

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

RAPHAEL HARRY FREDERICO RIBEIRO KRAMER

UMA ABORDAGEM HEURÍSTICA PARA O *POLLUTION-ROUTING PROBLEM*

JOÃO PESSOA

2014

RAPHAEL HARRY FREDERICO RIBEIRO KRAMER

UMA ABORDAGEM HEURÍSTICA PARA O *POLLUTION-ROUTING PROBLEM*

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção do Centro de Tecnologia da Universidade Federal da Paraíba, como requisito parcial para obtenção do Grau de Mestre em Engenharia de Produção. Área de Concentração: Pesquisa Operacional.

Orientador: Prof. Dr. Anand Subramanian

Co-Orientador: Prof. Dr. Lucídio dos Anjos Formiga Cabral

João Pessoa

2014

RAPHAEL HARRY FREDERICO RIBEIRO KRAMER

UMA ABORDAGEM HEURÍSTICA PARA O *POLLUTION-ROUTING PROBLEM*

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção do Centro de Tecnologia da Universidade Federal da Paraíba, como requisito parcial para obtenção do Grau de Mestre em Engenharia de Produção.

Aprovada em 14 de Fevereiro de 2014.

BANCA EXAMINADORA

Prof. Dr. Anand Subramanian — Orientador
Departamento de Engenharia de Produção — UFPB

Prof. Dr. Lucídio dos Anjos Formiga Cabral — Co-Orientador
Departamento de Computação Científica — UFPB

Prof. Dr. Roberto Quirino do Nascimento — Examinador Interno
Departamento de Computação Científica — UFPB

Prof. Dr. Artur Alves Pessoa — Examinador Externo
Departamento de Engenharia de Produção — UFF

AGRADECIMENTOS

Aos meus pais, pela luta diária e por me darem toda a estrutura necessária ao meu desenvolvimento e à busca pelos meus objetivos.

Aos meus irmãos, pela amizade, convivência e troca de experiências.

Ao meu orientador, Anand Subramanian, pelos conhecimentos transmitidos.

Ao meu orientador especial, Thibaut Vidal, pelas idéias propostas e pelo auxílio no desenvolvimento deste trabalho.

Aos professores Lucídio Cabral e Roberto Quirino, pela formação básica na área.

A minha querida e amada Vitória Mola, pelo companheirismo, paciência, confiança e amor compartilhado no dia-a-dia.

Aos amigos Walton Coutinho e Luciano Costa, pela sempre disponibilidade em ajudar.

Às amigas Robertinha, Flávia, Priscila e Ana pelos momentos de *des-stress* do mestrado.

Aos colegas do mestrado, Fernanda, Léo, Emanoela, Íris, Tálita, Elaine e Valtânia, em especial, pelo exemplo de dedicação e superação.

Ao pessoal do LabMeQA, Yuri, Acioli, Cathaline, Gracinha, Vitor, Bia, Teobaldo e Felipe, pela zueira e parceria nos congressos.

Aos professores e funcionários do DEP e do PPGEF da UFPB, em especial Luciano, Cláudia, Bueno, Silene, Nildo e Ana.

Ao CNPq pela bolsa de estudos.

À todos que de uma maneira ou outra contribuíram para o desenvolvimento deste trabalho.

RESUMO

Esta dissertação lida com o *Pollution-Routing Problem* (PRP), i.e. um Problema de Roteamento de Veículos (PRV) com considerações ambientais, recentemente introduzido na literatura por Bektaş e Laporte (2011). O objetivo consiste na minimização dos custos operacionais e ambientais, respeitando as restrições de carga dos veículos e janelas de tempo dos clientes. O custo é baseado no salário dos motoristas e no consumo de combustível, que depende de diversos fatores, como distância percorrida e carga transportada. As velocidades dos veículos são variáveis de decisão adicionais que complementam as decisões de roteamento. Tais velocidades interferem diretamente no custo total, nos tempos de viagem, bem como no conjunto de rotas viáveis. Uma abordagem híbrida que combina uma metaheurística baseada em busca local com uma abordagem exata e um algoritmo recursivo para otimizar as velocidades é proposta para solucionar o problema. Além do PRP, outras duas variantes do PRV com considerações ambientais são tratadas: o PRV considerando consumo de combustível e o PRV com minimização de energia. Os resultados obtidos se mostraram bastante favoráveis quando comparados com os melhores da literatura, e diversas soluções melhoradas são reportadas.

Palavras-chave: Roteamento de Veículos, Logística Verde, *Matheurística*, Otimização de Velocidades.

ABSTRACT

This dissertation deals with the Pollution-Routing Problem (PRP), a Vehicle Routing Problem (VRP) with environmental considerations, recently introduced in the literature by Bektaş e Laporte (2011). The objective is to minimize operational and environmental costs while respecting route-load constraints and service time windows. Costs are based on driver wages and fuel consumption, which depends on many factors, such as travel distance and vehicle load. Vehicle speeds are additional decision variables of the problem which complement routing decisions. They impact the total cost, the travel times between the locations, and thus the set of feasible routes. We propose a hybrid method that combines a local search-based metaheuristic with an exact approach and a recursive speed-optimization algorithm. Moreover, two other “green” VRP variants, the Fuel Consumption VRP (FCVRP) and the Energy Minimizing VRP (EMVRP), are addressed. The results obtained compare very favorably with those found in the literature, and many new improved solutions are reported.

Keywords: Vehicle Routing, Green Logistics, Matheuristic, Speed Optimization.

SUMÁRIO

1	INTRODUÇÃO	11
1.1	DEFINIÇÃO DO TEMA E DO PROBLEMA DE PESQUISA	11
1.2	JUSTIFICATIVA	13
1.3	OBJETIVOS	17
1.3.1	Objetivo Geral	17
1.3.2	Objetivos Específicos	17
2	FUNDAMENTAÇÃO TEÓRICA E REVISÃO DA LITERATURA	18
2.1	META-HEURÍSTICAS	18
2.2	LOGÍSTICA VERDE E PESQUISA OPERACIONAL	20
2.3	PROBLEMA DE ROTEAMENTO DE VEÍCULOS	23
2.3.1	<i>Green Traveling Salesman Problem</i>	27
2.3.2	<i>Green Vehicle Routing Problem</i>	27
2.3.3	<i>Pollution-Routing Problem</i>	28
2.3.4	<i>Surveys em PRVs com abordagens ambientais</i>	29
3	DESCRIÇÃO DO PROBLEMA	30
4	PROCEDIMENTOS METODOLÓGICOS	34
4.1	REPRESENTAÇÃO DA SOLUÇÃO	36
4.2	AVALIAÇÃO DO CUSTO COM PENALIZAÇÃO POR INVIABILIDADE	37
4.3	ESTRUTURAS DE DADOS AUXILIARES	38
4.4	GERAÇÃO DA SOLUÇÃO INICIAL	41
4.5	BUSCA LOCAL	42
4.5.1	Vizinhanças Entre Rotas	43
4.5.2	Vizinhanças Intra Rotas	43
4.6	PERTURBAÇÃO	44
4.7	ALGORITMO DE OTIMIZAÇÃO DAS VELOCIDADES — SOA	46
4.8	ABORDAGEM HÍBRIDA	50
4.8.1	Versão Final do Algoritmo — ILS-SP-SOA	52
5	RESULTADOS COMPUTACIONAIS	54
5.1	Instâncias	54
5.2	Resultados para o PRP	56
5.3	Resultados para o VRPTW	63
5.4	Resultados para o FCVRP	65
5.5	Resultados para o EMVRP	66
6	CONCLUSÃO	68
	Appendices	70

