UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

**Problema de Roteamento de Veículos Heterogêneos com
Múltiplos Entregadores e Coleta e Entrega Simultâneas**

Matheus Augusto Fernandes de Assunção

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências
de Computação e Matemática Computacional (PPG-CMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Matheus Augusto Fernandes de Assunção

Problema de Roteamento de Veículos Heterogêneos com Múltiplos Entregadores e Coleta e Entrega Simultâneas

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Maristela Oliveira dos Santos

USP – São Carlos
Agosto de 2020

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

F363p Fernandes de Assunção, Matheus Augusto
 Problema de Roteamento de Veículos Heterogêneos
 com Múltiplos Entregadores e Coleta e Entrega
 Simultâneas / Matheus Augusto Fernandes de Assunção;
 orientador Maristela Oliveira Dos Santos. -- São
 Carlos, 2020.
 93 p.

 Dissertação (Mestrado - Programa de Pós-Graduação
 em Ciências de Computação e Matemática
 Computacional) -- Instituto de Ciências Matemáticas
 e de Computação, Universidade de São Paulo, 2020.

 1. Roteamento de Veículos. 2. Otimização. 3.
 Logística. I. Dos Santos, Maristela Oliveira,
 orient. II. Título.

Matheus Augusto Fernandes de Assunção

Simultaneous Pickup and Delivery Heterogeneous Vehicle Routing Problem with Multiple Deliverymen

Master dissertation submitted to the Institute of
Mathematics and Computer Sciences – ICMC-USP,
in partial fulfillment of the requirements for the
degree of the Master Program in Computer Science
and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Profa. Dra. Maristela Oliveira dos Santos

USP – São Carlos
August 2020

*Dedicado àqueles que continuam tentando,
mesmo quando a vida se torna complicada,
e quando o tempo foge mais rápido do que se consegue correr*

AGRADECIMENTOS

Os agradecimentos principais são para a minha orientadora, a Professora Doutora Maristela Oliveira do Santos, cuja paciência e cujo direcionamento foram inestimáveis para o término desse trabalho. Agradeço também à minha mãe, Dulcinéia de Jesus Chaves Fernandes, e minha irmã Débora, pelo apoio e por acreditarem em mim. Agradeço ao meu amigo João Guilherme, que me ajudou tanto na implementação do método descrito nesse trabalho, na sua escrita, no modelo matemático, e em tantas outras coisas que não há espaço para listá-las todas. Esse trabalho não existiria sem a sua ajuda.

Agradeço também ao Instituto de Ciências Matemáticas e de Computação (ICMC) da USP São Carlos, ao Laboratório de Otimização (LOT) e todos os meus colegas, ao Centro de Ciências Matemáticas Aplicadas à Indústria (CeMEAI), que forneceram estrutura para que eu pudesse desenvolver esse projeto, em especial ao CeMEAI por fornecer o cluster Euler, recurso financiado pela FAPESP (proc. 2013/07375-0), utilizado para gerar os resultados dispostos neste trabalho. Agradeço finalmente à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela bolsa de mestrado a qual foi crucial para o desenvolvimento desse trabalho.

*“The scariest moment
is always just before you start.”
(Stephen King)*

RESUMO

ASSUNÇÃO, M. A. F. DE. **Problema de Roteamento de Veículos Heterogêneos com Múltiplos Entregadores e Coleta e Entrega Simultâneas**. 2020. 93 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

O Problema de Roteamento de Veículos com Janelas de Tempo e Múltiplos Entregadores trata de uma modalidade do Problema de Roteamento de Veículos com Janelas de Tempo onde os veículos, ao invés de necessariamente parar em cada cliente em sua rota, podem parar em pontos estratégicos próximos a alguns clientes, de onde a entrega é realizada à pé ou via algum veículo menor. A literatura existente comprova que a adição de múltiplos entregadores pode diminuir significativamente o tempo de entrega. Esse trabalho explora uma nova extensão desse problema, tratando de casos em que os clientes podem necessitar tanto de coleta, quanto de entrega, e a frota é composta de veículos de classes diferentes. Para resolver esse problema foi elaborado um modelo matemático utilizando técnicas de Programação Inteira Mista, e um método de Random Variable Neighborhood Descent. Os resultados encontrados pela heurística são comparados com resultados utilizando somente o *solver* a partir do modelo matemático, bem como com resultados utilizando a solução encontrada pelo RVND como solução inicial para o *solver*. Os resultados mostram a vantagem em utilizar o método RVND, pela qualidade das soluções encontradas e pelo tempo computacional necessário para encontrá-las.

Palavras-chave: Roteamento de Veículos, Otimização, Logística.

ABSTRACT

ASSUNÇÃO, M. A. F. DE. **Simultaneous Pickup and Delivery Heterogeneous Vehicle Routing Problem with Multiple Deliverymen**. 2020. 93 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

The Vehicle Routing Problem with Time Windows and Multiple Deliverymen describes a variant of the Vehicle Routing Problem with Time Windows in which the delivery vehicles, instead of necessarily stopping at each client, can stop at strategically chosen locations which are close to some of the clients, from where the delivery is done on foot or with a smaller vehicle to those clients. The existing literature shows that adding multiple deliverymen can minimize delivery times. This work explores a new extension of this problem, studying the case where the clients might have pickup and delivery needs, and the fleet is composed of different classes of vehicles. In order to solve this problem, a mathematical model using Mixed Integer Programming Techniques, and a Random Variable Neighborhood Descent method were developed. Results obtained by the heuristic are compared with results found using only the solver with the mathematical model, and also with the solutions obtained by using the solution found by the RVND as an initial solution for the solver. The results show a clear advantage in using the RVND method, for the quality of its solutions, and the computational time needed to find them.

Keywords: Vehicle Routing, Optimization, Logistics.

LISTA DE ILUSTRAÇÕES

Figura 1 – Esquema de uma possível solução do VRPTWMD.	24
Figura 2 – Antes do Or-Opt	47
Figura 3 – Depois do Or-Opt	48
Figura 4 – Antes do 2-Opt	48
Figura 5 – Depois do 2-Opt	49
Figura 6 – Antes do mix_routes	49
Figura 7 – Depois do mix_routes	50
Figura 8 – Antes do shift	50
Figura 9 – Depois do shift	50
Figura 10 – Antes do swap(1,1)	51
Figura 11 – Depois do swap(1,1)	51
Figura 12 – Antes do combine	51
Figura 13 – Depois do combine	52
Figura 14 – Valores da instância RCPD1001	87
Figura 15 – Roteamento do RCPD1001 com apenas 1 entregador por veículo	88
Figura 16 – Roteamento do RCPD1001 com até 3 entregadores por veículo	89
Figura 17 – Gráfico mostrando o desempenho dos modelos para as instâncias-teste	93

LISTA DE ALGORITMOS

Algoritmo 1 – Algoritmo RVND	44
Algoritmo 2 – Algoritmo Construtivo	45
Algoritmo 3 – Algoritmo de Perturbação	53

LISTA DE TABELAS

Tabela 1	– Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias R1	56
Tabela 2	– Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias C1	56
Tabela 3	– Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias RC1	57
Tabela 4	– Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias R2	57
Tabela 5	– Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias C2	57
Tabela 6	– Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias RC2	57
Tabela 7	– Modelos Alternativos	59
Tabela 8	– Resultados Comparativos para o modelo nosymnoc	61
Tabela 9	– Resultados Comparativos para o modelo sym2noc	61
Tabela 10	– Resultados Comparativos para o modelo sym1noc	62
Tabela 11	– Resultados Comparativos para o modelo sym12noc	62
Tabela 12	– Resultados Comparativos para o modelo nosymc	63
Tabela 13	– Resultados Comparativos para o modelo sym2c	63
Tabela 14	– Resultados Comparativos para o modelo sym1c	64
Tabela 15	– Resultados Comparativos para o modelo sym12c	64
Tabela 16	– Desempenho do RVND	66
Tabela 17	– Otimalidade das soluções	67
Tabela 18	– Tempos de Execução do CPLEX em segundos	68
Tabela 19	– Tempos de Execução do RVND em segundos	68
Tabela 20	– Tempos de Execução do CPLEX com RVND em segundos	69
Tabela 21	– Resultados das instâncias de 50 clientes <i>rcdp5004abc</i> e <i>rcdp5007abc</i> , parte 1	79
Tabela 22	– Resultados das instâncias de 50 clientes <i>rcdp5004abc</i> e <i>rcdp5007abc</i> , parte 2	79
Tabela 23	– Resultados das instâncias de 100 clientes R1, parte 1	80
Tabela 24	– Resultados das instâncias de 100 clientes R1, parte 2	80
Tabela 25	– Resultados das instâncias de 100 clientes R2, parte 1	81
Tabela 26	– Resultados das instâncias de 100 clientes R2, parte 2	81
Tabela 27	– Resultados das instâncias de 100 clientes C1, parte 1	82

Tabela 28 – Resultados das instâncias de 100 clientes C1, parte 2	82
Tabela 29 – Resultados das instâncias de 100 clientes C2, parte 1	82
Tabela 30 – Resultados das instâncias de 100 clientes C2, parte 2	83
Tabela 31 – Resultados das instâncias de 100 clientes RC1, parte 1	83
Tabela 32 – Resultados das instâncias de 100 clientes RC1, parte 2	83
Tabela 33 – Resultados das instâncias de 100 clientes RC2, parte 1	84
Tabela 34 – Resultados das instâncias de 100 clientes RC2, parte 2	84
Tabela 35 – Modelos Alternativos	87
Tabela 36 – Resultados obtidos pelo <i>solver</i> do CPLEX para o modelo homogêneo sem restrições adicionais, <i>nosymnoc</i>	88
Tabela 37 – Número de variáveis e de restrições para cada modelo homogêneo para a instância rcdp5001, com 1 entregador por veículo	91
Tabela 38 – Número de variáveis e de restrições para cada modelo homogêneo para a instância rcdp5001, com no máximo 3 entregadores por veículo	92

SUMÁRIO

1	INTRODUÇÃO	23
2	REVISÃO DA LITERATURA	27
2.1	O VRPTWMD	27
2.2	O VRSPD	30
2.3	<i>Random Variable Neighborhood Descent</i>	30
3	DEFINIÇÃO DO PROBLEMA E MODELO MATEMÁTICO	33
3.1	Modelagem do SPDHVRPMD	34
3.2	Desigualdades Válidas	38
3.2.1	<i>Restrições de quebra de simetria</i>	38
3.2.2	<i>Outras restrições de carga</i>	39
4	HEURÍSTICA DE EXPLORAÇÃO DE VIZINHANÇA RVND	41
4.1	Random Variable Neighborhood Descent	41
4.2	Funcionamento Geral do RVND para o SPDHVRPMD	42
4.2.1	<i>Heurística Construtiva</i>	44
4.2.2	<i>Heurística de Redução do Número de Rotas</i>	46
4.2.3	<i>Vizinhanças</i>	47
4.2.3.1	<i>Vizinhanças Intra-Rotas</i>	47
4.2.3.2	<i>Vinhanças Inter-Rotas</i>	49
4.2.4	<i>Algoritmo de Perturbação</i>	52
5	RESULTADOS COMPUTACIONAIS	55
5.1	Classes de Veículos e Instâncias	55
5.2	Testes Computacionais	58
5.2.1	<i>Resultados Comparativos</i>	59
6	PERSPECTIVAS	71
	REFERÊNCIAS	75
A	RESULTADOS COMPLEMENTARES	79
B	ESTUDO DE MODELO PARA O CASO HOMOGÊNEO	85

B.1	Parâmetros	85
B.2	Frota Homogênea	86
B.3	Resultados para instâncias de 5, 10, e 25 clientes	86
B.4	Resultados gerais	90

INTRODUÇÃO

O VRP (*Vehicle Routing Problem*, Problema de Roteamento de veículos) pode ser visto como uma extensão do Problema do Caixeiro Viajante onde existem vários veículos e uma demanda a ser entregue em vários clientes. O problema consiste em determinar rotas partindo de um depósito para um conjunto de clientes, de forma que suas demandas sejam satisfeitas de maneira a otimizar alguns fatores pré-definidos no problema. O objetivo pode ser de minimizar a distância percorrida ou o tamanho da frota (o número de veículos utilizados), maximizar o número de clientes atendidos (em contextos em que a entrega total não pode ser garantida), minimizar o tempo de operação, entre outros objetivos específicos à modalidade do problema tratado.

Também podem ser incorporadas restrições referentes à capacidade dos veículos, ao tempo máximo de duração de uma rota (por exemplo, uma jornada de 8 horas), às janelas de tempo em que o cliente pode ser atendido, às demandas do cliente (para coleta, entrega, ou ambas simultaneamente), número de clientes em cada rota, ou até mesmo a ordem de visita entre os clientes, dependendo do que o problema considerar ([LAPORTE, 1992](#)), ([BRAEKERS; RAMAEKERS; NIEUWENHUYSE, 2016](#)). Esse problema e todas as suas extensões são NP-Difíceis ([LENSTRA; KAN, 1981](#)), o que faz com que a maioria dos trabalhos na literatura não utilizem métodos exatos para resolvê-lo.

O Problema de Roteamento de Veículos com Janelas de Tempo e Múltiplos Entregadores (VRPTWMD, *Vehicle Routing Problem with Time Windows and Multiple Deliverymen*), abordado primeiro em [Pureza e Morabito \(2010\)](#), é uma modalidade do VRP inspirado em um serviço de entrega de refrigerantes em cidades onde os caminhões param em estacionamentos e os entregadores seguem à pé para os clientes próximos do local de parada. Essa modalidade de problema reflete a realidade de muitas empresas que fazem entregas em cidades grandes, onde parar o caminhão em cada cliente pode não ser possível ([PUREZA; MORABITO, 2010](#)).

Em [Pureza, Morabito e Reimann \(2012\)](#), os autores descrevem um modelo de Progra-

Figura 1 – Esquema de uma possível solução do VRPTWMD.

Fonte: (PUREZA; MORABITO; REIMANN, 2012)

mação Inteira Mista para este problema, e utilizam uma heurística de busca tabu e colônia de formigas para resolvê-lo. Neste artigo, é considerada somente coleta ou somente entrega, mas não coleta e entrega simultaneamente, e uma frota homogênea. A Figura 1 ilustra uma possível solução do problema. Nessa figura, podem-se ver como alguns pontos de demanda estão localizados de maneira agrupada (em *clusters*). Essa figura ilustra uma rota possível de um veículo saindo de um depósito (triângulo), estacionando nos pontos marcados em vermelho de onde os entregadores saem do veículo para atender os clientes próximos àquele ponto de estacionamento, e, após servir todos os clientes, retornando ao depósito.

A coleta e entrega simultâneas no contexto de roteamento de veículos, descrito em Min (1989), permite que os clientes possuam demandas de coleta ou de entrega, de modo que um veículo de entrega saia do depósito com toda a carga a ser entregue, e retorne ao depósito com toda carga coletada durante a rota. Em Dethloff (2001) é demonstrado que esse tipo de problema pode ser visto em contextos de logística reversa, como o caso de deixar produtos ou coletá-los de volta ao depósito para reciclagem ou reuso. No contexto que inspirou o VRPTWMD (*Vehicle Routing Problem with Time Windows and Multiple Deliverymen*), a entrega de bebidas em cidades grandes, essa abordagem se encaixa num sistema de entrega de engradados que são por sua vez retornados ao depósito para reuso.

Analisar esse mesmo problema com veículos heterogêneos o torna um pouco mais próximo da realidade. Em situações reais, nem todos os veículos de uma empresa serão iguais. Por serem diferentes, é possível que esses veículos tenham custo de utilização diferentes, carga máxima maior ou menor, velocidade ou tempo de serviço diferentes, número máximo de entregadores diferente, entre outros. Neste trabalho consideramos somente distinções de carga, custo,

e de número de entregadores, de modo a simplificar o problema porém ainda aproximando-nos de uma situação realista.

O modelo de matemático apresentado nesta dissertação é baseado principalmente no modelo de [Pureza, Morabito e Reimann \(2012\)](#), desenvolvido usando técnicas de Programação Inteira Mista, e validado utilizando instâncias para o VRP com Janelas de Tempo e Coleta e Entrega Simultâneas elaboradas por [Wang e Chen \(2012\)](#), além de instâncias para o VRP Heterogêneo por [Liu e Shen \(1999\)](#), ambas baseadas nas instâncias de [Solomon \(1987\)](#).

Também foi desenvolvida uma heurística de RVND para obter soluções para problemas maiores. Essa abordagem se baseia nos trabalhos de [Subramanian *et al.* \(2010\)](#) e [Alvarez e Munari \(2016\)](#), a serem debatidos no Capítulo 2. Essa heurística foi desenvolvida principalmente para fornecer soluções para instâncias grandes, de 100 clientes, dado que o solver encontra bastante dificuldade em resolver instâncias dessa dimensão, como pode ser visto no Capítulo 5.

O restante deste documento está organizado da seguinte forma: o Capítulo 2 contém um revisão dos trabalhos desenvolvidos em torno do VRPTWMD e em torno de VRP com Coleta e Entrega Simultâneas, discorrendo sobre seus objetivos e suas contribuições para a literatura; no Capítulo 3 está descrito o problema e sua modelagem matemática; no Capítulo 4 a heurística RVND é detalhada; no Capítulo 5 resultados computacionais para o problema heterogêneo são apresentados; por fim, algumas conclusões e perspectivas são apresentadas no Capítulo 6. Além disso, dois Apêndices são apresentados. O Apêndice A exibe os resultados encontrados para outras instâncias para as quais a única solução encontrada foi a solução inicial fornecida pela heurística RVND, e o Apêndice B detalha um estudo de modelo feito para o problema homogêneo.

REVISÃO DA LITERATURA

Esse trabalho descreve e modela o Problema de Roteamento de Veículos com Múltiplos Entregadores e Coleta e Entrega Simultâneas, além de resolver instâncias pequenas, médias e grandes, para validar a modelagem matemática. Para contextualizar esse problema, foi realizada uma revisão da literatura do VRPTWMD (*Vehicle Routing Problem with Time Windows and Multiple Deliverymen*, em português Problema de Roteamento de Veículos com Janelas de Tempo e Múltiplos Entregadores) e do VRPSPD (*Vehicle Routing Problem with Simultaneous Pickup and Delivery*, em português Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas).

2.1 O VRPTWMD

O VRPTWMD (*Vehicle Routing Problem with Time Windows and Multiple Deliverymen*) foi descrito pela primeira vez em [Pureza e Morabito \(2010\)](#). Neste artigo, os autores demonstram a vantagem de incorporar múltiplos entregadores ao alterar instâncias de [Solomon \(1987\)](#). Os autores dão como justificativa prática para essa abordagem o contexto de entrega de bebidas por uma empresa no interior de São Paulo, e explicam que a empresa precisa atender às seguintes restrições de planejamento:

- Disponibilidade de veículos da frota para cada período de entrega
- Capacidade dos veículos disponíveis
- Disponibilidade no estoque dos produtos solicitados (quantidade e mix)
- Disponibilidade de tempo para entrega dos produtos
- Prioridade de entrega do cliente
- Janelas de tempo em um ou mais clientes

Os autores notam também que quando há muitos clientes próximos uns aos outros (um *cluster*), o entregador estaciona o veículo e faz as entregas à pé (ou por outro veículo menor) àqueles clientes. O objetivo dessa estratégia é poupar tempo e o esforço de estacionar o veículo diretamente em cada local. Os autores concluem que, se houvesse mais um ou dois entregadores, poderia-se cortar o tempo de serviço num *cluster* significativamente, e é essa conclusão que os leva a propor o VRPTWMD.

Os autores [Pureza e Morabito \(2010\)](#) não modelam matematicamente o problema, mas buscam resolvê-lo utilizando heurísticas para construir rotas de custo mínimo, tentando também minimizar o número de entregadores em cada veículo, o tamanho da frota necessária, e a distância total percorrida. Os resultados desse método levam os autores a concluir que existe, de fato, vantagem em associar mais entregadores a uma rota.

Em [Pureza, Morabito e Reimann \(2012\)](#), o VRPTWMD é modelado matematicamente pela primeira vez e é resolvido utilizando heurísticas de Busca Tabu ([GLOVER, 1989](#)), e Colônia de Formigas ([DORIGO; GAMBARDELLA, 1997](#)). O modelo matemático descrito neste artigo e seus parâmetros foram amplamente utilizados na literatura e são também a base principal do modelo descrito no Capítulo 3. Esse modelo descreve um problema com frota homogênea (ou seja, com capacidades de carga, velocidades e tamanho máximo de tripulação iguais para toda a frota), com somente entrega ou coleta, exclusivamente, durante toda a operação, além das outras restrições como janelas de tempo, carga máxima, número máximo de entregadores disponíveis, e tempo máximo para o percurso. A função objetivo empregada nesse trabalho visa minimizar a quantidade de veículos na frota, o número de entregadores e a distância total percorrida.

Dentre as várias contribuições do artigo de [Pureza, Morabito e Reimann \(2012\)](#), existe também a definição da ordem de prioridade entre os objetivos, estabelecendo que a quantidade de veículos na frota é mais importante do que a quantidade total de entregadores na frota, que por sua vez é mais importante do que a distância total percorrida. Esse artigo utiliza as instâncias de [Solomon \(1987\)](#) alteradas para que os tempos de serviço sejam inversamente proporcionais ao número de entregadores no veículo, e diretamente proporcionais à demanda, e resolve o problema computacionalmente, mas não tenta resolver o problema de criação de *clusters*. Somente uma instância foi resolvida até a otimalidade pelo solver CPLEX, e todas as outras excederam o tempo máximo de 10 horas de execução. Para essas instâncias, tanto a Busca Tabu quanto o Colônia de Formigas conseguiram soluções melhores que as obtidas pelo CPLEX até o limite de tempo de 10 horas em tempo consideravelmente menor. Os autores concluem ainda que nenhuma das heurísticas dominou a outra quanto à qualidade das soluções.

Vários outros artigos trataram esse mesmo problema. [Alvarez e Munari \(2016\)](#) resolvem o problema com os mesmos parâmetros que [Pureza, Morabito e Reimann \(2012\)](#), porém utilizando duas meta-heurísticas, Busca Local Iterada (*Iterated Local Search*, ILS) e Busca em Vizinhança Grande (*Large Neighborhood Search*, LNS) propostas. Uma vizinhança é um conjunto de soluções alternativas atingíveis a partir de uma solução inicial utilizando um meca-

nismo (um algoritmo) de fácil replicação. Nesse trabalho, várias vizinhanças são estabelecidas de modo a diversificar a busca por uma solução minimizando os custos de utilização dos veículos, dos entregadores, e da distância total percorrida. São também definidos alguns operadores de destruição e reparação de solução, para o LNS, com o intuito de expandir a busca e possivelmente sair de mínimos locais. As meta-heurísticas são aplicadas em instâncias de [Solomon \(1987\)](#), e os resultados demonstram que, apesar de ambas as meta-heurísticas obterem bons resultados, uma não se sobressai a outra.

[Munari e Morabito \(2016\)](#) tentam uma abordagem diferente, utilizando um modelo de Set Partitioning, e um método exato de *branch-price-and-cut* para resolvê-lo. Os autores primeiramente expõem a modelagem original, e depois a modelagem alternativa de Set Partitioning, uma modelagem onde cada variável binária representa a presença ou ausência de uma possível rota na solução. Para gerar colunas para essa nova modelagem (novas rotas alternativas que podem entrar na solução), os autores desenvolvem um método exato de *branch-price-and-cut*. Novamente, as instâncias de [Solomon \(1987\)](#) (clássicas na literatura) são resolvidas com esse método, e, para avaliar a sua performance, esses resultados são comparados com resultados de outras heurísticas utilizadas na literatura para esse problema. Os autores concluem que o BPC obteve bons resultados para algumas classes de instâncias, mas para outras as soluções encontradas por métodos heurísticos na literatura as soluções do BPC foram significativamente piores. Os autores concluem portanto que um método híbrido tratando esse problema obteria resultados melhores. Então, [Alvarez e Munari \(2017\)](#) combinam o *branch-price-and-cut* com as meta-heurísticas ILS e LNS de [Alvarez e Munari \(2016\)](#) para criar um método híbrido. Nesse método, as meta-heurísticas fornecem as colunas iniciais para o BPC, a partir das quais uma nova solução é encontrada pelo método exato. Essa solução é então fornecida para as meta-heurísticas com o objetivo de melhorá-la e então fornecer mais colunas para o BPC. O método para ao encontrar a solução ótima ou quando o tempo limite de execução é atingido. As instâncias de [Solomon \(1987\)](#) são resolvidas por esse método híbrido, pelo BPC, e pelas meta-heurísticas, de modo a avaliar o seu desempenho. Os resultados demonstram que o método híbrido consegue consistentemente encontrar soluções de melhor qualidade do que o BPC ou as meta-heurísticas sozinhos, e até mesmo consegue encontrar a solução ótima mais rapidamente para a maior parte das instâncias que são resolvidas até a otimalidade.

[Neto e Pureza \(2016\)](#) modelam o MTVRPTWMD (*Multiple Trip VRPTWMD*), onde não somente utilizam-se vários entregadores, mas há também a reutilização de veículos heterogêneos para outras rotas subsequentes. Nesse artigo, o modelo foi alterado para permitir que nem sempre todos os clientes sejam atendidos, utilizando um método de premiação para tentar maximizar o número de clientes atendidos. Foi utilizada uma abordagem híbrida de uma heurística GRASP (*Greedy Randomized Adaptive Search Procedure*) que gera a solução inicial para o GAMS/C-PLEX que foi aplicada em dados de uma empresa real. Os resultados reportados no artigo conseguem reduzir em até 37% os custos da empresa. [Grancy e Reimann \(2016\)](#) comparam duas heurísticas para resolução do VRPTWMD (chamado nesse artigo de VRPTWMS, *VRPTW*

with Multiple Service Workers), o Colônia de Formigas e o GRASP, concluindo que o Colônia de Formigas tem performance muito melhor que o GRASP. [Ferreira e Pureza \(2012\)](#) exploram e comparam algumas heurísticas de economia para o VRPTWMD considerando novamente que nem todos os clientes precisam ser atendidos por uma frota homogênea. [Vega, Munari e Morabito \(2017\)](#) propõem uma abordagem robusta ao VRPTWMD, onde as demandas dos clientes são incertas. O problema é reformulado matematicamente, e resolvido utilizando um método heurístico estendendo a heurística I1 de [Solomon \(1987\)](#). Esse é o único trabalho na literatura que aborda incerteza, os outros são todos determinísticos.

A criação de *clusters* foi tratada por [Grancy \(2015\)](#) e [Grancy e Reimann \(2015\)](#), buscando construir tanto os clusters quanto as rotas de cada veículo. [Grancy e Reimann \(2015\)](#) utilizaram duas heurísticas de construção de *cluster* para então resolver o problema.

