, heurísticas e híbridas
Subramanian et al. (2013a)	ILS-RVND-SP
Li et al. (2015)	ILS_ANS
Koulaeian et al. (2015)	ICA e GA
Wang et al. (2021)	Modelo matemático + (k-médias, CW e NSGA-II)

2.6 Problema de Roteamento e Localização com Coleta e Entrega Simultâneas (PRLCES)

Dentre as referências que compõem o estado da arte do PRLCES, destacam-se os artigos do Karaoglan et al. (2011) e Karaoglan et al. (2012). A primeira utiliza os conjuntos de instâncias (com até 100 clientes) propostos por Prins et al. (2006) — ou ainda (Prins et al., 2004) e (Prodhon, 2006), as quais são referências relacionadas a esse conjunto de instâncias — e Barreto et al. (2007), gerando as demandas de coleta e entrega a partir das estratégias de separação de demandas propostas por Salhi e Nagy (1999) e Angelelli e Mansini (2002), para validar o algoritmo de Branch-and-Cut (B&C) que propõem. Eles comparam os resultados do B&C proposto com um modelo MIP para o problema, resolvido com o CPLEX 11.1, e com uma variante do algoritmo de B&C, a qual não faz uso da algoritmo SA para melhorar os UBs obtidos dos nós da árvore de enumeração. Como resultados, foram resolvidas 55 dentre as 148 instâncias consideradas, indicando que o método proposto é uma abordagem viável para instâncias de pequeno e médio portes.

Por sua vez, Karaoglan et al. (2012) também utilizaram o conjunto de instâncias do Prins et al. (2006), bem como as estratégias de separação de demandas propostas por Salhi e Nagy (1999) e Angelelli e Mansini (2002). Eles propuseram duas formulações MIP, baseadas em fluxo e nos nós (ou vértices), e uma família de inequações válidas para fortalecê-las. Além disso, propuseram também uma abordagem heurística de duas fases baseadas no SA (tp-SA) para resolução das instâncias consideradas grandes

e duas heurísticas de inicialização para o tp-SA. Como experimentos computacionais, compararam as formulações e heurísticas propostas entre elas mesmas, considerando o tamanho das instâncias testadas. Dentre os resultados obtidos, foi observado que a formulação baseada em fluxo teve um melhor desempenho do que a de nós, seja em termos de qualidade de solução ou de tempo computacional, para as instâncias pequenas, enquanto a segunda formulação é mais eficiente para as instâncias médias.

Continuando, Yu e Lin (2014) implementaram uma heurística *Multi-Start Simulated Annealing* (MSA) para resolver o PRLCES. Todos os experimentos computacionais foram feitos com as instâncias elaboradas por Karaoglan et al. (2012), obtendo-se resultados como a descoberta de novas melhores soluções para 126 dentre as 360 instâncias testadas.

Já Karaoglan e Altiparmak (2015) seguiram a mesma estratégia de Karaoglan et al. (2011), isto é, as mesmas instâncias e estratégias de separação são utilizadas. Além disso, fizeram comparações com o B&C dos mesmos, para validar o *memetic algorithm* que propõem, o qual é baseado em algoritmos genéticos, SA e em formulação de programação inteira. O algoritmo foi responsável pela obtenção de soluções viáveis com *gap* percentual médio de 0,43 para instâncias testadas e melhoria da melhor solução conhecida para uma delas.

Além disso, Yu e Lin (2016) também utilizaram o SA na resolução de instâncias do problema em questão. Os autores consideraram em seus experimentos as instâncias de Karaoglan et al. (2011), como também propuseram novas instâncias consideradas de grande porte, isto é, com mais de 100 clientes. Em relação aos resultados, quando comparado ao B&C com SA dos criadores das instâncias, o método proposto obteve um melhor desempenho.

Anos mais tarde, Zhao et al. (2019) propuseram uma abordagem *hyper-heuristic* baseada em *Iterated Local Search* para resolver o PRLCES. Em seus experimentos computacionais, utilizaram novamente as instâncias propostas por Prins et al. (2006) e Barreto et al. (2007), bem como as instâncias de Tuzun e Burke (1999), com pequenas modificações em relação ao cálculo das distâncias euclidianas, se comparado ao que foi feito por Karaoglan et al. (2011) e Yu e Lin (2016). Como estratégias de separação de demandas, também se basearam em (Angelelli e Mansini, 2002) e (Salhi e Nagy,

1999). O método proposto foi capaz de encontrar melhores resultados para as instâncias do Barreto et al. (2007), com redução de pelo menos 30% em relação aos tempos de execução, quando comparado a outras *hyper-heuristics* clássicas.

Huang et al. (2020) lidaram com uma variante do PRLCES, na qual, ao invés da simultaneidade, considera-se demandas de coleta e entre mistas, ou seja, PRLCEM. Para resolverem esse problema, os autores desenvolveram uma heurística denominada CWIGALNS, a qual combina o algoritmo CW, um método Iterativo Guloso (IG) e a meta-heurística ALNS. Além disso, uma formulação envolvendo dois modelos de programação inteira foi proposta para o problema. Experimentos computacionais foram realizados com instâncias derivadas de *benchmarks* clássicos do PRV, provando que heurística é capaz de encontrar boas soluções para o problema. Outra contribuição foi a utilização de um estudo de caso real para demonstrar a aplicabilidade prática dos modelos e método propostos.

No mesmo ano, Zhao et al. (2020) utilizou uma hiper-heurística baseada em evolução (*Evolution-based hyperheuristic*, EHH) para resolver o PRLCES, dando continuidade ao trabalho desenvolvido anteriormente (Zhao et al., 2019). Uma diferença quando comparado a este é a generalização da aplicação do método proposto para o PRL em geral, não se limitando apenas ao PRLCES. Durante os experimentos computacionais, foram utilizadas as instâncias de Prins et al. (2007), Barreto et al. (2007) e Tuzun e Burke (1999), com a estratégia de separação de demandas de Angelelli e Mansini (2002). Os resultados computacionais obtidos apontam o bom desempenho do método proposto.

Em relação às variantes do problema em questão nesta seção, Huang (2015) lidou com uma versão avançada de tal problema, denominada *Multi-compartment capacitated Location Routing Problem with Pickup-Delivery Routes and Stochastic Demands* (MLRPPDRSD), a qual se caracteriza por possuir conjuntos de rotas distintos para as demandas de coleta e entrega, além de demandas aleatórias. Para solucionar esse problema, foi proposto um método baseado em TS, bem como o conjunto de instâncias, tendo em vista que, até o momento de escrita desse artigo, não havia um *benchmark* padrão para essa variante. Outra variante foi abordada por Gianessi et al. (2016), *The Multicommodity-Ring Location Routing Problem* (MRLRP), cujo objetivo é localizar

um conjunto de centro de distribuição urbanos (*Urban Distribution Centers*, UDCs) e conectá-los em um anel, por meio do qual serão transportados os bens. Para resolvê-lo, utilizaram abordagens exatas, heurísticas e híbridas, a depender dos tamanhos das instâncias analisadas. Nos experimentos, derivaram um conjunto de instâncias para o MRLRP a partir do *benchmark* proposto por Prins et al. (2006)

Por sua vez, Karimi (2018) lidou com o PRLCES (*hub location*) considerando tempo de viagem predefinido, capacidade dos *hubs* e dos veículos, bem como demandas de coleta e entrega simultâneas. Propuseram uma formulação de programação inteira mista para lidar com a alocação, bem como uma heurística baseada no TS para determinar as alocações de *hubs* e as rotas dos veículos, simultaneamente. Nos testes, utilizaram os conjuntos de instâncias conhecidos na literatura como *Australian Post* (AP), disponíveis na OR-Library <http://people.brunel.ac.uk/~mastjjb/jeb/info.html>, e *Turkish Network* (TN), proposta por Tan e Kara (2007). Já Nadizadeh e Kafash (2019) lidaram com um variante do PRLCES, na qual as demandas dos clientes são variáveis *fuzzy*, utilizando um método de clusterização guloso (*Greedy Clustering Method*, GCM) para resolvê-la. Para a validação do método, adaptaram algumas instâncias de Karaoglan et al. (2011).

Por fim, Mamaghani e Davari (2020) incluíram janelas de tempo ao PRVCS. Tinham como objetivo determinar as melhores localizações para um conjunto de depósitos; alocar grupos de clientes para cada um desses depósitos; alocar os dias de atendimento aos clientes e encontrar rotas ótimas a serem percorridas por uma frota homogênea de veículos, sob os critérios de minimização da distância total percorrida e da violação dos limites de tempo dos clientes. Para resolverem o problema, foram utilizados algoritmos de solução bi-objetiva, denominados NSGA-II e NRGGA. Em seus experimentos computacionais, foram adaptadas 45 instâncias de Karaoglan et al. (2012), utilizando a estratégia de separação de demandas de Angelelli e Mansini (2002) e considerando apenas o parâmetro $\gamma = 0,8$.

Em suma, na Tabela 2.7 adiante está presente um resumo das informações acerca das referências apresentadas nesta seção. Novamente, as abordagens exatas são minoria perante à quantidade de trabalhos envolvendo métodos heurísticos. Apesar disso, devido ao fato de serem utilizadas as instâncias propostas por Karaoglan et al. (2011)

nos experimentos computacionais associados à presente pesquisa, o método exato sendo proposto é comparado ao algoritmo B&C com SA dos próprios autores responsáveis pelas instâncias, conforme abordado na Seção 5.10, a partir do qual já foram apresentados importantes resultados advindos da resolução do problema.

Tabela 2.7: Trabalhos relacionados ao PRLCES

Referência	Abordagem
Karaoglan et al. (2011)	B&C com SA
Karaoglan et al. (2012)	tp_SA
Yu e Lin (2014)	MSA
Karaoglan e Altiparmak (2015)	<i>Memetic Algorithm</i>
Yu e Lin (2016)	Heurística baseada no SA
Zhao et al. (2019)	<i>Hyper-heuristic</i> baseada em ILS
Huang et al. (2020)	CWIGALNS
Zhao et al. (2020)	EHH
Huang (2015)	TS
Gianessi et al. (2016)	Abordagens exatas, heurísticas e híbridas
Karimi (2018)	
Nadizadeh e Kafash (2019)	GCM
Mamaghani e Davari (2020)	NSGA-II e NRGa

2.7 Outras variantes

Nesta seção serão apresentadas outras variantes do PRVCES que foram consideradas nesta pesquisa. Dentre elas, encontram-se o PRVCESTL e o PRVCEMTL, os quais também foram abordados nos trabalhos de Subramanian (2012) e Subramanian et al. (2013a) juntamente com o PRVCES e o PRVCEM. Precisamente, foram aplicados os métodos ILS-RVND e ILS-RVND-SP nas instâncias do conjunto proposto por Salhi e Nagy (1999) que apresentam limites de duração de rota, cujos resultados implicaram o alcance das melhores soluções conhecidas até então ou mesmo a melhoria de algumas delas.

Além disso, como descrito na Seção 2.1, Rieck e Zimmermann (2013) também consideraram instâncias assimétricas, derivadas tanto de problemas reais quanto dos conjuntos Dethloff (2001) e Salhi e Nagy (1999), para validarem as formulações de fluxo que propuseram. Assim como eles, Agarwal e Venkateshan (2020) lidaram com o PRVCESA, propondo uma nova formulação matemática englobando quatro novas classes de inequa-

ções válidas para o problema, bem como fizeram a generalização para os *no-good cuts*. Ambas as contribuições implicaram em reduções no tamanho da árvore de *Branch-and-Bound* (B&B) e do tempo computacional necessário (Agarwal e Venkateshan, 2020). Ainda dentre as contribuições, encontram uma maneira de caracterizar o nível de dificuldade de um problema, sendo tal descoberta utilizada na geração de instâncias difíceis para o PRVCESA. Tais instâncias foram utilizadas durante os experimentos computacionais junto a algumas instâncias com 45 nós geradas randomicamente e àquelas utilizadas no trabalho de Rieck e Zimmermann (2013), correspondentes ao acréscimo da assimetria para os conjuntos de Dethloff (2001) e Salhi e Nagy (1999).

Por fim, Wang e Chen (2013) consideraram uma versão flexível do PCE com Janelas de Tempo, ao combinar clientes com demandas de coleta e/ou entrega, denominada PRVCES com Janelas de Tempo Flexíveis. Nesse problema, se um cliente possui ambos os tipos de demandas, pode ser requerido que as mesmas sejam atendidas em janelas de tempo distintas, ou seja, são estipuladas janelas de tempo distintas a cada um dos serviços. O objetivo dessa distinção está atrelado à busca pela minimização do tempo de acesso aos clientes. Para resolverem o problema, desenvolveram um algoritmo co-evolucionário (CEA), no qual são consideradas populações distintas para as operações de diversificação e intensificação. Para os testes realizados, foram construídas novas instâncias a partir do *benchmark* de Wang e Chen (2012). Vale ressaltar que uma formulação matemática para essa versão do problema também foi apresentada. A Tabela 2.8 resume as informações apresentadas nesta seção.

Tabela 2.8: Outras variantes do PRVCES

Referência	Variante	Abordagem
Subramanian (2012)	PRVCE(S,M)TL	Abordagens exatas, heurísticas e híbridas ILS-RVND-SP CEA
Subramanian et al. (2013a)	PRVCE(S,M)TL	
Wang e Chen (2013)	PRVCESJTF	
Rieck e Zimmermann (2013)	PRVCESA	Formulações (fluxo de veículos e de commodities)
Agarwal e Venkateshan (2020)	PRVCESA	Formulação. Generalização dos <i>no-good cuts</i>

Capítulo 3

Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas e suas variantes

Neste capítulo, será apresentada uma formulação matemática unificada englobando o PRVCES e suas variantes consideradas. Precisamente, na Seção 3.1, todas as dez variantes serão descritas como um único problema genérico, denominado *Problema de Roteamento e Localização com Coleta e Entrega Simultâneas Heterogêneo e com Janelas de Tempo* (PRLCESHJT). Definido o problema, uma formulação matemática para o PRLCESHJT será apresentada na Seção 3.2.

3.1 Descrição do problema

Seja $V_C = \{1, \dots, n\}$, $V_D = \{n+1, \dots, n+m\}$ e $V'_D = \{(n+1)+m, \dots, (n+m)+m\}$ os conjuntos de clientes, depósitos e de suas cópias, contendo n clientes, m depósitos and m cópias de depósitos, respectivamente. É importante ressaltar que os depósitos $n+b \in V_D$ e $(n+b)+m \in V'_D$, $b \in \{1, \dots, m\}$ são idênticos. Em relação aos arcos, define-se $A_d = \{(i, j) : i \in V_C \cup \{n+d\}, j \in V_C \cup \{n+d+m\}, i \neq j\}$, $d \in \{1, \dots, m\}$, como o conjunto de arcos que interligam os clientes, bem como tais clientes ao depósito

$(n + d) + m \in V'_D$ ou ao seu correspondente. Desse modo, seja o grafo $G = (V, A)$, com $V = V_C \cup V_D \cup V'_D$ and $A = \cup_{d=1}^m A_d$ representando o conjunto de todos os clientes e arcos, respectivamente, presentes na rede. Para cada arco $a = (i, j) \in A$, c_a e t_a denotam o custo e o tempo de viagem associados, respectivamente, estando incluso no segundo o tempo de serviço referente ao cliente i . Além disso, seja K o conjunto de todos os tipos de veículo disponíveis. Os veículos do tipo $k \in K$ possuem custos variável α_k e fixo β_k , enquanto $\gamma_d, d \in \{1, \dots, m\}$, representa o custo fixo associado ao depósito $n + d \in V_D$. São definidos ainda $Q_k, k \in K$ e $Q'_d, d \in \{1, \dots, m\}$ como as capacidades associadas a cada tipo de veículo e ao d -ésimo depósito, respectivamente; U_k o número máximo de veículos de cada tipo e U'_d o mesmo limite, contudo para cada depósito. Cada cliente $j \in V_C$ possui demandas não-negativas de coleta (p_j) e de entrega (q_j), bem como janelas de tempo $[e_j, l_j]$. As demandas associadas aos nós que representam os depósitos (isto é, nós pertencentes ao conjunto $V_D \cup V'_D$) são nulas e os mesmos possuem janelas de tempo equivalentes a $[0, T_{max}]$, em que T_{max} corresponde à duração máximo da rota.

O PRLCESHJT consiste em determinar quais depósitos serão utilizados, quais clientes estarão associadas a cada um dos depósitos escolhidos e as rotas de menor custo, atendendo as demandas de coleta e entrega de todos os clientes e satisfazendo as seguintes restrições:

- (i) cada cliente deve ser visitado exatamente uma vez;
- (ii) o mesmo veículo pode ser usado em no máximo uma rota;
- (iii) cada rota deve ser finalizada no mesmo depósito a partir do qual ela foi iniciada;
- (iv) as janelas de tempo de cada cliente devem ser satisfeitas;
- (v) a duração máxima da rota não deve ser excedida;
- (vi) o número de rotas deve respeitar os limites de veículos de cada tipo e em cada depósito;
- (vii) A capacidade de cada depósito não deve ser ultrapassada.

Além das restrições acima, dada uma rota $R = (v_0^R, v_1^R, v_2^R, \dots, v_\kappa^R, \dots, v_{r-1}^R, v_r^R)$, em que v_0^R e v_r^R representam o mesmo depósito, considere $p_{v_i^R}$ e $q_{v_i^R}$ como sendo as demandas de coleta e entrega, respectivamente, associadas ao vértice v_i^R pertencente à rota R e Q_R a capacidade do veículo associada à mesma rota. Logo, as seguintes restrições também devem ser satisfeitas:

$$(viii) \quad \sum_{i=1}^r q_{v_i^R} \leq Q_R;$$

$$(ix) \quad \sum_{i=1}^r p_{v_i^R} \leq Q_R;$$

$$(x) \quad \sum_{i=1}^{(\kappa-1)} p_{v_i^R} + \sum_{i=\kappa}^r q_{v_i^R} \leq Q_R, \kappa \in \{2, \dots, r\}$$

3.2 Formulação matemática

Para que seja construída uma formulação matemática para o PRLCESHJT, seja x_a^{kd} , $k \in K, d \in \{1, \dots, m\}, a \in A_d$, uma variável inteira não-negativa responsável por indicar o número de vezes em que o arco a foi atravessado por um veículo do tipo k que iniciou sua rota no depósito $n + d$. Seja também $y_d, d \in \{1, \dots, m\}$ uma variável binária que assume o valor 1 se o depósito $n + d$ for utilizado, sendo nula caso contrário. Além dessas variáveis, são definidas também as variáveis de fluxo P_a e D_a , $a \in A$, que determinam as cargas de coleta e entrega, respectivamente, contidas no veículo ao atravessar o arco $a = (i, j)$, em que a primeira corresponde ao total coletado até o cliente i , incluindo-o, enquanto a segunda equivale ao total a ser entregue a todos os clientes que sucedem o i , partindo-se do cliente j . Ademais, seja $T_j, j \in V$ uma variável contínua que indica o início do atendimento do vértice j . Para cada $d \in \{1, \dots, m\}$ and $j \in V$, define-se também os conjuntos $\delta_d^+(j) = \{(j, i) \in A_d\}$ e $\delta_d^-(j) = \{(i, j) \in A_d\}$, e, analogamente, $\delta^+(j) = \cup_{d=1}^m \delta_d^+(j)$ e $\delta^-(j) = \cup_{d=1}^m \delta_d^-(j)$. Por fim, seja $V_D(a)$ o conjunto de índices de depósitos que podem ser associados ao arco $a = (i, j) \in A$, dependendo se i ou j pertencem ou não aos conjuntos V_D and V_D' , respectivamente. Uma descrição formal de $V_D(a)$ é apresentada em (3.1).

$$V_D(a) = \begin{cases} \{1, \dots, m\}, & \text{if } i \notin V_D, j \notin V'_D \\ \{i - n\}, & \text{if } i \in V_D \\ \{j - n - m\}, & \text{if } j \in V'_D \end{cases}, \quad a = (i, j) \in A \quad (3.1)$$

Uma formulação matemática compacta para o problema pode ser definida conforme descrito abaixo:

$$(F1) \quad \min \sum_{k \in K} \sum_{d=1}^m \sum_{a \in \delta^+(n+d)} \beta_k x_a^{kd} + \sum_{d=1}^m \gamma_d y_d + \sum_{k \in K} \sum_{d=1}^m \sum_{a \in A_d} \alpha_k c_a x_a^{kd} \quad (3.2)$$

s.a.

$$\sum_{k \in K} \sum_{d=1}^m \sum_{a \in \delta_d^-(j)} x_a^{kd} = 1 \quad j \in V_C \quad (3.3)$$

$$\sum_{a \in \delta_d^+(i)} x_a^{kd} = \sum_{a \in \delta_d^-(i)} x_a^{kd} \quad k \in K, d \in \{1, \dots, m\}, i \in V_C \quad (3.4)$$

$$\sum_{a \in \delta^-(j)} D_a - \sum_{a \in \delta^+(j)} D_a = q_j \quad j \in V_C \quad (3.5)$$

$$\sum_{a \in \delta^+(j)} P_a - \sum_{a \in \delta^-(j)} P_a = p_j \quad j \in V_C \quad (3.6)$$

$$D_a + P_a \leq \sum_{k \in K} \sum_{d \in V_D(a)} Q_k x_a^{kd} \quad a \in A \quad (3.7)$$

$$\sum_{d=1}^m \sum_{a \in \delta_d^+(n+d)} x_a^{kd} \leq U_k \quad k \in K \quad (3.8)$$

$$\sum_{k \in K} \sum_{a \in \delta_d^+(n+d)} x_a^{kd} \leq U'_d \quad d \in \{1, \dots, m\} \quad (3.9)$$

$$\sum_{k \in K} \sum_{a=(i,j) \in A_d} q_i x_a^{kd} \leq Q'_d y_d \quad d \in \{1, \dots, m\} \quad (3.10)$$

$$\sum_{k \in K} \sum_{a=(i,j) \in A_d} p_i x_a^{kd} \leq Q'_d y_d \quad d \in \{1, \dots, m\} \quad (3.11)$$

$$T_j \geq T_i + t_a - \left(1 - \sum_{k \in K} \sum_{d \in V_D(a)} x_a^{kd}\right) Z \quad a = (i, j) \in A \quad (3.12)$$

$$e_j \leq T_j \leq l_j \quad j \in V \quad (3.13)$$

$$D_a, P_a \geq 0 \quad a \in A \quad (3.14)$$

$$x_a^{kd} \in \mathbb{Z}_+ \quad k \in K, d \in \{1, \dots, m\}, a \in A_d \quad (3.15)$$

$$y_d \in \{0, 1\} \quad d \in \{1, \dots, m\}. \quad (3.16)$$

A função objetivo (3.2) minimiza o custo total, incluindo o custo de deslocamento, bem como os custos variável e fixos associados aos veículos e depósitos. As restrições (3.3) e (3.4) impõem que cada cliente deve ser visitado exatamente uma vez e por um único veículo. As restrições (3.5) e (3.6) garantem a conservação de fluxo associada às demandas de entrega e coleta, respectivamente. As restrições (3.7) implicam que as cargas de coleta e entrega juntas não excederão a capacidade do veículo em qualquer ponto da rota. As restrições (3.8) e (3.9) são responsáveis por determinar que os limites de veículos por cada tipo ou depósito sejam respeitados, respectivamente. As restrições (3.10) estabelecem que a soma das demandas de entrega dos clientes associados ao d -ésimo depósito não excederá sua capacidade. As restrições (3.11) possuem finalidade semelhante, contudo para as demandas de coleta. As restrições (3.12) determinam os tempos de início dos atendimentos, sendo Z uma constante suficientemente grande. As restrições (3.13) garantem que as janelas de tempo dos clientes e depósitos serão respeitadas. Por fim, as restrições (3.14)-(3.16) definem o domínio das variáveis. Na Tabela 3.1 é mostrado como cada uma das dez variantes consideradas pode ser formulada como um caso particular do modelo genérico dado por (F1). Precisamente, é especificado, para cada variante, como os dados de entrada podem ser adaptados e quais restrições devem ser relaxadas, a fim de representar cada caso.

Tabela 3.1: Diferenças entre as variantes consideradas.