A	Exemplo da utilização das estruturas de dados auxiliares apresentadas na Seção 4.3	71
B	Concatenação com <i>time-warp</i>	77

LISTA DE FIGURAS

1.1	Representação gráfica do PRV	12
1.2	Emissões de CO ₂ do transporte rodoviário no setor de transporte por país	16
4.1	Representação de uma rota σ	36
4.2	Representação das velocidades de uma rota σ	37
4.3	Penalidades por uso do <i>time warp</i>	38
4.4	Vizinhanças entre rotas	44
4.5	Vizinhanças intra rotas	44
4.6	Consumo de combustível em função da velocidade	48
4.7	Cálculo dos tempos de chegada com o SOA	50
5.1	Solução da instância <i>UK10_01-PRPLIB</i>	61
5.2	Solução da instância <i>UK10_01-B</i>	61
5.3	Solução da instância <i>UK10_01-C</i>	61
5.4	Solução da instância <i>UK10_08-PRPLIB</i>	62
5.5	Solução da instância <i>UK10_08-B</i>	62
5.6	Solução da instância <i>UK10_08-C</i>	63
A.1	Exemplo <i>Shift(1,0)</i>	71
A.2	Concatenação dos clientes 2 e 3 ($\sigma' = 2 \oplus 3$)	73
A.3	Concatenação da subsequência σ' com o depósito final $0'$ ($\sigma_5 = \sigma' \oplus 0'$)	74
A.4	Concatenação das subsequências σ_2 e σ_5 ($\sigma'' = \sigma_2 \oplus \sigma_5$)	74
A.5	Concatenação das subsequências σ_4 e σ'' ($\sigma_2 = \sigma_4 \oplus \sigma''$)	75
B.1	Concatenação dos clientes 1 e 2 ($\sigma' = 1 \oplus 2$)	78
B.2	Concatenação da subsequência σ' e 3 ($\sigma'' = \sigma' \oplus 3$)	79

LISTA DE TABELAS

1.1	Emissões de CO ₂ no mundo do setor de transporte	15
3.1	Dados utilizados no modelo do PRP	33
5.1	Resultados para as instâncias da PRPLIB	58
5.2	Resultados para as instâncias propostas para o PRP — Conjunto B	59
5.3	Resultados para as instâncias propostas para o PRP — Conjunto C	60
5.4	Resultados do ILS-SP-SOA para as instâncias tipo <i>c</i> de Solomon (1987)	64
5.5	Resultados do ILS-SP-SOA para as instâncias tipo <i>r</i> de Solomon (1987)	64
5.6	Resultados do ILS-SP-SOA para as instâncias tipo <i>rc</i> de Solomon (1987)	65
5.7	Resultados para o FCVRP	66
5.8	Resultados para o EMVRP	67
A.1	Características dos clientes	71
A.2	Matriz de distâncias entre clientes	71
A.3	Valores das estruturas auxiliares para subsequências de um nó (exemplo)	72
B.1	Características dos clientes	77
B.2	Matriz de distâncias entre clientes	77

LISTA DE ABREVIACOES

AVRP	<i>Asymmetric Vehicle Routing Problem</i>
DAFP	<i>Dial-A-Flight-Problem</i>
DARP	<i>Dial-A-Ride-Problem</i>
EMVRP	<i>Energy Minimizing Vehicle Routing Problem</i>
EVRP	<i>Emissions Vehicle Routing Problem</i>
FCVRP	<i>Fuel Consumption Vehicle Routing Problem</i>
GRASP	<i>Greedy Randomized Adaptive Search Procedure</i>
GTSP	<i>Green Traveling Salesman Problem</i>
GVRP	<i>Green Vehicle Routing Problem</i>
HFVRP	<i>Heterogeneous Fleet Vehicle Routing Problem</i>
ILS	<i>Iterated Local Search</i>
PCV	Problema do Caixeiro Viajante
PRP	<i>Pollution-Routing Problem</i>
PRV	Problema de Roteamento de Veculos
SOA	<i>Speed Optimization Algorithm</i>
SOP	<i>Speed optimization Problem</i>
SP	<i>Set Partitioning</i>
SPP	<i>Set Partitioning Problem</i>
TDPRP	<i>Time Dependent Pollution-Routing Problem</i>
TDVRP	<i>Time Dependent Vehicle Routing Problem</i>
TSP	<i>Traveling Salesman Problem</i>
VRP	<i>Vehicle Routing Problem</i>
VRPPD	<i>Vehicle Routing Problem with Pickup and Delivery</i>
VRPSD	<i>Vehicle Routing Problem with Split Delivery</i>
VRPTW	<i>Vehicle Routing Problem with Time Windows</i>

1 INTRODUÇÃO

1.1 DEFINIÇÃO DO TEMA E DO PROBLEMA DE PESQUISA

O Problema de Roteamento de Veículos (PRV), do inglês *Vehicle Routing Problem* (VRP), é um dos problemas de distribuição logística mais conhecidos e estudados na área de otimização combinatória em virtude da sua aplicabilidade e importância, especialmente na cadeia de suprimentos. Devido a sua complexidade de resolução, o mesmo é considerado \mathcal{NP} -Difícil, isto é, a dificuldade para encontrar a solução ótima cresce exponencialmente a medida que o número de clientes aumenta.

O VRP em sua forma clássica é definido como segue: seja $G = (\mathcal{V}, \mathcal{E})$ um grafo não-orientado, onde $\mathcal{V} = \{0, 1, \dots, n\}$ é o conjunto de vértices de G e $\mathcal{E} = \{(i, j) : i, j \in \mathcal{V}, i < j\}$ é o conjunto de arestas de G . O vértice 0 de \mathcal{V} representa o depósito e os restantes representam os clientes. Cada cliente i possui uma demanda não-negativa, q_i , conhecida com antecedência e que não pode ser fracionada. Existem m veículos idênticos de capacidade Q e a cada aresta (i, j) está associado um custo não-negativo c_{ij} de deslocamento do cliente i ao cliente j . O VRP consiste em determinar um conjunto de rotas (cada uma associada a um veículo), tal que cada cliente seja visitado uma única vez por um único veículo de modo que o custo total de viagem seja minimizado e a demanda de cada cliente seja atendida. Cada veículo deve iniciar e terminar o percurso no depósito 0 e a demanda total de cada rota não deve exceder a capacidade do veículo. Tal problema é designado como Problema de Roteamento de Veículos Capacitado.

A literatura sobre o VRP e suas variantes é muito ampla. Uma cobertura do estado da arte em modelos e algoritmos de resolução pode ser encontrada em Cordeau et al (2007) e no livro de Golden et al (2008).

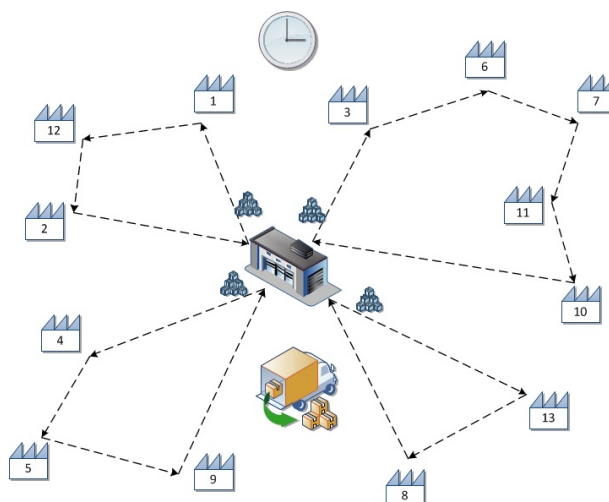


Figura 1.1: Representação gráfica do PRV
Fonte: Demir (2012)

Ao considerar questões ambientais no VRP, tais como emissão de gases do efeito estufa, especialmente o CO_2 , além de fatores como ruído, poluição dos solos, acidificação, etc., tem-se a variante do *Pollution-Routing Problem* (PRP).