2.2 O VRPSPD

O VRPSPD (*Vehicle Routing Problem with Simultaneous Pickup and Delivery*) considera a necessidade de alguns clientes de um serviço de entrega ou coleta de algum produto ([MIN, 1989](#)). O trabalho de [Dethloff \(2001\)](#) demonstra a importância de problemas de coleta e entrega simultâneas para contextos de logística reversa ([ROGERS; TIBBEN-LEMBKE, 2001](#)). Nesses problemas, os produtos precisam ser entregues para os clientes, além de também serem levados de volta para reciclagem. Essa modalidade de problema é bastante aproximada da realidade, especialmente no contexto de entrega de bebidas onde é possível deixar novos engradados e coletar engradados e garrafas vazias para reuso. O PRV com Coleta e Entrega Simultâneas é um problema bem estabelecido na literatura e vários trabalhos já trataram abordagens diferentes para resolvê-lo, como meta-heurísticas para busca local ([SUBRAMANIAN et al., 2010](#)), busca tabu ([MONTANÉ; GALVAO, 2006](#)), enxame de partículas ([GOKSAL; KARAOGLAN; ALTI-PARMAK, 2013](#)) ([KACHITVICHYANUKUL et al., 2009](#)), heurísticas de economia ([ÇATAY, 2010](#)), colônia de formigas ([GAJPAL; ABAD, 2009](#)), entre outras abordagens e extensões, como a adição de múltiplos depósitos ([NAGY; SALHI, 2005](#)). O trabalho de [Berbeglia et al. \(2007\)](#) relata a estrutura geral desses problemas e detalha um sistema de classificação entre eles.

O Capítulo 3 explica em mais detalhes o Problema de Roteamento de Veículo com Múltiplos Entregadores e Coleta e Entrega Simultâneas, e descreve também o modelo de Programação Inteira Mista desenvolvido durante o mestrado.

2.3 *Random Variable Neighborhood Descent*

RVND (*Random Variable Neighborhood Descent*) é uma heurística onde várias vizinhanças são aplicadas ordem aleatória sobre uma solução com o objetivo de atingir um mínimo local ([HANSEN; MLADENOVIC, 2001](#)), ([MLADENOVIC; HANSEN, 1997](#)). A vizinhança de uma

dada solução é o conjunto de soluções atingíveis a partir de uma alteração na solução original. Tem-se, então, para o RVND, uma solução inicial, e um conjunto de diferentes movimentos e alterações aplicáveis sobre esta, através dos quais encontram-se as diversas vizinhanças exploradas pelo algoritmo. O algoritmo, partindo da solução inicial, aplica cada vizinhança nessa solução em ordem aleatória, trocando a solução atual por outra sempre que houver melhora. O algoritmo para, então, quando ao passar por todas as vizinhanças não houver melhora.

As vizinhanças e o funcionamento geral da heurística RVND desenvolvida durante esse trabalho são inspiradas nos trabalhos de [Subramanian *et al.* \(2010\)](#) e [Alvarez e Munari \(2016\)](#). O RVND será demonstrado em maiores detalhes no Capítulo 4.

DEFINIÇÃO DO PROBLEMA E MODELO MATEMÁTICO

O Problema de Roteamento de Veículos Heterogêneos com Múltiplos Entregadores e com Coleta e Entrega Simultâneas (PRVHMECES, ou SPDHVRPMD, *Simultaneous Pickup and Delivery Heterogeneous Vehicle Routing Problem with Multiple Deliverymen*) é uma extensão do VRPTWMD, onde é preciso fazer entregas a clientes que possuem janelas de tempo para entrega e que também estão próximos uns aos outros (*clusterizados*). No VRPTWMD, um caminhão com vários entregadores estaciona em um ponto próximo a alguns clientes, de onde múltiplos entregadores saem e fazem entregas à pé para esses clientes. A adição de outros entregadores na rota é demonstrada na literatura como uma adição positiva ao tempo total operacional (PUREZA; MORABITO; REIMANN, 2012). No SPDHVRPMD, esses clientes possuem ainda demandas de coleta e entrega simultâneas, portanto a carga dos veículos não somente cresce, nem somente decresce, e o veículo deve sair do depósito com toda a carga necessária para entrega. Essa alteração, apesar de parecer pequena, repercute em várias mudanças ao longo do modelo, como pode ser visto na Seção 3.1.

O SPDHVRPMD difere do VRPTWMD ao considerarem-se coleta e entrega simultâneas, e veículos heterogêneos. A inspiração original do VRPTWMD, como explicado anteriormente, era um caso real de uma empresa de entregas de bebidas em São Paulo, onde, por conta do trânsito e por falta de estacionamento facilmente acessível, era preciso que os entregadores estacionassem o seu caminhão em pontos estratégicos e saíssem a pé para fazer entregas a pontos próximos.

É possível encontrar um contexto para o SPDHVRPMD análogo ao contexto do problema original. No mesmo contexto de entrega de bebidas, pode-se facilmente encaixar problemas de logística reversa (ROGERS; TIBBEN-LEMBKE, 2001). Esse tipo de logística foca no fluxo de produtos do consumidor ao produtor, como em problemas de reciclagem, coleta de lixo, consertos, remanufatura. Considerando-se essa necessidade não só de entregar engradados com

bebidas, mas também de coletar engradados com garrafas vazias para reciclagem, percebe-se que o problema é transformado completamente.

A modelagem apresentada a seguir utiliza como base duas fontes principais. A primeira é o artigo de [Pureza, Morabito e Reimann \(2012\)](#), que possui a modelagem do VRP com Janelas de Tempo e Múltiplos Entregadores. A segunda fonte é o artigo de [Kachitvichyanukul et al. \(2009\)](#), que descreve um modelo matemático para o VRPSPD (*Vehicle Routing Problem with Simultaneous Pickup and Delivery*). Além disso, após uma análise das soluções obtidas primariamente, percebeu-se que o problema com veículos homogêneos contém um alto grau de simetria. Caso uma solução utilize 5 veículos de um total de 25, existem $\frac{25!}{20!} = 6,375,600$ soluções alternativas que, na prática, são iguais, porém numericamente diferentes, com índices de veículos diferentes. Para evitar esse problema, foram utilizadas as restrições de quebra de simetria de [Coelho e Laporte \(2013\)](#), que garantem que os veículos sejam escolhidos em ordem crescente relativa aos seus índices e que os índices dos primeiros clientes em cada rota de cada veículo estejam em ordem crescente (daqueles clientes que são os primeiros em cada uma das rotas, a rota que começa com o cliente de menor índice será percorrida pelo veículo 1, a rota que começa com o cliente de segundo menor índice será percorrida pelo veículo 2, assim por diante). Essa análise pode ser estendida para o problema heterogêneo de maneira reduzida, considerando simetrias entre os veículos da mesma classe, homogêneos entre si.

Este capítulo discute um modelo matemático para o SPDHVRPMD, para o qual podem-se considerar algumas desigualdades válidas. O modelo matemático para o problema heterogêneo é apresentado na Seção 3.1. Nesse modelo, não é feita diferenciação entre *clusters* de clientes e os próprios clientes. O problema de agrupar os clientes em *clusters* pode ser visto como um pré-processamento que, para o modelo, já foi feito, mas que não é abordado nesse trabalho. A Seção 3.2 apresenta algumas restrições adicionais. O Capítulo 5 explora os resultados encontrados comparando os modelos obtidos combinando as desigualdades adicionais discutidas na Seção 3.2. O Apêndice B relata também uma análise rápida do caso homogêneo, mais simples.

3.1 Modelagem do SPDHVRPMD

Esta seção descreve o modelo de Programação Inteira Mista desenvolvido nessa dissertação. Esse modelo visa determinar um esquema de roteamento de veículos que serve todos os clientes de modo a minimizar o número de veículos utilizados, a quantidade de entregadores nas rotas, e a distância total percorrida pelos veículos. Os parâmetros desse modelo, suas variáveis, restrições, e função objetivo estão descritas a seguir.

Parâmetros

N : número de clientes mais o depósito, sendo 1 o índice do depósito

K : número de veículos no depósito, ou seja, disponíveis para entrega;

L^k : número máximo de entregadores em um veículo k ;
 M : número de entregadores disponíveis;
 q_i : demanda de entrega do cliente i ;
 p_i : demanda de coleta do cliente i ;
 $[a_i, b_i]$: intervalo de tempo para a realização do serviço no ponto i ;
 tv_{ij} : tempo de viagem entre os pontos i e j ;
 ts_{il} tempo de serviço com l entregadores no cliente i ;
 Q^k : capacidade de carga de cada veículo k ;
 B_{ijl} : um número grande o suficiente;
 c_1^k : custo associado ao uso de cada veículo do tipo k ;
 c_2 : custo associado à distância percorrida por cada veículo;
 c_3 : custo associado ao número de entregadores utilizados em cada veículo.

Além desses parâmetros, define-se $tv_{ij} = \frac{d_{ij}}{vel}$, em que d_{ij} é a distância euclidiana entre i e j e vel é a velocidade média do veículo, convencionada como 1. B_{ijl} é definido conforme estabelecido em [Pureza, Morabito e Reimann \(2012\)](#):

$$B_{ijl} = \max(b_i + ts_{il} + tv_{ij} - a_j, 0) \quad (3.1)$$

Define-se ts_{il} de acordo com o cálculo descrito em [Pureza, Morabito e Reimann \(2012\)](#), garantindo que o tempo de serviço seja diretamente proporcional à carga a ser entregue e coletada e inversamente proporcional ao número de entregadores, onde rs é a taxa de serviço de um entregador:

$$ts_{il} = \frac{1}{l} \min((q_i + p_i) * rs, b_1 - \max(a_i, tv_{1i}) - tv_{i1}) \quad (3.2)$$

Variáveis de decisão

x_{ijkl} : $\begin{cases} 1, & \text{se o veículo } k \text{ sai do ponto } i \text{ para o ponto } j \text{ com } l \text{ entregadores;} \\ 0, & \text{caso contrário.} \end{cases}$
 y_{ilk} : carga total no veículo k , operando com l entregadores, ao deixar o ponto i .
 t_{il} : instante em que se inicia o serviço em i operando com l entregadores.

A seguir, descrevem-se as restrições da modelagem de Programação Inteira Mista do SPDHVRPMD, agrupadas de acordo com suas características, e, ao final, a função objetivo a ser minimizada.

Restrições de fluxo

$$\sum_{\substack{i=1 \\ i \neq j}}^N \sum_{l=1}^{L^k} \sum_{k=1}^K x_{ijlk} = 1, \quad 2 \leq j \leq N; \quad (3.3)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^N \sum_{l=1}^{L^k} \sum_{k=1}^K x_{ijlk} = 1, \quad 2 \leq i \leq N; \quad (3.4)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^N x_{ijlk} = \sum_{\substack{j=1 \\ j \neq i}}^N x_{jilk}, \quad 1 \leq i \leq N, \quad 1 \leq l \leq L^k, \quad 1 \leq k \leq K; \quad (3.5)$$

$$\sum_{j=2}^N \sum_{l=1}^{L^k} x_{1jlk} \leq 1, \quad 1 \leq k \leq K; \quad (3.6)$$

As igualdades (3.3) e (3.4) garantem que um veículo k com l entregadores chega no ponto j e que apenas um veículo k , com l entregadores, parte do ponto i , respectivamente.

As equações (3.5) garantem a continuidade das rotas, ou seja, que se um veículo k operado com l entregadores entrar no nó j então este mesmo veículo deve sair do nó j operando em mesmo modo. Em conjunto, estas restrições garantem o fluxo dos veículos.

As restrições (3.6) limitam que cada veículo seja usado somente uma vez. Essas restrições, as restrições de tempo (3.7), e a restrição quanto ao número de entregadores, restrição (3.11), foram inspiradas no modelo original de [Pureza, Morabito e Reimann \(2012\)](#).

Restrições de tempo

$$t_{jl} \geq t_{il} + (ts_{il} + tv_{ij})x_{ijlk} - B_{ijl} * (1 - x_{ijlk}), \\ 2 \leq i \leq N, \quad 1 \leq j \leq N, \quad i \neq j, \quad 1 \leq l \leq L^k, \quad 1 \leq k \leq K; \quad (3.7)$$

As restrições (3.7) estabelecem os tempos de início de serviço em um nó j se um veículo k operando com l entregadores percorrer a aresta (i, j) , considerando o tempo de viagem de i até j , o tempo de serviço em i com l entregadores e o início do atendimento em i com l entregadores.

Restrições de carga

$$y_{jlk} \geq y_{ilk} + (p_j - q_j)x_{ijlk} - Q^k(1 - x_{ijlk}), \\ 1 \leq i \leq N, \quad 2 \leq j \leq N, \quad i \neq j, \quad 1 \leq l \leq L^k, \quad 1 \leq k \leq K; \quad (3.8)$$

$$y_{jlk} \leq y_{ilk} + (p_j - q_j)x_{ijlk} + Q^k(1 - x_{ijlk}),$$

$$1 \leq i \leq N, \quad 2 \leq j \leq N, \quad i \neq j, \quad 1 \leq l \leq L^k, \quad 1 \leq k \leq K; \quad (3.9)$$

$$y_{1lk} = \sum_{i=1}^N \sum_{\substack{j=1 \\ i \neq j}}^N q_i x_{ijlk}, \quad 1 \leq l \leq L^k, \quad 1 \leq k \leq K; \quad (3.10)$$

As inequações (3.8) e (3.9) atualizam a carga do veículo k quando ele deixa o cliente j após servir o cliente i , com l entregadores, calculando a carga total nova, considerando as demandas de coleta e de entrega em j . As restrições (3.10) exigem que a carga num veículo k ao sair do depósito seja igual a toda carga a ser entregue durante a rota desse veículo k com l entregadores. Essas restrições foram inspiradas no trabalho de [Kachitvichyanukul et al. \(2009\)](#).

Restrição quanto ao número de entregadores

$$\sum_{j=2}^N \sum_{l=1}^{L^k} \sum_{k=1}^K l x_{1jlk} \leq M \quad (3.11)$$

A restrição (3.11) assegura que o número total de entregadores que saem do depósito não seja maior que o número de entregadores disponíveis.

Domínio das variáveis

$$x_{ijlk} \in \{0, 1\}, \quad 1 \leq i \leq N, \quad 1 \leq j \leq N, \quad i \neq j, \quad 1 \leq l \leq L, \quad 1 \leq k \leq K; \quad (3.12)$$

$$y_{ilk} \in \mathbb{Z}^+, \quad 1 \leq i \leq N, \quad 1 \leq l \leq L, \quad 1 \leq k \leq K; \quad (3.13)$$

$$\max\{p_i - q_i, 0\} \leq y_{ilk} \leq Q^k, \quad 1 \leq i \leq N, \quad 1 \leq l \leq L, \quad 1 \leq k \leq K; \quad (3.14)$$

$$t_{il} \in \mathbb{R}^+, \quad 1 \leq i \leq N, \quad 1 \leq l \leq L; \quad (3.15)$$

$$a_i \leq t_{il} \leq b_i, \quad 1 \leq i \leq N, \quad 1 \leq l \leq L; \quad (3.16)$$

As restrições (3.12), (3.13), (3.14), (3.15), (3.16) definem o domínio das variáveis, além dos limites de tempo e cargas máximas e mínimas.

Desta forma, o modelo é escrito da seguinte maneira:

$$\min Z = c_1^k \sum_{j=2}^N \sum_{l=1}^{L^k} \sum_{k=1}^K x_{1jlk} + c_2 \sum_{i=1}^N \sum_{j=1, j \neq i}^N \sum_{l=1}^{L^k} \sum_{k=1}^K d_{ij} x_{ijlk} + c_3 \sum_{j=2}^N \sum_{l=1}^{L^k} \sum_{k=1}^K l x_{1jlk} \quad (3.17)$$

sujeito a: (3.3) ... (3.16)

A função objetivo (3.17) visa minimizar o custo total de operação, em que c_1 é o custo de utilização de um veículo, c_2 é o custo associado à distância total percorrida e c_3 é o custo unitário dos entregadores.

3.2 Desigualdades Válidas

O modelo definido na Seção 3.1 trata o problema SPDHVRPMD - *Simultaneous Pickup and Delivery Heterogeneous Vehicle Routing Problem with Multiple Deliverymen*. Porém, é possível acrescentar algumas restrições ao modelo de modo a apertar o espaço de busca e, possivelmente, obter soluções mais rapidamente quando consideram-se grupos de veículos homogêneos e outras características. Para tanto, nessa seção, apresentamos algumas restrições que podem ser acrescentadas no modelo sem perda de soluções factíveis com, na maioria das vezes, melhoria no desempenho quando usados para resolver o modelo pelos métodos exatos dos resolvidores de problemas inteiros mistos. Essas restrições podem ser acrescentadas tanto ao modelo heterogêneo, quanto ao homogêneo.

3.2.1 Restrições de quebra de simetria

O modelo apresentado da Seção 3.1 pode ser alterado para tratar com problemas de simetria de soluções. Soluções são consideradas simétricas se, apesar de apresentarem valores diferentes, do ponto de vista prático, estas soluções são idênticas. Considere uma situação em que temos 10 veículos e os veículos 4 e 5 sejam idênticos (grupo homogêneo de veículos), logo, não existe diferença entre determinar uma rota com o veículo 4 e determinar outra rota com o veículo 5. Simetrias em problemas de roteamento podem causar situações em que o *solver* já tenha encontrado uma solução ótima, porém ainda existam outras soluções alternativas que precisam ser encontradas antes de poder encerrar a busca. Portanto, lidar com simetria é fundamental para garantir a eficiência do método.

$$\sum_{j=2}^N \sum_{l=1}^{L^k} x_{1jlk} \leq \sum_{j=2}^N \sum_{l=1}^{L^k} x_{1jl,k-1}, \quad k \in K_h \setminus \{0\}; \quad (3.18)$$

$$\sum_{l=1}^{L^k} x_{1jlk} \leq \sum_{m=1}^j \sum_{l=1}^{L^k} x_{1ml,k-1}, \quad k \in K_h \setminus \{0\}, \quad 2 \leq j \leq N; \quad (3.19)$$

Neste trabalho, foram utilizadas as restrições de quebra de simetria de [Coelho e Laporte \(2013\)](#). A restrição (3.18) garante que um veículo k dentro de um conjunto K_h de veículos homogêneos seja utilizado se e somente se o veículo $k - 1$ desse mesmo conjunto K_h já tiver sido utilizado em outra rota. Existe ainda, porém, outra possibilidade de simetria. Por exemplo, caso existam 25 veículos iguais dos quais 10 são selecionados para o roteamento, é possível fazer qualquer permutação entre esses 10 veículos e gerar uma solução numericamente diferente, mas idêntica à original (por conta dos veículos serem iguais). Para esse segundo caso, as restrições (3.19) garantem que os primeiros clientes visitados por cada veículo estejam em ordem. Nesse caso, o veículo 1 pertenceria à rota cujo primeiro cliente tem o menor índice entre os primeiros clientes de cada rota, o veículo 2 pertenceria à rota cujo primeiro cliente possui o segundo menor índice entre os primeiros clientes de cada rota, e assim por diante. Dessa maneira, somente uma solução é possível, a que usa os primeiros k veículos cujos primeiros clientes em suas rotas estejam em ordem crescente.

3.2.2 Outras restrições de carga

As restrições (3.20) e (3.21) são adicionadas ao modelo para buscar melhores limitantes. As restrições (3.10) garantem que um veículo utilizado no roteamento saia do depósito carregado com tudo que é preciso ser entregue. Entretanto, é possível determinar um novo conjunto de restrições baseado nas restrições (3.10) que, ao invés de garantir que o veículo saia do depósito com toda carga entregue, garantem que o veículo chegue ao depósito com toda carga coletada ao longo de sua rota. Essas restrições estão descritas em (3.20) e (3.21).

$$y_{ilk} \geq \sum_{m=1}^N \sum_{\substack{j=2 \\ j \neq i}}^N p_m x_{mjlk} - Q^k (1 - x_{i1lk}); \quad 1 \leq i \leq N, \quad 1 \leq l \leq L^k, \quad 1 \leq k \leq K \quad (3.20)$$

$$y_{ilk} \leq \sum_{m=1}^N \sum_{\substack{j=2 \\ j \neq i}}^N p_m x_{mjlk} + Q^k (1 - x_{i1lk}); \quad 1 \leq i \leq N, \quad 1 \leq l \leq L^k, \quad 1 \leq k \leq K \quad (3.21)$$

Apesar dessas restrições serem redundantes, acrescentá-las ao modelo pode ter um efeito positivo na determinação de limitantes duais, por serem desigualdades válidas. O efeito dessas restrições e das restrições de quebra de simetria (3.18) e (3.19) é analisado em maior detalhe no Capítulo 5, e também no Apêndice B.

HEURÍSTICA DE EXPLORAÇÃO DE VIZINHANÇA RVND

Com o objetivo de obter soluções de boa qualidade para o problema em estudo em tempo computacional reduzido, foi desenvolvida uma heurística RVND baseada no trabalho de [Alvarez e Munari \(2016\)](#) e de [Subramanian *et al.* \(2010\)](#). O funcionamento dessa heurística, bem como as vizinhanças utilizadas e seus algoritmos internos são detalhados nas seções seguintes.

4.1 Random Variable Neighborhood Descent

Uma heurística RVND (*Random Variable Neighborhood Descent*) tem como objetivo, a partir de uma solução factível para um dado problema, encontrar soluções vizinhas, utilizando diversos critérios de seleção, e diversas vizinhanças, melhorando essa solução de acordo com algum objetivo ([HANSEN; MLADENOVIC, 2001](#)), ([MLADENOVIC; HANSEN, 1997](#)).

Denomina-se uma vizinhança o conjunto de soluções alcançáveis a partir da solução original por meio de um mecanismo pré-determinado, geralmente simples e replicável. A vizinhança de uma solução é encontrada, portanto, aplicando-se um algoritmo que transforma a solução original em uma outra solução alternativa, que pode ter ou não valor melhor para o objetivo alcançado.

Para problemas de roteamento, essas vizinhanças são em geral determinadas por movimentos que atuam dentro de uma rota específica da solução (intra-rota), ou entre duas ou mais rotas (inter-rotas). Uma vizinhança intra-rota tenta recombinar uma rota dada de modo a gerar outra rota a partir desta. Uma vizinhança inter-rota tenta combinar rotas, trocar nós, transferir nós, entre outros, a fim de gerar uma solução nova. As vizinhanças utilizadas nesse trabalho foram inspiradas em [Subramanian *et al.* \(2010\)](#) e [Alvarez e Munari \(2016\)](#).

Um algoritmo de RVND funciona de modo que, dada uma solução inicial básica, e um

conjunto de vizinhanças distintas, aplica-se cada vizinhaça, em ordem aleatória, nessa solução, sempre tentando melhorá-la. O algoritmo para quando a solução tiver passado por cada uma das vizinhanças sem melhoria ou quando for atingido o tempo limite.

O Problema de Roteamento de Veículos Heterogêneos com Múltiplos Entregadores e Coleta e Entrega Simultâneas foi resolvido com uma heurística de RVND, de modo a minimizar os custos da função objetivo descrita no Capítulo 3. As próximas Seções detalham o funcionamento desse método.

4.2 Funcionamento Geral do RVND para o SPDHVRPMD

O Algoritmo 1 descreve de modo geral o funcionamento do RVND. Primeiramente, utiliza-se de uma heurística construtiva para gerar a solução inicial do problema, passo 2. Esse algoritmo está descrito na Seção 4.2.1.

Criada a solução original, é aplicada então uma heurística de redução de rotas. Essa heurística é utilizada pois a heurística construtiva tende a gerar soluções com muitos veículos, preenchidos sequencialmente, o que pode significar que a solução encontrada é de pouca qualidade. A heurística de redução de rotas visa diminuir a quantidade de veículos destruindo rotas e realocando seus clientes em outras rotas. É importante notar que é crucial que a solução inicial seja de qualidade, afinal todas as soluções encontradas por meio das vizinhanças partirão desta. Esse procedimento será melhor detalhado na Seção 4.2.2. Após estas etapas, começa de fato o processo da RVND. Para encontrar outras soluções na esperança de melhoria, foram desenvolvidas algumas vizinhanças, que serão exploradas na Seção 4.2.3. As vizinhanças são ordenadas aleatoriamente, Linha 5, e a solução atual passa por cada uma delas.

O laço principal do método RVND é dado pela Linha 7 do Algoritmo 1. Os passos dentro deste se repetem enquanto houver a troca da solução atual por outra. O algoritmo finaliza quando nenhuma vizinhaça encontrar uma solução melhor partindo da solução atual, quando não houver um troca pela solução vinda do algoritmo de perturbação, ou quando o tempo limite for atingido.

Para cada vizinhaça, assim que uma melhoria é encontrada, essa solução substitui a atual, e a mesma vizinhaça é aplicada novamente, até não haver mais melhoria (Linha 11). Quando nenhum movimento possível pela vizinhaça em questão resulta em melhoria, o algoritmo move então para a próxima vizinhaça, encerrando um ciclo quando não há mais vizinhanças para explorar.

Os movimentos das vizinhanças não necessariamente geram soluções factíveis. Portanto, o RVND desenvolvido verifica sempre a factibilidade da solução encontrada pela vizinhaça, rejeitando soluções infactíveis.

Após a aplicação de todas as vizinhanças, na Linha 21, a solução incumbente é trocada pela nova solução, caso seja melhor do que a atual. Caso todas as vizinhanças sejam exploradas

e nenhuma melhoria ocorra, é aplicado um método de perturbação, Linha 25. Esse método, detalhado na Seção 4.2.4, é similar à heurística de redução do número de rotas.

O objetivo da perturbação é gerar uma solução, possivelmente pior, mas que possa no futuro levar a uma solução melhor do que a encontrada atualmente. Caso a solução obtida pela perturbação seja melhor, esta substitui a melhor solução atual e um novo ciclo começa. Caso esta seja igual, ou pior, existe uma chance de 10% dessa solução pior substituir a atual (Linha 26). Mesmo escolhendo uma solução pior, a melhor solução encontrada até então fica guardada até o fim da execução do método, e será a solução final caso o algoritmo nunca encontre uma melhor.

O processo termina quando nenhuma vizinhança melhora a solução atual, a perturbação não resulta numa solução melhor, e ela também não é escolhida para um novo ciclo, ou quando o tempo limite for atingido. A melhor solução encontrada pelo algoritmo é retornada, então, como a solução final do RVND, Linha 32.