Variante	Cientes	Tipos de veículo	Depósitos	Arcos	Restrições relaxadas
PRVCES	$p_j \geq 0, q_j \geq 0, e_j = 0, l_j = \infty$	$Q_k = Q, \alpha_k = 1, \beta_k = 0, U_k < \infty$	$Q'_d = \infty, \gamma_d = 0, U'_d = \infty$	$c_a = c_{a'}$	(3.10), (3.11), (3.12) (3.13), (3.16)
PRVCEM	$p_j > 0 \Rightarrow q_j = 0, p_j = 0 \Rightarrow q_j > 0$ $e_j = 0, l_j = \infty$				(3.10), (3.11), (3.16)
PRVCESTL	$p_j \geq 0, q_j \geq 0, e_j = 0, l_j = T_{max}$				
PRVCEMTL	$p_j > 0 \Rightarrow q_j = 0, p_j = 0 \Rightarrow q_j > 0$ $e_j = 0, l_j = T_{max}$				
PRVCESJT	$p_j \geq 0, q_j \geq 0, e_j \geq 0, l_j > 0$				
PRVCESJTF	$p_j > 0 \Rightarrow q_j = 0, p_j = 0 \Rightarrow q_j > 0$ $e_j \geq 0, l_j > 0$	$Q_k > 0, \alpha_k > 0, \beta_k > 0, U_k = \infty$	$Q'_d = \infty, \gamma_d = 0, U'_d < \infty$	$c_a = c_{a'}$	(3.10), (3.11), (3.12) (3.13), (3.16)
PRVCESFH	$p_j \geq 0, q_j \geq 0, e_j = 0, l_j = \infty$				(3.10), (3.11), (3.12) (3.13), (3.16)
PRVCEMMD	$p_j > 0 \Rightarrow q_j = 0, p_j = 0 \Rightarrow q_j > 0$ $e_j = 0, l_j = \infty$				(3.12), (3.13)
PRLCES	$p_j \geq 0, q_j \geq 0, e_j = 0, l_j = \infty$	$Q_k = Q, \alpha_k = 1, \beta_k = 0, U_k < \infty$	$Q'_d > 0, \gamma_d > 0, U'_d < \infty$	$c_a \neq c_{a'}$	(3.10), (3.11), (3.12) (3.13), (3.16)
PRVCESA			$Q'_d = \infty, \gamma_d = 0, U'_d = \infty$		

$j \in V_C, k \in K, d \in \{1, \dots, m\}, a = (i, j) \in A$, and $a' = (j, i) \in A$.

Capítulo 4

Modelo para o VRPSolver

Neste capítulo será feita uma breve explicação sobre o *VRPSolver* (v. Seção 4.1), focando-se no aspecto da modelagem, bem como será apresentado um modelo para o PRLCESHJT como uma aplicação desse *solver* (v. Seção 4.2).

4.1 Modelo Genérico

O *VRPSolver*, proposto por Pessoa et al. (2020); Sadykov e Vanderbeck (2021), consiste em um *solver* baseado em BCP, voltado a modelos genéricos, com ênfase no PRV e em suas variantes. Nele estão implementados elementos de destaque da literatura do PRV, tais como *ng-path* (Baldacci et al., 2011), corte *rank-1* com memória limitada (Bulhões et al., 2018) (Pecin et al., 2017b) (Petersen et al., 2008), enumeração de caminhos (Baldacci et al., 2008) e cortes de capacidade arredondada (Laporte e Nobert, 1983), abordados na definição de *packing sets* proposta pelos autores. Certamente, esses elementos o conferem destaque perante outras abordagens exatas para essa classe de problemas, justificando-se, assim, a escolha do mesmo para a pesquisa em questão. Nesse sentido, serão abordados, na Seção 4.1.1, a definição dos grafos e do subproblema associados ao *solver*; a formulação para o problema mestre, na Seção 4.1.2, e, por fim, a definição dos *packing sets*, na Seção 4.1.3.

4.1.1 Grafos e subproblema

No *VRPSolver*, os subproblemas de *pricing* são modelados como um problema de caminho mais curto com restrição de recursos. Desse modo, faz-se necessário definir a rede de grafos associados ao problema, com seus respectivos recursos. Nesse sentido, seja K o conjunto de subproblemas. Além disso, seja $G^k = (V^k, A^k)$, $k \in K$ um grafo direcionado que pode conter ciclos, $V = \cup_{k \in K} V^k$ e $A = \cup_{k \in K} A^k$. Cada grafo possui vértices especiais, denominados V_{source}^k e V_{sink}^k , responsáveis por determinar o início e o término de cada rota.

Quanto aos recursos, seja R^k , $k \in K$ o conjunto de todos eles para cada grafo G^k . Cada recurso $r \in R^k$ possui um consumo $q_{a^k, r} \in \mathbb{R}$ para todo arco $a^k = (v_i^k, v_j^k)$ in A^k que, por sua vez, deve respeitar um intervalo finito de consumo acumulado $[l_{v_i^k, 1}, u_{v_i^k, 1}]$. Um recurso pode ser classificado como *monótono* quando não há qualquer consumo negativo e *não-monótono* caso contrário. Além disso, o próprio conjunto R^k é dividido em dois subconjuntos R_M^k e R_N^k , associados aos recursos principais e secundários, respectivamente. Os principais obrigatoriamente são monótonos, enquanto os secundários podem ser monótonos ou não-monótonos. Por fim, os recursos podem ser ditos *disposable*, quando é permitido “desperdiçá-los” com o intuito de satisfazer o intervalo de consumo acumulado, ou *non-disposable*, quando não há tal permissão. Vale ressaltar que todo recurso principal sempre será *disposable*.

Em face de todas essas definições, as condições de viabilidade de um caminho $p = (V_{source}^k = v_0^k, a_1^k, v_1^k, \dots, a_{n-1}^k, v_{n-1}^k, a_n^k, v_n^k = V_{sink}^k)$ através do grafo G^k , com $n \geq 1$ arcos, $v_j^k \neq V_{source}^k$ e $v_j^k \neq V_{sink}^k$ são:

- O consumo acumulado $S_{j,r}^k$ ao visitar o vértice v_j^k , para todo recurso $r \in R^k$ classificado como *disposable*, sendo $S_{0,r}^k = 0$ e $S_{j,r}^k = \max\{l_{v_j^k, r}, S_{j-1,r}^k + q_{a_j^k, r}\}$ não deve exceder $u_{v_j^k, r}$.
- O consumo acumulado $S_{j,r}^k$ ao visitar o vértice v_j^k , para todo recurso $r \in R^k$ classificado como *non-disposable*, sendo $S_{0,r}^k = 0$ e $S_{j,r}^k = S_{j-1,r}^k + q_{a_j^k, r}$ deve pertencer ao intervalo $[l_{v_j^k, r}, u_{v_j^k, r}]$.

Note que p não necessariamente é um caminho elementar. Por fim, seja P^k o conjunto de todos os caminhos viáveis de G^k e $P = \cup_{k \in K} P^k$ a união de todos eles.

4.1.2 Formulação do problema mestre

Uma vez estabelecidas as definições inerentes ao subproblema, faz-se necessário o entendimento acerca da formulação do problema mestre. Para tanto, será explicado o conceito de mapeamento, proposto por Pessoa et al. (2020). Adiante, está descrita a formulação para o problema.

$$\sum_{j=1}^{n_1} c_j x_j + \sum_{s=1}^{n_2} f_s y_s \quad (4.1)$$

$$\text{s.a. } \sum_{j=1}^{n_1} \alpha_{ij} x_j + \sum_{s=1}^{n_2} \beta_{is} y_s \geq d_i, \quad i = 1, \dots, m, \quad (4.2)$$

$$x_j = \sum_{k \in K} \sum_{p \in P^k} \left(\sum_{a \in M(x_j)} h_a^p \right) \lambda_p, \quad j = 1, \dots, n_1, \quad (4.3)$$

$$L^k \leq \sum_{p \in P^k} \lambda_p \leq U^k, \quad k \in K, \quad (4.4)$$

$$\lambda_p \in \mathbb{Z}_+, \quad p \in P \quad (4.5)$$

$$x_j \in \mathbb{Z}, y_s \in \mathbb{Z}, \quad j = 1, \dots, \overline{n_1}, s = 1, \dots, \overline{n_2} \quad (4.6)$$

Na qual $x_j, 1 \leq j \leq n_1, y_s, 1 \leq s \leq n_2$ e $\lambda_p, p \in P$ são variáveis, enquanto as expressões (4.1) e (4.2) representam a função objetivo e restrições genéricas, respectivamente. Seja h_a^p o indicador de quantas vezes o arco a está presente no caminho p , para todo $a \in A$ e $p \in P$. Para toda variável $x_j, j = 1, \dots, n_1, M(x_j) \subseteq A$ define o mapeamento da mesma a um subconjunto de arcos não-vazio, os quais, por sua vez, podem ser associados a mais de uma variável. Com o mapeamento, é possível estabelecer uma relação entre as variáveis de caminho (λ_p) e as mapeadas (x_j), representada por (4.3). Uma solução viável para a formulação apresentada consiste em um conjunto de caminhos $p \in P$ limitados inferiormente e superiormente por L^k e $U^k, k \in K$, respectivamente, e decisões adicionais representadas pelas variáveis não-mapeadas y_s .

Substituindo todas as ocorrências das variáveis x_j por suas correspondências em termos de λ_p e relaxando as restrições de integralidade, tem-se a formulação linear para o problema mestre que de fato é resolvida por meio do método de geração de

colunas, a qual está descrita abaixo.

$$\sum_{k \in K} \sum_{p \in P^k} \left(\sum_{j=1}^{n_1} c_j \sum_{a \in M(x_j)} h_a^p \right) \lambda_p + \sum_{s=1}^{n_2} f_s y_s \quad (4.7)$$

$$\text{s.a.} \sum_{k \in K} \sum_{p \in P^k} \left(\sum_{j=1}^{n_1} \alpha_{ij} \sum_{a \in M(x_j)} h_a^p \right) \lambda_p + \sum_{s=1}^{n_2} \beta_{is} y_s \geq d_i, \quad i = 1, \dots, m, \quad (4.8)$$

$$L^k \leq \sum_{p \in P^k} \lambda_p \leq U^k, \quad k \in K, \quad (4.9)$$

$$\lambda_p \geq 0, \quad p \in P \quad (4.10)$$

Dessa formulação, usa-se as variáveis duais $\pi_i, 1 \leq i \leq m$, associadas às restrições (4.8), ν_+^k e $\nu_-^k, k \in K$, associadas às restrições (4.9), no cálculo do custo reduzido do arco $a \in A$ ($\bar{c}_a$) e, conseqüentemente, do caminho $p \in P$ ($\bar{c}_p$), expressos por (4.11) e (4.12), respectivamente.

$$\bar{c}_a = \sum_{j \in M^{-1}(a)} c_j - \sum_{i=1}^m \sum_{j \in M^{-1}(a)} \alpha_{ij} \pi_i \quad (4.11)$$

$$\bar{c}(p) = \sum_{j=1}^n \bar{c}_{a_j} - \nu_+^k - \nu_-^k \quad (4.12)$$

O *pricing*, portanto, consiste em encontrar o caminho $p \in P^k, k \in K$ com menor custo reduzido, sendo $M^{-1}(a)$ a função responsável por indicar quais são as variáveis x_j que estão mapeadas para o arco a , ou seja, a função inversa de $M(x_j)$.

4.1.3 Packing Sets

Os *Packing Sets* são um artifício de modelagem proposto por Pessoa et al. (2020), a fim de melhorar o algoritmo de BCP utilizado pelo *VRPSolver*. Eles variam de acordo com a aplicação e precisam ser especificados como parte da modelagem feita pelos usuários do *solver*. Nesse sentido, seja h_v^p o indicador de quantas vezes o vértice v está presente no caminho p e $\mathcal{B}^\mathcal{V} \subset 2^\mathcal{V}$ uma coleção de subconjuntos mutualmente disjuntos de V , tais que pelo menos uma solução ótima (x^*, y^*, λ^*) da formulação matemática

para o problema mestre (v. Seção 4.1.2) satisfaz o conjunto de restrições (4.13).

$$\sum_{p \in P} \left(\sum_{v \in B} h_v^p \right) \lambda_p \leq 1, \quad B \in \mathcal{B}^\nu \quad (4.13)$$

Desse modo, afirma-se que cada elemento de $\mathcal{B}^\nu$ representa um *packing set*, cuja definição implica que seus vértices só podem aparecer no máximo uma vez em uma solução, considerando todos os caminhos que a constituem.

4.2 Aplicação

Nesta seção será definido o modelo do PRLCESHJT implementado para o *VRP-Solver*. Precisamente, na Seção 4.2.1 serão indicados os grafos que compõem a rede. Por sua vez, nas Seções 4.2.2 e 4.2.3 são abordados os recursos e variáveis necessários à formulação, respectivamente. Por fim, na Seção 4.2.4, é mostrado modelo propriamente dito. É importante ressaltar que, para o modelo usando o *solver* em questão, são considerados os mesmos dados de entrada e conjuntos definidos na Seção 3.1.

4.2.1 Grafos

Seja o seguinte grafo: $G_d^k = (V_d^k, A_d^k)$, $\forall k \in K, d \in \{1, \dots, m\}$

- $V_d^k = V_C \cup \{n + d, (n + m) + d\}$: conjunto de todos os vértices pertencentes ao grafo G_d^k ;
- $A_d^k = A_d$: conjunto de todos os arcos pertencentes ao grafo G_d^k ;
- $V_{source}^{kd} = n + d$;
- $V_{sink}^{kd} = (n + m) + d$.

Note que é definido um grafo para cada par de depósito e tipo de veículo. Cada grafo contém cópias de todos os vértices clientes (pertencentes ao conjunto V_C) e os vértices referentes ao d -ésimo depósito, $n + d$ e $(n + m) + d$, sendo estes últimos os vértices especiais *source* e *sink* dos seus respectivos grafos. É importante ressaltar que

cada grafo é tratado de forma independente no *VRPSolver* e, portanto, todas as cópias de cada um dos vértices pertencentes ao conjunto V_C são elementos diferentes na rede, apesar de representarem o mesmo vértice na prática. Para simplificar a notação, os vértices de um grafo G_d^k serão referenciados como $i \in V_d^k$ ou $j \in V_d^k$, contudo, quando se fizer necessária a distinção explícita entre as cópias de um determinado cliente, estas serão nomeadas como $v_i^{kd} \in V_d^k$ ou $v_j^{kd} \in V_d^k$.

4.2.2 Recursos

Quanto aos recursos, sejam $R = \{1, 2, 3\}$, o quais estão associados às demandas de coleta e entrega e ao tempo, respectivamente. Em relação ao consumo de tais recursos, para o arco $a = (i, j) \in A_d^k$, $k \in K$, $d \in \{1, \dots, m\}$, este é estabelecido por (4.14), (4.15) e (4.16).

$$\bar{q}_{a,1} = p_j \quad (4.14)$$

$$\bar{q}_{a,2} = q_j \quad (4.15)$$

$$\bar{q}_{a,3} = t_a \quad (4.16)$$

Sendo $\bar{q}_{a,1}$, $\bar{q}_{a,2}$ e $\bar{q}_{a,3}$ as representações do consumo dos recursos 1, 2 e 3 no arco a , respectivamente, p_j e q_j as demandas de coleta e entrega, respectivamente, referentes ao vértice j e t_a o tempo de deslocamento associado ao arco a . É importante ressaltar que há um caso particular em relação ao recurso referente à entrega, o qual será tratado em mais detalhes ao final deste capítulo. Quanto aos intervalos de consumo acumulado para o três recursos, eles são definidos por (4.17), (4.18) e (4.19), para todo $i \in V_d^k$, $k \in K$, $d \in \{1, \dots, m\}$,

$$[\bar{l}_{i,1}^k, \bar{u}_{i,1}^k] = [0, Q_k]; \quad (4.17)$$

$$[\bar{l}_{i,2}^k, \bar{u}_{i,2}^k] = [0, Q_k]; \quad (4.18)$$

$$[\bar{l}_{i,3}^k, \bar{u}_{i,3}^k] = \begin{cases} [0, T_{max}], & \text{se } i = n + d \text{ ou } i = (n + m) + d \\ [e_i, l_i], & \text{c.c.} \end{cases} \quad (4.19)$$

nas quais $[\bar{l}_{i,1}^k, \bar{u}_{i,1}^k]$, $[\bar{l}_{i,2}^k, \bar{u}_{i,2}^k]$ e $[\bar{l}_{i,3}^k, \bar{u}_{i,3}^k]$ representam o intervalo de consumo acumulado dos recursos 1, 2 e 3, respectivamente, Q_k a capacidade dos veículos do tipo k , $[e_i, l_i]$ a janela de tempo do vértice i e T_{max} a duração máxima de cada uma das rotas.

Importante salientar que, apesar do modelo definido neste capítulo estar associado à formulação do PRLCESHJT, na prática os experimentos computacionais foram realizados com conjuntos de instâncias específicos de cada um dos problemas considerados, conforme será visto no Capítulo 5, devido à ausência de instâncias na literatura referentes ao problema unificado. Desse modo, os dois primeiros recursos são comuns a todos os problemas considerados e o terceiro é adicionado apenas às variantes que levam em consideração o tempo como característica do problema. Essa adição provoca uma alteração quanto à classificação desses recursos para essas variantes, devido à limitação do *solver* de criação de recursos principais (no máximo dois). Tal classificação está apresentada em (4.20), na qual R_M e R_N representam os recursos principais e secundários, respectivamente. Vale ressaltar que todos eles são *disposable*.

$$R = \begin{cases} R_M = \{1, 2\} \text{ and } R_N = \{\}, & \text{PRVCES e PRVCESA;} \\ R_M = \{3\} \text{ and } R_N = \{1, 2\}, & \text{variantes com o recurso de tempo;} \\ R_M = \{2\} \text{ and } R_N = \{1\}, & \text{demais variantes;} \end{cases} \quad (4.20)$$

4.2.3 Variáveis

Em relação às variáveis do modelo adaptado ao *VRPSolver*, define-se x_a^{kd} como inteiras, para todo arco $a \in A_d^k$, $k \in K$, $d \in \{1, \dots, m\}$, assim como definido na formulação proposta para o PRLCESHJT (v. Capítulo 3). Além disso, são necessárias também as variáveis binárias y_d , $d \in \{1, \dots, m\}$, responsáveis por indicar se o d -ésimo depósito é utilizado.

4.2.4 Modelo

De posse de todas as definições relacionadas aos grafos (v. Seção 4.2.1), recursos (v. Seção 4.2.2) e variáveis (v. Seção 4.2.3), tem-se a seguinte formulação matemática para o PRLCESHJT utilizando os conceitos inerentes à modelagem para o *VRPSolver*.

$$(F2) \quad \min \sum_{k \in K} \sum_{d=1}^m \sum_{a \in \delta^+(n+d)} \beta_k x_a^{kd} + \sum_{d=1}^m \gamma_d y_d + \sum_{k \in K} \sum_{d=1}^m \sum_{a \in A_d} \alpha_k c_a x_a^{kd} \quad (4.21)$$

s.a.

$$\sum_{k \in K} \sum_{d=1}^m \sum_{a \in \delta_d^-(j)} x_a^{kd} = 1 \quad j \in V_C \quad (4.22)$$

$$\sum_{k \in K} \sum_{a=(i,j) \in A_d} q_i x_a^{kd} \leq Q'_d y_d \quad d \in \{1, \dots, m\} \quad (4.23)$$

$$\sum_{k \in K} \sum_{a=(i,j) \in A_d} p_i x_a^{kd} \leq Q'_d y_d \quad d \in \{1, \dots, m\} \quad (4.24)$$

Apresentado o modelo, adiante serão definidos os limites $[L_d^k, U_d^k]$, os mapeamentos $M(x_a^{kd})$, o conjunto de *packing sets* $\mathcal{B}^V$ e os separadores de corte de capacidade (*Rounded Capacity Cut Separator, RCC*) associados à modelagem do problema.

$$[L_d^k, U_d^k] = [0, \min\{U_k, U'_d\}], \quad k \in K, d \in \{1, \dots, m\};$$

$$M(x_a^{kd}) = \{a\}, \quad a \in A_d^k, k \in K, d \in \{1, \dots, m\};$$

$$\mathcal{B}^V = \cup_{i \in V_C} \{\{v_i^{kd} : k \in K, d \in \{1, \dots, m\}\}\};$$

$$RCC: \begin{cases} (\cup_{i \in V_C} \{\{v_i^{kd} : k \in K, d \in \{1, \dots, m\}\}, p_i\}), \max_{k \in K} Q^k \\ (\cup_{i \in V_C} \{\{v_i^{kd} : k \in K, d \in \{1, \dots, m\}\}, d_i\}), \max_{k \in K} Q^k \end{cases}$$

Com esse modelo, objetiva-se a minimização do custo total, no qual estão inclusos a distância percorrida e os custos fixos e variáveis associados aos veículos e aos depósitos, como descrito em (4.21). Note que as restrições (4.22), (4.23) e (4.24), presentes na formulação F2, são idênticas às restrições (3.3), (3.10) e (3.11) presentes na formulação F1, respectivamente. Quanto aos limites $[L_d^k, U_d^k]$, estes estão associados ao número máximo de veículos de cada tipo ou depósito. Em relação aos mapeamentos, estabelece-

se que cada variável x_a^{kd} está associada a um arco $a \in A_d^k, k \in K, d = \{1, \dots, m\}$. Para os *packing sets*, tem-se que cada um desses conjuntos contém os vértices que representam um único cliente, ou seja, como os clientes estão replicados em cada um dos grafos, todas essas cópias são consideradas em um único conjunto, conforme indicado pela definição de $\mathcal{B}^\nu$. Já para o *RCC*, considera-se a capacidade máxima dentre todos os tipos de veículos, além de serem definidos separadores para cada tipo de demanda.

Todas as demais restrições do modelo de programação inteira para PRLCESHJT (v. Capítulo 3) estão presentes no subproblema e são implicitamente garantidas pelas definições dos grafos (v. Seção 4.2.1), dos recursos (v. Seção 4.2.2), dos limites L_d^k e U_d^k e dos mapeamentos $M(x_a^{kd}), \forall a \in A_d^k, k \in K, d \in \{1, \dots, m\}$. Uma única exceção deve ser feita para as restrições que estabelecem que as cargas de coleta e entrega, simultaneamente, não devem exceder a capacidade do veículo que as transporta. Note que, da forma como o modelo foi definido, garante-se apenas que o total de demandas coletadas e entregues não excede a capacidade do veículo, considerando-as isoladamente. No entanto, considerando-as simultaneamente, o modelo não é capaz de satisfazer essa condição, conforme o exemplo descrito adiante.

Considere $n = 3, m = 1, V = \{V_C = \{1, 2, 3\} \cup V_D = \{4\} \cup V'_D = \{5\}\}, |K| = 1$ e $Q_1 = 10$. Além disso, considere $\{q_1, q_2, q_3\} = \{5, 2, 3\}, \{p_1, p_2, p_3\} = \{4, 4, 1\}$ e D o somatório das demandas de entrega de todos os clientes de uma dada rota. Sejam Sq_j e Sp_j o consumo acumulado dos recursos referentes às demandas de entrega e coleta, respectivamente, até o cliente j , de acordo com (4.25) e (4.26).

$$Sq_j = \sum_{i=1}^j q_i \quad (4.25)$$

$$Sp_j = \sum_{i=1}^j p_i \quad (4.26)$$

Por questões de simplificação do exemplo, assume-se que todos os clientes, isto é $\{1, 2, 3\}$, fazem parte da mesma rota e são visitados na seguinte sequência: $4 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 5$. Logo, tem-se que:

$$D = q_1 + q_2 + q_3 = 5 + 2 + 3 = 10$$

$$Sq_1 = q_1 = 5 \leq Q_1$$

$$Sq_2 = q_1 + q_2 = 5 + 2 = 7 \leq Q_1$$

$$Sq_3 = q_1 + q_2 + q_3 = 5 + 2 + 3 = 10 \leq Q_1$$

$$Sp_1 = p_1 = 4 \leq Q_1$$

$$Sp_2 = p_1 + p_2 = 4 + 4 = 8 \leq Q_1$$

$$Sp_3 = p_1 + p_2 + p_3 = 4 + 4 + 1 = 9 \leq Q_1$$

Note que, ao longo de toda a rota, o consumo de nenhum dos recursos excede a capacidade do veículo, desde que considerados isoladamente. No entanto, considerando-os concomitantes ao longo de todo o trajeto, tem-se a seguinte situação:

$$(D - Sq_1) + Sp_1 = (10 - 5) + 4 = 5 + 4 = 9 \leq Q_1$$

$$(D - Sq_2) + Sp_2 = (10 - 7) + 8 = 3 + 8 = 11 > Q_1$$

$$(D - Sq_3) + Sp_3 = (10 - 10) + 9 = 0 + 9 = 9 \leq Q_1$$

Como todos os clientes estão sendo considerados pertencentes à mesma rota, há uma violação após a visita ao cliente 2, isto é, no arco (2, 3). Desse modo, a rota considerada torna-se inviável e, portanto, faz-se necessário a utilização de algum artifício para detectar essa inviabilidade e impedir a existência dessa rota. Para tanto, são utilizadas *lazy constraints*, a partir das quais é verificado se o somatório das cargas de coleta e entrega do veículo em todos os arcos que compõem as rotas de uma dada solução excede sua capacidade. Em caso afirmativo, a restrição (4.27) é adicionada ao modelo.

$$\sum_{a=(i,j) \in R} x_a^{k_R d_R} \leq |R| - 1, \quad \forall R \in \mathcal{R} \quad (4.27)$$

Sendo $\mathcal{R}$ a representação do conjunto de todas as rotas percorridas por um veículo do tipo k_R e iniciadas a partir do depósito cujo índice é d_R que são viáveis em relação às demandas de coleta e entrega isoladamente, mas não simultaneamente. A finalidade da restrição (4.27) é eliminar qualquer solução inteira que contenha rotas inviáveis em relação ao aspecto citado. Importante ressaltar que o uso das *lazy constraints* é viabilizado

pela implementação de uma função *callback* responsável por, a cada solução inteira encontrada, verificar a viabilidade das rotas, adicionando a restrição dinamicamente à formulação, se necessário.