O PRP pode ser definido da seguinte maneira. Seja $G = (\mathcal{V}, \mathcal{A})$ um grafo orientado completo, com $\mathcal{V} = \{0, 1, 2, \dots, n\}$ sendo o conjunto de vértices e $\mathcal{A} = \{(i, j) : i, j \in \mathcal{V}, i \neq j\}$ o conjunto de arcos entre cada par de vértices. O vértice $\{0\}$ representa o depósito, onde estão localizados inicialmente uma frota homogênea de m veículos com capacidade Q . Os vértices $\mathcal{V} - \{0\}$ representam os clientes, caracterizados por uma demanda não-negativa q_i , um tempo de atendimento τ_i e um intervalo de janela de tempo $[a_i, b_i]$ para o início do atendimento. Assume-se que a demanda e o tempo de atendimento do depósito são nulos, $q_0 = 0$ e $\tau_0 = 0$, respectivamente. Cada arco $(i, j) \in \mathcal{A}$ representa uma possível viagem de i para j a uma distância d_{ij} . Cada veículo emite uma certa quantidade de gases poluentes ao se deslocar ao longo de um arco (i, j) . Essa quantidade varia conforme um conjunto de fatores, tais como peso de carga e velocidade.

Problemas de roteamento de veículos de dimensões elevadas são geralmente tratados por métodos heurísticos. A palavra heurística é originada do grego *heuriskein*, que significa descobrir. Pode-se então definir heurística como sendo um conjunto de regras e métodos que conduzem à descoberta e à resolução de problemas. Algoritmos heurísticos visam encontrar uma solução mas não necessariamente a melhor, sendo geralmente aplicados à problemas cuja

obtenção da solução ótima é computacionalmente cara quando calculada através de métodos exatos.

Em decorrência do caráter específico e da ineficiência de muitos métodos heurísticos, começam a ser desenvolvidas as chamadas meta-heurísticas. Osman e Laporte (1996) definem meta-heurística como um processo iterativo que guia uma heurística subordinada combinando inteligentemente diferentes conceitos para explorar o espaço de busca de maneira mais eficiente, de forma a produzir soluções de alta qualidade em um tempo computacional aceitável. Dentre as existentes, pode-se citar: Busca Tabu, *Simulated Annealing*, Algoritmos Genéticos, Colônia de Formigas, *Variable Neighborhood Search (VNS)*, *Greedy Randomized Adaptive Search Procedure (GRASP)*, *Iterated Local Search (ILS)*, etc.

Dado o exposto, o seguinte questionamento vem a tona: **como determinar as rotas dos veículos visando a minimização dos custos associados à distância percorrida pelos veículos, de tal forma que as demandas dos clientes sejam atendidas nos horários disponíveis e que os impactos ambientais sejam reduzidos?**

1.2 JUSTIFICATIVA

O VRP desempenha uma função de grande importância na cadeia de suprimentos de diversas empresas envolvidas com o transporte de bens ou pessoas. A importância do estudo do problema é enaltecida pela enorme gama de aplicações em situações reais encaradas diariamente no ambiente empresarial, tais como: serviços de entrega de revistas e jornais, coleta de lixo, distribuição de comidas e bebidas, transporte de empregados, etc.

Toth e Vigo (2002) afirmam que os custos de transporte podem ser reduzidos na ordem de 5% a 20% com a aplicação de técnicas computacionais para a sua resolução. Tais reduções de custos podem ser verificadas em diversos estudos de caso apresentados por Golden et al (2002) e Baker (2002).

Formalmente, o VRP foi apresentado inicialmente por Dantzig e Ramser (1959) no final dos anos 50 ao abordar o problema de distribuição de gasolina para postos de combustível. Neste trabalho foi apresentada a primeira abordagem algorítmica e formulação de programação matemática para o problema. A partir daí, o número de estudos e pesquisas relacionados ao tema

creceu significativamente acarretando em uma grande disponibilidade de materiais na literatura.

Enfrentam-se atualmente sérios desafios no quesito desenvolvimento sustentável. A previsão de crescimento econômico nas principais economias globais é de quatro vezes mais até 2050 (PWC, 2013), podendo alcançar taxas ainda maiores em países em desenvolvimento, como a China e a Índia, ocasionando conseqüentemente um aumento considerável na emissão de gases poluentes. Se as questões ambientais não forem levadas em consideração ao se pensar em crescimento econômico, tensões insustentáveis nos recursos naturais e no meio ambiente serão inevitáveis.

Segundo dados da Agencia Internacional de Energia — IEA (2012), as maiores emissões de CO₂ no ano de 2010 se concentraram nos setores energético e de transporte, que em conjunto são responsáveis pela emissão de quase 2/3 do total global. Isoladamente, o setor de transporte é responsável por 22%, dos quais aproximadamente 75% das emissões são decorrentes da utilização do modal rodoviário. Verifica-se, porém, um decréscimo de 1,6% nestes dois setores em comparação com o ano anterior, redução esta ainda inferior às metas estabelecidas no Protocolo de Kyoto.

A Tabela 1.1 exhibe os principais países emissores de CO₂ de cada continente com as respectivas quantidades emitidas em sua totalidade e derivadas do setor de transporte, destacando-se a China como maior emissor em nível mundial. Ao contrário da maioria dos países que têm a maior parte das emissões de CO₂ advindas do setor elétrico, no Brasil, o setor de transportes é o que mais emite dióxido de carbono ao ambiente, sendo responsável por cerca de 43% do total nacional.

Na Europa aproximadamente 94% das emissões de CO₂ do setor de transporte é devido à utilização do transporte rodoviário. Nos Estados Unidos e na China esses percentuais são de 86% e 78%, respectivamente. No Brasil, por sua vez, este valor equivale a 89% (IEA, 2012). Conforme pode ser verificado na Figura 1.2, esta situação é encontrada em diversos países, enaltecendo a importância de um bom planejamento no uso deste modal de transporte.

O Brasil duplicou os valores de emissão de CO₂ em 2010, comparando-se com o ano de 1990 (IEA, 2012), sendo considerado o 15^o maior emissor de dióxido de carbono em nível mundial (Boden et al, 2012). A participação crescente de países emergentes, como Brasil, China

Tabela 1.1: Emissões de CO₂ no mundo do setor de transporte

País	Emissões* de CO ₂ em 2010				
	Total	% (Mundo)	Setor de Transporte	% (Transporte Global)	% (Total Nacional)
EUA	5368.6	17.73%	1621.7	24.00%	30.21%
Canadá	536.6	1.77%	169.7	2.51%	31.63%
México	416.9	1.38%	151.4	2.24%	36.32%
Brasil	387.7	1.28%	166	2.46%	42.82%
Reino Unido	483.5	1.60%	119.3	1.77%	24.67%
Alemanha	761.6	2.52%	145.5	2.15%	19.10%
França	357.8	1.18%	123.6	1.83%	34.54%
Japão	1143.1	3.78%	222.7	3.30%	19.48%
China	7258.5	23.97%	513.6	7.60%	7.08%
Índia	1625.8	5.37%	161.5	2.39%	9.93%
Indonésia	410.9	1.36%	105.8	1.57%	25.75%
Rússia	1581.4	5.22%	242	3.58%	15.30%
África do Sul	346.8	1.15%	38.2	0.57%	11.01%
Mundo	30276.1	100.00%	6755.8	100.00%	22.31%

*Em milhões de toneladas

Fonte: IEA, 2012

e Índia, no ranking dos maiores poluentes sugere que os mesmos devem modificar a postura em relação às questões climáticas e adotar políticas de maior responsabilidade em relação a emissão de CO₂.