Algoritmo 1 – Algoritmo RVND

```

1: procedimento RVND
2:   solucao  $\leftarrow$  ConstruirRotas()
3:   melhorSolucao  $\leftarrow$  solucao
4:   solucao  $\leftarrow$  reducaoDeRotas(solucao)
5:   vizRandList  $\leftarrow$  randomizarVizinhas()
6:   trocouDeSolucao  $\leftarrow$  True
7:   enquanto trocouDeSolucao ou tempo < TempoLimite faça
8:     trocouDeSolucao  $\leftarrow$  False
9:     para cada viz em vizRandList faça
10:      melhora  $\leftarrow$  True
11:      enquanto melhora faça
12:        melhora  $\leftarrow$  False
13:        solucaoVizinha  $\leftarrow$  aplicarVizinha(viz,solucao)
14:        se custo(solucaoVizinha) < custo(solucao) então
15:          solucao  $\leftarrow$  solucaoVizinha
16:          melhora  $\leftarrow$  True
17:          trocouDeSolucao  $\leftarrow$  True
18:        fim se
19:      fim enquanto
20:    fim para
21:    se custo(solucao) < custo(melhorSolucao) então
22:      melhorSolucao  $\leftarrow$  solucao
23:    fim se
24:    se trocouDeSolucao  $\neq$  True então
25:      solucaoPertubada  $\leftarrow$  perturbarSolucao(solucao)
26:      se custo(solucaoPertubada) < custo(solucao) ou chance(0.1) então
27:        solucao  $\leftarrow$  solucaoPertubada
28:        trocouDeSolucao  $\leftarrow$  True
29:      fim se
30:    fim se
31:  fim enquanto
32:  retorna melhorSolucao
33: fim procedimento

```

4.2.1 Heurística Construtiva

O algoritmo RVND necessita de uma heurística construtiva para gerar uma solução inicial de boa qualidade. Esse método constrói cada rota ao adicionar a cada rota de maneira gulosa o cliente não-visitado cuja adição provocará menor variação na carga total, mantendo factibilidade. O pseudo-código para esse método é descrito em Algoritmo 2.

Algoritmo 2 – Algoritmo Construtivo

```

1: procedimento CONSTRUIRROTAS
2:   LoadList  $\leftarrow$  OrdenarClientesMenorImpacto()
3:   Vis  $\leftarrow$  1 ▷ Inicializa Vis com o depósito
4:   para v de 1 até NumV faça ▷ Para cada veículo na frota
5:     route[v][0]  $\leftarrow$  1 ▷ O primeiro valor é o número de entregadores nessa rota
6:     para cada NewCl em LoadList faça
7:       se NewCl não está em Vis então ▷ Se client NewCl não tiver sido visitado ainda
8:         FinalPos  $\leftarrow$  null
9:         NumDel  $\leftarrow$  L[v] + 1 ▷ Número máximo de entregadores no veículo v
10:        Savings  $\leftarrow$  MaxInt
11:        para j de 0 até length(route[v]) faça
12:          fact, del  $\leftarrow$  VerificarFact(NewCl,j,route[v])
13:          se fact então
14:            pos  $\leftarrow$  j
15:            prev  $\leftarrow$  1
16:            se pos  $\neq$  0 então
17:              prev  $\leftarrow$  route[v][pos]
18:            fim se
19:            next  $\leftarrow$  1
20:            se pos  $\neq$  len(route[v] - 1) então
21:              next  $\leftarrow$  route[v][pos + 1]
22:            fim se
23:            sv  $\leftarrow$  dist[prev][NewCl] + dist[NewCl][next] - dist[prev][next]
24:            se sv < Savings ou del < NumDel então
25:              Savings  $\leftarrow$  sv
26:              FinalPos  $\leftarrow$  pos
27:              NumDel  $\leftarrow$  del
28:            fim se
29:          fim se
30:        fim para
31:        se FinalPos  $\neq$  null então ▷ Cliente i pode ser posto com sucesso na rota
32:          route[v]  $\leftarrow$  NewRoute (route[v],NumDel, FinalPos, NewCl)
33:          Vis  $\leftarrow$  Vis  $\cup$  NewCl
34:        fim se
35:      fim se
36:    fim para
37:  retorna route
38: retorna route
39: fim procedimento

```

O Algoritmo 2 detalha os passos para construir uma solução inicial para o HVRPMDSP. Na Linha 2, uma função chamada *OrdenarClientesMenorImpacto()* retorna uma lista com todos os clientes ordenados pelo diferença absoluta entre a carga entregue e a carga coletada, o seu impacto na carga. Portanto, o primeiro cliente em *LoadList* é aquele cuja inclusão na rota causa menos variação na carga. O cálculo do impacto de um cliente é descrito na Equação (4.1).

$$Impacto_{cliente} = |P_{cliente} - Q_{cliente}| \quad (4.1)$$

O laço *para* na Linha 4 olha para cada veículo e constrói sua rota. É importante notar que os veículos estão ordenados do mais barato para o mais caro. Isso significa que a heurística construtiva tenta encher os veículos mais baratos primeiro, e só depois olha para os veículos mais caros.

A Linha 5 garante que o primeiro elemento em *route[v]* refira-se ao número de entregadores naquela rota. Essa informação é crucial para a verificação de factibilidade na Linha 12.

O laço *para* na Linha 11 é chamado sempre que um cliente não tiver sido atribuído a uma rota. Esse laço tenta achar a melhor posição na rota para esse cliente, se essa posição existir.

A Linha 12 chama uma função que verifica se esse novo cliente *NewCl* pode ser inserido na posição *j* nessa rota, e, caso possa, quantos entregadores são necessários para que essa inserção seja factível.

A Linha 23 calcula o custo da distância extra percorrida ao inserir o cliente *NewCl* na posição *j*. A Linha 24 decide se essa inserção é a nova melhor posição possível onde o cliente *NewCl* pode ser inserido. O critério usado é que essa posição deve ser a que causa o menor aumento possível na distância total percorrida do arco (denotado como *Savings*), ou usar menos entregadores. Esse passo garante que, ao final do laço *para*, o método terá encontrado a melhor posição possível em que o cliente *NewCl* pode ser inserido na rota atual, se essa posição existir. Finalmente, se uma posição de inserção tiver sido encontrada, a Linha 32 atualiza a rota, e a Linha 33 atualiza a lista de clientes que já foram visitados para que ele não seja mais considerado em outras rotas.

4.2.2 Heurística de Redução do Número de Rotas

Esse algoritmo foi criado com o intuito de melhorar a solução da heurística construtiva. No melhor dos casos, a solução retornada por esse método será melhor do que a solução da heurística construtiva. No pior dos casos, essa heurística retorna uma solução que utiliza mais veículos, ou com um custo total maior, do que a solução construtiva. Não há garantias de melhoria, portanto antes desse método ser aplicado, a solução construtiva é guardada como a melhor solução atual, como visto no algoritmo da Seção 4.2.

O funcionamento é simples. Cada rota da solução construtiva é destruída, e seus clientes são reinseridos dentro de outras rotas já existentes, seguindo a regra de ordenar os clientes pela seu impacto na carga total ao serem inseridos descrito na Linha 2 do Algoritmo 2. Caso nenhum cliente possa ser reinserido de maneira factível em outra rota, a rota é reconstruída como era

originalmente. Caso alguns clientes possam ser reinseridos em outras rotas, os clientes que sobram vão para uma lista de clientes não-alocados.

Ao final dessa primeira parte do algoritmo, existem algumas rotas cujos clientes não puderam ser rearranjados em outras rotas existentes, e uma lista de clientes avulsos, ainda não alocados, cujas rotas originais foram destruídas mas não foi encontrada nenhuma rota existente onde estes possam ser inseridos de maneira factível. Utilizando então, novamente, o critério de ordenar os clientes pelo seu impacto na carga total, são criadas novas rotas para esses clientes que sobraram, preenchendo primeiro os veículos mais baratos.

A solução retornada por esse algoritmo é usada então para o RVND próprio, cujas vizinhanças são descritas na Seção 4.2.3.

4.2.3 Vizinhanças

Para tratar o SPDHVRPMD, foram adotadas 8 vizinhanças, das quais 2 são intra-rotas, e 6 são inter-rotas. Uma vizinhança intra-rota considera uma rota da solução, e faz um movimento factível nela de modo a transformar essa rota original em outra. Uma vizinhança inter-rota olha para duas ou mais rotas e realiza movimentos factíveis nelas para gerar novas rotas.

4.2.3.1 Vizinhanças Intra-Rotas

Dessa modalidade de vizinhança foram desenvolvidas somente duas, vizinhanças clássicas do VRP, as quais são descritas a seguir:

Or-Opt: Essa vizinhança recebe como entrada uma rota, e um cliente pertencente à mesma. É feita então uma tentativa de inserção desse mesmo cliente em todos os pontos possíveis da rota, retornando para o RVND a posição de inserção de menor custo. Por exemplo, na rota a, b, c, d, e , o cliente c é removido e reinserido na rota, formando a rota a, c, b, d, e , de menor custo. As Figuras 2 e 3 demonstram o funcionamento dessa vizinhança.

Figura 2 – Antes do Or-Opt

Fonte: Autor

2-Opt: Essa vizinhança recebe como entrada uma rota, e tenta quebrar dois arcos dentro da mesma e formar novos arcos, rearranjando os clientes dentro destes novos arcos. Essa

Figura 3 – Depois do Or-Opt

Fonte: Autor

vizinhaça consiste em escolher dois arcos da rota, (i, j) e (i^+, j^+) , formando arcos (i, i^+) e (j, j^+) . O RVND exaustivamente escolhe cada dois arcos sem clientes repetidos e aplica essa vizinhaça, até encontrar uma rota factível e de custo menor à original. Essa vizinhaça está ilustrada nas Figuras 4 e 5.

Figura 4 – Antes do 2-Opt

Fonte: Autor

É interessante notar que vizinhaças intra-rotas, apesar de frequentes na literatura, não possuem muitos benefícios para o SPDHVRPMD. Isso se dá pois o problema possui duas restrições que fazem com que a ordem em que os clientes são visitados importe para a factibilidade, sendo estas janelas de tempo, e coleta e entregas simultâneas. As janelas de tempo são afetadas diretamente pois a ordem de visita faz com que o tempo de chegada e saída de cada cliente mude tanto que talvez a rota não seja mais factível. Coleta e entrega simultâneas fazem com que a carga do veículo não só cresça, ou só diminua. A carga flutua, e, mesmo a carga total de entrega e a carga total de coleta estarem dentro da capacidade do veículo não garante que certas configurações de clientes não façam com que a capacidade estoure durante a rota.

Além disso, vizinhaças intra-rotas tendem a melhorar somente a distância percorrida, raramente permitindo que menos entregadores sejam necessários na rota. Como o custo da

Figura 7 – Depois do mix_routes

Fonte: Autor

shift: Essa vizinhança recebe duas rotas e transfere um cliente i de uma rota para a outra rota de maneira a diminuir os custos das duas. Essa vizinhança pode fundir as duas rotas caso a rota à qual o cliente i pertence só possua 1 cliente. As Figuras 8 e 9 ilustram o funcionamento do **shift**.

Figura 8 – Antes do shift

Fonte: Autor

Figura 9 – Depois do shift

Fonte: Autor

swap(1,1), swap(2,1), swap(2,2): Essas vizinhanças recebem duas rotas e dois parâmetros numéricos. A primeira entrada significa quantos clientes a primeira rota passará para a segunda, e a segunda significa quantos clientes a segunda passará para a primeira. Quando esse parâmetro é 2, os clientes escolhidos precisam necessariamente possuir um arco os ligando diretamente na rota original. As Figuras 10 e 11 exibem o funcionamento do **swap(1,1)**.

Figura 10 – Antes do swap(1,1)

Fonte: Autor

Figura 11 – Depois do swap(1,1)

Fonte: Autor

combine: Essa vizinhança tenta tirar vantagem da heterogeneidade dos veículos. Ela recebe duas rotas de entrada e as combina em uma só, alocando os clientes de ambas as rotas em um único veículo de custo menor ou igual à soma dos custos dos veículos anteriores. O **combine** é demonstrado nas Figuras 12 e 13.

Figura 12 – Antes do combine

Fonte: Autor

Essas vizinhanças possuem maior potencial para melhorias significativas, podendo esvaziar veículos, retirar clientes de certas rotas e os pondo em outras onde o estresse no

Figura 13 – Depois do combine

Fonte: Autor

custo poderia ser menor, diminuindo o número de entregadores. Um ponto importante sobre a vizinhança **combine** é que, infelizmente, dada à explosão combinatorial do esforço de tentar-se achar uma rota que acomode *todos* os clientes de ambas as rotas em um único veículo, foi-se estabelecido por meio de testes computacionais que o tamanho máximo de ambas as rotas somadas não poderia exceder 9 clientes. Isto é importante pois, caso fosse possível que alguns clientes não fossem encaixados nessa nova rota, teria que ser feito ou um reajuste de todas as outras rotas (o que poderia nulificar qualquer ganho) ou a criação de uma ou mais novas rotas contemplando os clientes que sobraram, o que certamente pioraria a solução.

4.2.4 Algoritmo de Perturbação

Durante o desenvolvimento e o processo de testes do RVND, percebeu-se que, até mesmo para instâncias de 100 clientes, havia bastante tendência de serem criadas rotas curtas, com 3 ou 4 clientes custosos, e também de se chegar a um mínimo local próximo do qual talvez houvesse soluções melhores. Tendo em mente essas possíveis soluções melhores ainda não-exploradas, foi feita uma variação do algoritmo de redução do número de rotas, um algoritmo que destrói a solução e a reconstrói quando nenhuma vizinhança consegue alguma melhoria na solução atual.

O funcionamento geral está descrito no Algoritmo 3. Primeiramente, todas as rotas que possuem até três clientes são destruídas (partindo da premissa de que rotas mais longas são melhores, por resultarem em menos veículos). Após esse passo, ordena-se cada cliente de acordo com o seu impacto na carga total (Linha 8). É feita então uma tentativa de encaixar cada cliente removido em uma das rotas existentes (Linha 9), utilizando a ordem de menor impacto determinada anteriormente. No final desse algoritmo, é possível que existam um conjunto de clientes para os quais nenhuma rota possuiu uma inserção factível. Para esses clientes são criadas novas rotas, seguindo o mesmo critério de menor impacto na carga total, utilizando primeiro os veículos mais baratos. Para tal, é feita uma versão alterada do algoritmo construtivo, Algoritmo 2, que recebe uma lista de clientes ordenados e uma solução pré-preenchida, e cria novas rotas para esses clientes nos veículos ainda vazios da solução.

Certamente, não há garantia de que esse algoritmo resultará numa solução que use menos rotas, porém essa solução nova será diferente da solução original e, aplicar as vizinhanças sobre ela pode levar a uma solução melhor do que a original no futuro, o que justifica aceitar, temporariamente, uma solução pior. Vale lembrar, porém, que a melhor solução encontrada pelo algoritmo não é esquecida, e caso o novo mínimo local encontrado não seja melhor que esta, o RVND retornará somente a melhor solução de todas.

Algoritmo 3 – Algoritmo de Perturbação

```

1: procedimento PERTURBARSOLUCAO
2:   clientes  $\leftarrow \{\}$  ▷ Inicializa lista vazia de clientes
3:   para v de 1 até NumV faça
4:     se tam(route[v])  $\leq 4$  então ▷ 3 clientes e o número de entregadores
5:       clientes  $\leftarrow$  ExtrairClientes(route[v])
6:     fim se
7:   fim para
8:   LoadList  $\leftarrow$  OrdenarClientesMenorImpacto(clientes)
9:   route  $\leftarrow$  EncaixarClientes(LoadList, route)
10:  route  $\leftarrow$  ConstruirRotas (LoadList, route)
11:  retorna route
12: fim procedimento
  
```

RESULTADOS COMPUTACIONAIS

Nesse capítulo, a heurística descrita no Capítulo 4 e o modelo definido no Capítulo 3 são usados em conjunto e separadamente para resolver um conjunto de problemas de Roteamento de Veículos Heterogêneos com Múltiplos Entregadores e Coleta e Entrega Simultâneas (SPDHVRPMD, *Simultaneous Pickup and Delivery Vehicle Routing Problem with Multiple Deliverymen*). O objetivo desse estudo é descobrir o comportamento e a performance da heurística RVND, especialmente quando comparada à abordagem exata utilizando somente o *solver* CPLEX a partir do modelo matemático.

Os resultados mostram que a heurística encontra resultados de qualidade consistente em segundos, sem sofrer maiores perdas em performance para instâncias de tamanho grande. Entretanto, a abordagem pura consegue resolver com sucesso instâncias pequenas, porém falha quando a dimensão do problema aumenta, resultando em tempos longos de execução sem encontrar uma única solução factível para o tempo máximo determinado. Esses resultados demonstram de maneira definitiva que o uso da heurística RVND é completamente validado e preferível ao uso do modelo puro, especialmente para instâncias maiores.

5.1 Classes de Veículos e Instâncias

Conforme discutido na introdução desse capítulo, considerar uma frota heterogênea traz o SPDHVRPMD mais próximo à realidade. Uma empresa real muito provavelmente possuirá em sua frota vários tipos diferentes de veículos, o que torna o problema homogêneo uma versão distante da realidade. A heterogeneidade é justificada, portanto, como uma maneira de tornar o problema mais fiel ao cenário real.

É preciso, portanto, que sejam determinadas as classes de veículo consideradas, bem como as diferenças entre cada uma destas. Para isto, escolheu-se definir essas classes conforme o trabalho de Liu e Shen (1999). Neste trabalho, para cada conjunto de instâncias de Solomon,

(SOLOMON, 1987), R1 e R2, C1 e C2, RC1 e RC2, foram definidas classes de veículos, variando em carga. Instâncias R possuem clientes aleatoriamente distribuídos. C possuem clientes clusterizados, e RC possuem clientes tanto aleatoriamente distribuídos, quando clusterizados. Além disso, para cada conjunto de instâncias, foram determinadas também três variações de custo de utilização para cada veículo. Essas variações consistem em veículos de custo c_1 alto (tipo *a*), médio (tipo *b*), e baixo (tipo *c*). No problema original, os custos eram $c_1 = 1$, $c_2 = 0.0001$, e $c_3 = 0.1$, portanto, para manter a proporcionalidade dos custos, os custos c_2 e c_3 foram definidos de acordo com o valor do veículo mais barato daquela instância:

$$c_2 = 0.0001 * \min(c_1) \quad (5.1)$$

$$c_3 = 0.1 * \min(c_1) \quad (5.2)$$

Essas instâncias foram adaptadas de modo a contemplarem variações no número máximo de entregadores que podem estar naquele veículo. Essa abordagem se diferencia da abordagem homogênea, discutida no Apêndice B, pois no problema homogêneo, há duas variações, uma onde todo veículo possui lotação máxima de 1 entregador, e em outra onde essa lotação máxima é de 3 entregadores. Nas Tabelas 1, 2, 3, 4, 5, e 6 exibem-se as adaptações feitas para as instâncias definidas por Liu e Shen (1999).

Finalmente, foi utilizado também o cálculo de tempo de serviço utilizado por Pureza, Morabito e Reimann (2012), descrito na Equação (3.2).

Tabela 1 – Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias R1

Classe	L	Carga	Custos		
			R1a	R1b	R1c
A	1	30	50	10	5
B	2	50	80	16	8
C	2	80	140	28	14
D	3	120	250	50	25
E	3	200	500	100	50

Tabela 2 – Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias C1

Classe	L	Carga	Custos		
			C1a	C1b	C1c
A	1	100	300	60	30
B	2	200	800	160	80
C	3	300	1350	270	135

Tabela 3 – Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias RC1

Classe	L	Carga	Custos		
			RC1a	RC1b	RC1c
A	1	40	60	12	6
B	2	80	150	30	15
C	2	150	300	60	30
D	3	200	450	90	45

Tabela 4 – Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias R2

Classe	L	Carga	Custos		
			R2a	R2b	R2c
A	1	300	450	90	45
B	2	400	700	140	70
C	2	600	1200	240	120
D	3	1000	2500	500	250

Tabela 5 – Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias C2

Classe	L	Carga	Custos		
			C2a	C2b	C2c
A	1	400	1000	200	100
B	2	500	1400	280	140
C	2	600	2000	400	200
D	3	700	2700	540	270

Tabela 6 – Classes de veículos variando e no número máximo de entregadores, carga, e custos alternativos para o conjunto de instâncias RC2

Classe	L	Carga	Custos		
			RC2a	RC2b	RC2c
A	1	100	150	30	15
B	1	200	350	70	35
C	2	300	550	110	55
D	2	400	800	160	80
E	3	500	1100	220	110
F	3	1000	2500	500	250

Vale notar que, para cada um dos tipos de instância, R1, C1, RC1, R2, C2 E RC2, o veículo de maior custo possui a carga máxima equivalente à de um veículo qualquer na sua respectiva instância original de Solomon (1987). Na resolução das instâncias, foi-se considerada uma frota composta por um número igual de veículos de cada classe, onde o número total equivale ao número de clientes da rota. Quando esse número não é divisível pela quantidade de classes de veículos, é feito um arredondamento favorecendo os veículos mais leves. Por exemplo, instâncias de 100 clientes do tipo C1 possuem frota composta de 34 veículos da classe A, 33 da classe B, e 33 da classe C. Finalmente, como pode ser visto nas Tabelas anteriores, os veículos de menor carga possuem menor custo, porém comportam menos entregadores, estimulando um *trade-off* entre o custo, a capacidade do veículo, e a quantidade máxima de entregadores.

São portanto um total de 68 instâncias de tamanhos de 5, 10, 25, 50, e 100 clientes, com 3 custos diferentes, para 8 modelos, somando um total de 1632 instâncias, das quais somente 240 foram resolvidas pelo CPLEX sem ajuda de uma solução do RVND, apresentadas na Seção 5.2.1. As 1392 instâncias restantes estão apresentadas no Apêndice A.

5.2 Testes Computacionais

Para validar o modelo e o método heurístico RVND desenvolvido, as instâncias descritas na Seção 5.1 foram resolvidas com auxílio de um *solver* CPLEX12.8 com um programa escrito em Python implementando o modelo matemático descrito no Capítulo 3, e outro programa escrito em Python implementando o algoritmo RVND descrito no Capítulo 4. Os problemas foram resolvidos em uma máquina com 2 processadores Intel XeonE5-2680v2 de 2.8 GHz com dez núcleos e 128 GB DDR3 1866MHz de memória. Para o CPLEX foi dado um tempo limite de 30 minutos, escolhido de modo a evitar estouro de memória. Para o RVND foi dado também um tempo limite de 30 minutos para ser equivalente ao tempo máximo do método exato. Além disso, houve um pequeno pré-processamento aplicado às variáveis no modelo matemático utilizado pelo CPLEX. Para tal, variáveis X_{ijk} para as quais $a_i + ts_{il} + tv_{ij} > b_j$ foram removidas, pois estes arcos não são possíveis já que as restrições de janela de tempo seriam quebradas se eles fossem ativos.

Foram explorados também 8 modelos alternativos, expostos na tabela 7. Esses modelos são definidos pelas Restrições (3.20) e (3.21), de carga, (3.18), de simetria, e (3.19), também de simetria. Essas restrições são detalhadas no Capítulo 3.

As instâncias foram resolvidas a princípio somente pelo CPLEX com o modelo matemático, sem solução inicial, depois foi utilizado o RVND para gerar uma solução inicial para os problemas. Assim, é possível comparar os resultados alcançados pelo CPLEX no tempo limite, com a solução inicial dada pelo RVND, com a solução encontrada pelo CPLEX com ajuda da solução inicial do RVND. Esse estudo comparativo é feito na Seção 5.2.1.

Tabela 7 – Modelos Alternativos

Modelos	Carga (3.20),(3.21)	Simetria 1 (3.18)	Simetria 2 (3.19)
nosymnoc	Não	Não	Não
sym2noc	Não	Não	Sim
sym1noc	Não	Sim	Não
sym12noc	Não	Sim	Sim
nosymc	Sim	Não	Não
sym2c	Sim	Não	Sim
sym1c	Sim	Sim	Não
sym12c	Sim	Sim	Sim

5.2.1 Resultados Comparativos

Todas as instâncias foram resolvidas inicialmente pelo CPLEX com o modelo matemático. Porém, somente para algumas instâncias foram obtidas soluções factíveis pelo CPLEX, portanto não é possível comparar o comportamento do mesmo para todas as instâncias dentro do tempo limite. As instâncias resolvidas foram as *rcdp0501abc*, *rcdp0504abc*, *rcdp0507abc*, de 5 clientes, *rcdp1001abc*, *rcdp1004abc*, *rcdp1007abc*, de 10 clientes, *rcdp2501abc*, *rcdp2504abc*, *rcdp2507abc*, de 25 clientes, e *rcdp5001abc*, de 50 clientes, seguindo os custos das instâncias RC1, descritos na Tabela 3.

Existem ainda as instâncias *rcdp5004abc*, *rcdp5007abc*, e as instâncias de 100 clientes R1 e R2, C1 e C2, RC1 e RC2, para as quais o CPLEX não conseguiu achar nenhuma solução no tempo limite, nem melhorar a solução dada pelo RVND. Esses resultados são exibidos no Apêndice A.

As Tabelas a seguir mostram os resultados comparativos entre as 3 soluções obtidas (CPLEX, RVND, RVND com CPLEX), para cada modelo alternativo. As Tabelas 8, 9, 10, 11, 12, 13, 14, e 15 exibem os resultados encontrados ao resolver essas instâncias desses três modos. Nessas tabelas existem as seguintes colunas:

- **Instância:** Denota a instância tratada. Por exemplo, a instância *rcdp0504b* é uma das três instâncias de 5 clientes com custo de veículos médio;
- **CPLEX:** Mostra os resultados encontrados pelo CPLEX sem ajuda de qualquer solução inicial;
 - F_1 : Apresenta o valor da função objetivo das soluções encontradas pelo CPLEX;
- **RVND:** Mostra os resultados encontrados pela heurística RVND;
 - F_2 : Apresenta o valor da função objetivo das soluções encontradas pela heurística RVND;

- **CPLEX + RVND:** Mostra os resultados encontrados pelo CPLEX quando provido de uma solução inicial encontrada pelo RVND;
 - F_3 : Apresenta o valor da função objetivo das solução encontradas pelo CPLEX quando provido de uma solução inicial encontrada pelo RVND;
- **NV:** Número de veículos utilizados em cada solução para cada instância de cada abordagem;
- **NE:** Número de entregadores utilizados em cada solução para cada instância de cada abordagem;
- **Dist:** Distância percorrida em cada solução para cada instância de cada abordagem;
- **GAPS:** Expõe os *Gaps* entre F_1 , F_2 , e F_3 , conforme calculado na Equação 5.3;
 - **Gap F_1, F_2 :** Compara as soluções encontradas pelo CPLEX com as encontradas pelo RVND;
 - **Gap F_1, F_3 :** Compara as soluções encontradas pelo CPLEX com as encontradas pelo CPLEX provido de uma solução inicial dada pelo RVND;
 - **Gap F_2, F_3 :** Compara as soluções encontradas pelo RVND com as encontradas pelo CPLEX provido das mesmas como solução inicial.