Por fim, é importante ressaltar um caso particular na definição do consumo do recurso referente à entrega, o qual está associado à simultaneidade entre os dois tipos de demanda. Considere agora $\{q_1, q_2, q_3\} = \{5, 2, 3\}$ e $\{p_1, p_2, p_3\} = \{6, 2, 2\}$, mantendo-se os demais dados do exemplo anterior. Novamente, não há violação de capacidade no consumo dos recursos separadamente, como descrito adiante:

$$D = q_1 + q_2 + q_3 = 5 + 2 + 3 = 10$$

$$Sq_1 = q_1 = 5 \leq Q_1$$

$$Sq_2 = q_1 + q_2 = 5 + 2 = 7 \leq Q_1$$

$$Sq_3 = q_1 + q_2 + q_3 = 5 + 2 + 3 = 10 \leq Q_1$$

$$Sp_1 = p_1 = 6 \leq Q_1$$

$$Sp_2 = p_1 + p_2 = 6 + 2 = 8 \leq Q_1$$

$$Sp_3 = p_1 + p_2 + p_3 = 6 + 2 + 2 = 10 \leq Q_1$$

No entanto, simultaneamente, os três clientes não poderiam pertencer à mesma rota, pois ocorreria uma violação após o primeiro cliente ser visitado, como indicado pelas seguintes expressões:

$$(D - Sq_1) + Sp_1 = (10 - 5) + 6 = 5 + 6 = 11 > Q_1$$

$$(D - Sq_2) + Sp_2 = (10 - 7) + 8 = 3 + 8 = 11 > Q_1$$

$$(D - Sq_3) + Sp_3 = (10 - 10) + 10 = 0 + 10 = 10 \leq Q_1$$

Esse seria mais um caso de uso das *lazy constraints* para proibir a existência dessa rota, contudo, para que seja evitada a adição de mais uma restrição ao modelo, opta-se por fazer com que os veículos saiam do depósito com um “excesso artificial” nos casos em que o primeiro cliente a ser visitado possui demanda de coleta maior do que a de

entrega. Precisamente, atribui-se o valor da demanda de coleta ao consumo da entrega para todo $a = (n + d, j) \in A_d^k, k \in K, d \in \{1, \dots, m\}$ tal que $p_j > q_j$. Aplicando-se esse artifício no exemplo anterior, tem-se a seguinte situação:

$$Sq_1 = q_1 = 6 \leq Q_1$$

$$Sq_2 = q_1 + q_2 = 6 + 2 = 8 \leq Q_1$$

$$Sq_3 = q_1 + q_2 + q_3 = 6 + 2 + 3 = 11 > Q_1$$

$$Sp_1 = p_1 = 6 \leq Q_1$$

$$Sp_2 = p_1 + p_2 = 6 + 2 = 8 \leq Q_1$$

$$Sp_3 = p_1 + p_2 + p_3 = 6 + 2 + 2 = 10 \leq Q_1$$

É possível perceber que, após essa mudança, a violação de capacidade ocorre considerando apenas o recurso de entrega e, assim, o cliente 3 não poderia ser atendido após os clientes 1 e 2 terem sido visitados nessa ordem. Portanto, a rota sendo analisada já seria proibida sem a necessidade de utilização dos *lazy cuts*, diminuindo, assim, a quantidade de cortes a serem utilizados.

Capítulo 5

Resultados computacionais

Neste capítulo serão descritos os experimentos computacionais realizados, como também apresentados os resultados obtidos para o PRVCEs e suas variantes consideradas. Todos eles foram executados em uma máquina *Super Server* com processador Intel® Xeon™ E5-2650 v4 2,20 GHz, 128 GB de memória RAM e sistema operacional Ubuntu 16.04.6 LTS de 64 bits. Quanto aos modelos, estes foram programados por meio da linguagem C++ (compilador g++, versão 9.4.0), sendo utilizada a versão 0.69 do *BaPCod*, com extensão para o *VRPSolver*.

Em relação aos parâmetros do *solver*, foram utilizados três arquivos de configuração (“GM_1”, “GM_10” e “GM_17”) para todas as variantes consideradas, os quais foram obtidos após a realização de experimentos de calibração com instâncias representativas de cada um dos problemas. Desse modo, na Tabela 5.1 estão apresentadas algumas informações básicas acerca de cada um dos *datasets* utilizados nos experimentos, bem como estão indicadas as associações entre os arquivos de configuração e os problemas estudados. Vale ressaltar que os conteúdos explícitos de cada um desses arquivos estão presentes no Apêndice B.

De posse das informações gerais, nas seções adiante serão explicadas as especificidades referentes aos testes realizados, bem como serão apresentados os resultados obtidos a partir deles, envolvendo cada um dos problemas considerados, ou seja, PRVCEs (Seção 5.1), PRVCEM (Seção 5.2), PRVCESTL (Seção 5.3), PRVCEMTL (Seção

Tabela 5.1: Descrição dos conjuntos de instâncias considerados.

Problema	Conjunto de Instâncias	Proposto por	Derivado de	#Conjunto de Instâncias	Arquivo de Configuração
PRVCES	SN	Salhi e Nagy (1999)	—	14	GM_10
	MG	Montané e Galvão (2006)	—	18	GM_17
PRVCEM	SN	Salhi e Nagy (1999)	—	21	GM_10
PRVCESTL	SN	Salhi e Nagy (1999)	—	14	GM_10
PRVCEMTL	SN	Salhi e Nagy (1999)	—	21	GM_10
PRVCESJT	W12_C	Wang e Chen (2012)	Solomon (1987)	17	GM_10
	W12_R			23	
	W12_RC			28	
PRVCESJTF	W13_C	Wang e Chen (2013)	Wang e Chen (2012)	17	GM_10
	W13_R			23	
	W13_RC			28	
PRVCESFH	Avci1	Avci e Topaloglu (2016)	—	14	GM_10
PRVCEMMD	SN	Salhi e Nagy (1999)	—	33	GM_10
PRVCESA	RZC6	Rieck e Zimmermann (2013)	Dethloff (2001)	40	GM_10
	RZC7	Rieck e Zimmermann (2013)	Salhi e Nagy (1999)	2	GM_10
PRLCES	Bar11	Karaoglan et al. (2011)	Barreto et al. (2007)	60	GM_1
	Prod11	Karaoglan et al. (2011)	Prins et al. (2004)	88	GM_10
	Prod12	Karaoglan et al. (2012)	Prins et al. (2004)	96	GM_10

5.4), PRVCESJT (Seção 5.5), PRVCESJTF (Seção 5.6), PRVCESA (Seção 5.7), PRVCESFH (Seção 5.8), PRVCEMMD (Seção 5.9) e PRLCES (Seção 5.10). Importante ressaltar que, da Tabela 5.2 à Tabela 5.22, C indica o número de clientes; TVD (VD) e TVU (VU) a quantidade de tipos de veículo (veículos) disponíveis e utilizados, respectivamente; DD e DU a quantidade de depósitos disponíveis e utilizados, respectivamente; RLB o *lower bound* obtido da raiz; RT o tempo necessário para a resolução da raiz; NN o número de nós da árvore de B&B; TT o tempo total do teste e LC o número de *lazy cuts* aplicados. Além disso, vale destacar também que os *bounds* para algumas instâncias específicas foram obtidos com a relaxação de um dos recursos, referente à entrega ou à coleta. Tais casos estão devidamente indicados em suas correspondentes tabelas.

5.1 PRVCES

Para os testes envolvendo o PRVCES, foram utilizados os conjuntos de instâncias propostos por Montané e Galvão (2006) e Salhi e Nagy (1999). Os resultados obtidos foram comparados com os reportados por Subramanian (2012), utilizando os algoritmos B&C com *lazy separations* (v. Tabelas 5.2 e 5.4) e BCP (v. Tabelas 5.3 e 5.5), para ambos os conjuntos.

Em relação ao conjunto proposto por Salhi e Nagy (1999), foram resolvidas as instâncias cmt2x e cmt2y, as quais estavam em aberto na literatura. Outro importante resultado está relacionado à melhoria dos limites para instâncias com 100, 150 e 199 clientes, quando comparado ao algoritmo B&C, conforme indicado na Tabela 5.2. Por outro lado, quando comparado ao algoritmo BCP, a abordagem proposta conseguiu superá-lo na resolução do nó raiz para todas as instâncias, exceto cmt3x, cmt3y, cmt11x, cmt11y e cmt4x (v. Tabela 5.3).

Quanto à análise das Tabelas 5.4 e 5.5, pode-se notar que todas instâncias contendo 100 clientes já foram resolvidas previamente, sendo a abordagem proposta também capaz de resolvê-las. No entanto, contribuições mais significativas foram obtidas dos testes com as instâncias que possuem 200 clientes. Mais precisamente, foram melhorados os LBs para as instâncias C_1_2_1, C_2_2_1, R_1_2_1 e RC_1_2_1, sendo algumas dessas melhorias bem expressivas, como indicado na Tabela 5.4. Além disso, percebe-se melhorias também em relação à resolução do nó raiz (*root*), quando comparado ao BCP, para todas as instâncias, exceto r201, rc201, C_2_2_1, R_2_2_1 e RC_2_2_1, conforme os dados da Tabela 5.5. Por fim, outro importante resultado alcançado foi propor, pela primeira vez, *bounds* para as instâncias com 400 clientes. Da Tabela 5.5, percebe-se que os *gaps* são inferiores a 3,50 % para todas essas instâncias, os quais são bastante interessantes, tendo em vista a quantidade de clientes a serem atendidos.

Tabela 5.2: Resultados computacionais das instâncias do PRVCES — SN — Comparação com o B&C com *lazy separations*.

Instância	C	B&C com <i>lazy constraints</i>			Abordagem com o <i>VRPSolver</i>							
		LB	UB ¹	GAP(%)	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
cmt1x	50	466,77	466,77	0,00	466,77	112,59	3	118,83	1	466,77	466,77	0,00
cmt1y	50	466,77	466,77	0,00	466,77	72,16	3	80,22	1	466,77	466,77	0,00
cmt2x	75	666,57	684,21	2,58	673,54	390,47	3695	470667,00	3078	684,21	684,21	0,00
cmt2y	75	666,69	684,21	2,56	675,64	246,23	159	18417,80	34	684,21	684,21	0,00
cmt3x	100	721,27	721,27	0,00	699,79	3479,50	3571	604800,00	0	700,27	721,27	2,91
cmt3y	100	721,27	721,27	0,00	697,56	648,28	3503	604800,00	0	701,15	721,27	2,79
cmt12x ³	100	643,76	662,22	2,79	643,25	—	—	—	—	644,47	662,22	2,68
cmt12y	100	644,10	662,22	2,74	643,25	37756,30	41	86400,00	0	644,47	662,22	2,68
cmt11x ²³	120	799,67	833,92	4,11	651,43	—	—	—	—	651,43	833,92	21,88
cmt11y ²	120	799,02	833,92	4,19	651,43	5374,95	1	5374,95	0	651,43	833,92	21,88
cmt4x	150	831,18	852,46	2,50	827,63	9381,37	203	86400,00	0	827,81	852,46	2,89
cmt4y	150	831,65	852,46	2,44	828,84	2194,16	255	86400,00	0	832,10	852,46	2,39
cmt5x	199	971,07	1029,25	5,65	1000,11	57202,90	5	86400,00	0	1000,50	1029,25	2,79
cmt5y	199	937,66	1029,25	8,90	1001,27	11475,00	41	86400,00	0	1002,64	1029,25	2,59

¹ Obtidos utilizando o método heurístico em (Subramanian, 2012). ² *Bounds* obtidos a partir da relaxação do recurso referente à entrega.

³ *Bounds* obtidos a partir dos testes com sua instância correspondente do tipo 'y'.

Tabela 5.3: Resultados computacionais das instâncias do PRVCES — SN — Comparação com o BCP.

Instância	C	BCP			Abordagem com o <i>VRPSolver</i>							
		RLB	RT(s)	LB	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
cmt1x	50	461,90	96,00	—	466,77	112,59	3	118,83	1	466,77	466,77	0,00
cmt1y	50	460,86	238,00	—	466,77	72,16	3	80,22	1	466,77	466,77	0,00
cmt2x	75	661,11	94,20	—	673,54	390,47	3695	470667,00	3078	684,21	684,21	0,00
cmt2y	75	659,76	308,00	—	675,64	246,23	159	18417,80	34	684,21	684,21	0,00
cmt3x	100	711,70	1404,00	—	699,79	3479,50	3571	604800,00	0	700,27	721,27	2,91
cmt3y	100	711,49	4519,00	—	697,56	648,28	3503	604800,00	0	701,15	721,27	2,79
cmt12x ²	100	638,72	3130,00	—	643,25	—	—	—	—	644,47	662,22	2,68
cmt12y	100	636,29	3608,00	—	643,25	37756,30	41	86400,00	0	644,47	662,22	2,68
cmt11x ¹²	120	766,22	9570,00	—	651,43	—	—	—	—	651,43	833,92	21,88
cmt11y ¹	120	765,47	9421,00	—	651,43	5374,95	1	5374,95	0	651,43	833,92	21,88
cmt4x	150	829,58	11433,00	—	827,63	9381,37	203	86400,00	0	827,81	852,46	2,89
cmt4y	150	827,50	791912,00	—	828,84	2194,16	255	86400,00	0	832,10	852,46	2,39
cmt5x	199	988,28	66063,00	—	1000,11	57202,90	5	86400,00	0	1000,50	1029,25	2,79
cmt5y	199	980,63	260737,00	—	1001,27	11475,00	41	86400,00	0	1002,64	1029,25	2,59

¹ Bounds obtidos a partir da relaxação do recurso referente à entrega. ² Bounds obtidos a partir dos testes com sua instância correspondente do tipo 'y'

Tabela 5.4: Resultados computacionais das instâncias do PRVCES — MG — Comparação com o B&C com *lazy separations*.

Instância	C	B&C com <i>lazy constraints</i>			Abordagem com o <i>VRPSolver</i>							
		LB	UB ¹	GAP(%)	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
c101	100	1202,59	1220,18	1,46	1219,54	2041,23	47	17365,50	17	1220,18	1220,18	0,00
c201	100	662,07	662,07	0,00	662,07	2621,28	1	2621,29	0	662,07	662,07	0,00
r101	100	978,28	1009,95	3,24	1008,57	2215,11	25	24511,80	1	1009,95	1009,95	0,00
r201	100	666,20	666,20	0,00	654,12	4235,12	7	22976,20	0	666,20	666,20	0,00
rc101	100	1041,06	1059,32	1,75	1056,71	788,33	143	14672,10	95	1059,32	1059,32	0,00
rc201	100	672,92	672,92	0,00	657,13	4142,82	29	38943,10	0	672,92	672,92	0,00
C1.2.1	200	2857,45	3629,89	27,03	3625,08	9057,19	853	604800,00	626	3626,07	3628,50	0,07
C2.2.1	200	1638,99	1726,59	5,34	1677,90	3122,28	43	86400,00	0	1689,99	1726,59	2,12
R1.2.1	200	2681,05	3360,02	25,32	3310,29	3140,52	49	86400,00	0	3316,69	3353,80	1,11
R2.2.1	200	1665,58	1665,58	0,00	1622,88	31799,30	5	44373,50	0	1623,23	1665,58	2,54
RC1.2.1	200	2656,78	3306,00	24,44	3265,11	3061,30	87	86400,00	0	3268,97	3303,70	1,05
RC2.2.1	200	1560,00	1560,00	0,00	1498,61	24432,50	1	24432,70	0	1498,61	1560,00	3,94
C1.4.1	400	—	—	—	11030,44	32539,00	1	32539,00	0	11030,44	11047,19	0,15
C2.4.1	400	—	—	—	3438,54	20491,20	1	20491,20	0	3438,54	3539,50	2,85
R1.4.1	400	—	—	—	9411,19	9407,60	1	9407,60	0	9411,19	9519,45	1,14
R2.4.1	400	—	—	—	3456,09	299928,00	1	299928,00	0	3456,09	3546,49	2,55
RC1.4.1	400	—	—	—	9331,23	12850,90	1	12850,90	0	9331,23	9447,53	1,23
RC2.4.1 ²	400	—	—	—	3293,37	26253,00	1	26253,00	0	3293,37	3403,70	3,24

¹ Obtidos utilizando o método heurístico em (Subramanian, 2012). ² Bounds obtidos a partir da relaxação do recurso referente à entrega.

Tabela 5.5: Resultados computacionais das instâncias do PRVCES — MG — Comparação com o BCP.

Instância	C	BCP			Abordagem com o <i>VRPSolver</i>							
		RLB	RT(s)	LB	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
c101	100	1208,99	3,99	1220,18	1219,54	2041,23	47	17365,50	17	1220,18	1220,18	0,00
c201	100	659,11	17086,00	—	662,07	2621,28	1	2621,29	0	662,07	662,07	0,00
r101	100	996,18	208,00	1009,95	1008,57	2215,11	25	24511,80	1	1009,95	1009,95	0,00
r201	100	661,95	9968,00	—	654,12	4235,12	7	22976,20	0	666,20	666,20	0,00
rc101	100	1049,84	78,50	1059,32	1056,71	788,33	143	14672,10	95	1059,32	1059,32	0,00
rc201	100	668,84	15021,00	—	657,13	4142,82	29	38943,10	0	672,92	672,92	0,00
C1_2_1	200	3609,16	78,40	—	3625,08	9057,19	853	604800,00	626	3626,07	3628,50	0,07
C2_2_1	200	1704,00	97487,00	—	1677,90	3122,28	43	86400,00	0	1689,99	1726,59	2,12
R1_2_1	200	3273,27	1094,00	—	3310,29	3140,52	49	86400,00	0	3316,69	3353,80	1,11
R2_2_1	200	—	—	—	1622,88	31799,30	5	44373,50	0	1623,23	1665,58	2,54
RC1_2_1	200	3241,98	888,00	—	3265,11	3061,30	87	86400,00	0	3268,97	3303,70	1,05
RC2_2_1	200	—	—	—	1498,61	24432,50	1	24432,70	0	1498,61	1560,00	3,94
C1_4_1	400	—	—	—	11030,44	32539,00	1	32539,00	0	11030,44	11047,19	0,15
C2_4_1	400	—	—	—	3438,54	20491,20	1	20491,20	0	3438,54	3539,50	2,85
R1_4_1	400	—	—	—	9411,19	9407,60	1	9407,60	0	9411,19	9519,45	1,14
R2_4_1	400	—	—	—	3456,09	299928,00	1	299928,00	0	3456,09	3546,49	2,55
RC1_4_1	400	—	—	—	9331,23	12850,90	1	12850,90	0	9331,23	9447,53	1,23
RC2_4_1 ¹	400	—	—	—	3293,37	26253,00	1	26253,00	0	3293,37	3403,70	3,24

¹ Bounds obtidos a partir da relaxação do recurso referente à entrega.

5.2 PRVCEM

Assim como o PRVCES, considerou-se as instâncias propostas por Salhi e Nagy (1999) para o PRVCEM, cujos resultados obtidos foram comparados aos método B&C com *lazy separations* e BCP de Subramanian (2012). Da análise conjunta das Tabelas 5.6 e 5.7, nota-se que foi resolvida apenas a instância cmt2q, dentre aquelas da literatura que estão em aberto. No entanto, foram reduzidos os *gaps* para as instâncias cmt4t, cmt5h, cmt5q e cmt5t (v. Tabela 5.6). Quanto à comparação com o BCP, melhores resultados para a resolução do nó raiz foram obtidos para as instâncias cmt1q, cmt2q, cmt2t, cmt3t, cmt4h, cmt4q, cmt4t, cmt5q e cmt5t, destacados na Tabela 5.7.

5.3 PRVCESTL

Para o PRVCESTL, Subramanian (2012) aplicou apenas os seus métodos heurístico (ILS-RVND) e híbrido (ILS-RVND-SP) nos testes envolvendo o conjunto de instâncias de Salhi e Nagy (1999) para o problema. Desse modo, estão sendo providos *bounds* inéditos para todas as 14 instâncias do conjunto (v. Tabela 5.8). Em especial, soluções ótimas foram encontradas pela primeira vez para as instâncias cmt6x, cmt6y, cmt14x e

Tabela 5.6: Resultados computacionais das instâncias do PRVCEM — Comparação com o B&C com *lazy separations*.

Instância	C	B&C com <i>lazy constraints</i>			Abordagem com o <i>VRPSolver</i>							
		LB	UB ¹	GAP(%)	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
cmt1h	50	465,02	465,02	0,00	459,15	296,19	179	14248,70	52	465,02	465,02	0,00
cmt1q	50	489,74	489,74	0,00	489,74	34,29	1	34,31	0	489,74	489,74	0,00
cmt1t	50	520,06	520,06	0,00	513,71	146,12	175	21349,20	0	520,06	520,06	0,00
cmt2h	75	653,79	662,63	1,33	647,82	560,09	315	86400,00	95	648,90	662,63	2,07
cmt2q	75	717,36	732,76	2,10	723,46	297,92	531	79473,40	44	731,26	731,26	0,00
cmt2t	75	770,35	782,77	1,59	782,58	194,25	7	296,69	2	782,77	782,77	0,00
cmt3h	100	700,94	700,94	0,00	687,31	846,94	711	86400,00	0	688,36	700,94	1,79
cmt3q	100	747,15	747,15	0,00	741,00	360,92	51	9140,69	2	747,15	747,15	0,00
cmt3t	100	798,07	798,07	0,00	791,94	658,97	161	33492,90	0	798,07	798,07	0,00
cmt12h	100	629,37	629,37	0,00	606,05	958,18	873	86400,00	0	608,76	629,37	3,27
cmt12q	100	729,25	729,25	0,00	717,44	450,76	3307	377709,00	1	729,25	729,25	0,00
cmt12t	100	787,52	787,52	0,00	787,52	343,61	3	348,76	2	787,52	787,52	0,00
cmt11h	120	806,46	820,35	1,69	726,49	68498,60	1	68498,60	0	726,49	818,05	11,19
cmt11q	120	939,36	939,36	0,00	885,60	1640,88	147	86400,00	0	888,02	939,36	5,47
cmt11t	120	989,32	998,80	0,95	977,35	367,75	749	86400,00	0	978,30	998,80	2,05
cmt4h	150	804,10	831,39	3,28	796,86	4405,78	83	86400,00	0	797,82	828,12	3,66
cmt4q	150	898,28	913,93	1,71	895,17	699,39	459	86400,00	0	898,18	913,93	1,72
cmt4t	150	956,54	990,39	3,42	974,56	602,91	237	86400,00	0	975,81	990,39	1,47
cmt5h	199	931,02	992,37	6,18	944,63	29221,90	13	86400,00	0	945,09	978,74	3,44
cmt5q	199	1056,74	1134,72	6,87	1078,80	980,84	261	86400,00	0	1086,07	1104,87	1,70
cmt5t	199	1145,63	1232,08	7,02	1202,97	1192,01	157	86400,00	0	1204,50	1218,77	1,17

¹Obtidos utilizando o método heurístico em (Subramanian, 2012).

Tabela 5.7: Resultados computacionais das instâncias do PRVCEM — Comparação com o BCP.