Assim sendo, deve existir uma preocupação não só com a Logística de Distribuição, mas também com o gerenciamento da emissão de poluentes. É neste contexto que emerge o conceito de Logística Verde. O termo Logística Verde, atualmente utilizado por pesquisadores, apesar de relacionado, possui características distintas da logística reversa. Como já se sabe, a logística reversa estuda meios para inserir produtos descartados novamente ao ciclo de negócios, agregando-lhes valor. A logística verde, ou logística ecológica, estuda meios de planejar e diminuir impactos ambientais da logística comum. Isso inclui, por exemplo, estudo de impacto com a inserção de um novo meio de transporte na cidade, projetos relacionados com o certificado ISO 14000, redução de energia nos processos logísticos e redução na utilização de materiais (Mason, 2002).

Juntamente com a importância prática, o interesse no VRP se justifica por ser um problema de difícil resolução, por estar enquadrado na classe \mathcal{NP} -Difícil, ou seja, não existe algoritmo capaz de encontrar soluções ótimas em tempo polinomial. Os algoritmos exatos existentes até o momento não são capazes de resolver instâncias com mais de 150 clientes de forma con-

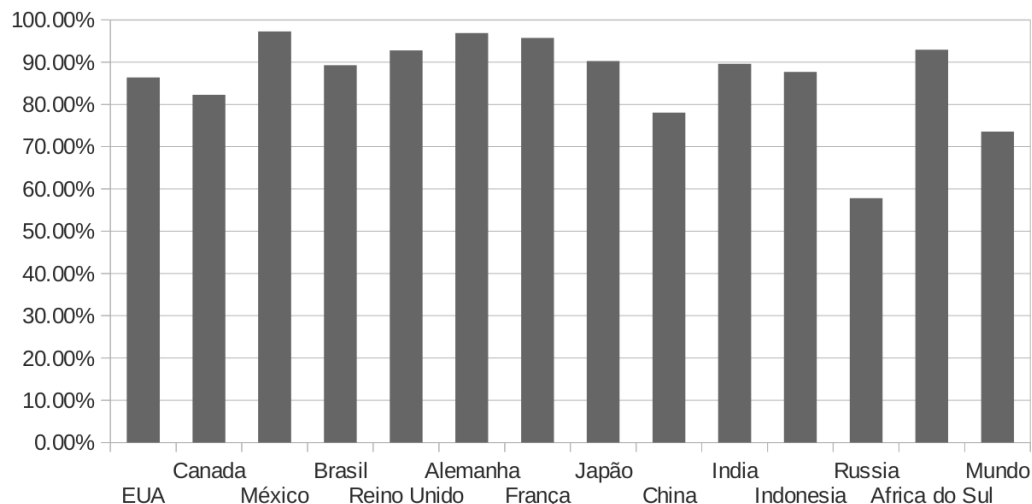


Figura 1.2: Emissões de CO₂ do transporte rodoviário no setor de transporte por país

sistente.

Quanto ao PRP, por ser uma variante recente do VRP (proposta em 2011), o número de trabalhos na literatura é relativamente baixo, com isso, verifica-se um enorme campo de pesquisa a ser explorado e de grande importância, justamente por abordar questões econômicas e ambientais de maneira interdependente. Por possuir o VRP clássico como uma particularidade, o PRP também é \mathcal{NP} -Difícil.

Além da definição das rotas, a principal característica do PRP consiste na definição das velocidades dos veículos durante a viagem entre cada par de clientes, dado que a quantidade de poluição emitida pode ser expressa como uma função da velocidade. O problema de otimização das velocidades, por si só, trata-se de um problema bastante complexo, enaltecendo ainda mais a importância do PRP.

Quanto a metodologia de resolução do problema, por se tratar de um problema encontrado em diversas empresas do setor industrial inseridas em um mercado cada vez mais competitivo, a necessidade de tomar decisões rápidas e eficientes é um objetivo de senso comum. Deste modo, a utilização de meta-heurísticas surge como a melhor estratégia a ser utilizada para obtenção de tal objetivo. Considerando a eficiência computacional e a qualidade das soluções obtidas com a utilização de meta-heurísticas, optou-se pela utilização da mesma para solucionar o PRP.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Propor um algoritmo heurístico para solucionar o *Pollution-Routing Problem* (PRP).

1.3.2 Objetivos Específicos

- Definir um procedimento construtivo para gerar uma solução inicial.
- Implementar uma(s) heurística(s) para refinar a solução inicialmente obtida.
- Averiguar diversas possibilidades de movimentos (estruturas de vizinhança) para melhoria das soluções.
- Examinar diferentes mecanismos de perturbação de uma solução que se encontra em um ponto de mínimo local.
- Implementar estruturas auxiliares para aumentar a eficiência do algoritmo.
- Implementar um algoritmo para resolver o Problema de Otimização de Velocidades (*Speed Optimization Problem*, SOP).
- Resolver o PRP e alguns casos particulares do PRP.
- Testar o algoritmo proposto em diferentes instâncias constantes na literatura.
- Comparar os resultados obtidos com os melhores encontrados na literatura.

2 FUNDAMENTAÇÃO TEÓRICA E REVISÃO DA LITERATURA

Nesta seção é apresentada inicialmente uma breve fundamentação teórica de conceitos e termos comumente utilizados ao lidar com meta-heurísticas com o intuito de facilitar o entendimento dos conteúdos a serem apresentados nas seções seguintes. Posteriormente, é apresentada uma revisão da literatura dos trabalhos relacionados à Logística Verde, em especial àqueles associados ao roteamento de veículos e à Pesquisa Operacional (PO).

2.1 META-HEURÍSTICAS

Em geral, a maioria das variantes do VRP é considerada \mathcal{NP} -Difícil, isto é, a resolução destes problemas por métodos exatos é uma tarefa bastante complexa na prática, sobretudo em instâncias de dimensões elevadas. Métodos heurísticos, por sua vez, são maneiras viáveis de gerar boas soluções para problemas de escala industrial, porém sem garantir a sua otimalidade.

Entende-se métodos heurísticos como sendo uma técnica inspirada em processos intuitivos (Souza, 2007). Dentre estes, destacam-se as heurísticas construtivas e as heurísticas de refinamento. As heurísticas construtivas objetivam gerar uma solução viável inicial. Dentre as heurísticas construtivas desenvolvidas para o VRP, pode-se citar a heurística das economias (*savings*) proposta por Clarke e Wright (1964), a heurística do vizinho mais próximo e a heurística da inserção mais barata. As heurísticas de refinamento, por sua vez, iniciam o processo de busca a partir de uma solução já existente, procurando, iterativamente, substituir a solução corrente por uma melhor, contida em sua vizinhança. Uma descrição mais detalhada a respeito das heurísticas construtivas e de refinamento desenvolvidas para o VRP pode ser encontrada em Laporte e Semet (2002).

Seja S o conjunto de todas as soluções possíveis para um dado problema e s uma solução qualquer, tal que $s \in S$. O conjunto $N(s)$ de soluções que podem ser obtidas a partir de um único movimento, partindo-se de s , é definido como vizinhança de s . Movimento pode ser definido como uma função que transforma uma solução s em outra, s' . Logo, cada solução $s' \in N(s)$ é chamado de vizinho de s . $N(s) \subset S$. De acordo com Souza (2007), de uma solução s do espaço de soluções deve ser sempre possível atingir qualquer outra solução em um número finito de passos, utilizando um determinado tipo ou tipos de movimentos.