A Equação 5.3 demonstra o cálculo do *Gap* entre dois valores f e g . Esse tipo de dado nos permite conhecer o quão distante uma solução está da outra. Se esse *Gap* estiver negativo, significa que o valor g é menor do que o valor f , ou seja, a solução encontrada g é melhor que a solução f .

$$GAP_{f,g} = \frac{(g - f)}{g} \quad (5.3)$$

Para os 3 *Gaps* calculados, para concluir-se um bom desempenho da heurística RVND, é preciso que o *Gap F_1, F_2* seja negativo e expressivo na maior parte dos casos (significando que o RVND encontra consistentemente soluções melhores do que o CPLEX). Para o *Gap F_2, F_3* , é desejável que este seja 0%, ou pouco negativo. O motivo disso é simples: caso a solução encontrada pelo RVND seja S_2 , a solução S_3 encontrada pelo CPLEX utilizando S_2 como solução inicial é no mínimo igual à S_2 , podendo haver melhorias. Caso o *Gap F_2, F_3* seja negativo, isso significa que houve uma melhora, portanto para determinar a qualidade da solução inicial oferecida pelo RVND espera-se que S_2 não esteja muito distante de S_3 . Para o *Gap F_1, F_3* , espera-se que este seja 0% ou negativo na maioria dos resultados. Instâncias onde esse *Gap* foi positivo implicam que a solução S_2 dada pelo RVND não só foi pior do que a encontrada pelo CPLEX, S_1 , mas também que ao se utilizar S_2 como solução inicial, S_3 encontrada pelo CPLEX ainda assim é pior do que S_1 . Nesse caso espera-se que S_1 não esteja muito distante de S_3 .

Além disso, é preciso que o tempo de execução do RVND seja menor do que o tempo do CPLEX, justificando o uso exclusivo da heurística. Ao analisar as tabelas comparativas, fica claro que os critérios de bom desempenho são alcançados com sucesso pela heurística RVND.

Tabela 8 – Resultados Comparativos para o modelo nosymnoc

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rdp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rdp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rdp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rdp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rdp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rdp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rdp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rdp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rdp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rdp1001a	524.667	5	7	444.533	704.096	4	7	590.186	524.667	5	7	444.533	25%	0%	-34%
rdp1001b	104.933	5	7	444.533	140.842	4	7	561.403	104.933	5	7	444.533	25%	0%	-34%
rdp1001c	52.467	5	7	444.533	76.446	5	7	544.441	52.467	5	7	444.533	31%	0%	-46%
rdp1004a	457.878	4	6	312.945	451.927	4	5	330.113	362.078	4	5	346.258	-1%	-26%	-25%
rdp1004b	72.416	4	5	346.258	90.454	4	5	451.452	72.451	4	5	375.645	20%	0%	-25%
rdp1004c	36.208	4	5	346.258	45.227	4	5	451.452	45.210	4	5	349.414	20%	20%	0%
rdp1007a	518.381	5	6	396.909	518.381	5	6	405.467	518.381	5	6	396.909	0%	0%	0%
rdp1007b	103.676	5	6	396.909	103.676	5	6	405.467	103.676	5	6	396.909	0%	0%	0%
rdp1007c	51.841	5	6	402.449	51.838	5	6	405.467	51.838	5	6	396.909	0%	0%	0%
rdp2501a	1115.891	11	15	981.883	1240.630	10	16	994.891	1115.257	11	15	876.181	10%	0%	-11%
rdp2501b	242.291	11	16	908.849	245.733	10	14	940.994	221.895	11	14	912.283	1%	-9%	-11%
rdp2501c	111.516	11	15	859.410	124.063	10	16	994.891	104.926	10	14	876.813	10%	-6%	-18%
rdp2504a	2400.282	10	19	1046.980	1132.964	10	13	995.637	1132.964	10	13	827.387	-112%	-112%	0%
rdp2504b	555.379	8	17	816.236	226.593	10	13	995.637	221.925	11	14	937.663	-145%	-150%	-2%
rdp2504c	172.737	9	17	894.952	113.935	10	14	1063.732	113.935	10	14	891.919	-52%	-52%	0%
rdp2507a	1680.191	12	19	1031.859	1368.536	12	17	1338.182	1368.372	12	17	1062.020	-23%	-23%	0%
rdp2507b	304.934	12	18	1111.285	326.197	9	16	1139.709	320.303	10	16	919.487	7%	5%	-2%
rdp2507c	151.797	11	17	994.865	151.799	12	17	1208.111	151.799	12	17	997.885	0%	0%	0%
rdp5001a	2128.276	19	28	1712.614	2433.901	21	29	1952.624	2254.030	21	29	1671.606	13%	6%	-8%
rdp5001b	498.214	23	33	2178.333	474.697	20	29	1910.471	450.922	21	29	1768.273	-5%	-10%	-5%
rdp5001c	232.800	22	31	1999.615	242.793	21	28	1923.291	224.800	21	28	1666.958	4%	-4%	-8%

Tabela 9 – Resultados Comparativos para o modelo sym2noc

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rdp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rdp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rdp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rdp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rdp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rdp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rdp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rdp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rdp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rdp1001a	524.667	5	7	444.533	704.208	4	7	561.403	524.667	5	7	444.533	25%	0%	-34%
rdp1001b	104.933	5	7	444.533	140.842	4	7	561.403	104.933	5	7	444.533	25%	0%	-34%
rdp1001c	52.467	5	7	444.533	70.446	4	7	603.892	52.467	5	7	444.533	26%	0%	-34%
rdp1004a	362.078	4	5	346.258	457.845	4	6	415.828	451.939	4	5	323.163	21%	20%	-1%
rdp1004b	72.416	4	5	346.258	91.569	4	6	415.828	90.464	4	5	386.605	21%	20%	-1%
rdp1004c	36.238	4	5	396.941	45.227	4	5	451.452	36.208	4	5	346.258	20%	0%	-25%
rdp1007a	518.381	5	6	396.909	518.381	5	6	405.467	518.381	5	6	396.909	0%	0%	0%
rdp1007b	103.676	5	6	396.909	107.097	3	4	255.739	103.676	5	6	396.909	0%	0%	0%
rdp1007c	52.399	5	7	331.029	51.838	5	6	405.467	51.838	5	6	396.909	-1%	-1%	0%
rdp2501a	1572.128	12	16	1021.280	1240.630	10	16	994.891	1049.189	10	14	864.770	-27%	-50%	-18%
rdp2501b	209.838	10	14	864.770	248.174	10	16	1053.873	221.927	11	14	939.175	15%	5%	-12%
rdp2501c	121.185	11	16	975.144	125.926	12	14	937.160	104.919	10	14	864.770	4%	-16%	-20%
rdp2504a	3310.758	10	16	792.933	1132.964	10	13	995.637	1132.944	10	13	824.028	-192%	-192%	0%
rdp2504b	435.584	12	17	986.408	238.643	10	13	1000.800	238.643	10	13	869.308	-83%	-83%	0%
rdp2504c	258.036	12	19	1059.705	113.296	10	13	995.637	113.294	10	13	824.039	-128%	-128%	0%
rdp2507a	2197.059	12	20	1176.546	1631.152	10	16	1143.306	1631.152	10	16	858.728	-35%	-35%	0%
rdp2507b	483.810	12	22	1175.007	297.493	10	17	1241.588	297.493	10	17	911.004	-63%	-63%	0%
rdp2507c	194.369	10	18	948.753	154.773	11	17	1209.700	154.772	11	17	953.403	-26%	-26%	0%
rdp5001a	3328.242	21	33	1706.965	2367.450	20	28	1852.946	2367.438	20	28	1573.026	-41%	-41%	0%
rdp5001b	1102.652	23	37	1876.625	473.501	20	28	1853.399	452.104	21	30	1753.316	-133%	-144%	-5%
rdp5001c	382.725	22	31	1874.235	221.706	19	28	1818.678	221.706	19	28	1509.932	-73%	-73%	0%

Tabela 10 – Resultados Comparativos para o modelo sym1noc

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rcdp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rcdp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rcdp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rcdp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rcdp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rcdp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rcdp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rcdp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rcdp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rcdp1001a	524.667	5	7	444.533	704.463	4	7	603.892	524.667	5	7	444.533	26%	0%	-34%
rcdp1001b	104.933	5	7	444.533	140.893	4	7	603.892	104.933	5	7	444.533	26%	0%	-34%
rcdp1001c	52.467	5	7	444.533	76.446	5	7	544.441	52.467	5	7	444.533	31%	0%	-46%
rcdp1004a	607.882	4	6	313.706	451.927	4	5	330.113	391.947	3	5	324.555	-35%	-55%	-15%
rcdp1004b	72.433	4	5	360.773	90.385	4	5	330.113	90.385	4	5	321.165	20%	20%	0%
rcdp1004c	45.800	4	6	333.026	45.785	4	6	415.828	36.208	4	5	346.258	0%	-26%	-26%
rcdp1007a	524.065	5	7	344.219	518.381	5	6	405.467	518.381	5	6	396.909	-1%	-1%	0%
rcdp1007b	103.676	5	6	396.909	103.676	5	6	405.467	103.676	5	6	396.909	0%	0%	0%
rcdp1007c	51.838	5	6	396.909	51.838	5	6	405.467	51.838	5	6	396.909	0%	0%	0%
rcdp2501a	1049.261	10	14	876.813	1240.726	10	16	985.051	1109.117	11	14	852.786	15%	5%	-12%
rcdp2501b	255.428	12	17	856.922	239.895	11	14	1040.227	239.895	11	14	912.799	-6%	-6%	0%
rcdp2501c	104.926	10	14	876.813	119.906	11	14	970.195	119.906	11	14	842.766	12%	12%	0%
rcdp2504a	2351.292	10	16	882.029	1103.329	11	13	1000.204	1103.329	11	13	888.156	-113%	-113%	0%
rcdp2504b	622.407	7	18	672.845	227.877	10	14	1092.042	227.877	10	14	897.623	-173%	-173%	0%
rcdp2504c	310.641	7	17	735.638	119.294	10	13	1020.941	119.294	10	13	823.598	-160%	-160%	0%
rcdp2507a	2190.343	11	19	1057.143	1374.161	12	18	1304.662	1374.121	12	18	1020.239	-59%	-59%	0%
rcdp2507b	409.343	13	20	1119.433	333.582	11	17	1235.957	303.658	11	17	1048.739	-23%	-35%	-10%
rcdp2507c	197.405	13	18	1007.620	161.351	11	18	1178.552	137.419	12	18	1031.190	-22%	-44%	-17%
rcdp5001a	2783.733	23	32	1955.561	2367.507	20	28	1853.399	2307.508	19	28	1584.589	-18%	-21%	-3%
rcdp5001b	484.599	22	32	1832.207	479.418	19	28	1819.774	479.417	19	28	1514.417	-1%	-1%	0%
rcdp5001c	455.870	23	33	1783.599	222.308	19	29	1903.142	222.308	19	29	1513.912	-105%	-105%	0%

Tabela 11 – Resultados Comparativos para o modelo sym12noc

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rcdp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rcdp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rcdp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rcdp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rcdp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rcdp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rcdp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rcdp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rcdp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rcdp1001a	524.667	5	7	444.533	704.208	4	7	561.403	524.667	5	7	444.533	25%	0%	-34%
rcdp1001b	104.933	5	7	444.533	166.119	5	8	580.476	104.933	5	7	444.533	37%	0%	-58%
rcdp1001c	52.467	5	7	444.533	76.446	5	7	544.441	52.467	5	7	444.533	31%	0%	-46%
rcdp1004a	362.124	4	5	353.991	512.456	5	5	409.363	391.870	3	5	311.730	29%	8%	-31%
rcdp1004b	78.376	3	5	313.456	91.569	4	6	415.828	91.569	4	6	307.572	14%	14%	0%
rcdp1004c	36.208	4	5	346.258	51.244	5	5	406.390	39.190	3	5	316.779	29%	8%	-31%
rcdp1007a	524.029	5	7	338.201	518.381	5	6	405.467	518.381	5	6	396.909	-1%	-1%	0%
rcdp1007b	103.692	5	6	410.100	103.676	5	6	405.467	103.676	5	6	396.909	0%	0%	0%
rcdp1007c	52.399	5	7	331.029	51.838	5	6	405.467	51.838	5	6	396.909	-1%	-1%	0%
rcdp2501a	1049.189	10	14	864.770	1235.071	10	15	1034.321	1109.470	11	14	911.657	15%	5%	-11%
rcdp2501b	221.882	11	14	901.885	246.959	10	15	963.004	221.905	11	14	920.494	10%	0%	-11%
rcdp2501c	104.933	10	14	888.823	123.468	10	15	944.415	120.495	11	15	825.639	15%	13%	-2%
rcdp2504a	2988.737	14	22	1122.875	1193.568	11	13	1028.996	1193.568	11	13	928.054	-150%	-150%	0%
rcdp2504b	572.269	10	16	890.564	190.644	10	13	1064.755	190.644	10	13	870.110	-200%	-200%	0%
rcdp2504c	356.983	11	19	972.488	110.333	11	13	1000.204	110.333	11	13	888.156	-224%	-224%	0%
rcdp2507a	1584.571	13	18	1095.203	1668.083	12	17	1169.466	1668.083	12	17	1013.860	5%	5%	0%
rcdp2507b	400.810	11	18	1008.016	316.929	13	18	1324.346	303.722	12	17	1101.823	-26%	-32%	-4%
rcdp2507c	229.318	12	21	1196.655	161.354	11	18	1184.732	161.321	11	18	869.121	-42%	-42%	0%
rcdp5001a	3203.213	21	32	1868.891	2427.975	21	28	1907.013	2314.544	22	29	1757.320	-32%	-38%	-5%
rcdp5001b	1014.245	26	38	2203.977	485.490	20	28	1852.946	485.488	20	28	1573.026	-109%	-109%	0%
rcdp5001c	468.386	18	34	1643.759	236.750	20	28	1852.686	227.765	20	28	1607.715	-98%	-106%	-4%

Tabela 12 – Resultados Comparativos para o modelo nosymc

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rdcp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rdcp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rdcp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rdcp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rdcp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rdcp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rdcp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rdcp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rdcp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rdcp1001a	524.667	5	7	444.533	764.462	5	7	544.441	524.667	5	7	444.533	31%	0%	-46%
rdcp1001b	104.933	5	7	444.533	152.892	5	7	544.441	104.933	5	7	444.533	31%	0%	-46%
rdcp1001c	52.467	5	7	444.533	70.421	4	7	561.403	52.467	5	7	444.533	25%	0%	-34%
rdcp1004a	391.919	3	5	319.903	451.927	4	5	330.113	391.919	3	5	319.903	13%	0%	-15%
rdcp1004b	90.465	4	5	387.591	90.385	4	5	330.113	78.376	3	5	313.456	0%	-15%	-15%
rdcp1004c	45.197	4	5	328.604	45.785	4	6	415.828	39.192	3	5	320.113	1%	-15%	-17%
rdcp1007a	518.381	5	6	396.909	518.381	5	6	405.467	518.381	5	6	396.909	0%	0%	0%
rdcp1007b	103.676	5	6	396.909	103.676	5	6	405.467	103.676	5	6	396.909	0%	0%	0%
rdcp1007c	51.846	5	6	410.100	51.838	5	6	405.467	51.838	5	6	396.909	0%	0%	0%
rdcp2501a	1115.140	11	15	856.613	1228.664	10	14	940.994	1049.271	10	14	878.474	9%	-6%	-17%
rdcp2501b	209.871	10	14	892.111	247.014	10	15	1034.321	209.938	10	14	948.457	15%	0%	-18%
rdcp2501c	121.131	11	16	884.481	122.866	10	14	940.994	104.933	10	14	888.512	1%	-15%	-17%
rdcp2504a	2778.784	12	22	1130.711	1133.739	10	13	1067.955	1104.148	11	13	1024.613	-145%	-152%	-3%
rdcp2504b	388.886	11	18	1071.647	226.620	10	13	1032.854	226.620	10	13	850.377	-72%	-72%	0%
rdcp2504c	204.039	10	19	1065.042	113.344	10	13	1078.018	110.986	11	14	977.256	-80%	-84%	-2%
rdcp2507a	1278.246	12	17	1040.976	1517.987	12	17	1208.111	1278.200	12	17	1033.378	16%	0%	-19%
rdcp2507b	364.955	12	18	1129.125	339.549	11	17	1276.293	273.564	12	17	969.869	-7%	-33%	-24%
rdcp2507c	152.472	12	18	1120.036	163.738	10	17	1190.742	145.792	11	17	985.953	7%	-5%	-12%
rdcp5001a	2777.130	21	31	1854.977	2367.450	20	28	1852.946	2098.370	20	28	1728.339	-17%	-32%	-13%
rdcp5001b	432.866	21	29	1721.690	474.702	20	29	1919.144	419.526	20	28	1605.298	9%	-3%	-13%
rdcp5001c	226.090	21	30	1816.336	237.351	20	29	1919.144	215.796	21	28	1659.951	5%	-5%	-10%

Tabela 13 – Resultados Comparativos para o modelo sym2c

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rdcp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rdcp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rdcp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rdcp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rdcp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rdcp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rdcp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rdcp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rdcp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rdcp1001a	524.667	5	7	444.533	704.208	4	7	561.403	524.667	5	7	444.533	25%	0%	-34%
rdcp1001b	104.933	5	7	444.533	140.893	4	7	603.892	104.933	5	7	444.533	26%	0%	-34%
rdcp1001c	52.467	5	7	444.533	70.421	4	7	561.403	52.467	5	7	444.533	25%	0%	-34%
rdcp1004a	452.269	4	5	378.207	289.713	3	3	285.426	289.713	3	3	285.426	-56%	-56%	0%
rdcp1004b	72.433	4	5	360.773	90.454	4	5	451.452	78.374	3	5	311.730	20%	8%	-15%
rdcp1004c	51.861	5	6	435.169	51.244	5	5	406.390	45.787	4	6	311.148	-1%	-13%	-12%
rdcp1007a	518.415	5	6	402.449	518.381	5	6	405.467	518.381	5	6	396.909	0%	0%	0%
rdcp1007b	104.874	5	7	395.094	103.676	5	6	405.467	103.676	5	6	396.909	-1%	-1%	0%
rdcp1007c	51.838	5	6	396.909	51.838	5	6	405.467	51.838	5	6	396.909	0%	0%	0%
rdcp2501a	1049.227	10	14	871.189	1240.870	10	16	1053.873	1109.523	11	14	920.539	15%	5%	-12%
rdcp2501b	274.803	12	18	1002.359	260.217	11	16	1011.205	209.897	10	14	914.531	-6%	-31%	-24%
rdcp2501c	104.919	10	14	864.770	124.097	10	16	1077.238	110.945	11	14	907.668	15%	5%	-12%
rdcp2504a	2982.657	14	21	1109.494	1103.087	11	13	959.858	1103.087	11	13	847.810	-170%	-170%	0%
rdcp2504b	361.040	9	15	866.884	239.839	10	14	1102.102	239.839	10	14	865.679	-51%	-51%	0%
rdcp2504c	192.638	11	15	1063.639	119.308	10	13	973.686	119.297	10	13	828.263	-61%	-61%	0%
rdcp2507a	2622.953	12	21	1158.854	1547.710	10	17	1284.324	1547.459	10	17	909.874	-69%	-70%	0%
rdcp2507b	460.775	11	18	978.906	338.196	10	16	1115.919	338.194	10	16	828.700	-36%	-36%	0%
rdcp2507c	270.643	12	20	1071.367	145.150	10	16	1157.456	145.150	10	16	916.906	-86%	-86%	0%
rdcp5001a	3328.242	21	33	1706.965	2487.944	21	28	1899.082	2487.943	21	28	1657.139	-34%	-34%	0%
rdcp5001b	748.470	21	32	1725.205	486.780	21	29	1952.624	486.780	21	29	1650.163	-54%	-54%	0%
rdcp5001c	382.725	22	31	1874.235	236.750	20	28	1854.341	230.730	19	28	1550.686	-62%	-66%	-3%

Tabela 14 – Resultados Comparativos para o modelo sym1c

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rcdp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rcdp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rcdp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rcdp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rcdp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rcdp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rcdp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rcdp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rcdp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rcdp1001a	524.667	5	7	444.533	704.208	4	7	561.403	524.667	5	7	444.533	25%	0%	-34%
rcdp1001b	104.933	5	7	444.533	140.842	4	7	561.403	104.933	5	7	444.533	25%	0%	-34%
rcdp1001c	52.467	5	7	444.533	76.446	5	7	544.441	52.467	5	7	444.533	31%	0%	-46%
rcdp1004a	458.067	4	6	344.531	451.927	4	5	330.113	362.078	4	5	346.258	-1%	-26%	-24%
rcdp1004b	72.416	4	5	346.258	102.488	5	5	406.390	102.488	5	5	406.390	29%	29%	0%
rcdp1004c	45.811	4	6	352.238	45.785	4	6	415.828	45.192	4	5	319.362	0%	-1%	-1%
rcdp1007a	518.381	5	6	396.909	518.381	5	6	405.467	518.381	5	6	396.909	12%	12%	0%
rcdp1007b	103.676	5	6	396.909	103.676	5	6	405.467	103.676	5	6	396.909	30%	30%	0%
rcdp1007c	51.838	5	6	396.909	51.838	5	6	405.467	51.838	5	6	396.909	12%	12%	0%
rcdp2501a	1109.366	11	14	894.414	1145.161	10	15	1013.237	1109.341	11	14	890.127	3%	0%	-3%
rcdp2501b	223.057	11	15	881.060	248.126	10	16	994.891	221.935	11	14	945.711	10%	-1%	-12%
rcdp2501c	136.785	12	17	975.289	122.869	10	14	944.617	104.919	10	14	864.770	-11%	-30%	-17%
rcdp2504a	2958.348	10	17	1057.930	1132.992	10	13	1022.793	1132.991	10	13	831.786	-161%	-161%	0%
rcdp2504b	626.310	10	21	925.155	226.687	10	13	1078.018	226.687	10	13	905.998	-176%	-176%	0%
rcdp2504c	476.927	13	19	878.925	119.291	10	13	978.798	119.291	10	13	818.765	-300%	-300%	0%
rcdp2507a	2705.965	12	20	994.184	1518.187	12	17	1247.970	1434.993	13	18	1165.422	-78%	-89%	-6%
rcdp2507b	748.013	15	22	1343.859	339.549	11	17	1276.293	309.517	11	17	930.653	-120%	-142%	-10%
rcdp2507c	254.033	14	22	1388.615	169.098	10	16	1115.919	160.722	11	17	870.139	-50%	-58%	-5%
rcdp5001a	2164.306	21	29	1717.734	2493.935	21	29	1961.722	2164.430	21	29	1738.282	13%	0%	-15%
rcdp5001b	465.471	22	31	1892.633	474.700	20	29	1916.038	452.081	21	30	1733.824	2%	-3%	-5%
rcdp5001c	262.712	24	31	1853.843	236.750	20	28	1854.341	215.790	21	28	1649.195	-11%	-22%	-10%

Tabela 15 – Resultados Comparativos para o modelo sym12c

Instancia	CPLEX				RVND				CPLEX + RVND				GAPS		
	F_1	NV	NE	Dist	F_2	NV	NE	Dist	F_3	NV	NE	Dist	Gap F_1, F_2	Gap F_1, F_3	Gap F_2, F_3
rcdp0501a	289.321	3	3	220.149	289.321	3	3	220.149	289.321	3	3	220.149	0%	0%	0%
rcdp0501b	57.864	3	3	220.149	57.864	3	3	220.149	57.864	3	3	220.149	0%	0%	0%
rcdp0501c	28.932	3	3	220.149	28.932	3	3	220.149	28.932	3	3	220.149	0%	0%	0%
rcdp0504a	229.367	2	3	227.886	229.454	2	3	242.324	229.367	2	3	227.886	0%	0%	0%
rcdp0504b	45.873	2	3	227.886	45.891	2	3	242.324	45.873	2	3	227.886	0%	0%	0%
rcdp0504c	22.937	2	3	227.886	22.945	2	3	242.324	22.937	2	3	227.886	0%	0%	0%
rcdp0507a	445.637	3	4	272.911	475.293	2	4	308.188	445.637	3	4	272.911	6%	0%	-7%
rcdp0507b	89.127	3	4	272.911	95.059	2	4	308.188	89.127	3	4	272.911	6%	0%	-7%
rcdp0507c	44.564	3	4	272.911	47.529	2	4	308.188	44.564	3	4	272.911	6%	0%	-7%
rcdp1001a	524.667	5	7	444.533	764.462	5	7	544.441	524.667	5	7	444.533	31%	0%	-46%
rcdp1001b	104.933	5	7	444.533	140.842	4	7	561.403	104.933	5	7	444.533	25%	0%	-34%
rcdp1001c	52.467	5	7	444.533	70.421	4	7	561.403	52.467	5	7	444.533	25%	0%	-34%
rcdp1004a	362.078	4	5	346.258	512.527	5	5	421.154	452.477	4	5	412.793	29%	20%	-13%
rcdp1004b	78.378	3	5	315.053	90.385	4	5	330.113	78.376	3	5	313.456	13%	0%	-15%
rcdp1004c	45.225	4	5	374.581	45.193	4	5	330.113	36.208	4	5	346.258	0%	-25%	-25%
rcdp1007a	518.381	5	6	396.909	518.381	5	6	405.467	518.381	5	6	396.909	0%	0%	0%
rcdp1007b	103.676	5	6	396.909	103.676	5	6	405.467	103.676	5	6	396.909	0%	0%	0%
rcdp1007c	52.402	5	7	336.569	51.838	5	6	405.467	51.838	5	6	396.909	-1%	-1%	0%
rcdp2501a	1049.189	10	14	864.770	1234.796	10	15	963.004	1049.189	10	14	864.770	15%	0%	-18%
rcdp2501b	209.876	10	14	896.521	248.174	10	16	1053.873	209.838	10	14	864.770	15%	0%	-18%
rcdp2501c	111.588	11	15	979.461	130.100	11	16	1021.176	127.787	12	17	977.833	14%	13%	-2%
rcdp2504a	3683.999	13	18	999.768	1193.547	11	13	1025.461	1193.547	11	13	924.519	-209%	-209%	0%
rcdp2504b	712.722	11	18	935.072	226.580	10	13	995.969	226.580	10	13	816.654	-215%	-215%	0%
rcdp2504c	302.383	9	18	971.819	113.935	10	14	1063.732	113.935	10	14	891.919	-165%	-165%	0%
rcdp2507a	2532.257	13	21	1042.761	1517.912	11	17	1235.957	1517.856	11	17	976.052	-67%	-67%	0%
rcdp2507b	410.709	14	21	1257.116	303.637	12	17	1247.970	303.637	12	17	1031.124	-35%	-35%	0%
rcdp2507c	239.307	9	18	844.507	166.088	8	16	1180.983	166.087	8	16	812.352	-44%	-44%	0%
rcdp5001a	2927.936	22	31	1989.355	2433.949	21	29	1964.828	2379.324	20	30	1553.987	-20%	-23%	-2%
rcdp5001b	609.396	19	31	1829.646	444.617	19	29	1903.142	444.616	19	29	1513.703	-37%	-37%	0%
rcdp5001c	292.794	22	31	1989.355	236.751	20	28	1853.399	236.751	20	28	1584.246	-24%	-24%	0%

Inicialmente não é possível notar tanta vantagem. Para as instâncias de 5 clientes, em todos os modelos foi encontrada pouca ou nenhuma diferença entre as 3 abordagens na maioria das vezes. As exceções são as instâncias *rcdp0507a*, *rcdp0507b*, e *rcdp0507c* onde o CPLEX encontra uma solução 6% melhor do que a solução encontrada pela heurística RVND. Essa diferença aumenta ainda mais para as instâncias de 10 clientes, chegando a uma piora de 37%, na instância *rcdp1001b* da Tabela 11. Não só isso, para o *Gap* F_2, F_3 , a melhora chega a ser -58% para a instância *rcdp1001b* da Tabela 9, significando que a solução encontrada pelo RVND foi cerca de uma vez e meia maior do que a encontrada pelo CPLEX quando utilizando a mesma como solução inicial ($F_2 = 166.119$, $F_3 = 104.933$).