Instância	C	BCP			Abordagem com o <i>VRPSolver</i>							
		RLB	RT(s)	LB	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
cmt1h	50	461,32	64,70	—	459,15	296,19	179	14248,70	52	465,02	465,02	0,00
cmt1q	50	488,26	31,20	—	489,74	34,29	1	34,31	0	489,74	489,74	0,00
cmt1t	50	515,20	5,81	—	513,71	146,12	175	21349,20	0	520,06	520,06	0,00
cmt2h	75	648,63	174,00	—	647,82	560,09	315	86400,00	95	648,90	662,63	2,07
cmt2q	75	710,00	14,00	—	723,46	297,92	531	79473,40	44	731,26	731,26	0,00
cmt2t	75	767,77	7,31	782,77	782,58	194,25	7	296,69	2	782,77	782,77	0,00
cmt3h	100	691,42	3376,00	—	687,31	846,94	711	86400,00	0	688,36	700,94	1,79
cmt3q	100	744,84	854,00	—	741,00	360,92	51	9140,69	2	747,15	747,15	0,00
cmt3t	100	784,75	346,00	—	791,94	658,97	161	33492,90	0	798,07	798,07	0,00
cmt12h	100	623,33	881,00	—	606,05	958,18	873	86400,00	0	608,76	629,37	3,27
cmt12q	100	722,35	948,00	—	717,44	450,76	3307	377709,00	1	729,25	729,25	0,00
cmt12t	100	787,52	52,70	—	787,52	343,61	3	348,76	2	787,52	787,52	0,00
cmt11h	120	776,36	7680,00	—	726,49	68498,60	1	68498,60	0	726,49	818,05	11,19
cmt11q	120	899,45	7831,00	—	885,60	1640,88	147	86400,00	0	888,02	939,36	5,47
cmt11t	120	985,75	2884,00	—	977,35	367,75	749	86400,00	0	978,30	998,80	2,05
cmt4h	150	795,22	8465,00	—	796,86	4405,78	83	86400,00	0	797,82	828,12	3,66
cmt4q	150	895,15	13090,00	—	895,17	699,39	459	86400,00	0	898,18	913,93	1,72
cmt4t	150	960,59	228,00	—	974,56	602,91	237	86400,00	0	975,81	990,39	1,47
cmt5h	199	948,46	276318,00	—	944,63	29221,90	13	86400,00	0	945,09	978,74	3,44
cmt5q	199	1069,91	97259,00	—	1078,80	980,84	261	86400,00	0	1086,07	1104,87	1,70
cmt5t	199	1175,62	755,00	—	1202,97	1192,01	157	86400,00	0	1204,50	1218,77	1,17

cmt14y. Outro resultado que chama atenção está relacionado aos *gaps* alcançados para as instâncias cmt10x e cmt10y, ambas com 199 clientes, cujos valores são inferiores a 1,00 %. Note que, para todos os testes realizados, não foi detectada nenhuma violação de capacidade e, portanto, não foram aplicados *lazy cuts*.

Tabela 5.8: Resultados computacionais das instâncias do PRVCESTL.

Instância	C	BKS	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
cmt6x	50	555,43	548,19	137,61	173	13457,70	0	555,43	555,43	0,00
cmt6y	50	555,43	548,21	170,31	167	13096,60	0	555,43	555,43	0,00
cmt7x	75	900,12	888,27	127,00	2227	86400,00	0	890,54	900,12	1,06
cmt7y	75	900,12	888,27	122,80	2829	86400,00	0	888,54	900,12	1,29
cmt8x	100	865,50	837,97	765,31	135	86400,00	0	844,16	865,50	2,47
cmt8y	100	865,50	831,05	478,78	267	86400,00	0	839,09	865,50	3,05
cmt14x	100	821,75	818,26	913,90	27	7567,95	0	821,75	821,75	0,00
cmt14y	100	821,75	819,37	1062,21	109	41446,70	0	821,75	821,75	0,00
cmt13x	120	1542,86	1450,40	2736,43	127	86400,00	0	1452,10	1542,86	5,88
cmt13y	120	1542,86	1450,79	3034,72	135	86400,00	0	1452,78	1542,86	5,84
cmt9x	150	1160,68	1135,03	1414,06	131	86400,00	0	1139,41	1160,68	1,83
cmt9y	150	1160,68	1135,17	1178,34	139	86400,00	0	1139,46	1160,68	1,83
cmt10x	199	1373,40	1359,52	4038,77	249	86400,00	0	1363,26	1373,40	0,74
cmt10y	199	1373,40	1352,49	2230,15	211	86400,00	0	1359,79	1373,40	0,99

5.4 PRVCEMTL

Similarmente ao PRVCESTL, foram aplicados apenas os métodos ILS-RVND e ILS-RVND nos testes com as instâncias do PRVCEMTL. Assim, são providos novamente *bounds* inéditos para todas as instâncias do conjunto (Tabela 5.9), dando-se destaque às instâncias cmt6h, cmt6q, cmt6t, cmt7q, cmt7t, cmt14h, cmt14q e cmt14t que foram resolvidas em otimalidade. Note que os *lazy cuts* também não foram necessários.

5.5 PRVCESJT

Tratando-se do PRVCESJT, experimentos computacionais foram realizados com as instâncias propostas por Wang e Chen (2012). Este conjunto, por sua vez, é subdividido em instâncias cujos clientes formam *clusters* (W12_C), estão dispersos (W12_R) e um misto entre agrupados e dispersos (W12_RC). Para o problema em questão, há uma

Tabela 5.9: Resultados computacionais das instâncias do PRVCEMTL.

Instância	C	BKS	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
cmt6h	50	555,43	548,31	209,70	219	25080,00	0	555,43	555,43	0,00
cmt6q	50	555,43	548,28	156,93	245	19898,90	0	555,43	555,43	0,00
cmt6t	50	555,43	549,93	147,60	123	12649,90	0	555,43	555,43	0,00
cmt7h	75	900,54	887,19	108,12	1515	86400,00	0	888,42	900,54	1,35
cmt7q	75	900,69	893,28	100,89	933	23514,70	0	900,69	900,69	0,00
cmt7t	75	903,05	900,54	82,98	41	818,33	0	903,05	903,05	0,00
cmt8h	100	865,50	831,01	493,18	187	86400,00	0	838,48	865,50	3,12
cmt8q	100	865,50	841,37	810,39	193	86400,00	0	844,77	865,50	2,39
cmt8t	100	865,54	846,14	496,40	295	86400,00	0	847,85	865,54	2,04
cmt14h	100	821,75	818,62	345,76	37	7767,58	0	821,75	821,75	0,00
cmt14q	100	821,75	819,82	539,26	7	2256,14	0	821,75	821,75	0,00
cmt14t	100	826,77	821,07	374,26	213	59050,90	0	826,77	826,77	0,00
cmt13h	120	1542,86	1452,3	3374,10	103	86400,00	0	1452,59	1542,86	5,85
cmt13q	120	1542,86	1450,23	2646,97	109	86400,00	0	1452,41	1542,86	5,86
cmt13t	120	1541,25	1450,76	692,27	173	86400,00	0	1458,98	1541,25	5,34
cmt9h	150	1160,68	1127,7	841,80	147	86400,00	0	1136,03	1160,68	2,12
cmt9q	150	1161,24	1135,47	892,13	207	86400,00	0	1140,03	1161,24	1,83
cmt9t	150	1162,55	1144,55	968,51	207	86400,00	0	1145,83	1162,55	1,44
cmt10h	199	1372,52	1350,31	2491,20	213	86400,00	0	1358,24	1372,52	1,04
cmt10q	199	1374,18	1351,34	1463,26	175	86400,00	0	1358,78	1374,18	1,12
cmt10t	199	1381,04	1357,96	911,28	255	86400,00	0	1365,94	1381,04	1,09

predominância de métodos heurísticos na literatura, com soluções ótimas já provadas apenas para as instancias rcdp1001, rcdp1004, rcdp1007, rcdp2501, rcdp5001, rdp101, rdp102, rdp105, rdp106, cdp105 e cdp106. Desse modo, de acordo com a Tabelas 5.10, 5.11 e 5.12, o método proposto foi capaz de encontrar soluções ótimas para 12 dentre 17 instâncias do conjunto W12_C, 22 dentre 28 instâncias do conjunto W12_RC e 13 dentre 23 instâncias do conjunto W12_R, respectivamente, totalizando 47 soluções ótimas provadas, das quais 36 são inéditas. Importante ressaltar que foram encontradas solução melhores do que o BKS para as instâncias rcdp105, rdp107, rdp108 e rdp110.

5.6 PRVCESJTF

Antes de serem apresentados os resultados relacionados ao PRVCESJTF, faz-se necessário destacar o aumento de dificuldade na resolução do problema, quando comparado ao PRVCESJT. Neste caso, como explicado, há janelas de tempo distintas para coleta e entrega. Desse modo, a resolução do problema implica a duplicação de todos

Tabela 5.10: Resultados computacionais das instâncias do PRVCESJT — W12_C.

Instância	C	VD	BKS		Abordagem com o <i>VRPSolver</i>								
			VU	UB	VU	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
cdp101	100	25	11	976,04	—	926,55	394,08	41	86400,00	28	930,81	976,04	4,63
cdp102	100	25	10	941,49	10	916,48	159,42	371	28239,50	308	941,49	941,49	0,00
cdp103	100	25	10	892,98	10	876,49	376,55	335	76051,50	149	892,98	892,98	0,00
cdp104	100	25	10	871,40	10	852,84	338,90	917	86400,00	473	856,20	871,40	1,74
cdp105	100	25	10	1053,12	10	982,52	15,51	363	2973,92	439	1053,12	1053,12	0,00
cdp106	100	25	10	963,45	10	932,98	26,13	307	2692,13	287	963,45	963,45	0,00
cdp107	100	25	10	987,64	10	918,04	41,66	1981	86400,00	2832	923,42	987,64	6,50
cdp108	100	25	10	932,49	—	863,09	243,81	565	86400,00	935	868,62	932,49	6,85
cdp109	100	25	10	909,27	—	887,05	229,49	1669	86400,00	1915	889,62	909,27	2,16
cdp201	100	25	3	591,56	3	591,56	55,10	1	55,10	0	591,56	591,56	0,00
cdp202	100	25	3	591,56	3	591,56	790,26	1	790,26	0	591,56	591,56	0,00
cdp203	100	25	3	591,17	3	591,17	2398,18	1	2398,18	0	591,17	591,17	0,00
cdp204	100	25	3	590,60	3	590,60	2086,44	1	2086,44	0	590,60	590,60	0,00
cdp205	100	25	3	588,88	3	588,88	55,82	1	55,82	0	588,88	588,88	0,00
cdp206	100	25	3	588,49	3	588,49	35,05	1	35,05	0	588,49	588,49	0,00
cdp207	100	25	3	588,29	3	588,29	127,91	1	127,91	0	588,29	588,29	0,00
cdp208	100	25	3	588,32	3	588,32	46,50	1	46,50	0	588,32	588,32	0,00

Tabela 5.11: Resultados computacionais das instâncias do PRVCESJT — W12_RC.

Instância	C	VD	BKS		Abordagem com o <i>VRPSolver</i>								
			VU	UB	VU	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
redp0501	5	25	—	—	3	220,15	0,01	1	0,01	0	220,15	220,15	0,00
redp0504	5	25	—	—	2	214,57	0,01	1	0,01	0	214,57	214,57	0,00
redp0507	5	25	—	—	2	211,83	0,01	1	0,01	0	211,83	211,83	0,00
redp1001	10	3	3	348,98	3	348,98	0,02	1	0,02	0	348,98	348,98	0,00
redp1004	10	3	2	216,69	2	216,69	0,04	1	0,04	0	216,69	216,69	0,00
redp1007	10	3	2	310,81	2	310,81	0,04	1	0,04	0	310,81	310,81	0,00
redp2501	25	10	5	551,05	5	551,05	0,20	1	0,20	0	551,05	551,05	0,00
redp2504	25	10	4	473,46	4	473,46	0,18	1	0,18	0	473,46	473,46	0,00
redp2507	25	10	5	540,87	5	540,87	0,37	1	0,37	0	540,87	540,87	0,00
redp5001	50	15	9	994,18	9	994,18	0,84	1	0,84	0	994,18	994,18	0,00
redp5004	50	15	6	733,21	6	733,21	16,00	1	16,01	0	733,21	733,21	0,00
redp5007	50	15	7	809,72	7	808,43	5,40	3	7,65	0	809,72	809,72	0,00
redp101	100	25	14	1708,21	14	1708,21	35,68	1	35,68	0	1708,21	1708,21	0,00
redp102	100	25	12	1570,28	12	1561,25	136,10	57	1499,24	61	1570,28	1570,28	0,00
redp103	100	25	11	1282,53	11	1269,89	117,99	857	48323,40	967	1282,53	1282,53	0,00
redp104	100	25	10	1171,37	10	1157,58	390,19	901	65697,70	965	1171,37	1171,37	0,00
redp105	100	25	13	1651,93	13	1631,34	43,12	9	76,19	6	1632,29	1632,29	0,00
redp106	100	25	12	1392,47	12	1378,83	237,38	783	38766,10	616	1392,47	1392,47	0,00
redp107	100	25	11	1252,79	11	1244,8	126,63	71	2412,86	67	1252,79	1252,79	0,00
redp108	100	25	10	1198,36	—	1143,8	2668,85	345	86400,00	253	1156,00	1198,36	3,53
redp201	100	25	4	1406,94	4	1406,94	305,85	1	305,85	0	1406,94	1406,94	0,00
redp202	100	25	3	1412,52	—	1158,9	1109,86	101	86400,00	0	1168,9	1412,52	17,25
redp203	100	25	3	1050,64	—	859,9	1039,37	73	78977,50	0	868,02	1050,64	17,38
redp204	100	25	3	798,46	—	744,3	3143,20	75	86400,00	0	752,2	798,46	5,79
redp205	100	25	4	1297,65	4	1297,65	491,21	1	491,21	0	1297,65	1297,65	0,00
redp206	100	25	3	1146,32	3	1100,64	541,53	335	72290,60	0	1146,32	1146,32	0,00
redp207	100	25	3	1061,84	—	981,41	1191,84	303	86400,00	0	990,72	1061,84	6,70
redp208	100	25	3	828,14	—	744,96	1161,46	121	86400,00	0	747,99	828,14	9,68

Tabela 5.12: Resultados computacionais das instâncias do PRVCESJT — W12_R.

Instância	C	VD	BKS		Abordagem com o <i>VRPSolver</i>									GAP (%)
			VU	UB	VU	RLB	RT(s)	NN	TT(s)	LC	LB	UB		
rdp101	100	25	19	1650,80	19	1650,80	2,65	1	2,65	0	1650,80	1650,80	0,00	
rdp102	100	25	17	1486,12	17	1486,12	3,82	1	3,82	0	1486,12	1486,12	0,00	
rdp103	100	25	13	1294,64	13	1294,61	20,05	3	24,41	0	1294,64	1294,64	0,00	
rdp104	100	25	10	984,81	10	979,68	351,48	17	2381,85	1	984,81	984,81	0,00	
rdp105	100	25	14	1377,11	14	1376,79	86,57	3	118,85	0	1377,11	1377,11	0,00	
rdp106	100	25	12	1252,03	12	1252,03	47,36	1	47,36	0	1252,03	1252,03	0,00	
rdp107	100	25	10	1121,86	10	1100,76	204,40	1591	69903,10	775	1112,55	1112,55	0,00	
rdp108	100	25	9	965,54	9	955,80	439,08	51	10069,00	15	965,22	965,22	0,00	
rdp109	100	25	11	1194,73	11	1187,69	209,59	11	1000,09	0	1194,73	1194,73	0,00	
rdp110	100	25	10	1148,20	10	1112,04	415,86	29	5499,80	0	1121,46	1121,46	0,00	
rdp111	100	25	10	1098,84	10	1098,62	358,57	3	426,30	0	1098,84	1098,84	0,00	
rdp112	100	25	9	1010,42	—	967,51	543,97	375	86400,00	41	973,40	1010,42	3,66	
rdp201	100	25	4	1252,37	4	1252,37	457,46	1	457,48	0	1252,37	1252,37	0,00	
rdp202	100	25	3	1191,70	—	1107,23	2020,19	183	86400,00	0	1116,26	1191,70	6,33	
rdp203	100	25	3	943,50	—	869,67	1228,35	17	21204,80	0	875,20	943,50	7,24	
rdp204	100	25	2	833,09	—	728,14	11952,50	9	86400,00	0	730,37	833,09	12,33	
rdp205	100	25	3	994,43	3	994,43	850,29	1	850,30	0	994,43	994,43	0,00	
rdp206	100	25	3	906,14	—	861,34	556,52	419	86400,00	0	865,07	906,14	4,53	
rdp207	100	25	2	890,61	—	808,68	31504,70	3	86400,00	0	811,93	890,61	8,83	
rdp208	100	25	2	726,82	—	691,18	6644,13	85	86400,00	0	692,95	726,82	4,66	
rdp209	100	25	3	909,16	—	867,03	1135,78	503	86400,00	0	870,32	909,16	4,27	
rdp210	100	25	3	939,37	—	904,67	959,40	305	86400,00	0	909,14	939,37	3,22	
rdp211 ¹	100	25	2	904,44	—	820,28	25152,80	1	25152,80	0	820,28	904,44	9,31	

¹ Bounds obtidos a partir da relaxação do recurso referente à entrega.

os clientes, um para cada tipo de demanda, o que aumenta sua dificuldade. Tendo em vista a necessidade de criação de clientes artificiais, optou-se por buscar soluções ótimas para as instâncias com até 50 clientes originalmente, enquanto foram gerados apenas *bounds* obtidos a partir da resolução do nó raiz para as demais instâncias.

Nas Tabelas 5.13 - 5.15 estão indicados os resultados obtidos para as instâncias do PRVCESJT. Soluções ótimas foram encontradas para todas as instâncias com até 50 clientes, exceto para RCf50104. Além disso, o método proposto foi capaz de provar a otimalidade para três instâncias com 100 clientes (Cf106, Rf101 e Rf103). Soluções já conhecidas para as instâncias Cf201, Cf202, Cf205 e Cf208 também foram classificadas como ótimas, contudo por meio da relaxação do recurso de entrega. Por fim, os resultados apresentados para as instâncias Cf206 e Rf211, respectivamente, são idênticos aos das suas correspondentes instâncias no PRVCESJT (cdp106 e rdp211). Os dados contidos em cada par eram os mesmos e, portanto, não foi possível a realização de testes com a versão flexível do PRVCESJT em tais casos.

Tabela 5.13: Resultados computacionais das instâncias do PRVCESJTF — W13_C.

Instância	C	VD	BKS		Abordagem com o <i>VRPSolver</i>									GAP (%)
			VU	UB	VU	RLB	RT(s)	NN	TT(s)	LC	LB	UB		
Cf101	100	25	10	858,24	—	842,65	2539,22	1	2539,22	6	842,65	858,24	1,82	
Cf102	100	25	10	897,23	—	852,6	11011,30	1	11011,30	0	852,60	897,23	4,97	
Cf103	100	25	10	850,10	—	838,23	13594,00	1	13594,00	0	838,23	850,10	1,40	
Cf104	100	25	10	898,04	—	829,93	16864,60	1	16864,60	0	829,93	898,04	7,58	
Cf105	100	25	10	922,05	—	868,4	3277,66	1	3277,66	0	868,40	922,05	5,82	
Cf106	100	25	10	862,08	10	862,08	3435,07	3	3436,74	1	862,08	862,08	0,00	
Cf107	100	25	10	854,69	—	846,23	3026,63	935	86400,00	1784	847,42	854,69	0,85	
Cf108	100	25	10	840,51	—	831,44	3110,04	1	3110,04	5	831,44	840,51	1,08	
Cf109	100	25	10	928,47	—	826,57	8958,23	1	8958,23	0	826,57	928,47	10,97	
Cf201 ¹	100	25	3	591,56	3	591,56	8232,88	1	8232,88	0	591,56	591,56	0,00	
Cf202 ¹	100	25	3	591,56	3	591,56	47125,20	1	47125,20	0	591,56	591,56	0,00	
Cf203 ¹	100	25	3	591,17	—	591,173	10797,60	1	10797,60	1	591,173	591,1749	0,0003	
Cf204 ¹	100	25	3	590,60	—	588,0	27012,80	1	27012,80	0	588,00	590,60	0,44	
Cf205 ¹	100	25	3	588,88	3	588,88	6042,67	1	6042,67	0	588,88	588,88	0,00	
Cf206 ²	100	25	3	588,49	3	588,49	35,05	1	35,05	0	588,49	588,49	0,00	
Cf207 ¹	100	25	3	588,29	—	588,29	33450,50	1	33450,50	1	588,286	588,29	0,0007	
Cf208 ¹	100	25	3	588,32	3	588,32	20990,80	1	20990,80	0	588,32	588,32	0,00	

¹ Bounds obtidos a partir da relaxação do recurso referente à entrega.² Dados de entrada idênticos aos da sua instância correspondente no PRVCESJT (cdp206).

5.7 PRVCESA

Em relação aos testes relacionados ao PRVCESA, foram resolvidas em otimalidade todas as 42 instâncias dos conjuntos RZC6 e RZC7 (v. Tabelas 5.16 e 5.17, respectivamente), das quais 20 soluções ótimas são inéditas. Vale ressaltar que não foi necessária a relaxação de nenhum recurso na resolução das mesmas. Além disso, percebe-se que não foi necessário utilizar os *lazy cuts* para as duas instâncias do conjunto RZC7, sendo ambas resolvidas no nó raiz, enquanto que, para a instância SCA8-9 (conjunto RZC6), foram bastante utilizados.

5.8 PRVCESFH

Para os experimentos computacionais envolvendo o PRVCESFH, foram consideradas as instâncias com até 100 clientes do conjunto proposto por Avci e Topaloglu (2016). Como não há nenhuma outra abordagem exata que resolva essas instâncias, apenas métodos heurísticos até o momento, não foi realizada nenhuma comparação direta. Apesar disso, de acordo com a Tabela 5.18, foram resolvidas 9 dentre as 14 instâncias consideradas, (PR_1_1 - PR_9_1), das quais eram conhecidas as soluções ótimas

Tabela 5.14: Resultados computacionais das instâncias do PRVCESJTF — W13_RC.

Instância	C	VD	BKS		Abordagem com o <i>VRPSolver</i>									GAP (%)
			VU	UB	VU	RLB	RT(s)	NN	TT(s)	LC	LB	UB		
RCf05101	5	3	3	220,15	3	220,15	0,03	1	0,03	0	220,15	220,15	0,00	
RCf05104	5	3	2	214,57	2	214,57	0,05	1	0,05	0	214,57	214,57	0,00	
RCf05107	5	3	2	211,83	2	211,83	0,34	1	0,34	0	211,83	211,83	0,00	
RCf10101	10	3	3	347,38	3	347,38	0,07	1	0,07	0	347,38	347,38	0,00	
RCf10104	10	3	2	216,69	2	216,69	0,73	1	0,73	0	216,69	216,69	0,00	
RCf10107	10	3	2	310,81	2	310,81	1,70	1	1,70	0	310,81	310,81	0,00	
RCf25101	25	12	5	529,13	5	529,13	3,27	1	3,27	0	529,13	529,13	0,00	
RCf25104	25	12	4	473,46	4	473,46	24,56	1	24,56	0	473,46	473,46	0,00	
RCf25107	25	12	4	567,15	4	567,15	153,27	1	153,27	0	567,15	567,15	0,00	
RCf50101	50	15	9	928,50	9	928,50	17,89	1	17,89	0	928,50	928,50	0,00	
RCf50104	50	15	6	738,30	—	664,99	2009,66	83	86400,00	0	671,03	738,30	9,11	
RCf50107	50	15	7	807,86	7	783,36	785,27	323	23354,50	166	793,84	793,84	0,00	
RCf101	100	25	14	1728,07	—	1570,23	1039,60	1	1039,60	0	1570,23	1728,07	9,13	
RCf102	100	25	13	1493,89	—	1374,94	2502,96	1	2502,96	0	1374,94	1493,89	7,96	
RCf103	100	25	12	1334,29	—	1203,52	10266,50	1	10266,50	0	1203,52	1334,29	9,80	
RCf104	100	25	12	1254,67	—	1115,14	43605,70	1	43605,70	0	1115,14	1254,67	11,12	
RCf105	100	25	14	1538,77	—	1434,11	2322,62	1	2322,62	0	1434,11	1538,77	6,80	
RCf106	100	25	13	1444,44	—	1285,47	5343,06	1	5343,06	0	1285,47	1444,44	11,01	
RCf107	100	25	12	1292,51	—	1177,38	14369,50	1	14369,50	0	1177,38	1292,51	8,91	
RCf108	100	25	11	1161,16	—	1069,77	31539,00	1	31539,00	0	1069,77	1161,16	7,87	
RCf201 ¹	100	25	4	1504,38	—	1259,26	17988,90	1	17988,90	0	1259,26	1504,38	16,29	
RCf202 ¹	100	25	4	1269,46	—	1032,02	56452,90	1	56452,90	0	1032,02	1269,46	18,70	
RCf203 ¹	100	25	4	1054,28	—	678,93	77146,30	1	77146,30	0	678,93	1054,28	35,60	
RCf204 ²	100	25	3	817,01	—	—	—	—	—	—	—	—	—	
RCf205 ¹	100	25	4	1417,18	—	1141,09	32687,90	1	32687,90	0	1141,09	1417,18	19,48	
RCf206 ¹	100	25	4	1128,03	—	997,24	61724,70	1	61724,70	0	997,24	1128,03	11,59	
RCf207 ¹	100	25	4	1053,44	—	728,71	86400,00	1	86400,00	0	728,71	1053,44	30,83	
RCf208 ²	100	25	3	1020,95	—	—	—	—	—	—	—	—	—	

¹ Bounds obtidos a partir da relaxação do recurso referente à coleta.² Mesmo com a relaxação dos recursos, não foi possível encontrar bounds para esta instância.