Uma solução mínima local, com respeito a uma estrutura de vizinhança N , é uma solução \hat{s} , tal que, para todo $s' \in N(\hat{s}) : f(\hat{s}) \leq f(s')$ (Blum e Roli, 2003).

Enquanto as heurísticas realizam uma busca limitada no espaço de soluções, as meta-heurísticas seguem os princípios de intensificação e diversificação. Intensificação visa uma exploração mais detalhada das áreas mais promissoras do espaço de busca, enquanto a diversificação evita que uma dada solução fique presa em um ótimo local durante o processo de busca (Laporte e Semet, 2002). A integração de algoritmos otimizadores baseados em meta-heurísticas com sistemas logísticos para a gestão da cadeia de suprimentos oferece grandes oportunidades para aplicação da PO no ramo industrial.

As meta-heurísticas podem ser classificadas de diferentes maneiras, dependendo das características consideradas para diferenciá-las. Blum e Roli (2003) apresentam as mais importantes formas de classificação como sendo:

- **Baseadas ou não-baseadas na natureza:** *baseadas na natureza* se enquadram aquelas que utilizam metodologias para geração de soluções baseadas em comportamentos da natureza e/ou dos seres-vivos, tais como: seleção natural, mutação, colônia de formigas, etc. Caso contrário, são *não-baseadas na natureza*.
- **Baseadas em população ou em indivíduos:** depende do número de soluções utilizadas ao mesmo tempo para a geração de novas soluções. Algoritmos que atuam sobre uma única solução são chamados de *métodos de trajetória*, onde se enquadram as meta-heurísticas baseadas em busca local. Já as meta-heurísticas baseadas em população desempenham um processo de busca que descrevem a evolução de um conjunto de pontos no espaço de busca.

- **Com função objetivo estática ou dinâmica:** depende de como a meta-heurística faz uso da função objetivo. Algumas modificam a função objetivo, utilizando-se de informações adquiridas durante o processo de busca para escapar de ótimos locais.
- **Com uma ou várias estruturas de vizinhança:** considera o número de estruturas de vizinhança utilizadas durante o procedimento de busca local.
- **Com ou sem uso de memória:** diz respeito à utilização ou não do histórico de busca. Meta-heurísticas sem uso de memória desempenham um processo Markoviano, pois usam o estado atual para determinar a próxima ação. Por outro lado, os procedimentos que fazem uso da memória utilizam o histórico dos últimos movimentos executados e/ou das soluções visitadas para tomar decisões.

Dentre as principais meta-heurísticas, destacam-se a *Greedy Randomized Adaptive Search Procedure* (GRASP), *Simulated Annealing* (SA), *Tabu Search* (TS), *Variable Neighborhood Search* (VNS) e a *Iterated Local Search* (ILS), baseadas em busca local, e a *Ant Colony Optimization* (ACO), *Artificial Bee Colony* (ABC) e *Genetic Algorithm* (GA), baseadas em busca populacional.

Uma explicação detalhada sobre diversas meta-heurísticas pode ser encontrada em Osman e Laporte (1996), Osman e Kelly (1996), Blum e Roli (2003) e Gendreau e Potvin (2010). Para aplicações em diversas variantes do PRV, consultar Gendreau et al (2008).

2.2 LOGÍSTICA VERDE E PESQUISA OPERACIONAL

A visão simplista de avaliar o rendimento de um processo logístico apenas em termos monetários já não é mais suficiente para garantir um bom desempenho na cadeia logística. As questões ambientais estão sendo consideradas cada vez mais pelas empresas ao tratarem de estratégias para conquistar o mercado, seja em termos de atendimento à legislação ou em termos da satisfação do cliente.

Com a quebra deste paradigma, torna-se necessário o desenvolvimento de mudanças nos processos logísticos, desde a fase de planejamento até a fase operacional, a fim de atender os requisitos impostos pelo ambiente externo (em termos políticos, legislativos ou sociais) à medida

em que se busca uma redução nos custos monetários.

A PO, por sua vez, bastante utilizada na logística auxiliando à tomada de decisões rápidas e eficientes, também tem ampliado seu escopo, passando a incorporar questões ambientais em suas aplicações (Bloemhof-Ruwaard et al, 1995; Daniel et al, 1997; Sbihi e Eglese, 2010; Dekker et al, 2012).

Um dos principais objetivos da logística é entregar os produtos aos consumidores com o menor custo possível, de maneira rápida e confiável. Ao lidar com questões ambientais o cumprimento desses objetivos torna-se mais difícil. Rodrigue et al (2001) mostram os paradoxos entre a logística verde e os princípios da logística comum, conforme apresentado a seguir:

- Redução de custos: a redução dos impactos ao ambiente geralmente são alcançados em prol de aumento nos custos monetários. Diminuição na quantidade de carga transportada por veículo e utilização de combustíveis eficientes e limpos reduzem os impactos ambientais mas aumentam os custos. Devido ao fato dos custos ambientais serem em sua grande maioria externalizados, isto é, estes são repassados a quem não está envolvido diretamente nas atividades em si, muitos empresários ignoram esta questão em prol da redução dos custos operacionais.
- Velocidade: aumento na velocidade de produção e entrega de produtos muitas vezes são alcançados a custas de maior estocagem e utilização de veículos mais rápidos, que na maioria dos casos são mais poluentes.
- Armazenamento: um dos escopos da logística é a redução no armazenamento de cargas, pois produtos estocados significam recursos ociosos. A redução de mercadorias estocadas acarreta um aumento maior de cargas em trânsito, em mais congestionamentos e mais poluição.
- Confiabilidade: baseado na habilidade de entregar as mercadorias de maneira pontual e sem danos. Caminhões e aviões são os meios de transporte mais propícios para o cumprimento deste objetivo, todavia, tais modais estão entre os que mais emitem CO₂.
- *E-Commerce*: baseado no aumento de oportunidades e diversificação na cadeia logística. As consequências na logística verde ainda são pouco conhecidas. Aumento no consumo

de energia e dificuldade na consolidação de cargas, gerando maior quantidade de *ton/km* transportada e maior empacotamento, são alguns dos viés identificados.

Embora os custos ambientais ainda não sejam tão onerosos para as finanças das empresas, nas últimas duas décadas a preocupação da sociedade com as questões ambientais tem aumentado, sendo cada vez mais levado em consideração como critério de escolha pelos consumidores.

Alguns governos já estão se mobilizando para responsabilizar as empresas pelos custos associados aos impactos ambientais, na grande maioria externalizados no cenário global. Tal situação pode ser verificada na Europa, onde verifica-se uma barreira aos produtos importados da China que não tenham certificados que garantam o comprometimento das empresas com o meio ambiente (hung Lai e Wong, 2012).

Deste modo, hung Lai e Wong (2012) consideram a busca por uma imagem verde como algo positivo para as empresas, em especial àquelas envolvidas com atividades de exportação, pois a obtenção de um *selo verde* as credenciariam a competir em mercados com rígidas restrições em termos de legislação ambiental. A partir de uma pesquisa empírica realizada com empresas chinesas, os autores mostram que o gerenciamento da logística verde afeta positivamente não só o desempenho em termos ambientais como também operacionais.

Diante do desafio das empresas em manter a lucratividade enquanto se busca a redução dos impactos ao ambiente, Salimifard et al (2012) ressaltam a importância da Pesquisa Operacional para este fim e a necessidade da academia investir esforços nesta questão. Em Dekker et al (2012), é possível encontrar um recorte de alguns trabalhos desenvolvidos recentemente e dos possíveis campos de pesquisa na área de logística que podem ser abordados através da PO em favor dos aspectos ambientais. Em muitos casos, com apenas um aumento marginal nos custos é possível garantir uma redução considerável da quantidade de gases poluentes emitidos (Dekker et al, 2012; Figliozzi, 2010; Saberi e Verbas, 2012).