O motivo dessas discrepâncias tão grandes talvez seja devido à convergência muito rápida em um mínimo local pela heurística RVND, ou talvez pela falta de um mecanismo que tenta utilizar veículos mais baratos para rotas já existentes. Isso é evidenciado de maneira clara nesse último exemplo. Tanto a solução do RVND ($F_2 = 107.097$) quanto a do CPLEX utilizando a solução do RVND como solução inicial ($F_3 = 59.097$) utilizaram 3 veículos e 4 entregadores. Fica claro que a diferença existe pois o CPLEX conseguiu de alguma forma escolher veículos mais baratos do que o RVND, quando este não conseguiu sair do mínimo local encontrado.

Outro ponto interessante a ser notado é que, para várias instâncias, o RVND encontra uma solução utilizando menos veículos, porém com custo maior. Isso pode ser visto, por exemplo, na instância *rcdp0507a* da Tabela 10, onde o RVND utiliza 2 veículos, enquanto o CPLEX e o CPLEX com solução inicial do RVND encontram a solução ótima utilizando 3 veículos ($F_1 = F_3 = 445.637$, $F_2 = 475.293$). Um motivo para isso, além da convergência muito rápida para um mínimo local, poderia ser a falta de uma vizinhança que visa partir rotas, utilizar mais veículos, porém de maneira mais inteligente e com menos custos.

Porém, as instâncias de 25 e 50 clientes, maiores e mais complexas, mostram um outro lado. Para estas, a solução encontrada pelo RVND consegue consistentemente ser melhor ou bem próxima da solução encontrada pelo CPLEX e da solução encontrada pelo CPLEX utilizando a mesma como solução inicial. Isso fica evidenciado em várias instâncias, mas mais expressivamente na instância *rcdp2504c* da Tabela 14, onde houve uma melhoria de -300% no *Gap* F_1, F_2 , significando a solução do RVND é 4 vezes menor que a solução encontrada pelo CPLEX ($F_1 = 476.927$, $F_2 = 119.291$). Para esse *Gap*, de um total de 9 instâncias com 25 clientes executadas com 8 modelos, portanto 72 resultados, para 48 a solução encontrada pelo RVND foi melhor ou igual à encontrada pelo CPLEX. Dessas 48, 32 obtiveram melhores de pelo menos -50%.

Analisando o *Gap* F_2, F_3 para as instâncias de 25 clientes, de 72 resultados, 37 foram tais onde o CPLEX não conseguiu melhorar em nada a solução inicial do RVND, e esse *Gap* é 0%. Dos restantes, a maior melhora encontrada pelo CPLEX foi de apenas -24%, na instância *rcdp2501b* da Tabela 13.

Para as instâncias de 50 clientes, vale lembrar que haviam também as instâncias *rcdp5004abc*

e *rcdp5007abc*, mas para as tais o CPLEX não conseguia encontrar soluções factíveis no tempo limite, portanto estas não serão analisadas aqui. Para as instâncias *rcdp5001abc*, o comportamento visto para as instâncias de 25 clientes pode ser visto novamente. Para o *Gap* F_1, F_2 , o pior *Gap* foi 17%, para a instância *rcdp5001a* na Tabela 12. O melhor *Gap* for -133% , na instância *rcdp5001b* da Tabela 9. Para 3 instâncias em 8 modelos, de 24 resultados, em 18 a solução encontrada pelo RVND é melhor do que a encontrada pelo CPLEX.

Para o *Gap* F_2, F_3 , a maior melhora foi de somente -15% , na instância *rcdp5001a* da Tabela 14. Dos 24 resultados obtidos, para 9 o CPLEX não conseguiu melhorar a solução inicial providenciada pelo RVND.

Essas análises de resultados mostram que a heurística RVND retorna soluções de qualidade consistente. Essas soluções conseguem ser consideravelmente melhores do que as soluções encontradas sem a heurística RVND, e, quando estas são utilizadas como soluções iniciais para o CPLEX, a solução final em geral estará próxima da inicial, especialmente para instâncias de maior tamanho. A Tabela 16 mostra o balanço geral dos *Gaps* analisados para todas as instâncias, de 5, 10, 25, e 50 clientes. Em 60% dos resultados, a solução encontrada pelo RVND foi menor ou igual à solução encontrada pelo CPLEX. Em 50.8% dos casos, esse resultado foi igual ao encontrado pelo CPLEX munido da solução inicial do RVND, ou seja, não houve melhora feita pelo CPLEX em cima da solução inicial, ou essa melhora foi de menos de 0.01% entre uma solução e outra. Finalmente, em 90.42% dos resultados a solução encontrada pelo CPLEX usando como solução inicial a solução encontrada pelo RVND foi melhor ou igual à solução encontrada pelo CPLEX. Isso significa que na grande maioria das vezes, aplicar o RVND foi benéfico para o CPLEX.

Tabela 16 – Desempenho do RVND

	$\text{count}(F_2 \leq F_1)$	$\text{count}(F_2 = F_3)$	$\text{count}(F_3 \leq F_1)$
nosymnoc	16	14	27
sym2noc	18	16	27
sym1noc	21	18	27
sym12noc	17	16	24
nosymc	15	10	30
sym2c	21	18	27
sym1c	17	13	27
sym12c	19	17	28
Total	144	122	217
Média entre 240	60%	50.83%	90.42%

A Tabela 17 mostra a quantidade de vezes em que a solução ótima foi encontrada para as instâncias. É possível ver nesta que, apesar do CPLEX encontrar com certa frequência a

Tabela 17 – Otimalidade das soluções

	Cplex	RVND	CPLEX + RVND
nosymnoc	14	3	15
sym2noc	14	3	16
sym1noc	14	3	15
sym12noc	12	3	15
nosymc	14	3	15
sym2c	13	4	16
sym1c	15	3	15
sym12c	14	3	15
Total	110	25	122
Média entre 240	40.74%	10.41%	50.83%

solução ótima, há uma melhora de cerca de 10% na quantidade de soluções ótimas encontradas quando se utiliza a solução inicial dada pelo RVND (que por si só consegue encontrar soluções ótimas para 10.41% das instâncias). Além disso, existem instâncias, como por exemplo todas as instâncias de 100 clientes, para as quais o CPLEX foi completamente incapaz de encontrar uma solução melhor do que a dada pelo RVND, sem conseguir provar a otimalidade desta. Isso sugere que o RVND talvez tenha fornecido mais soluções ótimas, porém, sem a comprovação que estas soluções iniciais são ótimas ou não, é impossível determinar o quão eficaz o RVND é em encontrar essas soluções.

A vantagem de utilizar o RVND é mais evidente ainda quando se considera o tempo de execução, como pode ser visto nas Tabelas 18, 19, e 20. O maior tempo em segundos que a heurística RVND levou para devolver a melhor solução encontrada foi de 120.65 segundos, tendo a sua maior média de execução meros 41.196 segundos. Em contrapartida, o CPLEX, mesmo quando utilizando a solução do RVND como solução inicial, atingiu o tempo limite de 30 minutos em praticamente todas as execuções, com notável exceção da instância *rcdp1004a* do modelo *sym2c*, onde a solução dada pelo RVND foi a ótima, e portanto houve a resolução em apenas 1.81 segundos. Pior ainda, mesmo para instâncias de 5 clientes, o CPLEX demora muito tempo para terminar, chegando a levar 1675 segundos para resolver a instância *rcdp0507b* na modalidade *sym12noc*. Em contraposto, utilizando a solução inicial provida pelo RVND, o CPLEX demorou somente 2.49 segundos. Isso não quer dizer que o CPLEX com o RVND consiga sempre tempos hábeis. Mesmo com essa abordagem o CPLEX ainda consistentemente demora muito, chegando a ter média de tempo de 607.783 segundos para a instância *rcdp0504b*, de somente 5 clientes. Essa discrepância entre os tempos de execução do CPLEX e do RVND é explicada pelo simples fato de que quanto mais clientes, mais complexo o modelo exato é, e mais esforço computacional é necessário para resolver o problema de maneira exata. Para o CPLEX,

Tabela 18 – Tempos de Execução do CPLEX em segundos

Instância	nosymnoc	sym2noc	sym1noc	sym12noc	nosymc	sym2c	sym1c	sym12c	Média
rcdp0501a	<1	<1	1.9	<1	<1	<1	3.51	<1	<1
rcdp0501b	<1	<1	1.83	<1	<1	<1	2.3	<1	<1
rcdp0501c	<1	<1	1.67	<1	<1	<1	1.36	<1	<1
rcdp0504a	476	482	1418	1166	1094	767	1313	378	886.75
rcdp0504b	330	849	1542	1044	1158	1057	510	1373	982.875
rcdp0504c	586	498	1	421	1257	816	1187	1.94	595.952
rcdp0507a	677	1325	761	563	1044	675	562	997	825.5
rcdp0507b	1131	720	574	1675	1037	544	590	755	878.25
rcdp0507c	577	1603	931	1824	848	653	572	974	997.75
rcdp1001a	<1	<1	2.99	<1	<1	<1	<1	6.79	1.46
rcdp1001b	<1	1.99	<1	2.59	<1	<1	2.12	3.87	1.481
rcdp1001c	<1	2.62	<1	<1	<1	<1	<1	3.09	<1
rcdp1004a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp1004b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp1004c	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp1007a	940	1447	TL	TL	1610	1201	659	TL	1475.375
rcdp1007b	883	TL	1108	TL	TL	TL	TL	TL	1718.25
rcdp1007c	1434	TL	1204	TL	TL	895	656	TL	1490.125
rcdp2501a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2501b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2501c	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2504a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2504b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2504c	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2507a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2507b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2507c	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp5001a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp5001b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp5001c	TL	TL	TL	TL	TL	TL	TL	TL	TL

Tabela 19 – Tempos de Execução do RVND em segundos

Instância	nosymnoc	sym2noc	sym1noc	sym12noc	nosymc	sym2c	sym1c	sym12c	Média
rcdp0501a	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0501b	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0501c	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0504a	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0504b	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0504c	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0507a	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0507b	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp0507c	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1001a	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1001b	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1001c	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1004a	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1004b	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1004c	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1007a	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1007b	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp1007c	<1	<1	<1	<1	<1	<1	<1	<1	<1
rcdp2501a	2.07	<1	1.14	6.46	4.36	4.71	7.69	4.98	4.038
rcdp2501b	6.64	3.47	2.89	4.69	2.86	2.63	<1	2.73	3.356
rcdp2501c	<1	3.09	7.34	2.49	3.8	6.62	3.14	4.17	3.941
rcdp2504a	12.93	12.14	30.71	<1	3.16	15.9	77.09	2.44	19.411
rcdp2504b	6.8	19.86	1.1	120.65	42.54	91.07	4.15	20.34	38.313
rcdp2504c	2.57	18.41	18.59	39.62	6.66	19.86	8.07	3.06	14.605
rcdp2507a	5.13	5.07	5.14	16.66	3.42	13.69	5.38	88.73	17.902
rcdp2507b	51.2	1.84	73.75	5.23	<1	54.09	<1	6.52	24.282
rcdp2507c	3.34	1.83	10.15	12.1	<1	32.76	50.43	42.35	19.207
rcdp5001a	29.38	10.75	26.93	32.61	10.84	7.36	2.67	25.9	18.305
rcdp5001b	48.01	37.68	47.59	6.88	30.16	19.24	41.16	54.44	35.645
rcdp5001c	38.73	22.35	34.76	26.72	30.91	89.57	43.12	43.41	41.196

Tabela 20 – Tempos de Execução do CPLEX com RVND em segundos

Instância	nosymnoc	sym2noc	sym1noc	sym12noc	nosymc	sym2c	sym1c	sym12c	Média
rcdp0501a	1.88	<1	1.54	1.7	<1	1.65	<1	<1	<1
rcdp0501b	1.76	<1	<1	1.58	<1	1.55	1.52	1.61	<1
rcdp0501c	<1	<1	<1	1.16	<1	<1	<1	<1	<1
rcdp0504a	578	512	<1	626	266	409	452	<1	355.406
rcdp0504b	478	798	<1	688	1143	954	801	<1	607.783
rcdp0504c	325	725	1.86	394	1143	502	515	1.45	450.913
rcdp0507a	<1	1.66	<1	1.65	<1	2.01	<1	1.68	<1
rcdp0507b	1.42	<1	1.84	2.49	1.29	1.94	<1	<1	1.152
rcdp0507c	1.29	1.93	1.39	2.3	1.4	<1	1.5	1.43	1.413
rcdp1001a	<1	1.61	2.18	3.01	<1	<1	<1	1.99	1.18
rcdp1001b	<1	2.48	<1	1.7	<1	<1	<1	2.81	<1
rcdp1001c	<1	2.33	<1	<1	<1	<1	<1	0.85	<1
rcdp1004a	TL	TL	TL	TL	TL	1.81	TL	TL	1631.976
rcdp1004b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp1004c	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp1007a	1437	1667	TL	TL	TL	TL	1276	TL	1727.625
rcdp1007b	1378	1.79	TL	TL	1618	TL	TL	TL	1678.098
rcdp1007c	1552	TL	TL	TL	1167	TL	1589	TL	1785.125
rcdp2501a	TL	993	TL	TL	TL	TL	TL	TL	TL
rcdp2501b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2501c	TL	TL	TL	TL	TL	TL	1416	TL	TL
rcdp2504a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2504b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2504c	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2507a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2507b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp2507c	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp5001a	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp5001b	TL	TL	TL	TL	TL	TL	TL	TL	TL
rcdp5001c	TL	TL	TL	TL	TL	TL	TL	TL	TL

a diferença entre resolver problemas com 26 clientes e problemas com 50 clientes é astronômica. Entretanto, para o RVND, a diferença entre trabalhar com 25, 50, ou até 100 clientes existe, porém é bem menos expressiva do que para o CPLEX.

Essas evidências claramente demonstram que vale a pena utilizar a heurística RVND, pois a mesma consegue encontrar resultados de qualidade garantida, em segundos.

Finalmente, ainda há as instâncias de 50 clientes *rcdp5004abc* e *rcdp5007abc*, e todas as instâncias de 100 clientes para as quais o CPLEX foi incapaz de encontrar um solução em tempo hábil, mas para as quais a heurística RVND encontrou soluções. Estas são um total de 1392 instâncias para as quais a única solução encontrada é a solução inicial do RVND.

O Apêndice A mostra em detalhe esses resultados. É importante notar que não é possível fazer uma análise de desempenho completa para estes precisamente pela ausência de outros resultados independentes, tanto na literatura quanto resultados encontrados pelo CPLEX. Não há, portanto, análise feita. O fato mais importante a ser constatado para elas é que, sem a heurística RVND, não seria encontrado resultado algum para essas 1392 instâncias. Isso, acompanhado de toda a evidência discutida nesse capítulo, demonstra a importância de utilizar uma heurística para a resolução de problemas dessa complexidade.

Além disso, para os resultados exibidos nesse capítulo, é difícil ver qualquer vantagem ou desvantagem no uso de um tipo de modelo sobre outro. O motivo disto talvez seja pois,

para o problema heterogêneo, as restrições de simetria são aplicadas somente entre os veículos iguais entre si, ou seja, são internas entre as classes de veículos. Isso é feito para que nenhuma solução válida não-simétrica seja cortada acidentalmente ao serem aplicadas essas restrições universalmente. A relação entre esses modelos é melhor ilustrada no Apêndice [B](#), que estuda o caso homogêneo brevemente.

PERSPECTIVAS

Este trabalho teve como objetivo principal estudar uma nova extensão do Problema de Roteamento de Veículos com Janelas de Tempo e Múltiplos Entregadores (VRPTWMD, *Vehicle Routing Problem with Time Windows and Multiple Deliverymen*).

A variação tratada é o *Simultaneous Pickup and Delivery Heterogeneous Vehicle Routing Problem with Multiple Deliverymen* (SPDHVRPMD), Problema de Roteamento de Veículos Heterogêneos com Múltiplos Entregadores e Coleta e Entrega Simultâneas. Nessa variação, não só é tratada a condição de múltiplos entregadores e janelas de tempo, mas também a de coleta e entrega simultâneas e de veículos heterogêneos.

Durante a elaboração desse trabalho, foi feita uma revisão da literatura, onde foi analisado o histórico do VRPTWMD e do Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas, além de uma breve análise da heurística *Random Neighborhood Variable Descent* (RVND). O VRPTWMD é um problema de roteamento inspirado na entrega de bebidas em cidades grandes onde estacionar em cada ponto de entrega pode não ser possível devido ao trânsito ou à falta de estacionamento (PUREZA; MORABITO, 2010). Os autores desse trabalho afirmam também que nesse tipo de cenário os pontos de demanda encontram-se várias vezes próximos uns aos outros, formando grupos, ou *clusters*. Por conta desses *clusters*, é possível planejar uma rota onde o veículo de entrega estaciona em um ponto estratégico (que pode ou não ser um cliente) em cada *cluster*, a partir do qual as entregas são realizadas à pé ou por outro veículo menor. Portanto, acrescentar mais entregadores pode diminuir o tempo gasto em cada *cluster*.

Na revisão da literatura mostrou-se, apesar da existência de vários trabalhos baseados no trabalho de Pureza, Morabito e Reimann (2012), nenhum destes tratou do problema de coletar e entregar produtos simultaneamente, e nem do problema com coleta e entrega simultâneas em conjunto com a heterogeneidade. Para auxiliar na resolução desse problema, foi desenvolvido um modelo de Programação Inteira Mista estendendo o modelo original de Pureza, Morabito e

Reimann (2012) para tratar o caso de coleta e entrega simultâneas e de veículos heterogêneos. Esse modelo foi também baseado no modelo para o Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas de Kachitvichyanukul *et al.* (2009) e nas restrições de quebra de simetria de Coelho e Laporte (2013).

Uma heurística RVND foi desenvolvida também para resolver o SPDHVRPMD. Esse problema, por conta de sua complexidade, demanda o uso de uma heurística rápida e eficaz, que seja capaz de produzir soluções factíveis de boa qualidade em tempo hábil. Essa heurística foi detalhada nesse trabalho e foi empregada na resolução de instâncias pequenas, médias e grandes, de até 100 clientes.

Para verificar o desempenho da heurística tratada, foram utilizadas as instâncias de VRPSPD de Wang e Chen (2012), juntamente com as instâncias de HVRP de Liu e Shen (1999), ambas baseadas nas classes de instâncias de VRP de Solomon (1987). Além disso, foram analisados oito modelos matemáticos alternativos, obtidos pela presença ou pela ausência de restrições adicionais ou seja, das desigualdades válidas. Para cada um dos 8 modelos, cada instância foi resolvida de 3 maneiras: a primeira sendo pelo CPLEX; a segunda pelo RVND; e por último pelo CPLEX auxiliado pela solução inicial dada pelo RVND.

Para as instâncias onde o CPLEX conseguiu achar soluções, os resultados demonstram uma clara vantagem na utilização da heurística, pois a mesma obteve soluções ótimas ou próximas da otimalidade (menos de 0.01% de distância), em tempos de execução curtos, cerca de 50.83% das vezes. Não só isso, o uso conjunto do CPLEX com o RVND conseguiu resultados melhores ou iguais aos do CPLEX 90.42% das vezes. Isso significa que na grande maioria das vezes, utilizar o RVND foi benéfico para o resultado final do CPLEX. Houve também um total de 1392 instâncias, de 50 e 100 clientes, para as quais o CPLEX não conseguiu achar uma solução em tempo hábil, nem conseguiu melhorar a solução inicial dada pelo RVND. Isso reforça ainda mais a vantagem em utilizar a heurística, pois esta consegue sempre encontrar uma solução de qualidade em tempo hábil.

Ao tratar o problema heterogêneo, é difícil ver alguma vantagem ou desvantagem em utilizar um tipo de modelo ou não. Para melhor ilustrar esse comportamento foi feita uma análise de modelo para o caso homogêneo, mais simples. Primeiro, foi considerado o problema com apenas um entregador por veículo, e depois com no máximo três entregadores por veículo, com o objetivo de fazer um comparativo entre os resultados. Os resultados apresentados demonstram que o modelo determina soluções válidas para o problema e também que existe vantagem na adição de mais entregadores nas rotas, conforme concluído no trabalho de Pureza, Morabito e Reimann (2012).

Esse trabalho teve como objetivo explorar o SPDHVRPMD, e sugerir uma heurística concreta que fosse capaz de resolver problemas de quaisquer tamanho. O CPLEX não é capaz de fornecer soluções factíveis para problemas de grande porte, pois o modelo matemático para um problema de 100 clientes é ordens de magnitude maior do que para 50, que é muito maior do

que para 25 clientes. A heurística RVND não sente tão expressivamente esse efeito, e portanto consegue soluções de qualidade comprovada pelos resultados em tempo hábil.

Um trabalho futuro possível seria a expansão dessa heurística para uma heurística mais complexa, melhor adaptada para esse problema específico, ou a criação de várias heurísticas para que o problema possa ser analisado de diversos ângulos. A criação de mais métodos exatos para resolver esse problema também seria interessante, visto que o CPLEX tem dificuldade em resolver instâncias de grande porte.

REFERÊNCIAS

ALVAREZ, A.; MUNARI, P. An exact hybrid method for the vehicle routing problem with time windows and multiple deliverymen. **Computers & Operations Research**, Elsevier, v. 83, p. 1–12, 2017. Citado na página 29.

ALVAREZ, D. A.; MUNARI, P. Abordagens metaheurísticas para o problema de roteamento de veículos com janelas de tempo e múltiplos entregadores. **Gestão e Produção**, SciELO Brasil, v. 23, p. 279–293, 2016. Citado nas páginas 25, 28, 29, 31 e 41.

BERBEGLIA, G.; CORDEAU, J.-F.; GRIBKOVSKAIA, I.; LAPORTE, G. Static pickup and delivery problems: a classification scheme and survey. **Top**, Springer, v. 15, n. 1, p. 1–31, 2007. Citado na página 30.

BRAEKERS, K.; RAMAEKERS, K.; NIEUWENHUYSE, I. V. The vehicle routing problem: State of the art classification and review. **Computers & Industrial Engineering**, Elsevier, v. 99, p. 300–313, 2016. Citado na página 23.

ÇATAY, B. A new saving-based ant algorithm for the vehicle routing problem with simultaneous pickup and delivery. **Expert Systems with Applications**, Elsevier, v. 37, n. 10, p. 6809–6817, 2010. Citado na página 30.

COELHO, L. C.; LAPORTE, G. The exact solution of several classes of inventory-routing problems. **Computers & Operations Research**, Elsevier, v. 40, n. 2, p. 558–565, 2013. Citado nas páginas 34, 39 e 72.

DETHLOFF, J. Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up. **OR-Spektrum**, Springer, v. 23, n. 1, p. 79–96, 2001. Citado nas páginas 24 e 30.

DOLAN, E. D.; MORÉ, J. J. Benchmarking optimization software with performance profiles. **Mathematical programming**, Springer, v. 91, n. 2, p. 201–213, 2002. Citado na página 90.

DORIGO, M.; GAMBARDILLA, L. M. Ant colony system: a cooperative learning approach to the traveling salesman problem. **IEEE Transactions on evolutionary computation**, IEEE, v. 1, n. 1, p. 53–66, 1997. Citado na página 28.

FERREIRA, V. d. O.; PUREZA, V. Some experiments with a savings heuristic and a tabu search approach for the vehicle routing problem with multiple deliverymen. **Pesquisa Operacional**, SciELO Brasil, v. 32, n. 2, p. 443–463, 2012. Citado na página 30.

GAJPAL, Y.; ABAD, P. An ant colony system (acs) for vehicle routing problem with simultaneous delivery and pickup. **Computers & Operations Research**, Elsevier, v. 36, n. 12, p. 3215–3223, 2009. Citado na página 30.

GLOVER, F. Tabu search—part i. **ORSA Journal on computing**, INFORMS, v. 1, n. 3, p. 190–206, 1989. Citado na página 28.

GOKSAL, F. P.; KARAOGLAN, I.; ALTIPARMAK, F. A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. **Computers & Industrial Engineering**, Elsevier, v. 65, n. 1, p. 39–53, 2013. Citado na página 30.

GRANCY, G. S. D.; REIMANN, M. Evaluating two new heuristics for constructing customer clusters in a vrptw with multiple service workers. **Central European Journal of Operations Research**, Springer, v. 23, n. 2, p. 479–500, 2015. Citado na página 30.

_____. Vehicle routing problems with time windows and multiple service workers: a systematic comparison between aco and grasp. **Central European Journal of Operations Research**, Springer, v. 24, n. 1, p. 29–48, 2016. Citado na página 29.

GRANCY, G. S. de. An adaptive metaheuristic for vehicle routing problems with time windows and multiple service workers. **J. UCS**, v. 21, n. 9, p. 1143–1167, 2015. Citado na página 30.

HANSEN, P.; MLADENOVIC, N. Variable neighborhood search: Principles and applications. **European journal of operational research**, Elsevier, v. 130, n. 3, p. 449–467, 2001. Citado nas páginas 30 e 41.

KACHITVICHYANUKUL, V. *et al.* A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. **Computers & Operations Research**, Elsevier, v. 36, n. 5, p. 1693–1702, 2009. Citado nas páginas 30, 34, 37 e 72.

LAPORTE, G. The vehicle routing problem: An overview of exact and approximate algorithms. **European journal of operational research**, Elsevier, v. 59, n. 3, p. 345–358, 1992. Citado na página 23.

LENSTRA, J. K.; KAN, A. Complexity of vehicle routing and scheduling problems. **Networks**, Wiley Online Library, v. 11, n. 2, p. 221–227, 1981. Citado na página 23.