Tabela 5.15: Resultados computacionais das instâncias do PRVCESJTF — W13_R.

Instância	C	VD	BKS		Abordagem com o <i>VRPSolver</i>									GAP (%)
			VU	UB	VU	RLB	RT(s)	NN	TT(s)	LC	LB	UB		
Rf101	100	25	18	1606,16	18	1574,92	162,87	1	162,87	0	1574,92	1574,92	0,00	
Rf102	100	25	16	1443,61	—	1399,43	1314,80	1	1314,80	0	1399,43	1443,61	3,06	
Rf103	100	25	14	1227,14	14	1208,74	2424,80	1	2424,80	0	1208,74	1208,74	0,00	
Rf104	100	25	10	1022,58	—	950,57	13607,60	1	13607,60	0	950,57	1022,58	7,04	
Rf105	100	25	14	1391,09	—	1343,81	1277,55	1	1277,56	0	1343,81	1391,09	3,40	
Rf106	100	25	13	1291,21	—	1208,09	2353,14	1	2353,14	0	1208,09	1291,21	6,44	
Rf107	100	25	11	1104,33	—	1042,25	5373,38	1	5373,38	0	1042,25	1104,33	5,62	
Rf108	100	25	10	974,99	—	913,16	18534,50	1	18534,50	0	913,16	974,99	6,34	
Rf109	100	25	12	1201,52	—	1125,25	1843,66	1	1843,66	0	1125,25	1201,52	6,35	
Rf110	100	25	12	1136,52	—	1049,69	5948,67	1	5948,67	0	1049,69	1136,52	7,64	
Rf111	100	25	11	1119,90	—	1028,12	6663,99	1	6663,99	0	1028,12	1119,90	8,20	
Rf112	100	25	10	976,69	—	924,40	12460,40	1	12460,40	0	924,40	976,69	5,35	
Rf201 ¹	100	25	4	1289,15	—	1138,67	10412,00	1	10412,00	0	1138,67	1289,15	11,67	
Rf202 ¹	100	25	4	1133,73	—	973,63	33097,90	1	33097,90	0	973,63	1133,73	14,12	
Rf203 ¹	100	25	3	974,37	—	734,57	89147,10	1	89147,10	0	734,57	974,37	24,61	
Rf204 ²	100	25	3	775,67	—	—	—	—	—	—	—	—	—	
Rf205 ¹	100	25	3	1101,47	—	938,62	55012,20	1	55012,20	0	938,62	1101,47	14,78	
Rf206 ¹	100	25	3	1011,14	—	850,58	86436,90	1	86436,90	0	850,58	1011,14	15,88	
Rf207 ²	100	25	3	839,75	—	—	—	—	—	—	—	—	—	
Rf208 ²	100	25	3	728,18	—	—	—	—	—	—	—	—	—	
Rf209 ¹	100	25	3	1008,49	—	657,94	26944,20	1	26944,20	0	657,94	1008,49	34,76	
Rf210 ¹	100	25	3	1076,93	—	866,23	86408,20	1	86408,20	0	866,23	1076,93	19,57	
Rf211 ¹³	100	25	3	826,04	—	820,28	25152,80	1	25152,80	0	820,28	904,44	9,31	

¹ Bounds obtidos a partir da relaxação do recurso referente à entrega.² Mesmo com a relaxação dos recursos, não foi possível encontrar bounds para esta instância.³ Dados de entrada idênticos aos da sua instância correspondente no PRVCESJT (rdp211).

Tabela 5.16: Resultados computacionais das instâncias do PRVCESA — RZC6.

Instância	C	Formulações de fluxo			Abordagem com o <i>VRPSolver</i>							
		UB	GAP(%)	M	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
CON3-0	50	667,46	0,00	m_1	664,79	121,64	11	628,27	0	667,46	667,46	0,00
CON3-1	50	590,82	0,00	m_1	587,84	551,60	15	5968,20	2	590,82	590,82	0,00
CON3-2	50	558,89	0,00	m_2	552,00	136,39	15	2357,08	0	558,89	558,89	0,00
CON3-3	50	634,93	0,00	m_1	634,93	57,82	1	57,83	0	634,93	634,93	0,00
CON3-4	50	627,95	0,00	m_1	626,03	274,25	7	1140,32	0	627,95	627,95	0,00
CON3-5	50	603,56	0,00	m_1	600,52	184,64	17	1864,26	8	603,56	603,56	0,00
CON3-6	50	539,58	0,00	m_1	535,46	269,38	57	17575,90	8	539,58	539,58	0,00
CON3-7	50	627,05	0,00	m_1	627,05	56,31	1	56,34	0	627,05	627,05	0,00
CON3-8	50	561,65	0,00	m_1	561,65	88,16	1	88,18	0	561,65	561,65	0,00
CON3-9	50	619,95	0,00	m_1	615,29	212,71	153	10633,90	80	619,95	619,95	0,00
CON8-0	50	918,21	1,80	m_1	907,13	18,49	145	417,64	141	918,07	918,07	0,00
CON8-1	50	772,44	1,00	m_2	772,44	16,23	1	16,23	1	772,44	772,44	0,00
CON8-2	50	738,99	0,60	m_2	738,99	11,92	1	11,92	0	738,99	738,99	0,00
CON8-3	50	857,35	1,30	m_2	855,26	14,14	17	39,32	7	857,35	857,35	0,00
CON8-4	50	816,81	1,50	m_1	814,78	15,44	13	33,93	5	816,81	816,81	0,00
CON8-5	50	798,07	1,50	m_2	791,69	21,06	15	60,12	12	795,79	795,79	0,00
CON8-6	50	718,36	2,20	m_1	711,18	42,87	173	1280,45	169	717,5	717,50	0,00
CON8-7	50	863,39	0,50	m_1	862,90	18,69	3	22,13	0	863,39	863,39	0,00
CON8-8	50	808,17	3,90	m_2	806,86	21,15	5	34,20	3	808,17	808,17	0,00
CON8-9	50	843,84	3,00	m_2	839,07	19,83	19	123,64	19	841,39	841,39	0,00
SCA3-0	50	677,35	0,00	m_1	663,95	127,78	59	11271,60	10	677,35	677,35	0,00
SCA3-1	50	758,90	0,00	m_1	740,61	138,86	75	15575,00	14	758,9	758,90	0,00
SCA3-2	50	735,18	0,00	m_1	731,91	46,90	47	228,27	38	735,18	735,18	0,00
SCA3-3	50	735,79	0,00	m_1	735,79	68,08	1	68,10	0	735,79	735,79	0,00
SCA3-4	50	741,75	0,00	m_1	736,39	253,01	17	3527,41	0	741,75	741,75	0,00
SCA3-5	50	702,45	0,00	m_1	683,67	76,96	435	114723,00	0	702,45	702,45	0,00
SCA3-6	50	707,72	0,00	m_1	687,92	189,07	177	21320,10	0	707,72	707,72	0,00
SCA3-7	50	708,24	0,00	m_1	702,15	67,68	35	960,35	18	708,24	708,24	0,00
SCA3-8	50	771,94	0,00	m_1	756,01	130,82	65	13238,50	0	771,94	771,94	0,00
SCA3-9	50	726,77	0,00	m_1	710,95	117,08	47	3245,79	5	726,77	726,77	0,00
SCA8-0	50	1026,79	3,90	m_1	1015,40	39,38	103	563,39	113	1026,79	1026,79	0,00
SCA8-1	50	1127,41	3,30	m_1	1113,96	28,57	109	995,19	64	1126,16	1126,16	0,00
SCA8-2	50	1126,12	2,20	m_1	1113,20	41,56	139	1614,96	60	1126,12	1126,12	0,00
SCA8-3	50	1062,99	3,00	m_2	1058,74	16,04	19	44,97	18	1062,99	1062,99	0,00
SCA8-4	50	1114,12	1,20	m_1	1114,12	14,02	1	14,02	0	1114,12	1114,12	0,00
SCA8-5	50	1085,96	1,40	m_1	1078,53	19,00	25	81,06	10	1085,96	1085,96	0,00
SCA8-6	50	1038,59	1,30	m_1	1023,24	20,43	483	1568,95	605	1033,5	1033,50	0,00
SCA8-7	50	1114,17	3,10	m_1	1096,05	17,74	129	524,30	131	1108,42	1108,42	0,00
SCA8-8	50	1165,08	4,40	m_1	1134,89	43,50	49	631,67	35	1143,84	1143,84	0,00
SCA8-9	50	1145,71	3,20	m_1	1124,71	30,74	2369	11996,30	3032	1144,07	1144,07	0,00

Tabela 5.17: Resultados computacionais das instâncias do PRVCESA — RZC7.

Instância	C	Formulações de fluxo			Abordagem com o <i>VRPSolver</i>							
		UB	GAP(%)	M	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
CMT1x	50	510,23	0,00	m_2	510,23	163,87	1	163,90	0	510,23	510,23	0,00
CMT1y	50	511,38	0,00	m_1	511,38	109,04	1	109,07	0	511,38	511,38	0,00

para apenas as três primeiras.

Tabela 5.18: Resultados computacionais das instâncias do PRVCESFH.

Instância	C	TVD	TVU	BKS	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
PR_1_1	10	2	2	620,2	620,23	0,37	1	0,37	0	620,23	620,23	0,00
PR_2_1	10	2	2	588,5	588,53	0,31	1	0,31	0	588,53	588,53	0,00
PR_3_1	15	3	2	445,1	445,13	1,81	1	1,81	0	445,13	445,13	0,00
PR_4_1	15	4	3	437,1	437,09	2,95	3	3,14	1	437,09	437,09	0,00
PR_5_1	20	3	2	494,0	490,7	10,63	15	29,02	12	493,99	493,99	0,00
PR_6_1	20	4	2	542,7	539,36	19,79	13	91,84	0	542,72	542,72	0,00
PR_7_1	35	3	2	1097,3	1086,33	322,00	191	4470,53	187	1092,22	1092,22	0,00
PR_8_1	35	3	3	1586,5 ¹	1590,32	49,59	21	222,36	10	1598,53	1598,53	0,00
PR_9_1	50	3	2	937,0	930,34	486,82	81	2700,61	40	931,73	931,73	0,00
PR_10_1	50	2	—	1169,2	1101,09	591,99	103	86400,00	0	1105,97	1169,2	5,41
PR_11_1	75	3	—	1531,9	1480,13	2747,42	55	86400,00	0	1481,93	1531,9	3,26
PR_12_1	75	2	—	925,6	883,82	7792,71	27	86400,00	0	885,67	925,6	4,31
PR_13_1	100	2	—	1299,5	1169,71	16762,00	17	86400,00	0	1170,51	1299,5	9,93
PR_14_1	100	2	—	1539,4	1404,71	71876,40	1	86400,00	0	1404,71	1539,4	8,75

¹ Em contato com os autores (Avci e Topaloglu, 2016), confirmou-se que o valor em questão foi reportado erroneamente.

5.9 PRVCEMMD

Para a variante com múltiplos depósitos, segue o mesmo procedimento já realizado para o PRVCEM, isto é, utilização das instâncias propostas por Salhi e Nagy (1999) especificamente para essa variante e comparação com os resultados reportados por Subramanian (2012). A única diferença aqui consiste no fato da comparação ser realizada apenas com o algoritmo B&C com *lazy separations*, tendo em vista que o método BCP não foi utilizado nos testes para o PRVCEMMD. Desse modo, na Tabela 5.19 está indicado que foram resolvidas 16 dentre as 21 instâncias do conjunto considerado, sendo obtidas as soluções ótimas para 12 delas pela primeira vez. Além disso, foram melhorados os LBs para as instâncias GJ04H, GJ06H, GJ07H e GJ04Q, bem como propostos *bounds* inéditos para as instâncias com 249 clientes.

Tabela 5.19: Resultados computacionais das instâncias do PRVCEMMD.

Instância	C	DD	DU	B&C com <i>lazy constraints</i>			Abordagem com o <i>VRPSolver</i>							
				LB	UB ¹	GAP(%)	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
GJ01H	50	4	4	480,43	499,12	3,74	490,06	127,73	2933	58625,60	3064	499,12	499,12	0,00
GJ02H	50	4	4	440,00	440,00	0,00	438,58	163,16	17	272,19	4	440,00	440,00	0,00
GJ03H	75	5	5	572,75	579,45	1,16	572,97	308,38	2783	203836,00	4445	579,45	579,45	0,00
GJ04H	100	2	—	754,33	789,19	4,42	760,07	540,43	1673	259200,00	1620	764,61	789,19	3,11
GJ05H	100	2	—	668,72	676,81	1,20	662,00	1338,33	1183	259200,00	0	662,81	676,81	2,07
GJ06H	100	3	—	709,71	742,18	4,38	724,28	930,88	1057	259200,00	2010	729,33	742,18	1,73
GJ07H	100	4	—	706,86	732,73	3,53	723,06	913,76	3481	432000,00	4591	725,43	732,73	1,00
GJ08H ²	249	2	—	—	—	—	3000,04	8964,68	1	8964,68	0	3000,04	3348,39	10,40
GJ09H ²	249	3	—	—	—	—	2748,16	8350,72	1	8350,72	0	2748,16	2990,92	8,12
GJ10H ²	249	4	—	—	—	—	2670,98	8421,47	1	8421,47	0	2670,98	2894,71	7,73
GJ11H ²	249	5	—	—	—	—	2644,04	9524,19	1	9524,19	0	2644,04	2842,79	6,99
GJ01Q	50	4	4	515,09	528,30	2,50	526,13	30,46	25	126,43	10	528,30	528,30	0,00
GJ02Q	50	4	4	449,72	449,72	0,00	449,72	35,92	1	35,93	0	449,72	449,72	0,00
GJ03Q	75	5	5	91,35	605,25	2,30	604,39	269,72	15	701,66	6	605,25	605,25	0,00
GJ04Q	100	2	—	841,74	874,78	3,78	865,38	384,46	3027	259200,00	456	866,31	874,78	0,97
GJ05Q	100	2	2	700,15	700,15	0,00	694,01	963,00	45	10428,50	1	700,15	700,15	0,00
GJ06Q	100	3	3	765,70	793,85	3,55	789,31	259,40	61	3685,74	1	793,85	793,85	0,00
GJ07Q	100	4	4	768,21	802,20	4,24	795,11	608,57	1265	130971,00	131	801,91	801,91	0,00
GJ08Q	249	2	—	—	—	—	3588,52	35569,20	1	35569,20	0	3588,52	3745,18	4,18
GJ09Q	249	3	—	—	—	—	3222,41	41491,10	1	41491,10	0	3222,41	3351,18	3,84
GJ10Q	249	4	—	—	—	—	3095,63	41693,10	1	41693,10	0	3095,63	3220,79	3,89
GJ11Q	249	5	—	—	—	—	3029,66	32911,20	1	32911,20	0	3029,66	3138,64	3,47
GJ01T	50	4	4	558,12	569,43	1,99	569,43	21,87	1	21,88	0	569,43	569,43	0,00
GJ02T	50	4	4	464,13	464,13	0,00	464,13	53,17	1	53,18	0	464,13	464,13	0,00
GJ03T	75	5	5	606,68	624,44	2,84	623,40	135,33	13	503,70	6	624,44	624,44	0,00
GJ04T	100	2	2	923,90	962,25	3,99	959,71	260,06	17	1395,35	0	962,25	962,25	0,00
GJ05T	100	2	2	723,35	733,17	1,34	727,34	590,19	141	26782,70	0	733,17	733,17	0,00
GJ06T	100	3	3	815,61	850,82	4,14	848,04	302,38	43	2318,99	0	850,82	850,82	0,00
GJ07T	100	4	4	816,31	853,54	4,36	851,62	343,47	17	1234,60	0	853,54	853,54	0,00
GJ08T	249	2	—	—	—	—	3948,22	13325,40	1	13325,40	0	3948,22	4110,78	3,95
GJ09T	249	3	—	—	—	—	3544,24	11247,70	1	11247,70	0	3544,24	3656,03	3,06
GJ10T	249	4	—	—	—	—	3357,93	14582,30	1	14582,30	0	3357,93	3470,70	3,25
GJ11T	249	5	—	—	—	—	3267,84	15461,90	1	15461,90	0	3267,84	3360,48	2,76

¹Obtidos utilizando o método heurístico em (Subramanian, 2012). ²Bounds obtidos a partir da relaxação do recurso referente à coleta.

5.10 PRLCES

Em relação ao PRLCES, experimentos computacionais foram realizados com as instâncias dos conjuntos Bar11, Prod11 e Prod12 (v. Tabela 5.1). Para os dois primeiros conjuntos, os resultados obtidos foram comparados com os reportados por Karaoglan et al. (2011), os quais utilizam um B&C com SA, enquanto que, para o terceiro, comparações foram feitas com uma formulação de fluxo proposta por Karaoglan et al. (2012). Antes da análise dos resultados, é importante ressaltar que a coluna *DSS* (*Demand Separation Strategy*) se refere às estratégias utilizadas para a geração das demandas de coleta a partir dos valores das demandas de entrega ou vice-versa. As letras X e Y fazem referência às estratégias propostas por Salhi e Nagy (1999), enquanto que as letras W e Z estão relacionadas àquelas propostas por Angelelli e Mansini (2002).

Analisando-se a Tabela 5.20, é possível notar que todas as instâncias com até 36 clientes já haviam sido resolvidas anteriormente, também sendo resolvidas pelo método proposto. No entanto, estão sendo provadas, pela primeira vez, soluções ótimas para todas as instâncias com 50 clientes e para as duas com 55 clientes que usam as estratégias de separação de demandas X e Y. Além disso, todas as instâncias contendo de 55 a 88 clientes que usam as estratégias W e Z tiveram seus LBs melhorados, implicando em reduções no *gap*.

Quanto aos testes com o conjunto Prod11 (v. Tabela 5.21), todas as instâncias com 20 clientes foram resolvidas, porém sem nenhuma confirmação de otimalidade inédita. Por outro lado, para as instâncias com 50 clientes, foram encontradas soluções ótimas para 17 delas, sendo 15 pela primeira vez. Para as instâncias com 100 clientes, mantém-se a ausência de soluções ótimas provadas, contudo, o método proposto foi capaz de melhorar os LBs para 15 delas. Vale salientar que, em alguns casos da Tabela 5.21, o LB encontrado pelo método proposto é superior ao UB reportado por Karaoglan et al. (2011). Para tais casos, a implementação do modelo descrito em (Karaoglan et al., 2011) foi resolvida por meio do CPLEX, ratificando os novos resultados. Além disso, a solução ótima encontrada para a instância *20-5-2* com DSS Y é inferior àquela reportada por Karaoglan et al. (2011). O CPLEX foi utilizado novamente, confirmando-se a solução encontrada pelo *VRPSolver*.

Por fim, quanto aos resultados do conjunto Prod12 apresentados na Tabela 5.22, foram resolvidas em otimalidade 78 dentre as 96 instâncias consideradas. Vale ressaltar que o conjunto completo é composto por 360 instâncias, cuja quantidade de clientes varia de 10 a 100. O motivo pelo qual foram consideradas apenas parte do conjunto está relacionado ao fato de ter sido aplicado um método de resolução exato apenas para as instâncias de pequeno porte (com até 30 clientes), dado o tamanho elevado do *dataset*. Precisamente, Karaoglan et al. (2012) utilizaram duas formulações, baseadas em nós e fluxo respectivamente, com o CPLEX e um limite de 2 horas para cada teste. Da análise dos resultados reportados, percebeu-se a dominância da formulação de fluxo sobre a de nós, e, por isso, apenas a primeira foi usada na comparação com o método proposto. Dentre as 78 soluções encontradas, 27 delas foram provadas serem ótimas pela primeira vez.

Tabela 5.20: Resultados computacionais das instâncias do PRLCES — Bar11.