Em termos logísticos, transporte refere-se ao deslocamento de cargas e pessoas ao longo da cadeia de suprimento. Dekker et al (2012) consideram quatro aspectos que devem ser considerados em prol do transporte verde: escolha do modal de transporte, transporte intermodal, escolha do equipamento e escolha do combustível. *Transporte verde (Green Transportation)*,

por sua vez, é definido por Björklund (2011) como sendo um serviço de transporte que causa menos impactos à saúde humana e ao ambiente quando comparado com outros serviços de transporte que possuem as mesmas finalidades.

2.3 PROBLEMA DE ROTEAMENTO DE VEÍCULOS

O VRP teve sua origem no trabalho de Dantzig e Ramser (1959), onde foi apresentada a primeira formulação matemática para o problema. O VRP é uma generalização do Problema do Caixeiro Viajante (PCV), do inglês *Traveling Salesman Problem* (TSP), apresentado por Flood (1956). Desde então, diversas variantes foram propostas a partir da versão original.

Dentre as variantes existentes, destacam-se o PRV capacitado (*Capacitated VRP*, CVRP), onde cada cliente tem um demanda por bens e os veículos disponíveis para o atendimento têm a mesma capacidade; a versão assimétrica (*Asymmetric VRP*, AVRP), onde o custo de deslocamento do cliente i para o cliente j pode diferir do custo de j para i ; o PRV com frota heterogênea (*Heterogeneous Fleet VRP*, HFVRP), que difere do tradicional por considerar veículos com capacidades distintas; o PRV com janelas de tempo (*VRP with Time Windows*, VRPTW), onde cada cliente deve ter sua demanda atendida dentro de um certo intervalo de tempo; o PRV com coleta e entrega (*VRP with Pickup and Delivery*, VRPPD), onde cada cliente tem demanda por entrega e/ou coleta de mercadorias; o PRV com entrega fracionada (*VRP with Split Delivery*, VRPSD), que permite mais de um veículo atender a demanda de um mesmo cliente; o PRV dependente do tempo (*Time-Dependent VRP*, TDVRP), motivado pelas condições de tráfego e congestionamento, onde a duração das viagens entre clientes depende do instante de tempo em que as mesmas se iniciam; entre outras.

Além destas, destacam-se o roteamento envolvendo pessoas, onde podem ser citadas o *Dial-A-Ride-Problem* (DARP) e o *Dial-A-Flight-Problem* (DAFP), para transportes terrestres e aéreos, respectivamente.

Nos últimos anos, considerações dos impactos ambientais derivados das atividades de transporte passaram a ser incorporadas aos modelos de roteamento de veículos, motivados não só pelo desejo de gerar uma imagem verde, mas também por razões econômicas. Devido a crescente preocupação da sociedade com os impactos ambientais causados pelas atividades hu-

manas, em especial no setor industrial, estima-se que, num futuro próximo, as emissões de gases causadores do efeito estufa passarão a ter um valor monetário (Saberri e Verbas, 2012). As empresas poderão emitir uma certa quantidade de gases causadores do efeito estufa e, caso necessitem emitir uma quantidade maior que a estipulada, deverão pagar uma quantidade monetária pelo montante excedente. Em roteamento de veículos, a quantidade de poluição emitida por um veículo depende, principalmente, da quantidade de carga transportada e da velocidade, além de outros fatores (Bektaş e Laporte, 2011).

A tese de doutorado de Palmer (2007) foi o primeiro trabalho a relacionar roteamento de veículos com questões ambientais (Dekker et al, 2012; Salimifard et al, 2012). Neste trabalho o autor investigou a variação da quantidade de gás carbônico emitida ao ambiente em função da velocidade do veículo e do consumo de combustível. Considerando diversos cenários de congestão e de janelas de tempo, o autor mostrou que com a modificação da função objetivo, de redução de tempo para redução da emissão de gás carbônico, foi possível obter uma redução média de 4,8% na quantidade de CO₂ emitida à custas de um aumento médio de apenas 0,5% nos custos totais.

Desde então, diversos trabalhos relacionando roteamento e poluição de veículos tem sido desenvolvidos. Dos diversos trabalhos disponíveis na literatura, verifica-se que a maioria destes contabilizam a poluição através do consumo de combustível, da velocidade do veículo e/ou da quantidade de carga transportada por distância. Para a maioria dos veículos, o consumo de combustível e a taxa de gás carbônico emitida por quilômetro diminuem quando o veículo trafega entre 88 e 105 km/h (Figliozzi, 2010).

Kara et al (2007) propuseram uma formulação matemática para o chamado *Energy Minimizing Vehicle Routing Problem* (EMVRP), que tem como objetivo a minimização da soma do produto entre carga e distância pra cada arco das rotas. Abordagens similares, i.e, aquelas que fazem uso da carga do veículo para minimizar o consumo de combustível e/ou emissões de CO₂, foram apresentadas por Peng e Wang (2009), Scott et al (2010), Ubeda et al (2011) e Xiao et al (2012).

Xiao et al (2012) introduziram o *Fuel Consumption Vehicle Routing Problem* (FCVRP), similar ao EMVRP, porém incluindo os custos fixos de uso dos veículos ao modelo. Kara et al

(2008) apresentaram o PRV com custos cumulativos (Cumulative VRP, CumVRP) que generaliza o EMVRP bem como os *Minimum Latency Problem*, o *m-Traveling Repairman Problem* e o *School-bus Routing Problem*. Kopfer et al (2013) também consideram a carga para estimar as emissões de CO₂, porém considerando uma frota heterogênea de veículos.

Entretanto, minimizar o consumo de combustível considerando apenas a carga e a distância não é suficiente. Outro importante fator relacionado ao consumo de combustível é a velocidade dos veículos. É uma questão comum em grandes cidades que interfere nas velocidades dos veículos é a questão do congestionamento das vias de tráfego.

Diante do exposto, Kuo (2010) propôs um modelo para o TDVRP tratando a duração da viagem e a velocidade dos veículos como dependentes do instante de tempo em que a viagem é realizada e os custos associados ao consumo de combustível como uma função da quantidade de carga transportada. Para solucionar o problema o autor utilizou a meta-heurística *Simulated Annealing* (Kirkpatrick et al, 1983), obtendo uma redução de 24,61% no consumo de combustível quando comparado com as soluções obtidas considerando apenas a redução do tempo total de viagem e de 22,69% em relação ao modelo de minimização da distância. No ano seguinte, o mesmo problema foi resolvido através de um algoritmo de Busca Tabu por Kuo e Wang (2011).

Vale ressaltar ainda que Kuo (2010) e Kuo e Wang (2011) abordaram o TDVRP satisfazendo as propriedades de “*non-passing*”¹, comumente ignoradas nos trabalhos que tratam do TDVRP. Mais tarde, Jabali et al (2012) também abordaram o TDVRP considerando as propriedades de “*non-passing*”, porém tratando os custos como função da velocidade, da carga e da duração das rotas, denominando-o de *Emissions-based Time-Dependent Vehicle Routing Problem* (E-TDVRP). As velocidades são escolhidas de acordo com um nível ótimo de emissões, exceto em horários de congestionamento onde são assumidas como constantes.