LIU, F.-H.; SHEN, S.-Y. The fleet size and mix vehicle routing problem with time windows. **Journal of the Operational Research society**, Taylor & Francis, v. 50, n. 7, p. 721–732, 1999. Citado nas páginas 25, 55, 56, 72 e 79.

MIN, H. The multiple vehicle routing problem with simultaneous delivery and pick-up points. **Transportation Research Part A: General**, Elsevier, v. 23, n. 5, p. 377–386, 1989. Citado nas páginas 24 e 30.

MLADENOVIC, N.; HANSEN, P. Variable neighborhood search. **Computers & operations research**, Elsevier, v. 24, n. 11, p. 1097–1100, 1997. Citado nas páginas 30 e 41.

MONTANÉ, F. A. T.; GALVAO, R. D. A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. **Computers & Operations Research**, Elsevier, v. 33, n. 3, p. 595–619, 2006. Citado na página 30.

MUNARI, P.; MORABITO, R. A branch-price-and-cut algorithm for the vehicle routing problem with time windows and multiple deliverymen. 2016. Citado na página 29.

NAGY, G.; SALHI, S. Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries. **European journal of operational research**, Elsevier, v. 162, n. 1, p. 126–141, 2005. Citado na página 30.

- NETO, J. F. d. S.; PUREZA, V. Modeling and solving a rich vehicle routing problem for the delivery of goods in urban areas. **Pesquisa Operacional**, SciELO Brasil, v. 36, n. 3, p. 421–446, 2016. Citado na página 29.
- PUREZA, V.; MORABITO, R. Designando entregadores extras no roteamento de veículos com janelas de tempo. **Anais do XLII Simpósio Brasileiro de Pesquisa Operacional**, 2010. Citado nas páginas 23, 27, 28 e 71.
- PUREZA, V.; MORABITO, R.; REIMANN, M. Vehicle routing with multiple deliverymen: Modeling and heuristic approaches for the vrptw. **European Journal of Operational Research**, Elsevier, v. 218, n. 3, p. 636–647, 2012. Citado nas páginas 23, 24, 25, 28, 33, 34, 35, 36, 56, 71, 72, 86 e 87.
- ROGERS, D. S.; TIBBEN-LEMBKE, R. An examination of reverse logistics practices. **Journal of business logistics**, Wiley Online Library, v. 22, n. 2, p. 129–148, 2001. Citado nas páginas 30 e 33.
- SOLOMON, M. M. Algorithms for the vehicle routing and scheduling problems with time window constraints. **Operations research**, Inform, v. 35, n. 2, p. 254–265, 1987. Citado nas páginas 25, 27, 28, 29, 30, 56, 58, 72 e 86.
- SUBRAMANIAN, A.; DRUMMOND, L. M. d. A.; BENTES, C.; OCHI, L. S.; FARIAS, R. A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery. **Computers & Operations Research**, Elsevier, v. 37, n. 11, p. 1899–1911, 2010. Citado nas páginas 25, 30, 31 e 41.
- VEGA, J. D. L.; MUNARI, P.; MORABITO, R. Robust optimization for the vehicle routing problem with multiple deliverymen. **Central European Journal of Operations Research**, Springer, p. 1–32, 2017. Citado na página 30.
- WANG, H.-F.; CHEN, Y.-Y. A genetic algorithm for the simultaneous delivery and pickup problems with time window. **Computers & Industrial Engineering**, Elsevier, v. 62, n. 1, p. 84–95, 2012. Citado nas páginas 25, 72, 86 e 87.

RESULTADOS COMPLEMENTARES

Esse Apêndice contém tabelas registrando os resultados encontrados para as instâncias *rcdp5004abc*, *rcdp5007abc* e as instâncias de 100 clientes R1, C1, RC1, R2, C2 e RC2. São um total de 1392 instâncias, e para nenhuma delas o CPLEX sozinho conseguiu achar uma solução em tempo viável, nem conseguiu melhorar a solução dada pelo RVND. Isso torna a análise desses resultados difícil, pois não há como comparar o comportamento dos modelos nem a qualidade das soluções fornecidas pelo RVND.

No que se refere a esses resultados, a conclusão mais pertinente que pode ser extraída é que a mera existência deles comprova o valor da heurística RVND, pois sem esta não haveria resultados a serem discutidos.

As instâncias a seguir seguem as características de problema expostas no Capítulo 5, das tabelas 1, 2, 3, 4, 5, e 6. Essas tabelas são adaptações feitas para as instâncias definidas por Liu e Shen (1999).

Tabela 21 – Resultados das instâncias de 50 clientes *rcdp5004abc* e *rcdp5007abc*, parte 1

Instância	nosymnoc					sym2noc					sym1noc					sym12noc				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rcdp5004a	2055.276	18	26	1897.760	189.630	2121.269	19	27	1935.578	160.480	2127.717	19	28	2020.681	174.940	2049.411	18	25	1926.861	154.180
rcdp5004b	411.100	18	26	1988.379	226.320	424.268	19	27	1956.841	138.530	423.080	19	26	1857.252	137.850	423.147	19	26	1937.924	89.610
rcdp5004c	205.501	18	26	1857.207	440.990	205.550	18	26	1988.379	127.090	206.170	18	27	2015.913	142.170	193.617	19	26	1985.277	127.570
rcdp5007a	2151.437	18	27	2010.253	52.580	2157.212	18	28	2085.478	83.170	2097.911	20	28	2061.017	50.980	2218.470	19	28	2181.808	15.060
rcdp5007b	418.306	17	27	2121.276	116.870	417.310	20	26	2081.156	394.270	454.443	20	27	2090.664	44.700	443.470	19	28	2025.136	96.010
rcdp5007c	227.228	20	27	2101.601	86.270	221.205	19	27	2099.401	82.240	221.732	19	28	2099.508	50.710	215.159	18	27	2090.980	132.100

Tabela 22 – Resultados das instâncias de 50 clientes *rcdp5004abc* e *rcdp5007abc*, parte 2

Instância	nosymc					sym2c					sym1c					sym12c				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	FO	NV	NE	Dist	Tempo (s)
rcdp5004a	2055.498	18	26	1988.379	196.870	2121.231	19	27	1864.155	96.570	2060.967	18	27	1887.651	268.180	2121.260	19	27	1955.373	71.480
rcdp5004b	411.135	18	26	1962.424	88.560	411.100	18	26	1988.379	241.970	411.080	18	26	1963.694	188.420	412.139	18	27	1804.570	171.260
rcdp5004c	204.941	18	25	1926.861	175.720	205.586	18	26	2015.490	76.910	205.500	18	26	1917.670	240.780	212.728	19	28	1933.730	394.240
rcdp5007a	2211.630	19	27	2065.943	133.910	2211.571	19	27	2049.391	116.390	2097.873	20	28	2110.082	513.990	2157.215	18	28	2145.676	178.930
rcdp5007b	443.456	19	28	2023.628	41.410	417.090	17	26	2011.278	72.080	442.381	19	27	2091.334	44.690	430.262	18	27	2057.001	74.070
rcdp5007c	215.167	18	27	2047.001	24.340	214.545	18	26	2020.342	178.070	203.146	19	27	2109.011	107.920	221.192	19	27	2116.123	142.770

Tabela 23 – Resultados das instâncias de 100 clientes R1, parte 1

Instância	nosymnoc					sym2noc					sym1noc					sym12noc				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rdp101a	3778.229	43	55	2843.755	17.020	3749.083	43	53	2963.322	12.370	3663.509	43	54	2898.315	59.170	3763.441	43	56	2914.548	29.770
rdp101b	752.676	43	56	2899.684	41.760	755.779	44	55	2997.025	21.700	751.719	43	55	2932.252	139.040	745.703	43	55	2916.542	38.310
rdp101c	373.302	43	56	2831.579	43.440	365.367	41	54	2970.227	49.530	376.346	43	56	2918.586	39.570	380.869	44	55	2986.765	21.970
rdp102a	3373.287	41	52	2854.328	12.450	3283.081	39	50	2778.451	14.930	3468.829	43	51	2876.971	8.520	3418.569	42	51	2855.372	12.610
rdp102b	668.704	40	52	2865.929	12.440	676.734	41	50	2856.144	10.320	690.770	43	48	2867.364	15.320	664.660	40	52	2829.855	15.190
rdp102c	327.315	39	48	2765.178	24.650	343.387	42	50	2901.233	11.820	333.787	39	49	2735.987	25.470	345.838	41	51	2823.175	17.330
rdp103a	3072.606	39	48	2641.738	24.110	3118.254	41	49	2758.397	18.900	3063.238	40	48	2798.701	14.47	3103.078	40	50	2745.884	9.48
rdp103b	606.676	40	48	2792.636	13.110	608.636	39	48	2761.201	12.590	617.670	40	47	2768.099	8.83	618.546	40	48	2663.782	17.09
rdp103c	311.343	41	48	2801.302	18.060	306.284	40	48	2668.287	12.180	306.803	40	49	2745.947	14.91	309.821	40	49	2773.686	8.12
rdp104a	3008.055	40	49	2796.029	11.120	3068.650	41	51	2860.790	8.060	3063.653	41	50	2888.639	7.24	3003.082	40	48	2736.075	9.43
rdp104b	618.549	40	50	2698.673	24.130	613.652	41	51	2786.589	8.960	602.740	40	50	2904.143	11.99	610.721	41	48	2825.258	8.69
rdp104c	306.850	41	51	2821.196	10.910	301.274	40	50	2697.120	17.900	306.824	41	51	2799.098	11.19	306.868	41	51	2884.953	8.41
rdp105a	3213.704	40	48	2892.055	14.130	3243.214	40	54	2882.964	14.610	3178.355	39	51	2872.031	8.71	3182.980	39	52	2785.145	24.95
rdp105b	635.651	39	51	2828.996	11.050	655.672	41	51	2828.797	10.590	644.700	40	50	2840.443	12.58	656.781	41	52	2947.084	11.62
rdp105c	323.353	40	52	2869.808	22.520	322.881	40	51	2935.475	14.690	332.825	42	51	2776.005	11.98	327.340	41	50	2803.757	11.2
rdp106a	3253.957	42	54	3023.778	11.200	3244.194	42	52	3020.590	11.320	3189.053	41	51	3020.645	14.39	3199.434	41	53	3124.515	10.58
rdp106b	639.885	41	53	3075.443	10.830	647.822	42	51	3003.384	12.020	651.916	42	55	3147.656	7.02	654.810	41	50	2989.430	24.5
rdp106c	325.421	42	54	3109.713	12.060	314.384	40	52	3008.843	16.880	320.385	41	54	2979.166	13.98	323.923	42	51	2962.419	10.08
rdp107a	3208.813	40	53	2958.146	6.680	3253.268	41	52	2836.142	11.810	3213.723	40	48	2870.437	26.6	3303.937	42	52	2925.926	10.52
rdp107b	671.672	43	53	2813.207	10.550	638.589	40	50	2756.059	15.270	649.724	41	51	2874.716	7.59	650.653	41	52	2842.679	7.76
rdp107c	330.336	42	52	2852.213	9.470	325.318	39	48	2772.114	32.540	323.922	41	49	2962.576	19.25	329.351	42	50	2817.160	8.64
rdp108a	2968.272	39	51	2850.629	10.360	3008.202	40	49	2812.401	9.490	3058.604	41	49	2864.470	10.11	2963.444	39	50	2849.175	10.66
rdp108b	593.847	39	51	3026.767	5.610	602.669	40	50	2795.576	9.940	611.670	41	49	2791.592	9.78	591.799	39	49	2946.504	7.7
rdp108c	296.360	39	50	2877.655	10.740	295.870	39	49	2871.108	10.250	312.879	42	47	2815.971	23.77	304.848	41	47	2808.069	7.36
rdp109a	3229.838	41	53	3192.053	7.990	3229.382	41	53	3108.729	8.000	3228.894	41	53	2986.195	11.53	3218.292	41	51	2831.645	13.7
rdp109b	644.709	41	52	2938.601	11.240	676.769	41	50	2917.505	26.970	653.763	41	49	2920.461	14.29	645.894	41	53	3104.964	61.07
rdp109c	327.379	42	52	2949.027	7.360	346.332	43	50	2779.689	21.500	322.418	41	52	3032.243	8.72	337.283	41	52	2763.539	28.14
rdp110a	3227.921	41	53	2803.076	15.160	3183.466	40	54	2925.413	7.790	3278.500	42	53	2861.599	7.02	3238.834	41	55	3021.615	11.62
rdp110b	654.668	42	52	2852.517	12.270	656.718	42	54	2896.190	10.140	634.649	40	52	2856.986	7.49	646.757	41	54	2983.223	7.18
rdp110c	326.838	42	51	2800.987	9.090	323.311	41	54	2818.081	9.200	322.811	41	53	2812.157	8.12	323.331	41	54	2862.706	7
rdp111a	3468.569	43	51	2867.356	8.490	3317.995	40	51	2776.824	11.410	3423.206	42	52	2818.561	6.32	3468.101	43	51	2746.150	9.94
rdp111b	673.637	41	51	2813.765	10.620	674.663	41	52	2865.893	9.700	675.590	41	53	2791.649	12.14	682.693	42	50	2811.422	7.45
rdp111c	336.800	41	51	2750.988	8.540	336.302	41	50	2736.065	13.210	341.318	42	50	2743.660	9.75	347.331	43	52	2802.619	10.03
rdp112a	3232.454	39	50	2662.551	10.510	3092.419	39	52	2730.694	27.250	3177.715	38	49	2742.646	15.65	3227.095	39	49	2587.842	19.27
rdp112b	627.412	37	51	2666.521	11.880	636.477	38	50	2660.955	10.760	627.659	37	51	2909.915	49.3	608.296	35	52	2626.429	24.03
rdp112c	309.791	39	53	2858.599	28.310	312.713	37	49	2686.451	17.500	312.685	37	49	2593.346	16.86	313.196	37	50	2619.095	21.99

Tabela 24 – Resultados das instâncias de 100 clientes R1, parte 2

Instância	nosymc					sym2c					sym1c					sym12c				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rdp101a	3753.843	43	54	2929.315	48.580	3683.542	42	56	2979.092	115.870	3809.030	44	55	2973.023	98.720	3753.644	43	54	2953.956	18.250
rdp101b	725.608	41	55	2872.240	85.620	745.714	43	55	2923.520	53.760	760.795	44	54	2964.913	23.650	746.768	43	56	3011.332	39.010
rdp101c	375.878	43	55	2975.562	121.520	375.396	44	56	2996.870	26.430	367.855	42	55	2958.820	93.460	380.380	44	54	2933.968	10.070
rdp102a	3332.958	40	50	2714.979	14.420	3318.374	40	51	2891.642	7.300	3393.352	41	52	2835.107	29.090	3263.002	39	50	2782.843	9.460
rdp102b	673.673	41	51	2799.139	9.230	681.764	42	49	2889.559	11.650	685.761	42	53	2937.765	11.370	682.559	40	48	2635.906	48.110
rdp102c	343.867	42	51	2861.942	18.850	339.375	42	46	2820.350	20.790	340.882	42	49	2889.559	12.530	340.356	42	48	2808.971	14.870
rdp103a	3068.424	40	49	2820.875	17.350	3102.549	40	50	2705.217	12.480	3112.912	41	48	2695.001	15.44	3098.026	40	49	2757.029	12.95
rdp103b	627.600	42	49	2719.783	32.130	612.674	40	48	2836.142	10.110	624.699	41	50	2881.448	15.78	606.613	39	46	2744.282	11.87
rdp103c	309.293	40	48	2743.323	12.790	303.322	39	46	2785.872	11.460	308.371	41	48	2894.432	11.98	308.320	41	48	2725.834	17.04
rdp104a	3058.229	41	49	2772.285	13.270	3068.266	41	51	2774.381	10.970	3068.797	41	51	2907.237	11.32	3058.658	41	49	2861.971	8.76
rdp104b	613.646	41	51	2796.334	11.270	604.635	40	52	2840.951	8.600	604.737	40	52	2929.517	22.42	611.497	41	49	2612.126	12.03
rdp104c	300.814	40	49	2725.343	12.070	301.813	40	51	2794.917	11.070	301.833	40	51	2817.299	43.97	301.830	40	51	2788.047	10.39
rdp105a	3278.012	41	51	2747.869	10.370	3283.864	41	52	2975.449	13.010	3243.649	40	54	2943.874	18.33	3218.323	40	49	2814.500	10.87
rdp105b	645.815	40	51	2974.096	11.320	649.685	41	51	2815.682	23.010	646.709	40	52	2895.021	11.49	624.612	38	50	2795.881	22.38
rdp105c	327.860	41	51	2887.818	11.460	312.325	38	50	2875.329	13.270	322.859	40	51	2880.333	15.3	318.342	39	52	2870.215	9.75
rdp106a	3233.885	42	50	2924.458	13.730	3139.271	40	51	3049.944	9.820	3194.534	41	52	3113.166	7.48	3153.835	40	54	3035.176	10.03
rdp106b	647.710	42	51	2864.415	13.380	646.729	42	50	2866.615	17.710	647.692	42	51	2856.184	13.39	648.750	42	52	2976.626	13.88
rdp106c	323.966	42	51	3051.487	10.790	323.507	42	50	3143.501	8.500	319.921	41	53	3044.876	11.91	319.398	41	52	2981.502	12.69
rdp107a	3288.770	42	49	2878.392	9.480	3288.930	42	49	2895.381	8.610	3142.863	39	50	2744.188	11.98	3293.534	42	50	2842.863	6.76
rdp107b	659.646	42	51	2782.725	14.570	647.587	41	49	2721.505	15.860	670.740	43	52	2878.081	8.5	638.609	40	50	2757.540	13.12
rdp107c	329.339	42	50	2786.993	15.680	324.871	41	51	2894.779	7.600	333.839	43	49	2755.918	10.25	329.916	42	51	2921.346	9.64
rdp108a	2948.138	39	47	2739.528	16.950	3002.837	40	48	2677.816	10.760	2968.429	39	51	2879.373	6.64	3062.721	41	50	2672.047	8.27
rdp108b	590.577	39	48	2726.505	12.360	579.616	38	47	2739.052	11.320	591.576	39	49	2710.562	11.4	592.659	39	50	2824.135	7.83
rdp108c	305.803	41	49	2722.652	7.600	300.855	40	49	2857.578	7.540	301.383	40	50	2928.418	9.24	293.267	39	50	2685.632	18.95
rdp109a	3229.653	41	53	3169.794	7.740	3279.160	42	53	3027.188	6.300	3273.630	42	52	2904.343	10.25	3223.730	41	52	2939.942	9.14
rdp109b	655.878	42	53	3070.021	7.060	643.830	41	51	2997.466	8.040	643.831	41	51	2985.103	9.36	654.820	42	52	2992.257	8.09
rdp109c	322.907	41	53	3034.248	7.560	323.372	41	54	2973.626	9.010	322.947	41	53	3104.964	8.51	328.381	42	54	2970.283	7.85
rdp110a	3233.280	41	54	2859.533	6.410	3278.333	42	53	2852.396	9.710	3228.675	41	53	2923.794	10.55	3233.280	41	54	2859.533	6.62
rdp110b	644.632	41	52	2799.677	8.190	653.619	42	51	2746.048	9.250	642.735	41	50	2884.677	8.05	644.701	41	52	2870.609	9.7
rdp110c	323.332	41	54	2883.553	12.870	327.857	42	53	2913.254	10.040	322.329	41	52	2813.019	7.31	318.337	40	54	2918.121	6.4
rdp111a	3423.054	42	52	2791.835	12.510	3413.052	42	50	2724.331	7.530	3423.130	42	52	2797.596	9.29	3469.009	43	51	2929.748	8.18
rdp111b	683.626	41	51	2782.271	9.230	703.712	44	51	2810.635	7.100	661.612	40	49	2709.710	11.83	664.624	40	52	2838.644	10.4
rdp111c	351.316	44	50	2710.597	7.180	341.325	42	50	2780.489	10.700	331.813	40	51	2767.449	14.7	351.787	44	51	2652.913	16.51
rdp112a	3232.797	39	50	2738.221	9.690	3127.649	37	49	2766.472	48.580	3047.510	38	53	2764.469	27.52	3231.942	39	50	2606.043	12.35
rdp112b	635.449	38	49	2658.722	10.340	636.507	38	50	2705.996	12.250	625.431	37	49	2651.888	19.39	635.623	38	49	2833.718	14
rdp112c	312.728	37	49	2685.327	11.780	319.785	41	53	2776.266	14.780	317.733	38	49	2694.342	21.55	312.764	37	49	2770.627	13.79

Tabela 25 – Resultados das instâncias de 100 clientes R2, parte 1

Instância	nosymnoc					sym2noc					sym1noc					sym12noc				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rdp201a	5762.829	11	11	1507.312	33.790	5512.372	11	11	1497.151	11.260	5804.135	11	12	1429.801	19.620	5515.593	11	11	1568.739	7.160
rdp201b	1053.705	10	10	1522.786	15.890	1104.093	11	11	1677.010	7.110	1152.920	11	11	1546.670	13.080	1161.316	11	12	1508.407	15.140
rdp201c	551.201	11	11	1489.190	10.480	580.488	11	12	1471.686	11.520	531.236	10	11	1510.268	14.390	551.205	11	11	1489.956	8.830
rdp202a	5010.875	10	10	1352.769	11.130	5009.343	10	10	1318.728	10.660	5007.630	10	10	1280.672	10.450	5007.507	10	10	1277.942	9.600
rdp202b	1001.088	10	10	1231.950	10.790	1001.915	10	10	1323.895	14.070	953.069	9	9	1341.038	15.750	1001.772	10	10	1307.953	13.890
rdp202c	476.215	9	9	1269.947	16.140	500.836	10	10	1296.928	13.480	500.542	10	10	1231.547	10.420	481.114	9	10	1375.583	18.550
rdp203a	4507.672	9	9	1170.482	9.040	4507.902	9	9	1175.598	9.950	4507.700	9	9	1171.120	10.91	4506.507	9	9	1144.601	10.9
rdp203b	900.981	9	9	1109.055	12.610	901.426	9	9	1158.442	10.520	901.434	9	9	1159.353	9.66	901.818	9	9	1202.054	7.38
rdp203c	450.896	9	9	1199.075	7.690	450.450	9	9	1100.049	17.740	450.674	9	9	1149.871	8.64	450.837	9	9	1185.996	12.54
rdp204a	4001.767	8	8	928.165	15.240	4003.015	8	8	955.888	22.590	4002.674	8	8	948.321	14.41	4006.099	8	8	1024.423	11.19
rdp204b	801.108	8	8	1012.000	13.470	801.325	8	8	1036.121	13.840	800.926	8	8	991.736	14.83	801.282	8	8	1031.369	22.32
rdp204c	400.459	8	8	990.813	10.700	400.327	8	8	961.488	15.030	400.389	8	8	975.238	26.57	400.204	8	8	934.200	21.74
rdp205a	5008.613	10	10	1302.518	7.120	5004.433	10	10	1209.623	16.460	5006.744	10	10	1260.986	9.46	4509.427	9	9	1209.490	9.39
rdp205b	952.145	9	9	1238.327	21.300	1001.685	10	10	1298.337	7.590	902.557	9	9	1284.129	14.33	1001.102	10	10	1233.551	10.42
rdp205c	500.540	10	10	1231.173	10.680	451.409	9	9	1313.164	12.780	451.326	9	9	1294.701	8.53	501.045	10	10	1343.274	14.27
rdp206a	5001.256	10	10	1139.030	14.820	4307.229	8	9	1183.069	20.880	5001.499	10	10	1144.416	16.03	4307.573	8	9	1185.158	16.9
rdp206b	1000.342	10	10	1149.161	15.690	960.390	9	10	1181.151	19.660	900.879	9	9	1097.715	12.71	901.145	9	9	1127.238	18.49
rdp206c	475.612	9	9	1136.036	11.680	500.101	10	10	1133.591	11.740	500.104	10	10	1134.257	18.03	450.611	9	9	1135.737	18.45
rdp207a	4009.895	8	8	1108.786	9.400	4005.180	8	8	1003.999	15.890	4006.802	8	8	1040.049	10.37	4007.178	8	8	1048.404	15.9
rdp207b	801.489	8	8	1054.361	18.670	801.246	8	8	1027.342	12.190	801.766	8	8	1085.085	11.7	801.358	8	8	1039.771	13.07
rdp207c	400.701	8	8	1044.751	15.340	400.844	8	8	1076.550	13.510	400.794	8	8	1065.326	9.53	400.432	8	8	984.961	15.62
rdp208a	4000.644	8	8	903.200	39.150	4003.018	8	8	955.955	24.500	4001.747	8	8	927.707	17.02	4002.034	8	8	934.086	18.72
rdp208b	800.250	8	8	916.661	14.570	800.926	8	8	991.769	21.230	800.909	8	8	989.887	22.25	800.149	8	8	905.442	11.81
rdp208c	400.010	8	8	891.034	21.400	400.157	8	8	923.687	23.390	400.203	8	8	934.076	13.62	400.204	8	8	934.330	49.01
rdp209a	4507.452	9	9	1165.611	12.240	4506.998	9	9	1155.509	9.340	4510.883	9	9	1241.843	14.01	4511.187	9	9	1248.604	8.98
rdp209b	901.877	9	9	1208.595	11.410	901.766	9	9	1196.265	7.910	901.915	9	9	1212.729	7.49	902.271	9	9	1252.309	12.98
rdp209c	450.918	9	9	1203.920	11.930	450.923	9	9	1205.101	7.650	450.943	9	9	1209.509	10.29	450.957	9	9	1212.730	6.79
rdp210a	4758.083	9	9	1179.617	9.300	5002.053	10	10	1156.728	10.410	5006.228	10	10	1249.505	7.24	4762.087	9	9	1268.610	21.48
rdp210b	901.847	9	9	1205.246	9.160	1000.830	10	10	1203.382	11.800	1000.561	10	10	1173.494	15.9	951.941	9	9	1215.697	13.09
rdp210c	476.125	9	9	1249.942	9.590	500.458	10	10	1212.930	10.760	500.243	10	10	1165.048	12.34	500.502	10	10	1222.741	10.68
rdp211a	4501.893	9	9	1042.070	14.870	4501.434	9	9	1031.871	15.190	4506.500	9	9	1144.434	18.61	4501.539	9	9	1034.199	15.71
rdp211b	900.441	9	9	1049.020	11.350	900.448	9	9	1049.806	13.650	900.601	9	9	1066.795	7.38	900.565	9	9	1062.831	12.86
rdp211c	450.456	9	9	1101.307	11.050	450.200	9	9	1044.337	15.620	450.281	9	9	1062.484	12.59	450.221	9	9	1049.005	14.14