Instância	DSS	C	DD	DU	B&C com SA		Abordagem com o <i>VRPSolver</i>							
					UB	GAP(%)	RLB	RT(s)	NN	TT(s)	LC	LB	UB	GAP (%)
Srivastava86_8x2	W	8	2	2	873,58	0,00	835,88	0,03	5	0,08	0	873,58	873,58	0,00
Srivastava86_8x2	Z	8	2	2	806,06	0,00	766,99	0,03	5	0,07	0	806,06	806,06	0,00
Srivastava86_8x2	X	8	2	2	625,43	0,00	574,03	0,18	5	0,30	0	625,43	625,43	0,00
Srivastava86_8x2	Y	8	2	2	625,43	0,00	574,03	0,14	5	0,25	0	625,43	625,43	0,00
Perl83_12x2	W	12	2	1	243,98	0,00	227,19	0,33	5	0,69	1	243,98	243,98	0,00
Perl83_12x2	Z	12	2	1	243,98	0,00	222,9	0,38	5	1,19	1	243,98	243,98	0,00
Perl83_12x2	X	12	2	1	242,41	0,00	200,76	1,83	5	2,41	0	242,41	242,41	0,00
Perl83_12x2	Y	12	2	1	242,41	0,00	200,76	5,27	7	6,11	1	242,41	242,41	0,00
Gaskell67_21x5	W	21	5	2	528,42	0,00	494,76	9,53	499	561,24	378	528,42	528,42	0,00
Gaskell67_21x5	Z	21	5	2	513,30	0,00	484,79	5,65	15	15,62	0	513,30	513,30	0,00
Gaskell67_21x5	X	21	5	2	454,48	0,00	400,98	13,94	29	218,54	0	454,48	454,48	0,00
Gaskell67_21x5	Y	21	5	2	454,48	0,00	400,98	23,77	29	341,99	0	454,48	454,48	0,00
Gaskell67_22x5	W	22	5	1	653,80	0,00	613,12	29,52	9	160,61	0	653,80	653,80	0,00
Gaskell67_22x5	Z	22	5	1	653,80	0,00	612,67	25,51	9	127,82	0	653,80	653,80	0,00
Gaskell67_22x5	X	22	5	1	629,51	0,00	588,41	258,82	207	11704,90	0	629,51	629,51	0,00
Gaskell67_22x5	Y	22	5	1	629,51	0,00	584,24	386,50	353	20533,30	0	629,51	629,51	0,00
Min92_27x5	W	27	5	2	3142,02	0,00	2767,18	9,93	21	61,78	4	3142,02	3142,02	0,00
Min92_27x5	Z	27	5	2	3142,02	0,00	2767,18	11,71	17	54,02	2	3142,02	3142,02	0,00
Min92_27x5	X	27	5	1	2998,80	0,00	2560,28	220,51	43	6660,18	0	2998,80	2998,80	0,00
Min92_27x5	Y	27	5	1	2998,80	0,00	2560,28	66,50	27	1934,32	0	2998,80	2998,80	0,00
Gaskell67_29x5	W	29	5	2	592,10	0,00	508,69	28,36	13	91,86	0	592,10	592,10	0,00
Gaskell67_29x5	Z	29	5	2	592,10	0,00	502,38	23,04	15	197,36	0	592,10	592,10	0,00
Gaskell67_29x5	X	29	5	1	490,34	0,00	420,55	231,12	31	1669,27	0	490,34	490,34	0,00
Gaskell67_29x5	Y	29	5	1	490,34	0,00	420,55	131,55	23	822,43	0	490,34	490,34	0,00
Gaskell67_32x5_1	W	32	5	1	696,38	0,00	639,37	130,07	1049	15357,10	602	696,38	696,38	0,00
Gaskell67_32x5_1	Z	32	5	1	643,37	0,00	585,36	36,91	15	128,96	2	643,37	643,37	0,00
Gaskell67_32x5_1	X	32	5	1	563,47	0,00	524,71	98,27	13	157,13	0	563,47	563,47	0,00
Gaskell67_32x5_1	Y	32	5	1	563,47	0,00	514,67	258,68	13	967,90	0	563,47	563,47	0,00
Gaskell67_32x5_2	W	32	5	1	595,27	0,00	560,31	160,43	13	269,05	0	595,27	595,27	0,00
Gaskell67_32x5_2	Z	32	5	1	564,33	0,00	539,9	60,31	9	770,03	0	564,33	564,33	0,00
Gaskell67_32x5_2	X	32	5	1	507,03	0,00	476,06	149,02	9	557,64	0	507,03	507,03	0,00
Gaskell67_32x5_2	Y	32	5	1	507,03	0,00	464,41	218,44	29	6343,89	0	507,03	507,03	0,00
Gaskell67_36x5	W	36	5	1	540,37	0,00	462,35	30,27	47	593,54	30	540,37	540,37	0,00
Gaskell67_36x5	Z	36	5	1	540,37	0,00	462,25	41,12	43	794,18	15	540,37	540,37	0,00
Gaskell67_36x5	X	36	5	1	494,86	0,00	425,3	148,47	31	2018,86	2	494,86	494,86	0,00
Gaskell67_36x5	Y	36	5	1	494,86	0,00	425,21	561,46	39	4181,94	1	494,86	494,86	0,00
Ch69_50x5	W	50	5	2	708,37	4,52	591,47	157,92	981	39298,90	319	708,37	708,37	0,00
Ch69_50x5	Z	50	5	2	701,91	3,67	582,64	147,26	4479	203533,00	1359	700,72	700,72	0,00
Ch69_50x5	X	50	5	1	576,69	2,40	503,93	706,00	281	41234,90	16	576,69	576,69	0,00
Ch69_50x5	Y	50	5	1	578,97	2,69	492,87	687,23	1631	200222,00	15	576,69	576,69	0,00
Perl83_55x15	W	55	15	—	1330,85	20,93	1048,11	98,94	667	86400,00	0	1082,88	1330,85	18,63
Perl83_55x15	Z	55	15	—	1338,30	20,81	1041,26	97,48	533	86400,00	0	1069,39	1338,30	20,09
Perl83_55x15	X	55	15	2	985,56	2,94	804,66	324,64	393	106044,00	8	982,55	982,55	0,00
Perl83_55x15	Y	55	15	2	988,06	3,14	804,66	1394,58	581	207138,00	14	982,55	982,55	0,00
Ch69_75x10	W	75	10	—	1177,65	17,55	954,25	968,66	221	43200,00	202	1023,70	1177,65	13,07
Ch69_75x10	Z	75	10	—	1108,82	12,81	915,93	511,46	517	43200,00	191	987,89	1108,82	10,91
Ch69_75x10	X	75	10	—	888,16	10,33	726,01	1851,03	141	43200,00	0	766,41	888,16	13,71
Ch69_75x10	Y	75	10	—	884,00	8,90	701,14	1928,07	83	43200,00	0	754,50	884,00	14,65
Perl83_85x7	W	85	7	—	1901,09	23,53 ¹	1446,87	227,64	155	43200,00	0	1476,45	1901,09	22,34
Perl83_85x7	Z	85	7	—	1893,28	23,41	1440,4	157,14	155	43200,00	0	1486,39	1893,28	21,49
Perl83_85x7	X	85	7	—	1381,57	5,88	1050,45	1016,02	247	43200,00	66	1108,18	1381,57	19,79
Perl83_85x7	Y	85	7	—	1368,64	4,96	1044,94	1525,81	117	43200,00	0	1101,99	1368,64	19,48
Daskin95_88x8	W	88	8	—	533,37	21,60 ¹	453,11	12608,10	1	12608,20	0	453,11	533,37	15,05
Daskin95_88x8	Z	88	8	—	487,81	14,87 ¹	431,53	4359,62	37	43200,00	0	431,53	487,81	11,54
Daskin95_88x8	X	88	8	—	399,17	11,20	305,41	19457,80	1	19457,90	0	305,41	399,17	23,49
Daskin95_88x8	Y	88	8	—	401,98	0,00	315,49	11157,40	5	43200,00	0	315,49	401,98	21,52
Ch69_100x10	W	100	10	—	1079,41	12,74 ¹	867,34	2781,19	33	43200,00	0	907,34	1079,41	15,94
Ch69_100x10	Z	100	10	—	1038,26	9,26	861,08	1775,25	73	43200,00	0	900,33	1038,26	13,28
Ch69_100x10	X	100	10	—	880,18	7,45	741,08	2444,41	31	43200,00	0	780,68	880,18	11,30
Ch69_100x10	Y	100	10	—	880,89	5,59 ¹	742,28	30317,20	3	43200,00	0	742,28	880,89	15,74

¹ GAP obtido usando o algoritmo B&C proposto por Karaoglan et al. (2011) sem o SA.

Tabela 5.21: Resultados computacionais das instâncias do PRLCES — Prod11.

Instância	DSS	C	DD	DU	B&C com SA		Abordagem com o <i>VRPSolver</i>							
					UB	GAP (%)	RLB	RT (s)	NN	TT (s)	LC	LB	UB	GAP (%)
20-5-1	W	20	5	3	26457,70	0,00	21814,07	8,94	31	122,82	2	26468,86 ²	26468,86	0,00
20-5-1	Z	20	5	3	26461,30	0,00	21355,07	12,53	29	189,35	1	26461,30	26461,30	0,00
20-5-1	X	20	5	2	16816,50	0,00	9561,59	201,88	551	5760,99	596	16816,53	16816,53	0,00
20-5-1	Y	20	5	2	16816,00	0,00	9560,45	195,72	49	3368,08	33	16811,96	16811,96	0,00
20-5-1b	W	20	5	2	18718,80	0,00	10590,32	200,73	177	4510,41	152	18718,81	18718,81	0,00
20-5-1b	Z	20	5	2	18703,00	0,00	10661,59	38,42	29	2348,89	2	18702,96	18702,96	0,00
20-5-1b	X	20	5	1	9167,14	0,00	5245,50	461,98	9	1747,95	0	9167,14	9167,14	0,00
20-5-1b	Y	20	5	1	9167,14	0,00	5245,41	1002,54	43	28681,20	0	9175,06 ²	9175,06	0,00
20-5-2	W	20	5	3	27988,70	0,00	21825,36	3,90	27	176,99	2	27988,67	27988,67	0,00
20-5-2	Z	20	5	3	27980,60	0,00	22030,91	9,20	27	130,83	3	27980,64	27980,64	0,00
20-5-2	X	20	5	2	17814,70	0,00	12518,20	145,93	17	960,12	2	17814,67	17814,67	0,00
20-5-2	Y	20	5	2	17814,70	0,00	12515,05	88,30	15	3075,32	1	17793,43 ²	17793,43	0,00
20-5-2b	W	20	5	2	17125,50	0,00	11838,10	249,35	57	3128,57	21	17125,46	17125,46	0,00
20-5-2b	Z	20	5	2	17120,50	0,00	11577,84	41,44	55	736,20	33	17120,54	17120,54	0,00
20-5-2b	X	20	5	1	10257,30	0,00	7568,00	75,78	5	95,56	0	10257,34	10257,34	0,00
20-5-2b	Y	20	5	1	10257,30	0,00	7568,00	65,00	5	81,33	0	10257,34	10257,34	0,00
50-5-1	W	50	5	—	33786,20 ¹	3,18	28241,74	67,69	3201	86400	3692	32801,93	33786,20	2,91
50-5-1	Z	50	5	2	32804,50	0,30	27938,21	51,53	229	2634,62	168	32768,39	32768,39	0,00
50-5-1	X	50	5	1	16350,00	0,17	12903,43	387,79	105	8055,06	140	16346,65	16346,65	0,00
50-5-1	Y	50	5	—	16355,20	0,22	12879,93	441,35	1489	86400	2139	16315,34	16355,20	0,24
50-5-1b	W	50	5	2	26541,40	0,21	21092,69	679,27	963	36914,20	453	26528,93	26528,93	0,00
50-5-1b	Z	50	5	2	26530,80	0,18	21555,54	732,76	19	6891,70	0	26520,13	26520,13	0,00
50-5-1b	X	50	5	1	13132,90	0,04	9381,48	7119,84	23	14123,30	0	13132,93	13132,93	0,00
50-5-1b	Y	50	5	1	13132,90	0,04	9374,51	5555,99	427	86400	361	13126,01	13132,71	0,05
50-5-2	W	50	5	3	42860,40	2,54	34291,63	63,39	363	6095,07	264	41875,94	41875,94	0,00
50-5-2	Z	50	5	3	41825,80	0,16	33806,32	65,64	187	2322,40	207	41807,62	41807,62	0,00
50-5-2	X	50	5	2	26395,60	0,10	21599,02	370,60	47	12595,70	7	26390,94	26390,94	0,00
50-5-2	Y	50	5	2	26392,70	0,09	21589,09	121,22	1339	40032,10	1514	26382,58	26382,58	0,00
50-5-2b	W	50	5	—	35661,90	0,13	27386,16	1128,71	243	86400	9	30619,65	35661,90	14,14
50-5-2b	Z	50	5	—	35648,40	0,11	27560,19	672,66	483	86400	6	30185,12	35648,40	15,33
50-5-2b	X	50	5	2	22268,50	0,00	17467,65	821,68	61	9023,49	15	22268,53	22268,53	0,00
50-5-2b	Y	50	5	2	22268,50	0,00	17467,04	913,11	131	12344,00	86	22269,08 ²	22269,08	0,00
50-5-3	W	50	5	—	23533,10	4,65	21530,51	12195,30	7219	86400	11623	22585,14	22669,90	0,37
50-5-3	Z	50	5	2	23501,50	0,26	22604,64	60,00	9	166,59	2	23485,34	23485,34	0,00
50-5-3	X	50	5	1	11624,20	0,24	11417,32	448,42	1211	84265,20	913	11621,21	11621,21	0,00
50-5-3	Y	50	5	1	11626,60	0,25	11426,63	381,72	87	5989,47	14	11626,33	11626,33	0,00
50-5-3b	W	50	5	—	17182,00	0,21	16124,65	936,44	653	86400	300	17175,65	17182,00	0,04
50-5-3b	Z	50	5	2	17174,10	0,13	16280,96	845,74	751	12835,80	218	17173,31	17173,31	0,00
50-5-3b	X	50	5	1	8472,39	0,12	7958,57	6567,55	109	20976,40	31	8469,87	8469,87	0,00
50-5-3b	Y	50	5	1	8469,87	0,03	7961,77	3656,50	9	5550,71	1	8470,67 ²	8470,67	0,00
100-5-1	W	100	5	3	160277,00	1,36	118862,21	267,93	2771	86400	6191	119618,25	158314,91	24,44

Continua na próxima página

Tabela 5.21 – Continuação da página anterior

Instância	DSS	C	DD	DU	B&C com SA		Abordagem com o <i>VRPSolver</i>							GAP (%)
					UB	GAP (%)	RLB	RT (s)	NN	TT (s)	LC	LB	UB	
100-5-1	Z	100	5	—	159200,00	0,73	117838,30	254,88	515	86400	152	118540,98	159200,00	25,54
100-5-1	X	100	5	—	102388,00	0,18	65578,02	1200,63	169	86400	0	102290,88	102388,00	0,09
100-5-1	Y	100	5	—	102381,00	0,17	65450,26	1165,74	169	86400	281	68493,32	102381,00	33,10
100-5-1b	W	100	5	—	145697,00	0,78	104889,90	5734,62	61	86400	0	105491,20	145697,00	27,60
100-5-1b	Z	100	5	—	145650,00	0,76	103860,17	1465,34	95	86400	18	104576,39	145650,00	28,20
100-5-1b	X	100	5	—	94884,00	0,12	57161,33	29644,60	27	86400	0	62321,07	94884,00	34,32
100-5-1b	Y	100	5	—	94878,80 ¹	0,11	57107,84	12504,00	87	86400	0	94785,26	94878,80	0,10
100-5-2	W	100	5	—	168371,00	28,20	117356,31	234,93	639	86400	173	120754,54	168371,00	28,28
100-5-2	Z	100	5	—	172418,00 ¹	29,90	117165,48	293,70	471	86400	0	120539,99	172418,00	30,09
100-5-2	X	100	5	—	105655,00	0,11	74814,46	931,55	189	86400	0	105608,34	105655,00	0,04
100-5-2	Y	100	5	2	105655,00	0,11	74789,05	837,06	243	86400	21	105581,07	105585,85	0,004
100-5-2b	W	100	5	—	153827,00	29,57	104238,34	9208,43	67	86400	0	107970,97	153827,00	29,81
100-5-2b	Z	100	5	—	153776,00	29,55	104209,69	1614,18	81	86400	0	107949,31	153776,00	29,80
100-5-2b	X	100	5	—	97213,80	0,08	65403,11	10335,80	69	86400	0	97180,13	97213,80	0,03
100-5-2b	Y	100	5	—	97206,00	0,07	65393,03	5725,44	85	86400	0	97168,17	97206,00	0,04
100-5-3	W	100	5	—	114517,00	3,65	109998,40	234,04	371	43200	38	110605,23	114517,00	3,42
100-5-3	Z	100	5	—	112467,00	1,89	110060,56	227,00	265	43200	30	110449,26	112467,00	1,79
100-5-3	X	100	5	—	56552,10	0,17	53103,92	4003,58	195	86400	0	56491,23	56552,10	0,11
100-5-3	Y	100	5	—	56581,90	0,22	53128,34	5289,39	161	86400	0	56490,94	56581,90	0,16
100-5-3b	W	100	5	—	100006,00	2,13	97005,00	6702,12	37	43200	0	97937,57	100006,00	2,07
100-5-3b	Z	100	5	—	98972,50	1,11	97221,76	1424,25	49	43200	0	97923,97	98972,50	1,06
100-5-3b ³	X	100	5	—	50224,60	0,11	47525,02	2347,38	1	2347,38	0	47525,02	50224,60	5,38
100-5-3b ³	Y	100	5	—	50220,80	0,09	46754,43	762,76	1	762,76	0	46754,43	50220,80	6,90
100-10-1	W	100	10	—	224445,00	1,41	185638,73	700,70	271	86400	9	190219,59	224445,00	15,25
100-10-1	Z	100	10	—	222434,00	0,53	184297,10	508,89	383	86400	0	188880,39	222434,00	15,08
100-10-1	X	100	10	—	109785,00	1,07	85855,01	3153,64	53	86400	0	88906,52	109785,00	19,02
100-10-1	Y	100	10	—	109787,00	1,07	85749,30	3959,69	57	86400	0	88802,93	109787,00	19,11
100-10-1b	W	100	10	—	209084,00	0,10	172129,67	4672,84	79	86400	0	174066,04	209084,00	16,75
100-10-1b	Z	100	10	—	208958,00	0,04	171819,20	2808,28	83	86400	0	172790,13	208958,00	17,31
100-10-1b	X	100	10	—	102430,00	0,08	79284,44	82156,20	1	94084,10	0	79284,44	102430,00	22,60
100-10-1b	Y	100	10	—	102426,00	0,07	79249,44	87087,40	1	87087,40	0	79249,44	102426,00	22,63
100-10-2	W	100	10	—	218395,00	22,30	163912,68	955,86	273	86400	510	165847,12	218395,00	24,06
100-10-2	Z	100	10	—	218243,00	22,42	166387,08	465,51	237	86400	0	168297,54	218243,00	22,89
100-10-2	X	100	10	—	155190,00	32,08	101479,39	2996,02	47	86400	0	102317,07	155190,00	34,07
100-10-2	Y	100	10	—	107521,00 ¹	1,96	101465,65	3124,91	37	86400	0	105238,05	107521,00	2,12
100-10-2b	W	100	10	—	159507,00 ¹	1,74	150836,29	18544,70	43	86400	0	153087,36	159507,00	4,02
100-10-2b	Z	100	10	—	202371,85	22,78	153554,13	2992,50	53	86400	0	155816,24	202371,85	23,00
100-10-2b	X	100	10	—	99140,10	1,08	93919,58	86935,80	1	86400	0	93919,58	99140,10	5,27
100-10-2b	Y	100	10	—	99138,40	1,08	93918,88	83312,30	1	99267,30	0	93918,88	99138,40	5,26
100-10-3	W	100	10	—	210355,00	20,87	166646,80	718,96	233	86400	29	169705,86	210355,00	19,32
100-10-3	Z	100	10	—	166404,22	1,89	161988,61	510,67	275	86400	0	163369,73	166404,22	1,82
100-10-3	X	100	10	—	100702,00	1,11	77717,76	3105,33	51	86400	0	81089,08	100702,00	19,48

Continua na próxima página

Tabela 5.21 – Continuação da página anterior

Instância	DSS	C	DD	DU	B&C com SA		Abordagem com o <i>VRPSolver</i>							
					UB	GAP (%)	RLB	RT (s)	NN	TT (s)	LC	LB	UB	GAP (%)
100-10-3	Y	100	10	—	99913,60	0,33	77712,56	4140,66	45	86400	0	81087,90	99913,60	18,84
100-10-3b	W	100	10	—	196017,00	21,58	152851,81	5662,61	49	86400	0	156102,27	196017,00	20,36
100-10-3b	Z	100	10	—	152088,46 ¹	0,76	149182,40	2000,37	65	86400	0	150953,12	152088,46	0,75
100-10-3b ³	X	100	10	—	93450,50	0,07	71853,84	2372,63	1	2372,64	0	71853,84	93450,50	23,11
100-10-3b	Y	100	10	—	93475,00	0,09	71255,13	86976,50	1	86400	0	71255,13	93475,00	23,77

¹ GAP obtido usando o algoritmo B&C proposto por Karaoglan et al. (2011) sem o SA

² Divergência entre as soluções ótimas encontradas e provadas por Karaoglan et al. (2011).

³ Bounds obtidos a partir da relaxação do recurso referente à coleta.

Tabela 5.22: Resultados computacionais das instâncias do PRLCES — Prod12.

Instância	DSS	C	DD	DU	Formulação de fluxo			Abordagem com o <i>VRPSolver</i>							
					LB	UB	GAP (%)	RLB	RT (s)	NN	TT (s)	LC	LB	UB	GAP (%)
10-20-5-1	W	10	3	2	20123,20	20123,20	0,00	12292,65	0,51	11	6,08	1	20123,24	20123,24	0,00
10-20-5-1b	W	10	3	1	9141,61	9141,61	0,00	5760,70	3,76	7	5,20	0	9141,61	9141,61	0,00
10-20-5-2	W	10	3	2	20846,60	20846,60	0,00	14012,47	1,04	9	1,99	1	20846,60	20846,60	0,00
10-20-5-2b	W	10	3	1	11157,40	11157,40	0,00	6668,25	8,67	15	13,00	9	11157,38	11157,38	0,00
10-20-5-1	X	10	3	1	8263,62	8263,62	0,00	5870,72	8,53	3	8,76	0	8263,62	8263,62	0,00
10-20-5-1b	X	10	3	1	8131,74	8131,74	0,00	2754,95	109,07	7	271,42	0	8131,74	8131,74	0,00
10-20-5-2	X	10	3	1	10257,20	10257,20	0,00	7405,07	7,25	3	7,92	0	10257,22	10257,22	0,00
10-20-5-2b	X	10	3	1	6953,43	6953,43	0,00	2786,37	41,95	5	56,47	0	6953,43	6953,43	0,00
10-20-5-1	Y	10	3	1	8239,42	8239,42	0,00	5585,49	8,33	3	8,54	0	8239,42	8239,42	0,00
10-20-5-1b	Y	10	3	1	8131,74	8131,74	0,00	2664,84	96,29	7	259,55	0	8131,74	8131,74	0,00
10-20-5-2	Y	10	3	1	10261,40	10261,40	0,00	6822,99	15,39	49	45,44	30	10261,44	10261,44	0,00
10-20-5-2b	Y	10	3	1	6953,43	6953,43	0,00	3454,06	102,14	5	134,82	0	6953,43	6953,43	0,00
10-20-5-1	Z	10	3	2	20111,90	20111,90	0,00	11446,51	1,85	11	5,76	0	20111,93	20111,93	0,00
10-20-5-1b	Z	10	3	1	9141,61	9141,61	0,00	5807,33	4,41	7	5,48	0	9141,61	9141,61	0,00
10-20-5-2	Z	10	3	2	20839,80	20839,80	0,00	14143,22	1,03	7	2,35	1	20839,79	20839,79	0,00
10-20-5-2b	Z	10	3	1	6953,43	6953,43	0,00	5472,32	3,41	5	9,40	0	6953,43	6953,43	0,00
15-20-5-1	W	15	4	2	17961,90	17966,60	0,03	15980,62	3,02	5	9,36	0	17966,59	17966,59	0,00
15-20-5-1b	W	15	4	1	9199,18	9199,18	0,00	7450,43	5,91	5	12,18	0	9199,18	9199,18	0,00
15-20-5-2	W	15	4	2	20379,30	20379,30	0,00	17472,71	2,08	11	14,34	3	20379,34	20379,34	0,00
15-20-5-2b	W	15	4	1	10217,60	10217,60	0,00	8335,04	36,20	5	57,87	0	10217,57	10217,57	0,00
15-20-5-1	X	15	4	1	8272,08	8272,08	0,00	7835,92	4,27	9	8,38	3	8272,08	8272,08	0,00
15-20-5-1b	X	15	4	1	8140,97	8140,97	0,00	3931,98	188,32	11	1306,83	0	8140,97	8140,97	0,00
15-20-5-2	X	15	4	1	10187,00	10187,00	0,00	9157,93	9,13	5	21,54	1	10187,02	10187,02	0,00
15-20-5-2b	X	15	4	1	6984,07	6984,07	0,00	4721,99	120,03	7	169,00	0	6984,07	6984,07	0,00
15-20-5-1	Y	15	4	1	8268,69	8268,69	0,00	7355,04	54,74	3	55,87	0	8268,69	8268,69	0,00
15-20-5-1b	Y	15	4	1	8140,97	8140,97	0,00	4052,0	228,48	9	711,92	0	8140,97	8140,97	0,00