Fagerholt et al (2010) abordaram o problema de roteamento de navios de carga objetivando a minimização das emissões e do consumo de combustível como um problema de caminho mais curto (*shortest path problem*) a partir da discretização dos tempos de chegada em

¹também conhecida por FIFO (*first-in-first-out*). Garante que, ao se deslocar do cliente i para o cliente j , um tempo de partida (de i) mais cedo ocasionará em um tempo de chegada mais cedo (em j). A maioria das versões *time-dependent* assumem que as velocidades no arco (i, j) são fixas e iguais às velocidades do instante de partida de i , de modo que, se a velocidade no instante de tempo $k + 1$ for maior que a velocidade no instante de tempo k , uma viagem iniciando mais tarde (em $k + 1$) possivelmente terminaria antes que uma viagem iniciando mais cedo (em k), ocorrendo uma situação de “*passing*” (Kuo, 2010).

cada nó. Através da otimização da velocidade dos navios em uma rota já determinada, com os clientes/portos sujeitos à determinadas janelas de tempo, foi possível obter uma redução média de até 21% nas instâncias testadas.

Norstad et al (2011) resolveram instâncias similares às de Fagerholt et al (2010) através de dois algoritmos: uma heurística baseada em busca local *multi-start* para geração das rotas e um algoritmo recursivo suavizado (*recursive smoothing algorithm - RSA*) para otimizar a velocidade. Quando a função de consumo de combustível depende apenas da velocidade, o *RSA* mostrou-se mais eficiente que o apresentado por Fagerholt et al (2010). Todavia, ao considerar outras variáveis, além da velocidade, a abordagem de caminho mais curto com discretização dos tempos é mais viável. No entanto, a exatidão do *RSA* só foi provada posteriormente, por Hvattum et al (2013).

A minimização do consumo de combustível e das emissões em uma rota de navios também foi tratada por Qi e Song (2012), porém os autores consideraram aspectos probabilísticos quanto ao tempo de chegada nos portos, bem como a questão de penalidade por atraso. O problema foi resolvido através de simulação baseada em métodos de aproximação estocástica.

O problema de otimização das velocidades, objetivando a minimização das emissões, também foi estudado por Qian e Eglese (2014). Ao contrário dos problemas abordados por Fagerholt et al (2010), Norstad et al (2011) e Qi e Song (2012), onde as rotas já estão definidas, o problema abordado por Qian e Eglese (2014) consiste na determinação da sequência de clientes e das velocidades entre cada par de clientes. Para tanto, os autores propuseram dois métodos: um exato baseado em programação dinâmica e outro heurístico. A heurística proposta consiste em duas fases: (i) definição das rotas; e (ii) ajuste das velocidades. Para a definição das rotas, os autores utilizaram um procedimento similar ao de Eglese et al (2006) e para o ajuste das velocidades, foi utilizado um algoritmo de programação dinâmica aproximado (*approximate dynamic programming algorithm*).

Ao focar na minimização das emissões e no consumo de combustível como objetivos primário ou secundário do VRP, Figliozzi (2010) propôs uma nova variante ao VRP, chamado *Emissions Vehicle Routing Problem* (EVRP). O EVRP proposto possui restrições de capacidade e janelas de tempo, além de considerar a duração da viagem como dependente do tempo (*time-*

dependent travel times). O autor propôs duas formulações para o problema: uma com função multiobjetivo que inclui os custos dos veículos, da distância percorrida, da duração das rotas e das emissões; e outra mais tradicional com abordagem hierárquica. Para solucionar o problema o autor utilizou um algoritmo heurístico para resolver o TDVRP e, com as rotas obtidas, utilizou outro algoritmo para minimizar as emissões através da otimização dos tempos de saída dos clientes (*departure times*).

Devido a complexidade computacional e a dificuldade em gerar soluções ao modelo discreto, Saberi e Verbas (2012) abordaram o EVRP através de um modelo de aproximação contínua, resolvendo diversos problemas-testes baseados nas instâncias de Solomon (Solomon, 1987). O problema foi resolvido para horários de pico e fora de pico e os resultados sugerem que com apenas um pequeno acréscimo nos custos é possível reduzir uma quantidade considerável de CO₂.

2.3.1 *Green Traveling Salesman Problem*

Com o objetivo de contabilizar os custos ambientais juntamente com os custos operacionais, Özceylan et al (2011) apresentaram uma nova variante para o conhecido TSP. Nesta variante, nomeada de *Green-TSP* (GTSP), eles consideram além da distância, os custos derivados da emissão de CO₂, do consumo de combustível e do tempo de duração das viagens. Uma formulação matemática não-linear inteira mista é apresentada para o problema e um conjunto de instâncias-teste são geradas e resolvidas através do Lingo 11.0 e de um algoritmo contendo as meta-heurísticas *Ant Colony Optimization* (ACO) e *Artificial Bee Colony* (ABC).

Urquhart et al (2010) também trataram o TSP por meio de uma abordagem *verde*. Os autores utilizaram dois métodos para mensurar as emissões de CO₂ e resolveram o problema através de um algoritmo evolucionário. Os resultados dos dois métodos foram comparados entre si e com os resultados obtidos pelo método clássico de minimização das distâncias.

2.3.2 *Green Vehicle Routing Problem*

Assim como o desenvolvimento da versão *green* do TSP, era de se esperar o surgimento do *Green-VRP* (GVRP). Entretanto, o primeiro modelo apresentado para o GRVP, por Erdoğan

e Miller-Hooks (2012), difere dos princípios do GTSP. Ao considerar os desafios associados ao problema de operacionalização de uma frota de veículos de combustível alternativo (biodiesel, eletricidade, etanol, hidrogênio, gás natural, etc) devido a pouca existência de pontos de reabastecimento, os autores incorporam ao modelo do VRP paradas nos postos de abastecimento (*Alternative Fueling Stations*, AFSs), evitando a falta de combustível durante as viagens e mantendo baixos custos operacionais.

2.3.3 *Pollution-Routing Problem*

Bektaş e Laporte (2011) apresentaram uma extensão ao VRP clássico de forma a contabilizar na função objetivo os custos advindos da quantidade emitida de gases poluentes, do consumo de combustível e do tempo total de viagem, além da distância, denominando-a de *Pollution-Routing Problem* (PRP). Além da formulação do PRP, são propostas três diferentes variantes com objetivos distintos: a) minimização da distância; b) minimização da carga ponderada; e c) minimização da energia. Neste trabalho os autores mostram ainda, através de um exemplo ilustrativo, a variação na quantidade de energia utilizada (em *kWh*) por cada abordagem, bem como o efeito na distância, energia e nos custos ao incorporar restrições de janelas de tempo e variáveis de velocidade ao modelo.

Demir et al (2012) abordaram o PRP de maneira heurística em dois estágios. Inicialmente resolveram o VRPTW através da meta-heurística ALNS (*Adaptive Large Neighborhood Search*) utilizando 5 operadores de inserção e 12 de remoção, dos quais 4 são novos (1 de inserção e 3 de remoção), tendo a meta-heurística *Simulated Annealing* como busca local. No segundo estágio, utilizaram um algoritmo exato recursivo para otimizar a velocidade em cada arco de uma dada solução, de modo a minimizar o consumo de combustível e os custos com motoristas. Para verificar a eficiência do algoritmo ALNS proposto, foram resolvidas as instâncias de Solomon (1987). Por fim, foram resolvidos 9 conjuntos de instâncias para o PRP e os resultados foram comparados com os obtidos através do resolvidor CPLEX.

Outros trabalhos baseados no PRP podem ser encontrados na tese de doutorado de Demir (2012) e em Franceschetti et al (2013), onde é apresentado o *Time-Dependent Pollution-Routing Problem* (TDPRP).