Tabela 26 – Resultados das instâncias de 100 clientes R2, parte 2

Instância	nosymc					sym2c					sym1c					sym12c				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rdp201a	5764.336	11	11	1540.793	7.370	5316.239	10	11	1611.239	20.910	5316.239	10	11	1611.239	22.650	5763.887	11	11	1530.812	13.060
rdp201b	1103.994	11	11	1665.946	8.090	1151.967	11	11	1440.737	10.660	1102.197	11	11	1466.337	14.140	1161.379	11	12	1513.257	12.130
rdp201c	580.701	11	12	1508.412	20.630	580.565	11	12	1472.199	13.930	581.107	11	12	1605.994	14.110	580.509	11	12	1472.328	9.510
rdp202a	5009.019	10	10	1311.544	8.710	5008.767	10	10	1305.932	8.820	5009.516	10	10	1322.580	9.830	5009.612	10	10	1324.710	12.870
rdp202b	1002.151	10	10	1350.094	9.660	1002.108	10	10	1345.279	13.350	1002.179	10	10	1353.179	13.000	1002.070	10	10	1341.139	12.910
rdp202c	501.012	10	10	1336.081	9.180	500.756	10	10	1279.085	9.840	476.313	9	9	1291.769	20.740	476.278	9	9	1284.100	17.460
rdp203a	4508.404	9	9	1186.762	9.840	4509.162	9	9	1203.611	11.870	4506.683	9	9	1148.519	13.42	4506.054	9	9	1134.527	10.920
rdp203b	901.131	9	9	1125.700	21.830	901.380	9	9	1153.330	10.370	901.250	9	9	1138.878	10.92	901.120	9	9	1124.498	16.17
rdp203c	450.693	9	9	1154.004	13.220	450.638	9	9	1141.831	8.980	450.762	9	9	1169.298	14.49	450.515	9	9	1114.540	15.06
rdp204a	4002.076	8	8	935.015	22.230	4009.734	8	8	1105.192	15.550	4002.830	8	8	951.782	15.32	4005.983	8	8	1021.849	9.64
rdp204b	800.539	8	8	948.761	9.790	800.874	8	8	985.951	19.850	800.930	8	8	992.239	13.19	800.951	8	8	994.539	17.77
rdp204c	400.328	8	8	961.813	25.620	400.612	8	8	1024.812	15.370	400.280	8	8	951.120	18.83	400.202	8	8	933.679	19.7
rdp205a	5004.931	10	10	1220.679	8.710	5008.362	10	10	1296.942	9.320	5004.307	10	10	1206.816	15.12	4762.130	9	9	1269.548	11.3
rdp205b	960.493	9	10	1192.632	16.310	1001.097	10	10	1233.030	7.570	1000.803	10	10	1200.358	9.4	1000.921	10	10	1213.441	16.21
rdp205c	500.713	10	10	1269.572	12.580	500.772	10	10	1282.591	7.270	500.457	10	10	1212.729	9.06	500.292	10	10	1176.035	15.45
rdp206a	4759.255	9	9	1205.677	18.630	4755.121	9	9	1113.804	15.260	4801.783	9	10	1180.438	11.82	5004.695	10	10	1215.440	11.83
rdp206b	1000.220	10	10	1135.561	14.510	951.218	9	9	1135.304	14.820	862.303	8	9	1284.803	17.43	1000.236	10	10	1137.360	13.48
rdp206c	500.528	10	10	1228.543	11.750	500.055	10	10	1123.342	19.780	500.181	10	10	1151.270	10.5	499.989	10	10	1108.767	18.38
rdp207a	4005.942	8	8	1020.927	16.070	4007.506	8	8	1055.691	19.230	4011.424	8	8	1142.747	19.22	4007.045	8	8	1045.455	18.21
rdp207b	801.208	8	8	1023.125	18.340	801.537	8	8	1059.713	16.000	801.650	8	8	1072.230	10.89	801.226	8	8	1025.107	11.54
rdp207c	400.886	8	8	1085.689	29.460	401.044	8	8	1120.834	10.860	400.507	8	8	1001.572	44.45	400.822	8	8	1071.570	15.04
rdp208a	4002.605	8	8	946.774	13.350	4002.284	8	8	939.655	15.150	4001.032	8	8	911.818	18.86	4003.749	8	8	972.202	14.56
rdp208b	800.297	8	8	921.890	14.120	800.242	8	8	915.827	14.580	800.072	8	8	896.855	16.22	800.526	8	8	947.279	16.83
rdp208c	400.307	8	8	957.153	23.650	400.228	8	8	939.575	15.580	400.126	8	8	916.916	31.95	400.115	8	8	914.367	13.86
rdp209a	4509.254	9	9	1205.638	13.460	4509.456	9	9	1210.125	12.910	4507.745	9	9	1172.118	10.23	4507.804	9	9	1173.414	15.3
rdp209b	902.543	9	9	1282.527	11.130	901.750	9	9	1216.714	13.820	902.412	9	9	1268.027	9.95	902.087	9	9	1231.941	12.34
rdp209c	951.099	9	9	1244.154	12.870	450.770	9	9	1171.089	12.630	450.894	9	9	1198.735	17.76	450.879	9	9	1195.329	15.12
rdp210a	5001.482	10	10	1144.038	17.210	4082.653	9	10	1199.784	15.230	4763.128	9	9	1291.731	18.47	5003.407	10	10	1186.832	10.22
rdp210b	951.797	9	9	1199.686	10.050	1001.249	10	10	1249.933	7.110	1001.500	10	10	1271.744	9.42	1001.076	10	10	1230.701	10.87
rdp210c	450.945	9	9	1210.030	8.940	500.754	10	10	1278.571	10.480	451.048	9	9	1232.896	16.85	500.556	10	10	1234.620	11.48
rdp211a	4502.629	9	9	1058.416	17.450	4501.991	9	9	1044.246	7.830	4502.435	9	9	1054.104	14.56	4504.616	9	9	1102.581	18.4
rdp211b	900.596	9	9	1066.180	19.040	900.539	9	9	1059.885	6.620	900.569	9	9	1063.170	21.45	900.574	9	9	1063.728	16.98
rdp211c	450.413	9	9	1091.674	38.340	450.338	9	9	1075.008	11.110	450.204	9	9	1045.386	15.11	450.363	9	9	1080.730	18.55

Tabela 27 – Resultados das instâncias de 100 clientes C1, parte 1

Instância	nosymnoc					sym2noc					sym1noc					sym12noc				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
cdp101a	7314.219	22	22	1807.284	30.360	7314.193	22	22	1806.443	34.170	7312.522	22	22	1750.724	91.030	7315.595	22	22	1853.182	35.890
cdp101b	1462.848	22	22	1808.039	43.160	1462.755	22	22	1792.461	69.300	1462.195	22	22	1699.180	39.920	1462.714	22	22	1785.687	36.690
cdp101c	731.098	22	22	1699.180	38.570	731.436	22	22	1812.103	34.120	731.170	22	22	1723.236	44.480	731.506	22	22	1835.434	51.530
cdp102a	6652.882	20	20	1762.722	16.040	6654.521	20	20	1817.375	18.620	6654.799	20	20	1826.628	23.610	6656.061	20	20	1868.688	23.570
cdp102b	1330.526	20	20	1754.292	17.430	1330.307	20	20	1717.815	31.470	1330.716	20	20	1785.939	19.430	1329.965	20	20	1660.861	36.360
cdp102c	665.368	20	20	1789.378	22.930	665.052	20	20	1683.920	28.390	665.544	20	20	1848.067	23.290	665.081	20	20	1697.785	35.640
cdp103a	6979.134	21	21	1637.807	30.420	6977.662	21	21	1588.728	41.980	6979.462	21	21	1648.730	35.4	6978.395	21	21	1613.164	36.23
cdp103b	1396.129	21	21	1688.097	44.220	1395.915	21	21	1652.553	50.150	1395.747	21	21	1624.552	25.89	1395.726	21	21	1621.003	45.66
cdp103c	697.802	21	21	1600.791	28.040	697.844	21	21	1614.811	40.640	697.693	21	21	1564.294	27.25	698.148	21	21	1715.926	28.36
cdp104a	6648.231	20	20	1607.697	26.650	6979.712	21	21	1657.079	40.290	6646.267	20	20	1542.232	25.26	6979.036	21	21	1634.518	31.98
cdp104b	1395.594	21	21	1598.974	67.740	1395.515	21	21	1585.800	39.960	1395.765	21	21	1627.504	38.09	1395.338	21	21	1556.259	43.19
cdp104c	698.050	21	21	1683.229	29.540	697.988	21	21	1662.650	24.580	697.625	21	21	1541.519	42.03	697.875	21	21	1625.129	27.89
cdp105a	6987.060	21	21	1901.996	44.370	6986.358	21	21	1878.593	30.400	6985.394	21	21	1846.455	28.54	6987.412	21	21	1913.744	41.47
cdp105b	1397.263	21	21	1877.113	26.290	1397.544	21	21	1924.059	30.180	1397.878	21	21	1979.608	17.61	1397.367	21	21	1894.483	46.46
cdp105c	698.833	21	21	1944.500	37.580	698.989	21	21	1996.192	25.080	698.709	21	21	1902.838	32.31	698.668	21	21	1889.407	34.58
cdp106a	6657.039	20	20	1901.290	21.530	6655.870	20	20	1862.318	27.420	6660.392	20	20	2013.074	16.25	6655.378	20	20	1845.949	36.8
cdp106b	1331.607	20	20	1934.452	33.750	1332.770	20	20	2128.354	27.130	1331.004	20	20	1833.992	21.21	1331.022	20	20	1837.070	28.37
cdp106c	665.739	20	20	1913.136	36.470	665.573	20	20	1857.538	37.180	665.667	20	20	1889.123	35.14	665.456	20	20	1818.626	27.97
cdp107a	6982.885	21	21	1762.845	25.110	6983.520	21	21	1784.010	38.230	6982.224	21	21	1740.801	24.36	6983.055	21	21	1768.506	28.06
cdp107b	1396.943	21	21	1823.892	27.170	1397.287	21	21	1881.203	22.770	1396.753	21	21	1792.084	26.15	1397.035	21	21	1839.117	29.7
cdp107c	698.532	21	21	1844.058	48.530	698.226	21	21	1741.949	34.660	698.509	21	21	1836.466	27.64	697.824	21	21	1608.150	28.51
cdp108a	6654.706	20	20	1823.527	19.580	6652.292	20	20	1743.082	28.260	6652.372	20	20	1745.742	15.54	6650.706	20	20	1690.199	27.95
cdp108b	1331.801	20	20	1966.911	33.340	1331.172	20	20	1861.996	24.590	1329.707	20	20	1617.804	23.64	1331.118	20	20	1852.939	33.26
cdp108c	665.470	20	20	1823.289	27.240	665.195	20	20	1731.524	31.650	665.556	20	20	1851.896	20.54	665.209	20	20	1736.214	23.79
cdp109a	6652.160	20	20	1738.656	27.700	6653.494	20	20	1783.141	18.020	6652.359	20	20	1745.288	24.24	6651.603	20	20	1720.113	36.84
cdp109b	1330.207	20	20	1701.234	31.360	1330.851	20	20	1808.525	24.490	1330.133	20	20	1688.752	28.17	1330.529	20	20	1754.875	18.68
cdp109c	665.405	20	20	1801.641	20.110	665.286	20	20	1761.910	21.380	665.148	20	20	1715.888	32.8	665.064	20	20	1687.844	32.37

Tabela 28 – Resultados das instâncias de 100 clientes C1, parte 2

Instância	nosymc					sym2c					sym1c					sym12c				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
cdp101a	7317.175	22	22	1905.830	91.350	7310.975	22	22	1699.180	56.950	7316.994	22	22	1899.793	26.840	7318.902	22	22	1963.408	24.770
cdp101b	1463.223	22	22	1870.488	46.210	1462.943	22	22	1823.782	33.730	1462.606	22	22	1767.608	43.890	1463.492	22	22	1915.358	50.350
cdp101c	731.354	22	22	1784.744	47.730	731.768	22	22	1922.800	39.280	731.544	22	22	1847.859	34.850	731.715	22	22	1905.045	42.310
cdp102a	6653.326	20	20	1777.537	31.450	6655.999	20	20	1866.627	17.200	6651.782	20	20	1726.057	25.060	6658.697	20	20	1956.572	18.910
cdp102b	1331.319	20	20	1886.533	27.220	1332.157	20	20	2026.208	14.980	1331.821	20	20	1970.114	25.080	1331.127	20	20	1854.434	18.790
cdp102c	665.234	20	20	1744.544	26.030	665.777	20	20	1925.732	16.260	665.675	20	20	1891.810	20.910	665.313	20	20	1770.857	24.130
cdp103a	6979.182	21	21	1639.398	52.090	6647.704	20	20	1590.125	30.320	6980.225	21	21	1674.168	31.86	6650.675	20	20	1689.158	32.79
cdp103b	1395.817	21	21	1636.185	48.250	1395.903	21	21	1650.439	34.980	1395.778	21	21	1629.733	40.61	1395.928	21	21	1654.598	35.38
cdp103c	697.949	21	21	1649.678	44.600	697.841	21	21	1613.778	25.450	697.735	21	21	1578.290	36.01	697.886	21	21	1628.589	35.94
cdp104a	6979.572	21	21	1652.398	50.810	6651.308	20	20	1710.261	28.150	6647.833	20	20	1594.446	42.27	6980.091	21	21	1669.715	58.71
cdp104b	1330.525	20	20	1754.167	28.990	1329.521	20	20	1586.908	29.670	1395.516	21	21	1585.934	22.68	1395.638	21	21	1606.346	45.26
cdp104c	697.671	21	21	1556.915	43.610	697.758	21	21	1586.146	31.550	697.857	21	21	1619.014	43.72	697.988	21	21	1662.688	22.37
cdp105a	6987.643	21	21	1921.439	72.610	6985.699	21	21	1856.635	51.570	6985.184	21	21	1839.474	43.68	6989.224	21	21	1974.143	24.38
cdp105b	1398.092	21	21	2015.403	42.690	1397.285	21	21	1880.810	29.830	1397.262	21	21	1876.955	38.77	1397.929	21	21	1988.100	30.44
cdp105c	698.407	21	21	1802.245	31.130	698.829	21	21	1942.837	37.230	698.706	21	21	1902.055	27.55	698.964	21	21	1988.100	27.38
cdp106a	6658.899	20	20	1963.309	21.330	6658.210	20	20	1940.334	18.590	6660.750	20	20	2025.014	19.2	6656.014	20	20	1867.137	28.66
cdp106b	1330.875	20	20	1812.449	32.360	1331.465	20	20	1910.815	25.920	1331.419	20	20	1903.114	22.79	1332.345	20	20	2057.563	17.66
cdp106c	665.756	20	20	1918.752	38.700	665.878	20	20	1959.314	25.330	665.590	20	20	1863.492	33.53	665.597	20	20	1865.588	30.65
cdp107a	6987.571	21	21	1919.038	49.830	6985.677	21	21	1855.884	41.300	6982.657	21	21	1755.223	30	6985.914	21	21	1863.807	23.28
cdp107b	1396.501	21	21	1750.213	27.410	1396.912	21	21	1818.743	24.530	1396.311	21	21	1718.443	29.61	1396.820	21	21	1803.284	37.66
cdp107c	698.618	21	21	1872.812	32.260	698.075	21	21	1691.633	30.350	698.290	21	21	1763.370	39.87	698.464	21	21	1821.416	23.5
cdp108a	6654.665	20	20	1822.182	26.750	6653.118	20	20	1770.611	19.300	6649.686	20	20	1656.196	24.04	6650.542	20	20	1684.733	24.45
cdp108b	1329.958	20	20	1659.592	31.190	1330.149	20	20	1691.565	21.030	1329.994	20	20	1665.706	27.06	1330.179	20	20	1696.466	23.69
cdp108c	665.053	20	20	1684.345	29.890	664.788	20	20	1596.100	36.580	665.570	20	20	1856.631	16.81	664.981	20	20	1660.404	30.12
cdp109a	6651.018	20	20	1700.611	28.780	6653.614	20	20	1787.126	26.200	6651.775	20	20	1725.841	21.28	6653.328	20	20	1777.611	20.66
cdp109b	1330.327	20	20	1721.171	31.200	1330.007	20	20	1667.799	31.410	1330.216	20	20	1702.648	34.69	1330.344	20	20	1723.953	18.54
cdp109c	665.194	20	20	1731.418	33.580	664.934	20	20	1644.660	23.270	665.407	20	20	1802.291	17.79	665.159	20	20	1719.774	23.3

Tabela 30 – Resultados das instâncias de 100 clientes C2, parte 2

Instância	nosymc					sym2c					sym1c					sym12c				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
cdp201a	5584.526	5	5	845.265	6.450	5605.426	5	5	1054.258	7.940	5589.650	5	5	896.505	14.280	5594.255	5	5	942.551	10.580
cdp201b	1121.521	5	5	1076.065	11.200	1119.629	5	5	981.446	9.070	1116.745	5	5	837.235	7.430	1119.797	5	5	989.852	11.550
cdp201c	561.102	5	5	1110.216	12.130	558.574	5	5	857.384	8.480	559.961	5	5	996.051	8.010	558.175	5	5	817.490	12.890
cdp202a	5604.091	5	5	1040.906	8.690	5627.652	5	5	1276.518	9.560	5586.679	5	5	866.794	21.680	5599.138	5	5	991.381	8.640
cdp202b	1119.971	5	5	998.567	9.890	1118.111	5	5	905.530	17.530	1121.454	5	5	1072.700	10.740	1118.251	5	5	912.533	15.620
cdp202c	558.156	5	5	815.608	17.860	559.508	5	5	950.761	13.330	557.752	5	5	775.233	17.950	557.768	5	5	776.819	15.400
cdp203a	5596.653	5	5	966.526	13.540	5588.669	5	5	886.687	12.490	5592.955	5	5	929.552	11.9	5577.904	5	5	779.044	16.34
cdp203b	1118.663	5	5	933.148	13.630	1119.063	5	5	953.156	16.210	1117.295	5	5	864.725	14.55	1119.789	5	5	989.450	13.26
cdp203c	559.486	5	5	948.619	15.100	557.802	5	5	780.172	14.630	559.378	5	5	937.758	12.63	560.484	5	5	1048.417	17.31
cdp204a	5596.137	5	5	961.368	10.090	5597.195	5	5	971.953	17.240	5599.591	5	5	995.914	12.42	5582.614	5	5	826.141	22.7
cdp204b	1117.713	5	5	885.626	17.010	1116.753	5	5	837.646	22.330	1116.663	5	5	833.148	18.66	1117.950	5	5	897.476	13.14
cdp204c	559.001	5	5	900.059	16.500	558.226	5	5	822.604	19.360	558.133	5	5	813.281	21.75	559.745	5	5	974.548	19.59
cdp205a	5597.299	5	5	972.985	11.540	5585.375	5	5	853.748	14.730	5597.271	5	5	972.705	9.69	5592.682	5	5	926.822	13.12
cdp205b	1118.435	5	5	921.747	9.740	1116.814	5	5	840.715	11.030	1119.460	5	5	972.985	13.68	1118.594	5	5	929.724	9.36
cdp205c	559.138	5	5	913.847	10.340	558.742	5	5	874.239	15.590	560.655	5	5	1065.456	8.03	560.995	5	5	1099.507	9.39
cdp206a	5599.909	5	5	999.085	13.620	5593.494	5	5	934.942	9.390	5596.058	5	5	960.576	19.82	5592.327	5	5	923.273	13.53
cdp206b	1121.000	5	5	1049.979	14.890	1117.683	5	5	884.133	10.360	1118.305	5	5	915.226	12.24	1117.745	5	5	887.239	10.84
cdp206c	558.499	5	5	849.950	21.640	559.837	5	5	983.702	10.370	557.392	5	5	739.241	11.66	559.054	5	5	905.448	12.98
cdp207a	5574.855	5	5	748.547	18.520	5572.122	5	5	721.220	11.890	5597.784	5	5	977.843	15.4	5583.532	5	5	835.319	12.27
cdp207b	1118.989	5	5	949.474	12.870	1118.441	5	5	922.058	13.170	1117.877	5	5	893.866	19.79	1114.790	5	5	739.496	17.58
cdp207c	558.141	5	5	814.080	16.750	559.025	5	5	902.454	18.470	558.972	5	5	897.207	16.66	558.007	5	5	800.679	20.53
cdp208a	5605.166	5	5	1051.660	16.040	5589.027	5	5	890.272	14.210	5588.348	5	5	883.475	8.27	5591.514	5	5	915.142	8.57
cdp208b	1117.484	5	5	874.175	11.680	1117.608	5	5	880.382	17.660	1118.050	5	5	902.485	14.68	1118.225	5	5	911.259	13.21
cdp208c	557.952	5	5	795.223	13.250	559.784	5	5	978.434	17.080	559.108	5	5	910.826	21.22	558.400	5	5	839.957	13.62

Tabela 31 – Resultados das instâncias de 100 clientes RC1, parte 1

Instância	nosymnoc					sym2noc					sym1noc					sym12noc				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rcdp101a	4940.671	45	55	3989.985	32.940	4639.052	39	55	4050.626	48.000	4633.387	40	54	4055.065	96.840	4506.395	37	53	3965.799	139.510
rcdp101b	966.308	42	57	4069.786	41.770	915.708	38	55	4037.461	529.210	938.713	41	54	3882.568	143.220	964.162	43	55	4140.122	29.380
rcdp101c	457.943	39	55	4143.664	99.340	470.470	40	56	3937.973	17.930	445.194	37	54	3941.516	456.290	454.914	40	55	3935.660	600.790
rcdp102a	4272.829	37	54	4147.266	196.700	4326.277	38	53	3913.946	338.130	4145.972	35	53	3968.464	368.900	4332.465	37	54	4119.257	141.040
rcdp102b	843.676	36	55	4247.592	442.080	866.528	38	54	3965.316	237.480	853.200	36	53	3898.987	182.370	895.434	39	53	3976.000	396.170
rcdp102c	432.615	37	53	3935.960	311.880	427.153	37	54	3869.263	322.440	434.424	38	56	4002.343	290.330	433.224	37	54	4009.613	237.510
rcdp103a	4338.735	39	55	3935.412	502.230	4345.392	41	56	4089.292	227.260	4140.811	37	52	4036.264	187.92	4272.366	38	54	3888.920	485.3
rcdp103b	829.303	37	53	4063.365	192.190	841.184	38	53	3878.653	311.480	828.351	38	52	4122.044	258.18	841.389	39	53	3961.122	326.39
rcdp103c	421.821	39	55	3977.780	341.200	420.102	39	52	3915.089	297.940	420.655	39	53	3858.210	527.59	414.176	38	52	4122.044	343.69
rcdp104a	4175.764	39	53	3719.644	387.540	3923.964	36	51	3840.710	239.760	3983.645	36	51	3765.669	216.01	3989.417	37	52	3778.818	494.71
rcdp104b	785.801	36	52	3720.787	515.040	785.781	36	52	3673.229	378.500	785.857	36	52	3744.164	320.41	811.198	38	53	3845.989	327.8
rcdp104c	404.317	37	51	3620.257	297.210	404.947	37	52	3726.212	201.950	410.990	38	52	3878.405	596.73	404.414	37	51	3798.239	432.15
rcdp105a	4555.696	40	56	4219.274	269.510	4279.368	40	55	3988.088	164.280	4435.515	40	56	4077.977	188.95	4441.384	41	57	4108.693	221.15
rcdp105b	885.850	40	55	3957.792	92.220	875.139	40	56	4059.722	112.110	893.039	40	56	4055.016	308.6	875.115	40	56	4040.030	174.94
rcdp105c	436.374	40	54	3915.082	119.990	430.911	39	55	4032.433	253.510	450.186	42	57	4093.060	168.28	438.126	40	57	4087.278	131.03
rcdp106a	4296.617	37	53	3907.839	982.590	4301.881	38	53	3824.085	116.830	4297.040	39	53	3878.475	178.91	4121.770	36	54	3928.367	499
rcdp106b	903.610	38	55	3888.069	453.720	834.091	37	52	3912.923	473.450	824.312	36	54	3948.560	317.46	859.226	38	53	3769.518	310.8
rcdp106c	424.225	37	54	3992.173	138.660	436.807	38	55	3804.155	961.880	430.307	39	54	4017.352	444.67	417.582	37	53	3840.976	251.5
rcdp107a	4175.429	37	53	3762.349	530.880	4115.251	36	53	3805.266	673.990	4055.512	38	53	3715.285	576.68	4121.703	36	54	3900.550	468.47
rcdp107b	809.687	35	52	3658.076	209.820	836.152	37	54	3685.951	269.640	823.092	36	53	3834.671	467.94	823.014	36	53	3758.823	334.88
rcdp107c	411.509	36	53	3716.734	387.210	404.892	35	52	3760.979	564.770	411.507	36	53	3758.823	298.85	412.148	36	54	3917.570	551.37
rcdp108a	4043.402	37	51	3642.020	533.290	4019.778	36	52	3763.734	301.240	4242.712	36	53	3651.845	390.65	4139.380	37	52	3712.184	77.83
rcdp108b	791.791	35	52	3713.174	212.720	815.935	36	52	3771.231	253.700	829.299	37	53	3925.162	264.13	804.928	36	53	3742.748	413.8
rcdp108c	414.025	37	52	3887.438	237.690	407.943	37	52	3602.320	88.950	400.716	35	50	3635.515	565.68	408.565	37	53	3832.252	268.17