Continua na próxima página

Tabela 5.22 – Continuação da página anterior

Instância	DSS	C	DD	DU	Formulação de fluxo			Abordagem com o <i>VRPSolver</i>							
					LB	UB	GAP (%)	RLB	RT (s)	NN	TT (s)	LC	LB	UB	GAP (%)
15-20-5-2	Y	15	4	1	10159,10	10159,10	0,00	9358,72	3,85	5	6,34	0	10159,12	10159,12	0,00
15-20-5-2b	Y	15	4	1	6984,07	6984,07	0,00	4594,6	131,67	7	249,39	0	6984,07	6984,07	0,00
15-20-5-1	Z	15	4	2	17948,20	17948,20	0,00	15958,10	1,79	5	5,26	0	17948,18	17948,18	0,00
15-20-5-1b	Z	15	4	1	9194,18	9194,18	0,00	7515,38	6,23	5	9,68	0	9194,18	9194,18	0,00
15-20-5-2	Z	15	4	2	20371,50	20371,50	0,00	17529,38	2,31	29	17,26	0	20371,51	20371,51	0,00
15-20-5-2b	Z	15	4	1	10217,60	10217,60	0,00	8281,25	10,80	5	16,52	0	10217,57	10217,57	0,00
20-20-5-1	W	20	5	3	26456,90	26468,90	0,05	21814,07	7,66	31	106,71	2	26468,86	26468,86	0,00
20-20-5-1b	W	20	5	2	18718,80	18718,80	0,00	10590,32	191,19	187	4802,51	132	18718,81	18718,81	0,00
20-20-5-2	W	20	5	3	27975,30	28003,50	0,10	21825,36	3,37	27	156,73	2	27988,67	27988,67	0,00
20-20-5-2b	W	20	5	2	17125,50	17125,50	0,00	11838,13	228,30	49	2529,57	15	17125,46	17125,46	0,00
20-20-5-1	X	20	5	2	16816,50	16816,50	0,00	9562,00	213,76	613	5787,54	644	16816,53	16816,53	0,00
20-20-5-1b	X	20	5	1	9167,14	9167,14	0,00	5245,5	411,33	9	1537,29	0	9167,14	9167,14	0,00
20-20-5-2	X	20	5	2	17807,10	17814,70	0,04	12518,15	140,96	17	1000,56	2	17814,67	17814,67	0,00
20-20-5-2b	X	20	5	1	10257,30	10257,30	0,00	7568,0	68,96	5	85,85	0	10257,34	10257,34	0,00
20-20-5-1	Y	20	5	2	16812,00	16812,00	0,00	9560,7	217,72	45	3279,33	18	16811,96	16811,96	0,00
20-20-5-1b	Y	20	5	—	9175,06	9175,06	0,00	5245,41	988,63	9	7200,00	0	8299,90	9175,06	9,54
20-20-5-2	Y	20	5	2	17793,40	17793,40	0,00	12514,87	93,89	15	3561,55	0	17793,43	17793,43	0,00
20-20-5-2b	Y	20	5	1	10257,30	10257,30	0,00	7568,0	76,86	5	90,25	0	10257,34	10257,34	0,00
20-20-5-1	Z	20	5	3	26437,50	26505,60	0,26	21355,07	11,60	29	178,72	1	26461,30	26461,30	0,00
20-20-5-1b	Z	20	5	2	18703,00	18703,00	0,00	10661,59	34,96	31	3380,35	3	18702,96	18702,96	0,00
20-20-5-2	Z	20	5	3	27980,60	27980,60	0,00	22030,9	7,83	27	114,14	2	27980,64	27980,64	0,00
20-20-5-2b	Z	20	5	2	17120,50	17120,50	0,00	11577,84	36,42	65	715,27	38	17120,54	17120,54	0,00
25-50-5-1	W	25	5	1	16258,00	16363,00	0,64	12653,99	21,68	53	196,19	50	16314,10	16314,10	0,00
25-50-5-1b	W	25	5	1	13064,90	13083,20	0,14	9315,22	181,61	13	1060,26	2	13083,17	13083,17	0,00
25-50-5-2	W	25	5	2	24284,00	24322,70	0,16	16617,1	11,82	29	330,64	0	24311,11	24311,11	0,00
25-50-5-2b	W	25	5	2	21218,00	21228,60	0,05	13493,92	263,95	35	7200,00	3	13814,04	21227,61	34,92
25-50-5-3	W	25	5	1	11559,70	12626,90	8,45	11186,12	10,03	21	53,18	5	11604,48	11604,48	0,00
25-50-5-3b	W	25	5	1	8395,03	8398,07	0,04	7982,95	46,57	3	49,32	0	8398,07	8398,07	0,00
25-50-5-1	X	25	5	1	9391,78	9425,03	0,35	7509,08	36,23	741	7200,00	895	9409,01	9415,56	0,07
25-50-5-1b	X	25	5	1	7296,59	7296,59	0,00	5399,12	445,47	17	1119,77	6	7296,59	7296,59	0,00
25-50-5-2	X	25	5	1	13217,70	13226,70	0,07	9856,12	276,81	17	2095,82	7	13226,73	13226,73	0,00
25-50-5-2b	X	25	5	—	11194,40	11194,40	0,00	7823,58	420,01	5	2611,65	4	9305,92	11194,40	16,87
25-50-5-3	X	25	5	—	9438,57	9438,57	0,00	6157,63	407,08	13	7200,00	0	8587,32	9438,57	9,02
25-50-5-3b	X	25	5	1	7345,01	7345,01	0,00	4371,56	1671,46	5	3059,88	0	7345,01	7345,01	0,00
25-50-5-1	Y	25	5	1	8367,59	8367,59	0,00	6297,01	23,16	29	97,36	32	8367,59	8367,59	0,00
25-50-5-1b	Y	25	5	1	7297,41	7297,41	0,00	5223,57	275,63	237	7200,00	187	7296,06	7297,50	0,02
25-50-5-2	Y	25	5	1	13225,70	13225,70	0,00	9798,79	38,01	871	3399,04	1080	13225,65	13225,65	0,00
25-50-5-2b	Y	25	5	1	11195,60	11195,60	0,00	7772,56	435,88	139	3763,47	108	11195,60	11195,60	0,00
25-50-5-3	Y	25	5	—	9434,55	9434,55	0,00	6064,63	174,59	13	7200,00	0	8530,95	9434,55	9,58
25-50-5-3b	Y	25	5	—	7345,53	7345,53	0,00	4313,74	3754,64	3	7200,00	1	4558,52	7345,53	37,94
25-50-5-1	Z	25	5	1	16250,30	16348,20	0,60	12723,88	12,99	9	71,96	0	16289,16	16289,16	0,00
25-50-5-1b	Z	25	5	1	13067,20	13071,70	0,03	9472,17	150,54	9	1779,19	0	13071,67	13071,67	0,00
Continua na próxima página															

Continua na próxima página

Tabela 5.22 – Continuação da página anterior

Instância	DSS	C	DD	DU	Formulação de fluxo			Abordagem com o <i>VRPSolver</i>							
					LB	UB	GAP (%)	RLB	RT (s)	NN	TT (s)	LC	LB	UB	GAP (%)
25-50-5-2	Z	25	5	2	24264,50	24325,20	0,25	16631,83	9,82	29	315,72	0	24300,69	24300,69	0,00
25-50-5-2b	Z	25	5	—	21216,90	21228,60	0,06	13518,54	168,63	35	7200,00	5	13843,64	21227,6	34,78
25-50-5-3	Z	25	5	1	11557,50	11599,40	0,36	11199,51	12,45	3	13,99	0	11590,21	11590,21	0,00
25-50-5-3b	Z	25	5	1	8398,07	8398,07	0,00	8007,37	68,09	3	70,49	0	8398,07	8398,07	0,00
30-50-5-1	W	30	5	—	22349,40	23434,60	4,63	15740,17	22,39	389	7200,00	0	18400,56	23411,70	21,40
30-50-5-1b	W	30	5	—	19112,90	19145,90	0,17	12349,33	209,29	295	7200,00	287	19139,91	19145,90	0,03
30-50-5-2	W	30	5	2	26342,40	27399,50	3,86	21184,7	12,79	25	205,11	1	26401,60	26401,60	0,00
30-50-5-2b	W	30	5	—	22232,80	22324,90	0,41	17042,2	451,62	33	7200,00	0	18024,36	22293,00	19,15
30-50-5-3	W	30	5	2	18187,90	19276,50	5,65	13193,51	18,16	49	512,19	20	18252,40	18252,40	0,00
30-50-5-3b	W	30	5	2	15005,60	15036,50	0,21	9951,76	339,55	41	7200,00	0	10605,37	15029,51	29,44
30-50-5-1	X	30	5	1	9417,81	9476,75	0,62	8035,56	200,05	243	7200,00	214	9458,19	9464,20	0,06
30-50-5-1b	X	30	5	1	7307,64	7315,76	0,11	5890,12	812,97	11	2752,75	5	7314,95	7314,95	0,00
30-50-5-2	X	30	5	1	13304,50	13388,90	0,63	11410,01	68,02	7	634,37	6	13325,08	13325,08	0,00
30-50-5-2b	X	30	5	1	11213,50	11221,70	0,07	9329,21	496,38	41	2262,68	39	11219,58	11219,58	0,00
30-50-5-3	X	30	5	—	9440,52	9459,62	0,20	7355,14	758,37	11	7200,00	0	9248,82	9457,57	2,21
30-50-5-3b	X	30	5	1	7350,27	7350,27	0,00	5218,93	2999,39	5	5587,34	0	7350,27	7350,27	0,00
30-50-5-1	Y	30	5	1	9413,24	9431,28	0,19	7972,92	31,61	59	426,32	38	9430,16	9430,16	0,00
30-50-5-1b	Y	30	5	1	7314,72	7314,72	0,00	5859,11	657,00	41	5270,90	12	7314,72	7314,72	0,00
30-50-5-2	Y	30	5	1	14316,00	14321,00	0,03	12506,09	26,25	745	2147,72	947	14320,95	14320,95	0,00
30-50-5-2b	Y	30	5	—	11221,10	11221,10	0,00	9461,37	345,99	161	7200,00	134	11218,59	11221,10	0,02
30-50-5-3	Y	30	5	—	9449,50	9453,20	0,04	7448,39	207,78	23	7200,00	0	9296,85	9453,20	1,65
30-50-5-3b	Y	30	5	1	7352,29	7352,29	0,00	5294,48	3213,68	59	6157,65	29	7352,29	7352,29	0,00
30-50-5-1	Z	30	5	2	22337,20	23400,30	4,54	15557,52	18,04	167	1037,82	123	22399,63	22399,63	0,00
30-50-5-1b	Z	30	5	2	19114,10	19141,40	0,14	12275,59	142,42	25	6132,11	0	19135,89	19135,89	0,00
30-50-5-2	Z	30	5	2	25333,30	26437,80	4,18	20075,34	10,73	29	150,88	22	25439,56	25439,56	0,00
30-50-5-2b	Z	30	5	2	22233,00	22255,10	0,10	16881,21	106,50	31	6063,34	6	22245,65	22245,65	0,00
30-50-5-3	Z	30	5	2	18187,10	19260,50	5,57	13262,61	20,52	19	238,21	1	18230,84	18230,84	0,00
30-50-5-3b	Z	30	5	2	15003,30	15031,70	0,19	10044,93	299,23	23	7200,00	3	10814,91	15031,05	28,05

Capítulo 6

Considerações Finais

O presente trabalho aborda o desenvolvimento de uma abordagem exata baseada em BCP e utilizando o *VRPSolver* para resolver 10 variantes do PRVCES. Precisamente, foram considerados o problema clássico (PRVCES) e variantes com coleta e entrega mistas (PRVCEM), frota de veículos heterogênea (PRVCESFH), duração de rota (PRVCESTL e PRVCEMTL), múltiplos depósitos (MDVRPMPD), localização de facilidades (PRLCES), custos assimétricos (PRVCESA) e janelas de tempo (PRVCESJT e PRVCESJTf).

Da análise dos resultados obtidos dos extensivos experimentos computacionais realizados, observou-se que o método proposto foi capaz de melhorar os limites duais para 199 dentre 557 instâncias testadas, dos quais 152 corresponderam a comprovações inéditas de otimalidade. Além disso, outras 165 soluções ótimas foram encontradas, contudo as mesmas já haviam sido provadas por métodos anteriores. Por fim, foram fornecidos limites pela primeira vez para 111 instâncias, enquanto resultados piores do que os da literatura ocorreram em apenas 78 casos (14% do total de instâncias).

Devido aos importantes resultados conseguidos, são sugeridas as seguintes linhas de pesquisa como trabalhos futuros: (i) o desenvolvimento de um algoritmo de *pricing* que considere a interdependência entre os recursos de coleta e entrega, sem a necessidade de utilização dos *lazy cuts* para esse fim, e (ii) a generalização do método proposto de modo a incluir variantes do PRVCES com incertezas.

REFERÊNCIAS BIBLIOGRÁFICAS

- AGARWAL, Y. K.; VENKATESHAN, P. A new model for the asymmetric vehicle routing problem with simultaneous pickup and deliveries. *Operations Research Letters*, v. 48, p. 48–54, 2020.
- AGARWAL, Y. K.; VENKATESHAN, P. New valid inequalities for the symmetric vehicle routing problem with simultaneous pickup and deliveries. *Networks*, v. 79, n. 4, p. 537–556, 2021.
- AGIUS, M.; ABSI, N.; FEILLET, D.; GARAIX, T. A branch-and-price algorithm for a routing problem with inbound and outbound requests. *Computers & Operations Research*, v. 146, p. 105896, 2022.
- ANGELELLI, E.; MANSINI, R. The vehicle routing problem with time windows and simultaneous pick-up and delivery. KLOSE, A.; SPERANZA, M. G.; VAN WASSENHOVE, L. N. (Eds.), *Quantitative Approaches to Distribution Logistics and Supply Chain Management*, p. 249–267, Berlin, Heidelberg. Springer Berlin Heidelberg, 2002.
- AUGERAT, P.; BELENGUER, J. M.; BENAVENT, E.; CORBERÁN, A.; NADDEF, D.; RINALDI, G. Computational results with a branch and cut code for the capacitated vehicle routing problem. Tech. Rep. 949-M, Université Joseph Fourier, Grenoble, France., 1995.
- AVCI, M.; TOPALOGLU, S. An adaptive local search algorithm for vehicle routing problem with simultaneous and mixed pickups and deliveries. *Computers and Industrial Engineering*, v. 83, p. 15–29, 2015.

- AVCI, M.; TOPALOGLU, S. A hybrid metaheuristic algorithm for heterogeneous vehicle routing problem with simultaneous pickup and delivery. *Expert Systems with Applications*, v. 53, p. 160–171, 2016.
- BALDACCI, R.; CHRISTOFIDES, N.; MINGOZZI, A. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming*, v. 115, p. 351–385, 2008.
- BALDACCI, R.; MINGOZZI, A.; ROBERTI, R. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, v. 59, n. 5, p. 1269–1283, 2011.
- BARRETO, S.; FERREIRA, C.; PAIXÃO, J.; SANTOS, B. S. Using clustering analysis in a capacitated location-routing problem. *European Journal of Operational Research*, v. 179, p. 968–977, 2007.
- BATTARRA, M.; CORDEAU, J.-F.; IORI, M. *Chapter 6: Pickup-and-Delivery Problems for Goods Transportation*, Chapter 6, p. 161–191. SIAM, 2014.
- BELLOSO, J.; JUAN, A. A.; MARTINEZ, E.; FAULIN, J. A biased-randomized metaheuristic for the vehicle routing problem with clustered and mixed backhauls. *Networks*, v. 69, n. 3, p. 241–255, 2017.
- BELMECHERI, F.; PRINS, C.; YALAOUI, F.; AMODEO, L. Particle swarm optimization algorithm for a vehicle routing problem with heterogeneous fleet, mixed backhauls, and time windows. *Journal of Intelligent Manufacturing*, v. 24, p. 775–789, 2013.
- BERBEGLIA, G.; CORDEAU, J.-F.; GRIBKOVSKAIA, I.; LAPORTE, G. Static pickup and delivery problems: a classification scheme and survey. *TOP*, v. 15, p. 1–31, 2007.
- BERBEGLIA, G.; CORDEAU, J. F.; LAPORTE, G. Dynamic pickup and delivery problems. *European Journal of Operational Research*, v. 202, p. 8–15, 2010.

- BERGHIDA, M.; BOUKRA, A. Quantum inspired algorithm for a vrp with heterogeneous fleet mixed backhauls and time windows. *International Journal of Applied Metaheuristic Computing (IJAMC)*, v. 7, n. 4, p. 18–38, 2016.
- BOUANANE, K.; EL AMRANI, M.; BENADADA, Y. The vehicle routing problem with simultaneous delivery and pickup: A taxonomic survey. *International Journal of Logistics Systems and Management*, v. 1, p. 1, 2020.
- BRAEKERS, K.; RAMAEKERS, K.; VAN NIEUWENHUYSE, I. The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, v. 99, p. 300–313, 2016.
- BULHÕES, T.; PESSOA, A.; PROTTI, F.; UCHOA, E. On the complete set packing and set partitioning polytopes: Properties and rank 1 facets. *Operations Research Letters*, v. 46, n. 4, p. 389–392, 2018.
- CASCO, D.; GOLDEN, B. L.; WASIL, E. *Vehicle Routing with Backhauls: Models, algorithms, and case studies*, volume 16, p. 127–147. Studies in Management Science and Systems, North-Holland, Amsterdam, 1988.
- CLARKE, G.; WRIGHT, J. W. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, v. 12, n. 4, p. 568–581, 1964.
- CRISPIM, J.; BRANDÃO, J. Metaheuristics applied to mixed and simultaneous extensions of vehicle routing problems with backhauls. *Journal of the Operational Research Society*, v. 56, n. 11, p. 1296–1302, 2005.
- CRUZ, R. C.; SILVA, T. C.; SOUZA, M. J.; COELHO, V. N.; MINE, M. T.; MARTINS, A. X. Genvns-ts-cl-pr: A heuristic approach for solving the vehicle routing problem with simultaneous pickup and delivery. *Electronic Notes in Discrete Mathematics*, v. 39, p. 217–224, 2012.
- DANTZIG, G. B.; RAMSER, J. H. The truck dispatching problem. *Management Science*, v. 6, n. 1, p. 80–91, 1959.

- DELL'AMICO, M.; RIGHINI, G.; SALANI, M. A branch-and-price approach to the vehicle routing problem with simultaneous distribution and collection. *Transportation Science*, v. 40, n. 2, p. 235–247, 2006.
- DETHLOFF, J. Relation between vehicle routing problems: An insertion heuristic for the vehicle routing problem with simultaneous delivery and pick-up applied to the vehicle routing problem with backhauls. *Journal of the Operational Research Society*, v. 53, p. 115–118, 2002.
- DETHLOFF, J. Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up. *OR-Spektrum*, v. 23, n. 1, p. 79–96, 2001.
- FAN, J. The vehicle routing problem with simultaneous pickup and delivery based on customer satisfaction. *Procedia Engineering*, v. 15, p. 5284–5289. CEIS 2011, 2011.
- FUKASAWA, R.; LONGO, H.; LYSGAARD, J.; ARAGÃO, M. P. D.; REIS, M.; UCHOA, E.; WERNECK, R. F. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming*, v. 106, p. 491–511, 2006.
- GAJPAL, Y.; ABAD, P. Saving-based algorithms for vehicle routing problem with simultaneous pickup and delivery. *Journal of the Operational Research Society*, v. 61, n. 10, p. 1498–1509, 2010.
- GAJPAL, Y.; ABAD, P. An ant colony system (acs) for vehicle routing problem with simultaneous delivery and pickup. *Computers and Operations Research*, v. 36, p. 3215–3223, 2009.
- GEHRING, H.; HOMBERGER, J. A parallel hybrid evolutionary metaheuristic for the vehicle routing problem with time windows. *Proceedings of EUROGEN99*, volume 2, p. 57–64. Citeseer, 1999.
- GIANESSI, P.; ALFANDARI, L.; LÉTOCART, L.; CALVO, R. W. The multicommodity-ring location routing problem. *Transportation Science*, v. 50, p. 541–558, 2016.

- GOKSAL, F. P.; KARAOGLAN, I.; ALTIPARMAK, F. A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. *Computers and Industrial Engineering*, v. 65, p. 39–53, 2013.
- GOLDEN, B.; BAKER, E.; ALFARO, J.; SCHAFFER, J. The vehicle routing problem with backhauling: two approaches. *Proceedings of the twenty-first annual meeting of the SE TIMS, Myrtle Beach, SC, USA*, 1985.
- HOF, J.; SCHNEIDER, M. An adaptive large neighborhood search with path relinking for a class of vehicle-routing problems with simultaneous pickup and delivery. *Networks*, v. 74, p. 207–250, 2019.
- HORNSTRA, R. P.; SILVA, A.; ROODBERGEN, K. J.; COELHO, L. C. The vehicle routing problem with simultaneous pickup and delivery and handling costs. *Computers and Operations Research*, v. 115, 2020.
- HUANG, S. H. Solving the multi-compartment capacitated location routing problem with pickup-delivery routes and stochastic demands. *Computers and Industrial Engineering*, v. 87, p. 104–113, 2015.
- HUANG, Z.; HUANG, W.; GUO, F. Integrated sustainable planning of micro-hub network with mixed routing strategy. *Computers and Industrial Engineering*, v. 149, 2020.
- JEPSEN, M.; PETERSEN, B.; SPOORENDONK, S.; PISINGER, D. Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research*, v. 56, n. 2, p. 497–511, 2008.
- JUN, Y.; KIM, B. I. New best solutions to vrpspd benchmark problems by a perturbation based algorithm. *Expert Systems with Applications*, v. 39, p. 5641–5648, 2012.
- KALAYCI, C. B.; KAYA, C. An ant colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery. *Expert Systems with Applications*, v. 66, p. 163–175, 2016.

- KARAOGLAN, I.; ALTIPARMAK, F. A memetic algorithm for the capacitated location-routing problem with mixed backhauls. *Computers and Operations Research*, v. 55, p. 200–216, 2015.
- KARAOGLAN, I.; ALTIPARMAK, F.; KARA, I.; DENGİZ, B. A branch and cut algorithm for the location-routing problem with simultaneous pickup and delivery. *European Journal of Operational Research*, v. 211, p. 318–332, 2011.
- KARAOGLAN, I.; ALTIPARMAK, F.; KARA, I.; DENGİZ, B. The location-routing problem with simultaneous pickup and delivery: Formulations and a heuristic approach. *Omega*, v. 40, p. 465–477, 2012.
- KARIMI, H. The capacitated hub covering location-routing problem for simultaneous pickup and delivery systems. *Computers and Industrial Engineering*, v. 116, p. 47–58, 2018.
- KASSEM, S.; CHEN, M. Solving reverse logistics vehicle routing problems with time windows. *International Journal of Advanced Manufacturing Technology*, v. 68, p. 57–68, 2013.
- KOÇ, Ç.; LAPORTE, G.; TÜKENMEZ, İ. A review of vehicle routing with simultaneous pickup and delivery. *Computers and Operations Research*, v. 122, 2020.
- KOCH, H.; SCHLÖGELL, M.; BORTFELDT, A. A hybrid algorithm for the vehicle routing problem with three-dimensional loading constraints and mixed backhauls. *Journal of Scheduling*, v. 23, p. 71–93, 2020.
- KOULAEIAN, M.; SEIDGAR, H.; KIANI, M.; FAZLOLLAHTABAR, H. A multi depot simultaneous pickup and delivery problem with balanced allocation of routes to drivers. *The International Journal of Industrial Engineering: Theory, Applications and Practice*, v. 22, p. 294–313, 2015.
- LAPORTE, G.; NOBERT, Y. A branch and bound algorithm for the capacitated vehicle routing problem. *OR Spektrum*, v. 5, p. 77–85, 1983.

- LI, J.; PARDALOS, P. M.; SUN, H.; PEI, J.; ZHANG, Y. Iterated local search embedded adaptive neighborhood selection approach for the multi-depot vehicle routing problem with simultaneous deliveries and pickups. *Expert Systems with Applications*, v. 42, p. 3551–3561, 2015.
- LIU, R.; XIE, X.; AUGUSTO, V.; RODRIGUEZ, C. Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care. *European Journal of Operational Research*, v. 230, p. 475–486, 2013.
- LIU, S.; TANG, K.; YAO, X. Memetic search for vehicle routing with simultaneous pickup-delivery and time windows. *Swarm and Evolutionary Computation*, v. , p. 100927, 2021.
- MAMAGHANI, E. J.; DAVARI, S. The bi-objective periodic closed loop network design problem. *Expert Systems with Applications*, v. 144, 2020.
- MIN, H. The multiple vehicle routing problem with simultaneous delivery and pick-up points. *Transportation Research Part A: General*, v. 23, n. 5, p. 377–386, 1989.
- MINGYONG, L.; ERBAO, C. An improved differential evolution algorithm for vehicle routing problem with simultaneous pickups and deliveries and time windows. *Engineering Applications of Artificial Intelligence*, v. 23, p. 188–195, 2010.
- MONTANÉ, F. A. T.; GALVÃO, R. D. A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. *Computers and Operations Research*, v. 33, p. 595–619, 2006.
- MOSHEIOV, G. Vehicle routing with pick-up and delivery: tour-partitioning heuristics. *Computers & Industrial Engineering*, v. 34, n. 3, p. 669–684, 1998.
- NADIZADEH, A.; KAFASH, B. Fuzzy capacitated location-routing problem with simultaneous pickup and delivery demands. *Transportation Letters*, v. 11, p. 1–19, 2019.
- NAGY, G.; SALHI, S. Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries. *European Journal of Operational Research*, v. 162, n. 1, p. 126–141. *Logistics: From Theory to Application*, 2005.