Tabela 32 – Resultados das instâncias de 100 clientes RC1, parte 2

Instância	nosymc					sym2c					sym1c					sym12c				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rcdp101a	4524.072	38	56	4027.210	87.250	4692.936	40	54	3916.030	53.380	4940.464	44	55	4026.518	192.700	4765.397	41	56	4044.702	40.670
rcdp101b	927.962	40	55	4152.923	53.790	939.876	41	55	3987.954	50.580	926.795	40	54	4023.947	90.350	938.683	40	54	4003.307	48.060
rcdp101c	488.080	44	55	4042.985	20.720	476.595	41	56	4117.315	25.470	464.541	40	56	4104.565	75.120	476.516	41	56	4006.604	45.460
rcdp102a	4332.232	37	54	3998.805	302.160	4332.598	37	54	4141.077	136.260	4332.240	37	54	4009.613	335.690	4278.133	37	55	4023.340	456.650
rcdp102b	841.244	36	53	3969.805	315.210	867.757	38	55	4071.605	239.970	854.415	37	54	4048.479	280.580	878.419	38	54	3902.856	89.210
rcdp102c	433.855	38	55	4081.038	337.390	433.240	37	54	4075.690	118.280	426.061	37	52	3943.758	268.250	433.270	38	54	3975.934	152.740
rcdp103a	4207.365	39	53	4023.945	348.840	4213.040	39	54	4099.023	302.160	4210.434	39	52	4042.424	301.84	4302.558	37	54	3973.946	178.12
rcdp103b	829.251	38	53	4010.170	243.980	814.819	36	51	3907.640	351.340	829.350	38	53	3962.366	396.44	830.458	38	54	4021.837	512.31
rcdp103c	409.174	37	54	4002.684	273.610	411.749	39	53	4029.475	603.960	414.608	38	53	3941.599	799.4	417.695	39	53	3943.511	264.58
rcdp104a	4055.770	38	53	3776.995	437.790	3989.744	37	52	3824.153	840.640	4043.852	37	51	3858.353	450.81	4049.873	37	52	3775.185	357.77
rcdp104b	797.847	37	52	3656.261	457.840	811.038	38	53	3664.112	445.290	808.684	37	51	3690.578	369.15	772.707	35	51	3848.951	549.21
rcdp104c	398.320	37	51	3653.376	319.150	398.950	37	52	3823.710	309.810	392.952	36	52	3786.913	619.58	392.949	36	52	3815.458	551.32
rcdp105a	4375.535	40	56	4119.422	213.940	4381.483	40	57	4149.295	126.660	4387.250	40	58	4101.816	353.27	4441.330	41	57	4101.525	185.02
rcdp105b	876.282	40	57	4185.068	225.940	873.869	39	55	3957.213	383.360	877.443	40	58	4184.132	189.97	963.940	42	55	3997.986	512.36
rcdp105c	430.921	39	55	4036.821	148.310	456.128	40	57	4216.410	397.890	436.934	40	55	4056.071	209.81	461.609	40	56	4233.156	349.41
rcdp106a	4241.777	38	54	3835.669	214.070	4206.706	38	53	4034.297	701.580	4236.108	37	53	3886.778	319.31	4368.844	38	55	3993.694	448.56
rcdp106b	884.641	40	54	3988.968	116.360	847.257	38	53	3811.809	299.650	848.513	38	53	3881.952	110.95	847.224	37	53	3876.866	200.88
rcdp106c	428.567	40	56	4195.535	956.210	429.695	38	53	3891.118	178.040	430.226	38	54	3908.044	411.76	435.745	39	53	4061.319	103.1
rcdp107a	4181.410	37	54	3836.781	335.210	4121.393	36	54	3887.003	430.110	4186.853	37	55	3730.715	262.82	4181.542	37	54	3833.397	328.67
rcdp107b	824.215	36	54	3768.487	510.060	810.958	35	53	3744.942	348.930	842.432	38	54	3911.836	284.61	836.262	37	54	3721.077	237.85
rcdp107c	424.155	38	54	3837.999	375.450	411.507	36	53	3736.433	547.230	411.499	36	53	3713.248	547.87	418.143	37	54	3766.443	373.12
rcdp108a	4012.860	35	51	3679.888	93.810	4018.809	36	52	3614.315	388.840	3958.745	35	52	3643.791	681.7	4084.782	37	53	3584.728	296.42
rcdp108b	805.021	36	53	3699.223	601.230	818.120	37	54	3605.689	426.670	818.279	37	54	3758.459	425.53	842.387	39	54	3802.558	115.35
rcdp108c	420.065	38	52	3871.911	35.550	414.522	37	53	3772.294	400.100	421.182	39	54	3794.211	459.63	401.914	36	52	3690.817	349.26

Tabela 33 – Resultados das instâncias de 100 clientes RC2, parte 1

Instância	nosymnoc					sym2noc					sym1noc					sym12noc				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rcdp201a	3601.742	18	18	2116.161	15.760	3600.298	18	18	2019.853	20.540	3601.697	18	18	2113.166	28.320	3601.248	18	18	2083.178	27.980
rcdp201b	720.116	18	18	2038.717	25.490	720.376	18	18	2125.234	16.010	720.506	18	18	2168.802	18.300	720.531	18	18	2177.131	19.350
rcdp201c	360.149	18	18	2099.448	19.180	360.053	18	18	2035.356	30.110	360.173	18	18	2115.422	21.840	360.367	18	18	2244.790	21.660
rcdp202a	3235.899	17	17	2059.919	12.750	3237.653	17	17	2176.879	16.570	3235.995	17	17	2066.312	13.770	3237.960	17	17	2197.318	11.960
rcdp202b	647.189	17	17	2063.037	13.090	647.459	17	17	2152.894	12.960	647.938	17	17	2312.809	14.700	647.795	17	17	2265.008	13.650
rcdp202c	323.850	17	17	2233.230	17.310	323.627	17	17	2084.668	18.810	323.787	17	17	2191.002	17.750	323.718	17	17	2145.571	10.330
rcdp203a	3397.776	18	18	1851.752	17.990	3397.937	18	18	1862.460	25.700	3397.095	18	18	1806.348	29.07	3396.087	18	18	1739.165	24.33
rcdp203b	712.411	19	19	1803.503	25.270	679.697	18	18	1899.074	27.680	679.470	18	18	1823.377	18.87	679.846	18	18	1948.772	35.69
rcdp203c	339.720	18	18	1813.569	21.890	339.918	18	18	1945.143	27.360	356.354	19	19	1902.365	24.23	356.216	19	19	1810.417	31.68
rcdp204a	3229.951	17	17	1663.390	16.410	3230.497	17	17	1699.771	18.000	3232.001	17	17	1800.051	14.7	3229.691	17	17	1646.095	13.39
rcdp204b	645.980	17	17	1660.068	18.490	646.134	17	17	1711.199	12.990	646.030	17	17	1676.715	13.25	646.223	17	17	1740.924	20.91
rcdp204c	323.254	17	17	1836.014	15.610	322.980	17	17	1653.402	16.720	323.174	17	17	1782.699	15.11	306.672	16	16	1781.561	11.39
rcdp205a	3241.672	17	17	2444.830	12.560	3235.896	17	17	2059.729	19.430	3237.198	17	17	2146.542	16.43	3244.698	17	17	2646.546	9.12
rcdp205b	647.859	17	17	2286.231	13.130	647.930	17	17	2310.104	14.500	647.486	17	17	2162.050	18.05	647.348	17	17	2116.047	14.7
rcdp205c	324.074	17	17	2382.658	9.070	323.688	17	17	2125.336	21.930	323.874	17	17	2249.598	12.73	323.923	17	17	2281.750	13.17
rcdp206a	3398.875	18	18	1924.992	23.810	3400.080	18	18	2005.358	14.250	3399.227	18	18	1948.481	18.65	3399.023	18	18	1934.852	25.78
rcdp206b	679.867	18	18	1955.776	18.300	679.819	18	18	1939.706	13.570	679.939	18	18	1979.754	24.25	680.111	18	18	2036.947	15.06
rcdp206c	339.909	18	18	1939.529	19.520	339.915	18	18	1943.407	25.740	339.901	18	18	1933.802	24.63	339.947	18	18	1964.881	21.67
rcdp207a	3203.287	18	18	2219.132	13.510	3201.877	18	18	2125.159	12.950	3200.600	18	18	2039.980	16.54	3201.662	18	18	2110.814	14.79
rcdp207b	639.946	18	18	1982.131	15.410	639.502	18	18	1833.975	23.610	639.654	18	18	1884.807	26.06	640.282	18	18	2094.167	12.86
rcdp207c	320.161	18	18	2107.667	16.980	319.979	18	18	1986.077	18.080	320.125	18	18	2083.598	21.98	320.000	18	18	2000.091	17.57
rcdp208a	3197.052	18	18	1803.494	18.670	3200.995	18	18	2066.342	15.220	3198.859	18	18	1923.944	16.25	3198.525	18	18	1901.693	16.63
rcdp208b	639.833	18	18	1944.174	16.220	639.770	18	18	1923.220	19.210	639.973	18	18	1990.884	14.71	639.600	18	18	1866.808	19.07
rcdp208c	319.939	18	18	1959.614	22.730	319.761	18	18	1840.964	15.830	319.712	18	18	1808.093	16.14	319.875	18	18	1916.792	18.68

Tabela 34 – Resultados das instâncias de 100 clientes RC2, parte 2

Instância	nosymc					sym2c					sym1c					sym12c				
	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)	F	NV	NE	Dist	Tempo (s)
rcdp201a	3601.061	18	18	2070.733	15.370	3604.193	18	18	2279.539	14.620	3436.579	17	17	2105.290	12.770	3603.838	18	18	2255.855	17.500
rcdp201b	721.091	18	18	2363.630	18.610	720.183	18	18	2060.902	27.290	720.194	18	18	2064.702	17.450	719.995	18	18	1998.454	17.310
rcdp201c	360.116	18	18	2077.189	18.080	360.319	18	18	2212.474	23.000	360.328	18	18	2218.857	9.930	360.043	18	18	2028.885	21.810
rcdp202a	3233.928	17	17	1928.566	16.250	3237.865	17	17	2191.033	17.430	3238.054	17	17	2203.587	7.670	3235.023	17	17	2001.504	12.140
rcdp202b	647.122	17	17	2040.693	14.740	648.305	17	17	2434.836	10.240	647.080	17	17	2026.710	12.800	648.233	17	17	2411.075	16.770
rcdp202c	323.722	17	17	2148.043	14.430	323.585	17	17	2056.572	14.800	324.124	17	17	2416.019	18.280	323.514	17	17	2009.609	18.400
rcdp203a	3397.983	18	18	1865.503	25.210	3399.035	18	18	1935.675	20.090	3397.238	18	18	1815.839	27.08	3396.766	18	18	1784.386	23.35
rcdp203b	712.402	19	19	1800.560	23.490	679.404	18	18	1801.269	20.110	679.556	18	18	1851.923	36.83	679.421	18	18	1807.146	26.06
rcdp203c	339.939	18	18	1959.553	19.470	339.841	18	18	1894.039	30.200	339.926	18	18	1950.584	21.63	339.908	18	18	1938.691	20.19
rcdp204a	3230.758	17	17	1717.197	14.000	3232.081	17	17	1805.389	16.750	3232.468	17	17	1831.169	16.99	3232.152	17	17	1810.113	15.79
rcdp204b	646.368	17	17	1789.266	18.090	646.275	17	17	1758.371	16.500	646.438	17	17	1812.814	16.82	646.344	17	17	1781.370	12.06
rcdp204c	323.093	17	17	1728.941	17.710	323.207	17	17	1804.405	17.440	323.075	17	17	1716.726	16.67	323.153	17	17	1768.615	15.4
rcdp205a	3239.320	17	17	2287.975	13.540	3240.348	17	17	2356.534	8.240	3238.591	17	17	2239.405	12.48	3239.579	17	17	2305.234	10.12
rcdp205b	647.343	17	17	2114.404	20.730	647.251	17	17	2083.784	17.680	648.223	17	17	2407.690	9.23	647.604	17	17	2201.404	12.22
rcdp205c	323.803	17	17	2201.948	11.960	323.998	17	17	2331.800	10.740	323.773	17	17	2182.320	16.94	324.093	17	17	2395.098	12.98
rcdp206a	3399.577	18	18	1971.817	19.090	3399.337	18	18	1955.793	17.830	3399.356	18	18	1957.057	20.59	3398.926	18	18	1928.386	24.48
rcdp206b	680.202	18	18	2067.210	16.830	680.152	18	18	2050.690	18.100	679.554	18	18	1851.425	23.41	679.654	18	18	1884.772	23.81
rcdp206c	340.093	18	18	2061.981	17.950	339.986	18	18	1990.956	22.840	340.023	18	18	2015.540	17.66	339.892	18	18	1927.707	31.11
rcdp207a	3199.000	18	18	1933.323	24.940	3200.732	18	18	2048.791	20.000	3203.168	18	18	2211.199	17.76	3200.740	18	18	2049.321	16.2
rcdp207b	640.028	18	18	2009.295	18.080	639.759	18	18	1919.582	17.490	640.173	18	18	2057.826	12.43	640.010	18	18	2003.258	14.75
rcdp207c	319.851	18	18	1900.379	16.220	320.021	18	18	2014.044	16.700	320.084	18	18	2056.282	37.44	320.143	18	18	2095.209	15.23
rcdp208a	3198.213	18	18	1880.858	15.750	3198.188	18	18	1879.196	18.860	3198.572	18	18	1904.775	12.83	3199.726	18	18	1981.720	24.11
rcdp208b	639.916	18	18	1972.086	17.520	640.044	18	18	2014.511	15.440	639.752	18	18	1917.267	16.78	639.832	18	18	1943.966	11.74
rcdp208c	319.747	18	18	1831.337	19.680	319.935	18	18	1956.592	17.380	319.994	18	18	1996.005	14.19	319.761	18	18	1840.425	18.13

ESTUDO DE MODELO PARA O CASO HOMOGÊNEO

Esse apêndice trata de um estudo de modelo para o caso mais simples, o caso homogêneo. O problema homogêneo é uma variação do problema heterogêneo onde todos os veículos possuem os mesmos custos e cargas máximas, além de suportarem a mesma quantidade máxima de entregadores.

Ao analisar o caso homogêneo, é possível não somente investigar o impacto das restrições de simetria e de carga apresentadas no Capítulo 3, mas também observar o comportamento das soluções encontradas quando se limita o número máximo de entregadores a 1 por veículo (transformando o problema em um simples problema de coleta e entrega simultâneas com janelas de tempo).

Uma bateria de testes foi feita sobre o modelo elaborado no Capítulo 3 para instâncias homogêneas. Nota-se que o modelo desenvolvido anteriormente pode ser facilmente reduzido para um modelo homogêneo ao considerar-se que existe somente um tipo de veículo, e que todos os veículos possuem os mesmos L^k , Q^k , e c_1^k .

Algumas instâncias pequenas, médias, e grandes foram selecionadas para esses testes, sob parâmetros discutidos na Seção B.1.

B.1 Parâmetros

Foram resolvidas instâncias pequenas, médias, e grandes utilizando o *solver* CPLEX 12.8 com um programa escrito em Python implementando o modelo matemático descrito no Capítulo 3. Os problemas foram resolvidos em uma máquina com 2 processadores Intel Xeon E5-2680v2 de 2.8 GHz com dez núcleos e 128 GB DDR3 1866MHz de memória, com tempo limite de 30 minutos. Foi escolhido esse tempo limite pois em 1 hora várias instâncias estouravam a

memória, sem gerar nenhum resultado. Foram resolvidas as instâncias de Wang e Chen (2012) rcdp05, rcdp10, rcdp25, rcdp50, cdp1, e rdp1, de 5, 10, 25, 50, e as últimas de 100 clientes respectivamente. Cada instância foi resolvida limitando o número de entregadores a 1 por veículo a princípio, e depois permitindo até 3 por veículo. Essas instâncias foram adaptadas de Solomon (1987) e possuem uma coluna extra denotando a demanda de coleta. Para o tempo de serviço em cada nó foi utilizado o cálculo em Pureza, Morabito e Reimann (2012) demonstrado na Equação (3.2). Para os testes, foram utilizados como parâmetros os valores utilizados na literatura, $c_2 = 0.0001$, $c_3 = 0.1$, $M = N - 1$, $vel = 1$, $rs = 2$, e $K = N - 1$, o número de nós sem contar o depósito. Foi providenciada também uma solução inicial trivial, onde cada veículo é alocado a exatamente um cliente. Além disso, para minimizar o número de variáveis e de restrições, foi feito um pré-processamento manual. Nesse pré-processamento, as variáveis X_{ijkl} descrevendo um percurso onde $a_i + ts_{il} + tv_{ij} > b_j$ não foram criadas, por não serem arcos possíveis de acordo com as janelas de tempo.

Além desse pré-processamento, para cada execução do método exato foi dada uma solução inicial simples. Essa solução adiciona primeiramente um cliente à rota do primeiro veículo, cuja rota está vazia a princípio. Após esse passo, o algoritmo tenta encaixar mais um cliente à rota desse veículo, respeitando as restrições. Obtendo sucesso ou não, o algoritmo segue para o próximo veículo, adiciona um cliente à sua rota, e repete o processo anterior. Isso é feito para todos os veículos ou até não sobra mais clientes. Desse modo, no melhor caso, metade dos veículos visitam dois clientes, e no pior caso, cada veículo visita exatamente um cliente.

Para os testes homogêneos, foram considerados $c_1 = 1$, e $L = 3$. Para os testes heterogêneos, esses valores variam entre as diferentes classes de veículos, como é discutido no capítulo 5.

Para cada tipo de problema, homogêneo e heterogêneo, foram determinados 8 variações de modelos diferentes, definidos pela presença ou ausência das restrições (3.18), (3.19), (3.20) e (3.21). Os modelos tratados são descritos na Tabela 35. Desse modo, é possível investigar a fundo o impacto que cada restrição auxiliar no desempenho do *solver*.

B.2 Frota Homogênea

Nesta seção são relatados os resultados para as instâncias de Wang e Chen (2012) para o problema homogêneo considerando os modelos definidos na Tabela 35. Esses resultados foram obtidos pelo *solver* para as instâncias de VRP com Coleta e Entrega Simultâneas.

B.3 Resultados para instâncias de 5, 10, e 25 clientes

Para todos os modelos descritos na Tabela 35, instâncias de 5, 10, e 25 clientes foram resolvidos na otimalidade, por conta do seu tamanho. Nesta subseção esses resultados serão

Tabela 35 – Modelos Alternativos

Modelos	Carga (3.20),(3.21)	Simetria 1 (3.18)	Simetria 2 (3.19)
nosymnoc	Não	Não	Não
sym2noc	Não	Não	Sim
sym1noc	Não	Sim	Não
sym12noc	Não	Sim	Sim
nosymc	Sim	Não	Não
sym2c	Sim	Não	Sim
sym1c	Sim	Sim	Não
sym12c	Sim	Sim	Sim

apresentados em detalhe, de modo a observar o comportamento do modelo quando permitem-se que mais entregadores sejam alocados aos veículos.

Figura 14 – Valores da instância RCPD1001

RCdp1001

CUSTOMER
NUMBER
10

VEHICLE
NUMBER
3

CAPACITY
200

CUSTOMER CUST NO.	XCOORD.	YCOORD.	DDEMAND	PDEMAND	READY TIME	DUE DATE	SERVICE TIME
0	40	50	0	0	0	240	0
30	88	30	10	10	74	104	10
48	42	5	10	40	151	181	10
50	72	35	30	10	116	146	10
59	10	20	19	20	42	72	10
71	65	55	14	23	65	95	10
82	27	43	9	16	42	72	10
87	12	24	13	23	90	120	10
91	49	42	13	3	167	197	10
94	57	48	23	25	95	125	10
100	31	67	3	26	180	210	10

Fonte: (WANG; CHEN, 2012)

As Figuras 15 e 16 mostram o resultado alcançado pelo *solver* do CPLEX para a instância RCPD1001, que possui 10 clientes, como pode ser visto na Figura 14. A instância RCPD1001 originalmente pode ser resolvida com apenas 3 veículos, porém, ao utilizar-se o cálculo de tempo de serviço adaptado de Pureza, Morabito e Reimann (2012), descrito na Equação (3.2), essa solução não é mais possível.

Como pode-se ver na Figura 15, ao fixar-se o número de entregadores em 1 para cada veículo, são utilizados 7 veículos diferentes de um total de 10, dos quais 4 fazem rotas que atendem somente 1 cliente e retornam para o depósito. Ao permitirem-se mais entregadores

Figura 15 – Roteamento do RCPD1001 com apenas 1 entregador por veículo

Fonte: Autor

(no máximo 3), o número de veículos utilizados cai para 4, dos quais 2 utilizam 2 entregadores (indicados pela legenda " $l = 2$ "), formando rotas que antes não eram possíveis por conta das janelas de tempo, como é demonstrado na Figura 16.

Tabela 36 – Resultados obtidos pelo *solver* do CPLEX para o modelo homogêneo sem restrições adicionais, *nosymnoc*

Instância	L	Veículos	Entregadores	Distância	Tempo(s)	Variáveis	Restrições
RCPD0501	1	3	3	220,14	<1	106	236
RCPD0501	3	3	3	220,14	<1	338	586
RCPD1001	1	7	7	541,16	<1	481	931
RCPD1001	3	4	6	375,38	<1	1703	3511
RCPD2501	1	14	14	991,04	<1,45	4676	10876
RCPD2501	3	6	15	601,23	1913,51	18378	45526

Na Tabela 36 mostramos os resultados obtidos pelo *solver* do CPLEX para as instâncias RCPD0501, RCPD1001, e RCPD2501, que têm 5, 10, e 25 clientes respectivamente. Os valores apresentados foram obtidos com o modelo *nosymnoc*, sem restrições de simetria e sem a restrição adicional de carga. Cada instância foi resolvida inicialmente fixando o número de entregadores

Figura 16 – Roteamento do RCPD1001 com até 3 entregadores por veículo

Fonte: Autor

em 1 para cada veículo, e depois permitindo até 3 entregadores por veículo. Na coluna Instância, define-se a instância tratada. A coluna L representa a quantidade máxima de entregadores por veículo. A coluna Veículos mostra a quantidade de veículos na frota da solução ótima. A coluna Entregadores exibe o número de entregadores utilizados no total na solução ótima, notando que, quando $L = 1$, essa quantidade é igual à quantidade de veículos. A coluna Distância mostra a distância total percorrida pelos veículos. A coluna Tempo(s) exibe o tempo total de execução em segundos. As colunas Variáveis e Restrições contêm a quantidade de variáveis e restrições nos problemas, respectivamente. Todas as instâncias foram resolvidas até a otimalidade.

Na Tabela 36 é possível ver que permitir mais entregadores por veículo é atraente para as instâncias de 10 e 25 clientes, já a instância de 5 clientes manteve a mesma solução ao permitirem-se mais entregadores, por conta do seu tamanho. Na instância com 25 clientes, percebe-se ainda que com o custo extra de 1 entregador a mais, consegue-se cortar o número de veículos por mais da metade, o que é muito significativo, considerando que o custo de um veículo é 10 vezes maior que o custo de um entregador para os fins destes testes ($c_1 = 1$, $c_3 = 0.1$).

B.4 Resultados gerais

A Figura 17, baseada nos gráficos apresentados em [Dolan e Moré \(2002\)](#), mostra o desempenho dos oito modelos, descritos na Tabela 35, quando aplicados a todas as instâncias de 5, 10, 25, 50 e 100 clientes. Esse tipo de gráfico permite a comparação direta entre os modelos definidos anteriormente.

Define-se um conjunto M de modelos e S de instâncias. Depois, coleta-se cada resultado obtido pelo modelo m para a instância s , r_{ms} . Após definidos esses valores, define-se o menor valor encontrado para uma dada instância s .

$$\min r_s = \min(r_{ms}, \forall m \in M) \quad \forall s \in S \quad (\text{B.1})$$

Definidos os melhores valores encontrados para cada instância, é calculada uma razão entre o resultados encontrados pelos modelos e o melhor valor de todos os resultados para uma determinada instância.

$$Tr_{ms} = \frac{r_{ms}}{\min r_s} \quad \forall m \in M \quad \forall s \in S \quad (\text{B.2})$$

Após esse passo, determina-se a maior razão entre as calculadas, ou seja, a maior distância entre a melhor e a pior soluções para uma determinada instância.

$$\max Tr = \max(Tr_{ms}, \forall m \in M, \forall s \in S) \quad (\text{B.3})$$

Então essa distância é dividida em 100 intervalos iguais, determinando, portanto, o tamanho do passo dado pelo gráfico a cada ponto no eixo horizontal, considerando que a menor razão entre a melhor solução e a solução encontrada é 1, caso sejam iguais.

$$P = \frac{\max Tr - 1}{100} \quad (\text{B.4})$$

Finalmente, o gráfico é desenhado. Cada ponto no eixo horizontal é um percentil, calculado a seguir.

$$H(1) = 1 \quad (\text{B.5})$$

$$H(n) = H(n-1) + P, \quad 2 \leq n \leq 100 \quad (\text{B.6})$$

Cada imagem no eixo vertical mostra a porcentagem de soluções obtidas por aquele determinado modelo cujas distâncias (razões) são menores que o valor correspondente no eixo horizontal. A função FDM é, portanto, cumulativa, e possui valor máximo de 100%.

$$FDM(m, n) = \frac{\sum_{Tr_{ms} \leq H(n)} Tr_{ms}}{|S|} \quad (B.7)$$

Na Figura 17, pode-se ver que cerca de 75% das soluções encontradas pelo modelo *sym1c* foram equivalentes às melhores soluções obtidas entre todos os outros modelos.

Analisando o gráfico, é possível ver que o primeiro modelo a chegar a 100% é o modelo *nosymnoc*, que não possui nenhuma das restrições adicionais. Entretanto, a maior parte dos modelos não conseguiu melhorar a solução dada inicialmente. Isso significa que o modelo tem muitas restrições, favorecendo um modelo menor, como o *nosymnoc*, que possui bem menos restrições.

Ao estudar as soluções encontradas pelos modelos, também foi possível perceber que, para instâncias *rdp1*, os modelos *sym2noc*, *sym12noc*, *sym2c*, e *sym12c* não conseguiram sequer uma vez melhorar a solução trivial. Para as instâncias *cdp1*, esses modelos conseguiram algumas melhorias, porém todas dentro de 5% da solução trivial (de 110 para 106, no melhor caso). Esse comportamento comprova que a segunda restrição de simetria, (3.19), apenas piora o desempenho, tornando o modelo mais pesado. Analisando as restrições (3.18) e (3.19), fica claro que (3.18) é mais forte do que (3.19), e consegue eliminar mais casos de simetria, e que, quando utilizadas em conjunto, o modelo fica grande demais em número de restrições para o *solver*. Esse comportamento pode ser visto nas Tabelas 37 e 38.

Tabela 37 – Número de variáveis e de restrições para cada modelo homogêneo para a instância *rcdp5001*, com 1 entregador por veículo

Modelo	Variáveis	Restrições
<i>nosymnoc</i>	25751	64701
<i>sym2noc</i>	25751	67151
<i>sym1noc</i>	25751	64750
<i>sym12noc</i>	25751	67200
<i>nosymc</i>	25751	69701
<i>sym2c</i>	25751	72151
<i>sym1c</i>	25751	69750
<i>sym12c</i>	25751	72200

Além disso, ao analisarmos o comportamento dos modelos que possuem as restrições adicionais de carga (3.20) e (3.21), houve pouca ou nenhuma diferença entre estes e os modelos

Tabela 38 – Número de variáveis e de restrições para cada modelo homogêneo para a instância rcdp5001, com no máximo 3 entregadores por veículo

Modelo	Variáveis	Restrições
nosymnoc	109053	289201
sym2noc	109053	291651
sym1noc	109053	289250
sym12noc	109053	291700
nosymc	109053	304201
sym2c	109053	306651
sym1c	109053	304250
sym12c	109053	306700

que não possuem essas restrições adicionais. Estudando os resultados, percebe-se que existe uma pequena vantagem nos resultados obtidos sem essas restrições adicionais, justamente por sua ausência retirar diminuir o tamanho do modelo.

Estudando o modelo, percebe-se que essas análises são pertinentes, o modelo que obteve consistentemente os piores resultados foi o modelo *sym12c*, que empregou todas as restrições adicionais (3.18), (3.19), (3.20), e (3.21), a grande custo de processamento.

Figura 17 – Gráfico mostrando o desempenho dos modelos para as instâncias-teste