- NEPOMUCENO, N.; SABOIA, R. B.; PINHEIRO, P. R. A fast randomized algorithm for the heterogeneous vehicle routing problem with simultaneous pickup and delivery. *Algorithms*, v. 12, 2019.
- PARK, H.; SON, D.; KOO, B.; JEONG, B. Waiting strategy for the vehicle routing problem with simultaneous pickup and delivery using genetic algorithm. *Expert Systems with Applications*, v. 165, 2021.
- PARRAGH, S. N. Introducing heterogeneous users and vehicles into models and algorithms for the dial-a-ride problem. *Transportation Research Part C: Emerging Technologies*, v. 19, p. 912–930, 2011.
- PARRAGH, S. N.; DOERNER, K. F.; HARTL, R. F. A survey on pickup and delivery problems. *Journal fur Betriebswirtschaft*, v. 58, p. 21–51, 2008a.
- PARRAGH, S. N.; DOERNER, K. F.; HARTL, R. F. A survey on pickup and delivery problems: Part ii: Transportation between pickup and delivery locations. *Journal fur Betriebswirtschaft*, v. 58, p. 81–117, 2008b.
- PECIN, D.; PESSOA, A.; POGGI, M.; UCHOA, E. Improved branch-cut-and-price for capacitated vehicle routing. *Mathematical Programming Computation*, v. 9, p. 61–100, 2017a.
- PECIN, D.; PESSOA, A.; POGGI, M.; UCHOA, E.; SANTOS, H. Limited memory rank-1 cuts for vehicle routing problems. *Operations Research Letters*, v. 45, n. 3, p. 206–209, 2017b.
- PESSOA, A.; SADYKOV, R.; UCHOA, E.; VANDERBECK, F. A generic exact solver for vehicle routing and related problems. *Mathematical Programming*, v. 183, p. 483–523, 2020.
- PETERSEN, B.; PISINGER, D.; SPOORENDONK, S. *Chvátal-Gomory Rank-1 Cuts Used in a Dantzig-Wolfe Decomposition of the Vehicle Routing Problem with Time Windows*, p. 397–419. Springer US, Boston, MA, 2008.

- PINTO, T.; ALVES, C.; CARVALHO, J. V. D.; MOURA, A. An insertion heuristic for the capacitated vehicle routing problem with loading constraints and mixed linehauls and backhauls. *FME Transactions*, v. 43, p. 311–318, 2015.
- PINTO, T.; ALVES, C.; DE CARVALHO, J. V. Variable neighborhood search algorithms for pickup and delivery problems with loading constraints. *Electronic Notes in Discrete Mathematics*, v. 58, p. 111–118, 2017.
- PINTO, T.; ALVES, C.; VALÉRIO DE CARVALHO, J. Variable neighborhood search algorithms for the vehicle routing problem with two-dimensional loading constraints and mixed linehauls and backhauls. *International Transactions in Operational Research*, v. 27, n. 1, p. 549–572, 2020.
- PRINS, C.; PRODHON, C.; CALVO, R. Nouveaux algorithmes pour le problème de localisation et routage sous contraintes de capacité. *MOSIM'04*, v. 2, p. 1115–1122, 2004.
- PRINS, C.; PRODHON, C.; CALVO, R. W. Solving the capacitated location-routing problem by a grasp complemented by a learning process and a path relinking. *4OR*, v. 4, p. 47–64, 2006.
- PRINS, C.; PRODHON, C.; RUIZ, A.; SORIANO, P.; CALVO, R. W. Solving the capacitated location-routing problem by a cooperative lagrangean relaxation-granular tabu search heuristic. *Transportation Science*, v. 41, p. 470–483, 2007.
- PRODHON, C. *LE PROBLÈME DE LOCALISATION-ROUTAGE*. PhD thesis, Université de Technologie de Troyes, Troyes, France, 2006.
- QU, Y.; BARD, J. F. The heterogeneous pickup and delivery problem with configurable vehicle capacity. *Transportation Research Part C: Emerging Technologies*, v. 32, p. 1–20, 2013.
- QU, Y.; BARD, J. F. A branch-and-price-and-cut algorithm for heterogeneous pickup and delivery problems with configurable vehicle capacity. *Transportation Science*, v. 49, p. 254–270, 2015.

- REIL, S.; BORTFELDT, A.; MÖNCH, L. Heuristics for vehicle routing problems with backhauls, time windows, and 3d loading constraints. *European Journal of Operational Research*, v. 266, p. 877–894, 2018.
- RIECK, J.; ZIMMERMANN, J. Exact solutions to the symmetric and asymmetric vehicle routing problem with simultaneous delivery and pick-up. *Business Research*, v. 6, p. 77–92, 2013.
- ROGERS, D. S.; TIBBEN-LEMBKE, R. S. *Going Backwards*. Reverse Logistics Executive Council, Pittsburgh, 1999.
- ROPKE, S.; PISINGER, D. A unified heuristic for a large class of vehicle routing problems with backhauls. *European Journal of Operational Research*, v. 171, p. 750–775, 2006.
- SADYKOV, R.; VANDERBECK, F. BaPCod - a generic branch-and-price code. Technical report, Inria Bordeaux Sud-Ouest. HAL-03340548, 2021.
- SALHI, S.; NAGY, G. A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling. *The Journal of the Operational Research Society*, v. 50, n. 10, p. 1034–1042, 1999.
- SANTOS, M. J.; AMORIM, P.; MARQUES, A.; CARVALHO, A.; PÓVOA, A. The vehicle routing problem with backhauls towards a sustainability perspective: a review. *TOP*, v. 28, p. 358–401, 2020.
- SHI, Y.; BOUDOUH, T.; GRUNDER, O. An efficient tabu search based procedure for simultaneous delivery and pick-up problem with time window. *IFAC-PapersOnLine*, v. 51, p. 241–246, 2018.
- SHI, Y.; ZHOU, Y.; BOUDOUH, T.; GRUNDER, O. A lexicographic-based two-stage algorithm for vehicle routing problem with simultaneous pickup–delivery and time window. *Engineering Applications of Artificial Intelligence*, v. 95, 2020.
- SOLOMON, M. M. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, v. 35, n. 2, p. 254–265, 1987.

- SOUZA, M. J. F.; MINE, M.; DE SOUZA ALVES SILVA, M.; OCHI, L.; SUBRAMANIAN, A. A hybrid heuristic, based on iterated local search and genius, for the vehicle routing problem with simultaneous pickup and delivery. *International Journal of Logistics Systems and Management*, v. 10, p. 142–157, 2011.
- SRIVASTAVA, S. K. Network design for reverse logistics. *Omega*, v. 36, n. 4, p. 535–548. Special Issue on Logistics: New Perspectives and Challenges, 2008.
- SUBRAMANIAN, A.; DRUMMOND, L. M.; BENTES, C.; OCHI, L. S.; FARIAS, R. A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery. *Computers and Operations Research*, v. 37, p. 1899–1911, 2010a.
- SUBRAMANIAN, A. Metaheurística iterated local search aplicada ao problema de roteamento de veículos com coleta e entrega simultânea. Master's thesis, Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal da Paraíba, João Pessoa, PB, 2008.
- SUBRAMANIAN, A. *Heuristic, Exact and Hybrid Approaches for Vehicle Routing Problems*. PhD thesis, Universidade Federal Fluminense, Niterói, RJ, 2012.
- SUBRAMANIAN, A.; UCHOA, E.; OCHI, L. S. New lower bounds for the vehicle routing problem with simultaneous pickup and delivery. FESTA, P. (Ed.), *Experimental Algorithms*, p. 276–287, Berlin, Heidelberg. Springer Berlin Heidelberg, 2010b.
- SUBRAMANIAN, A.; UCHOA, E.; OCHI, L. S. A hybrid algorithm for a class of vehicle routing problems. *Computers and Operations Research*, v. 40, p. 2519–2531, 2013a.
- SUBRAMANIAN, A.; UCHOA, E.; PESSOA, A. A.; OCHI, L. S. Branch-and-cut with lazy separation for the vehicle routing problem with simultaneous pickup and delivery. *Operations Research Letters*, v. 39, p. 338–341, 2011.
- SUBRAMANIAN, A.; UCHOA, E.; PESSOA, A. A.; OCHI, L. S. Branch-cut-and-price for the vehicle routing problem with simultaneous pickup and delivery. *Optimization Letters*, v. 7, p. 1569–1581, 2013b.

- TAN, P. Z.; KARA, B. Y. A hub covering model for cargo delivery systems. *Networks*, v. 49, n. 1, p. 28–39, 2007.
- TASAN, A. S.; GEN, M. A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries. *Computers and Industrial Engineering*, v. 62, p. 755–761, 2012.
- TCHOUPPO, M. N.; YALAOUI, A.; AMODEO, L.; YALAOUI, F.; FLORI, P.; LUTZ, F. An efficient column-generation algorithm for a new fleet size and mix pickup and delivery problem with time windows. *IFAC-PapersOnLine*, v. 51, p. 440–445, 2018.
- TOTH, P.; VIGO, D. *Vehicle Routing*. Society for Industrial and Applied Mathematics 3600 University City Science Center, Philadelphia, 2 edição, 2014.
- TÜTÜNCÜ, G. Y.; CARRETO, C. A.; BAKER, B. M. A visual interactive approach to classical and mixed vehicle routing problems with backhauls. *Omega*, v. 37, p. 138–154, 2009.
- TUZUN, D.; BURKE, L. I. A two-phase tabu search approach to the location routing problem. *European Journal of Operational Research*, v. 116, n. 1, p. 87–99, 1999.
- VIDAL, T.; CRAINIC, T. G.; GENDREAU, M.; PRINS, C. A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research*, v. 234, p. 658–673, 2014.
- WADE, A.; SALHI, S. An investigation into a new class of vehicle routing problem with backhauls. *Omega*, v. 30, n. 6, p. 479–487, 2002.
- WADE, A.; SALHI, S. *An Ant System Algorithm for the Mixed Vehicle Routing Problem with Backhauls*, p. 699–719. Springer US, Boston, MA, 2004.
- WANG, C.; MU, D.; ZHAO, F.; SUTHERLAND, J. W. A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup-delivery and time windows. *Computers and Industrial Engineering*, v. 83, p. 111–122, 2015.
- WANG, C.; ZHAO, F.; MU, D.; SUTHERLAND, J. W. Simulated annealing for a vehicle routing problem with simultaneous pickup-delivery and time windows. PRABHU,

- V.; TAISCH, M.; KIRITSIS, D. (Eds.), *Advances in Production Management Systems. Sustainable Production and Service Supply Chains*, p. 170–177, Berlin, Heidelberg. Springer Berlin Heidelberg, 2013.
- WANG, H. F.; CHEN, Y. Y. A genetic algorithm for the simultaneous delivery and pickup problems with time window. *Computers and Industrial Engineering*, v. 62, p. 84–95, 2012.
- WANG, H. F.; CHEN, Y. Y. A coevolutionary algorithm for the flexible delivery and pickup problem with time windows. *International Journal of Production Economics*, v. 141, p. 4–13, 2013.
- WANG, Y.; RAN, L.; GUAN, X.; ZOU, Y. Multi-depot pickup and delivery problem with resource sharing. *Journal of Advanced Transportation*, v. 2021, p. 1–22, 2021.
- WASSAN, N. A.; NAGY, G.; AHMADI, S. A heuristic method for the vehicle routing problem with mixed deliveries and pickups. *Journal of Scheduling*, v. 11, p. 149–161, 2008.
- YANG, S.; NING, L.; SHANG, P.; TONG, L. C. Augmented lagrangian relaxation approach for logistics vehicle routing problem with mixed backhauls and time windows. *Transportation Research Part E: Logistics and Transportation Review*, v. 135, 2020.
- YANG, S.; NING, L.; TONG, L. C.; SHANG, P. Optimizing electric vehicle routing problems with mixed backhauls and recharging strategies in multi-dimensional representation network. *Expert Systems with Applications*, v. 176, 2021.
- YOUSSEFIKHOSHBAKHT, M.; DIDEHVAR, F.; RAHMATI, F. A combination of modified tabu search and elite ant system to solve the vehicle routing problem with simultaneous pickup and delivery. *Journal of Industrial and Production Engineering*, v. 31, n. 2, p. 65–75, 2014.
- YU, V. F.; LIN, S. W. Multi-start simulated annealing heuristic for the location routing problem with simultaneous pickup and delivery. *Applied Soft Computing Journal*, v. 24, p. 284–290, 2014.

- YU, V. F.; LIN, S. Y. Solving the location-routing problem with simultaneous pickup and delivery by simulated annealing. *International Journal of Production Research*, v. 54, p. 526–549, 2016.
- ZACHARIADIS, E. E.; KIRANOUDIS, C. T. A local search metaheuristic algorithm for the vehicle routing problem with simultaneous pick-ups and deliveries. *Expert Systems with Applications*, v. 38, p. 2717–2726, 2011.
- ZACHARIADIS, E. E.; TARANTILIS, C. D.; KIRANOUDIS, C. T. An adaptive memory methodology for the vehicle routing problem with simultaneous pick-ups and deliveries. *European Journal of Operational Research*, v. 202, p. 401–411, 2010.
- ZHAO, Y.; LENG, L.; ZHANG, C. A novel framework of hyper-heuristic approach and its application in location-routing problem with simultaneous pickup and delivery. *Operational Research*, v. 21, p. 1299–1332, 2019.
- ZHAO, Y.; LENG, L.; ZHANG, J.; ZHANG, C.; WANG, W. Evolutionary hyperheuristics for location-routing problem with simultaneous pickup and delivery. *Complexity*, v. 2020, 2020.

Apêndice A

Lazy Cuts

Neste apêndice é feita uma comparação entre o tipo de *lazy cut* utilizado nesta pesquisa (4.27) com a generalização proposta por Agarwal e Venkateshan (2020), os quais passarão a ser chamados, respectivamente, de *Lazy Cuts 1* e *Lazy Cuts 2*. Nesta comparação, foram consideradas algumas instâncias do PRVCES e do PRVCEM, das quais haviam sido obtidas soluções ótimas, bem como com os menores de tempo de execução.

Tratando-se das configurações da máquina na qual os testes foram realizados, esta consiste em um computador com processador Intel[®] Core[™] i5-8265U 1,6 GHz, 8 GB de memória RAM e sistema operacional Ubuntu 18.04.5 LTS de 64 bits. Além disso, os modelos com os dois tipos de *lazy cuts* foram implementados por meio da linguagem Julia (versão 1.4.2), sendo utilizados o *VRPSolver* e o *BaPCod* em suas versões 0.4.1 e 0.52e, respectivamente. Por fim, os valores dos parâmetros do *VRPSolver* que foram utilizados se encontram descritos em *Arquivo de configuração 1*.

Nas Tabelas A.1 e A.2, estão indicados quantos cortes de cada tipo foram necessários para a resolução das instâncias do PRVCES e PRVCEM consideradas, sendo tais quantidades utilizadas como critério de comparação, bem como o tempo total de resolução. As colunas *Diff. TT(s)* e *Diff. LC* indicam as diferenças entre os tempos, em segundos, e a quantidade de cortes adicionadas, respectivamente. Da análise das tabelas, percebe-se que mais cortes do tipo 1 foram adicionados, porém não houve um

domínio de um tipo sobre o outro no que se refere à redução do tempo de execução.

Tabela A.1: Comparação entre os tipos de *Lazy Cuts* — PRVCES.

Instância	C	<i>Lazy Cuts 1</i>			<i>Lazy Cuts 2</i>			Diff. TT(s)	Diff. LC
		UB	TT(s)	LC	UB	TT(s)	LC		
r101	100	1009,95	2238,00	4	1009,95	1817,25	0	420,75	4
r201	100	666,20	3899,34	0	666,20	2957,98	0	941,36	0
c101	100	1220,18	536,75	13	1220,18	1315,37	4	-778,62	9
c201	100	662,07	415,80	0	662,07	431,24	0	-15,44	0
rc101	100	1059,32	1820,96	170	1059,32	620,07	42	1200,89	128
rc201	100	672,92	4016,09	0	672,92	4266,60	0	-250,51	0

Tabela A.2: Comparação entre os tipos de *Lazy Cuts* — PRVCEM.

Instância	C	<i>Lazy Cuts 1</i>			<i>Lazy Cuts 2</i>			Diff. TT(s)	Diff. LC
		UB	TT(s)	LC	UB	TT(s)	LC		
cmt1h	50	465,02	2842,42	41	465,02	2568,93	41	273,49	0
cmt1q	50	489,74	18,37	0	489,74	20,94	0	-2,57	0
cmt2t	75	782,77	109,95	1	782,77	102,75	1	7,2	0
cmt12t	100	787,52	845,89	1	787,52	868,45	1	-22,56	0

Arquivo de configuração 1 — VRPSPD_1

Parameters open for the user with default values (notation in the Math.Prog. paper in comments)

RCSPstopCutGenTimeThresholdInPricing = 5 // τ^{soft}

RCSPhardTimeThresholdInPricing = 10 // τ^{hard}

RCSPuseBidirectionalSearch = 1 // ϕ^{bidir}

Ng-path parameters

RCSPmaxNGneighbourhoodSize = 16 // η^{max}

Rank-1 cuts parameters

RCSPrankOneCutsMaxNumRows = 7 // θ^{rows}

RCSPrankOneCutsMemoryType = 1 // θ^{mem} , 0 = auto, 1 = arc, 2 = vertex

Enumeration parameters

RCSPmaxNumOfLabelsInEnumeration = 0

RCSPmaxNumOfEnumeratedSolutions = 10000000 // ω^{routes}

Strong Branching parameters

StrongBranchingPhaseOneCandidatesNumber = 50 // ζ_1^{num}

StrongBranchingPhaseOneTreeSizeEstimRatio = 0.1 // ζ_1^{estim}

StrongBranchingPhaseTwoTreeSizeEstimRatio = 0.1 // ζ_2^{estim}

Tailing off parameters

CutTailingOffThreshold = 0.015 // δ^{gap}

Apêndice B

Parâmetros

Neste apêndice estão descritos os parâmetros associados ao *VRPSolver* que foram utilizados nos testes realizados. Durante o processo de calibração desses parâmetros, foram obtidos 17 arquivos de configuração distintos. Dentre eles, foram selecionados os três que se mostraram mais eficazes na resolução de todos os problemas considerados, conforme apresentado na descrição dos resultados (v. Capítulo 5). Tais arquivos foram denominados como *GM_1*, *GM_10* e *GM_17* — sendo *GM* um acrônimo para *Generic Model* — e estão apresentados em *Arquivo de configuração 2, 3 e 4*, respectivamente. Importante ressaltar que alguns parâmetros estão descritos juntamente com seu símbolo matemático correspondente, a fim de estabelecer a correlação entre a notação utilizada por Pessoa et al. (2020) e sua nomenclatura prática.

Arquivo de configuração 2 — GM_1

Parameters fixed in VRPSolver but with values different from default

MaxNbOfStagesInColGenProcedure = 3
 colGenSubProbSolMode = 3
 MipSolverMultiThread = 1
 ApplyStrongBranchingEvaluation = true

Parameters open for the user with default values (notation in the Math.Prog. paper in comments)

RCSPstopCutGenTimeThresholdInPricing = 5 // τ^{soft}
 RCSPhardTimeThresholdInPricing = 10 // τ^{hard}
 RCSPuseBidirectionalSearch = 1 // ϕ^{bidir}

Enumeration parameters

RCSPmaxNumOfLabelsInEnumeration = 0
 RCSPmaxNumOfEnumeratedSolutions = 10000000 // ω^{routes}

Capacity cuts parameters

RCSPcapacityCutsSeparator = 1 // 0 - CVRPSEP

Tailing off parameters

CutTailingOffThreshold = 0.015 // δ^{gap}
 CutTailingOffCounterThreshold = 5

Strong Branching parameters

StrongBranchingPhaseOneCandidatesNumber = 50 // ζ_1^{num}
 StrongBranchingPhaseOneTreeSizeEstimRatio = 0.1 // ζ_1^{estim}
 StrongBranchingPhaseTwoCandidatesNumber = 5 // ζ_2^{num}
 StrongBranchingPhaseTwoTreeSizeEstimRatio = 0.1 // ζ_2^{estim}

Rank-1 cuts parameters

RCSPrankOneCutsMaxNumPerRound = 100
 RCSPrankOneCutsMaxNumRows = 3 // θ^{rows}
 RCSPrankOneCutsLSnumIterations = 1000
 RCSPrankOneCutsMemoryType = 1 // θ^{mem} , 0 = auto, 1 = arc, 2 = vertex

Ng-path parameters

RCSPdynamicNGmode = 1
 RCSPmaxNGneighbourhoodSize = 16 // η^{max}
 RCSPmaxNGaverNeighbourhoodSize = 14
 RCSPinitNGneighbourhoodSize = 2

Others parameters

DEFAULTPRINTLEVEL = 0
 MaxNbOfBBtreeNodeTreated = 10000
 colGenDualPriceSmoothingAlphaFactor = 1
 RCSPuseExactComplBoundsInEnumeration = true
 GlobalTimeLimit = 86400

Arquivo de configuração 3 — GM_10

Parameters fixed in VRPSolver but with values different from default

MaxNbOfStagesInColGenProcedure = 3
 colGenSubProbSolMode = 3
 MipSolverMultiThread = 1
 ApplyStrongBranchingEvaluation = true

Parameters open for the user with default values (notation in the Math.Prog. paper in comments)

RCSPstopCutGenTimeThresholdInPricing = 5 // τ^{soft}
 RCSPhardTimeThresholdInPricing = 20 // τ^{hard}
 RCSPuseBidirectionalSearch = 1 // ϕ^{bidir}

Enumeration parameters

RCSPmaxNumOfLabelsInEnumeration = 0
 RCSPmaxNumOfEnumeratedSolutions = 10000000 // ω^{routes}

Capacity cuts parameters

RCSPcapacityCutsSeparator = 1 // 0 - CVRPSEP

Tailing off parameters

CutTailingOffThreshold = 0.015 // δ^{gap}
 CutTailingOffCounterThreshold = 5

Strong Branching parameters

StrongBranchingPhaseOneCandidatesNumber = 100 // ζ_1^{num}
 StrongBranchingPhaseOneTreeSizeEstimRatio = 2 // ζ_1^{estim}
 StrongBranchingPhaseTwoCandidatesNumber = 5 // ζ_2^{num}
 StrongBranchingPhaseTwoTreeSizeEstimRatio = 0.2 // ζ_2^{estim}

Rank-1 cuts parameters

RCSPrankOneCutsMaxNumPerRound = 100
 RCSPrankOneCutsMaxNumRows = 4 // θ^{rows}
 RCSPrankOneCutsLSnumIterations = 1000
 RCSPrankOneCutsMemoryType = 2 // θ^{mem} , 0 = auto, 1 = arc, 2 = vertex

Ng-path parameters

RCSPdynamicNGmode = 1
 RCSPmaxNGneighbourhoodSize = 16 // η^{max}
 RCSPmaxNGaverNeighbourhoodSize = 14
 RCSPinitNGneighbourhoodSize = 2

Bucket parameters

RCSPnumberOfBucketsPerVertex = 25
 RCSPdynamicBucketSteps = 1

Others parameters

DEFAULTPRINTLEVEL = 0
 MaxNbOfBBtreeNodeTreated = 10000
 colGenDualPriceSmoothingAlphaFactor = 1
 RCSPuseExactComplBoundsInEnumeration = true
 GlobalTimeLimit = 86400

Arquivo de configuração 4 — GM_17

Parameters fixed in VRPSolver but with values different from default

MaxNbOfStagesInColGenProcedure = 3
 colGenSubProbSolMode = 3
 MipSolverMultiThread = 1
 ApplyStrongBranchingEvaluation = true

Parameters open for the user with default values (notation in the Math.Prog. paper in comments)

RCSPstopCutGenTimeThresholdInPricing = 10 // τ^{soft}
 RCSPhardTimeThresholdInPricing = 40 // τ^{hard}
 RCSPuseBidirectionalSearch = 1 // ϕ^{bidir}

Enumeration parameters

RCSPmaxNumOfLabelsInEnumeration = 0
 RCSPmaxNumOfEnumeratedSolutions = 10000000 // ω^{routes}

Capacity cuts parameters

RCSPcapacityCutsSeparator = 1 // 0 - CVRPSEP

Tailing off parameters

CutTailingOffThreshold = 0.015 // δ^{gap}
 CutTailingOffCounterThreshold = 5

Strong Branching parameters

StrongBranchingPhaseOneCandidatesNumber = 50 // ζ_1^{num}
 StrongBranchingPhaseOneTreeSizeEstimRatio = 0.1 // ζ_1^{estim}
 StrongBranchingPhaseTwoTreeSizeEstimRatio = 0.1 // ζ_2^{estim}

Rank-1 cuts parameters

RCSPrankOneCutsMaxNumRows = 5 // θ^{rows}
 RCSPrankOneCutsMemoryType = 1 // θ^{mem} , 0 = auto, 1 = arc, 2 = vertex

Ng-path parameters

RCSPdynamicNGmode = 1
 RCSPmaxNGneighbourhoodSize = 16 // η^{max}
 RCSPmaxNGaverNeighbourhoodSize = 14
 RCSPinitNGneighbourhoodSize = 2

Others parameters

DEFAULTPRINTLEVEL = 0
 MaxNbOfBBtreeNodeTreated = 10000
 colGenDualPriceSmoothingAlphaFactor = 1
 RCSPuseExactComplBoundsInEnumeration = true
 MinNumOfCutRoundsBeforeStopBySp = 20
 GlobalTimeLimit = 86400