Baseado na versão original do PRP, Franceschetti et al (2013) apresentaram o *Time-Dependent Pollution-Routing Problem* (TDPRP), que estende o PRP ao considerar o efeito do congestionamento no roteamento de veículos. Diferente do PRP, onde as velocidades podem ser escolhidas livremente (até um limite superior), no TDPRP as velocidades e, consequentemente os custos, dependem do instante de tempo em que a viagem é efetuada. Os impactos do congestionamento foram modelados de maneira similar ao trabalho de Jabali et al (2012).

Dado que o PRP objetiva a minimização dos custos ambientais e operacionais, que são conflitantes entre si, recentemente Demir et al (2014a) apresentaram uma versão bi-objetiva para o PRP: sendo uma função objetivo para a minimização do combustível e a outra para a minimização da duração das rotas.

2.3.4 *Surveys* em PRVs com abordagens ambientais

Recentemente dois *surveys* em PRV com abordagens ambientais, propostos por Lin et al (2014) e Demir et al (2014b), foram publicados. No primeiro *survey* os autores propõem uma classificação para os PRVs com abordagens ambientais e fazem um levantamento dos trabalhos recentemente publicados nesta área. Além disto, uma atualização dos *surveys* publicados para as variantes clássicas de PRVs é apresentada. O segundo *survey*, por sua vez, além de apresentar diversos trabalhos de PRV na área de logística verde, faz um levantamento de diversos modelos para estimar a emissão de poluentes pelos veículos através da mensuração do consumo de combustível.

3 DESCRIÇÃO DO PROBLEMA

Conforme apresentado anteriormente, na Seção 1.1, o PRP pode ser definido da seguinte maneira. Seja $G = (\mathcal{V}, \mathcal{A})$ um grafo orientado completo, com $\mathcal{V} = \{0, 1, 2, \dots, n\}$ sendo o conjunto de vértices e $\mathcal{A} = \{(i, j) : i, j \in \mathcal{V}, i \neq j\}$ o conjunto de arcos entre cada par de nós. O vértice $\{0\}$ representa o depósito, onde estão localizados inicialmente uma frota homogênea de m veículos com capacidade Q . Os vértices $\mathcal{V} - \{0\}$ representam os clientes, caracterizados por uma demanda não-negativa q_i , um tempo de atendimento τ_i e um intervalo de janela de tempo $[a_i, b_i]$ para o início do atendimento. Assume-se que a demanda e o tempo de atendimento do depósito são nulos, $q_0 = 0$ e $\tau_0 = 0$, respectivamente. Cada arco $(i, j) \in \mathcal{A}$ representa uma possível viagem de i para j a uma distância d_{ij} .

Uma particularidade do PRP reside no fato de que a velocidade v_{ij} em cada arco (i, j) é uma variável de decisão contínua, definida no intervalo $[v_{min}, v_{max}]$. Além disso, cada veículo emite uma certa quantidade de gases causadores do efeito estufa, tal como CO₂, cuja quantidade depende de um conjunto de fatores, tais como peso de carga e velocidade. O PRP objetiva encontrar uma matriz de velocidades $(\mathbf{v})_{ij}$ para os arcos, e um conjunto de $r \leq m$ rotas $\mathbf{R} = \{\sigma^1, \dots, \sigma^r\}$ de modo a atender todos os clientes minimizando os custos ambientais e operacionais. Cada rota $\sigma = (\sigma_1, \dots, \sigma_{|\sigma|})$ deve iniciar e terminar no depósito, i.e., $\sigma_1 = 0$ e $\sigma_{|\sigma|} = 0$, a demanda total de cada rota não pode exceder a capacidade do veículo, e cada cliente deve ser visitado durante sua respectiva janela de tempo.

A Equação (3.1) define, para um rota σ , a carga do veículo $f_{\sigma_i \sigma_{i+1}}$ enquanto trafega no arco (σ_i, σ_{i+1}) , e a Equação (3.2) define recursivamente o tempo t_{σ_i} de chegada nos clientes, sabendo que cada rota inicia no tempo zero ($t_{\sigma_1} = 0$) e que é permitido chegar mais cedo no

cliente i e esperar pelo início de sua janela de tempo a_i . Chegadas tardias, porém, não são permitidas.

$$f_{\sigma_i\sigma_{i+1}} = \sum_{k=i+1}^{|\sigma|} q_{\sigma_k}, \quad i = 1, \dots, |\sigma| - 1 \quad (3.1)$$

$$\begin{cases} t_{\sigma_1} = 0 \\ t_{\sigma_i} = \max\{a_{\sigma_{i-1}}, t_{\sigma_{i-1}}\} + \tau_{\sigma_{i-1}} + \frac{d_{\sigma_{i-1}\sigma_i}}{v_{\sigma_{i-1}\sigma_i}}, \quad i = 2, \dots, |\sigma| \end{cases} \quad (3.2)$$

A função objetivo do PRP é baseada na suposição de que as emissões de CO₂ são proporcionais ao consumo de combustível. O cálculo das emissões em função da velocidade, em um dado arco (i, j) , baseia-se no modelo de Barth et al (2005), Scora e Barth (2006) e Barth e Boriboonsomsin (2008), conforme apresentado na Equação (3.3), onde $\lambda = \xi/\kappa\psi$ e $\gamma = 1/1000\eta_t f \eta$ são constantes relacionadas às propriedades do combustível; $\beta = 0,5C_d\rho A$ e w são constantes associadas às características dos veículos; e α_{ij} é uma constante que depende das características da rodovia e da aceleração do veículo. Mais precisamente, $\alpha_{ij} = \tau_{ij} + g \sin \theta_{ij} + gC_r \cos \theta_{ij}$, onde τ_{ij} e g são as acelerações do veículo e da gravidade, respectivamente; e θ_{ij} é a inclinação da rodovia. O significado dos demais parâmetros são apresentados na Tabela 3.1 juntamente com os valores adotados por Demir et al (2012).

$$\begin{aligned} F_{\sigma_i\sigma_{i+1}}^F(v_{\sigma_i\sigma_{i+1}}) &= \lambda(kNV + w\gamma\alpha_{\sigma_i\sigma_{i+1}}v_{\sigma_i\sigma_{i+1}} + \gamma\alpha_{\sigma_i\sigma_{i+1}}f_{\sigma_i\sigma_{i+1}}v_{\sigma_i\sigma_{i+1}} \\ &\quad + \beta\gamma v_{\sigma_i\sigma_{i+1}}^3)d_{\sigma_i\sigma_{i+1}}/v_{\sigma_i\sigma_{i+1}} \end{aligned} \quad (3.3)$$

Considerando a aceleração do veículo, τ_{ij} , e a inclinação das ruas, θ_{ij} , como nulas, e agrupando as constantes, conforme apresentado nas Equações (3.4)-(3.7), a Equação (3.3) pode ser reescrita como a Equação (3.8).

$$w_1 = \lambda kNV \quad (3.4)$$

$$w_2 = \lambda w\gamma g C_r \quad (3.5)$$

$$w_3 = \lambda\gamma g C_r \quad (3.6)$$

$$w_4 = \lambda\beta\gamma \quad (3.7)$$

Por fim, o objetivo geral do PRP, considerando o consumo de combustível, a um custo unitário de ω_{FC} (por litro consumido), e os custos com motoristas, a um custo de ω_{FD} (por unidade de tempo), pode ser calculado pela Equação (3.9).

$$F_{\sigma_i\sigma_{i+1}}^F(v_{\sigma_i\sigma_{i+1}}) = d_{\sigma_i\sigma_{i+1}} \left(\frac{w_1}{v_{\sigma_i\sigma_{i+1}}} + w_2 + w_3 f_{\sigma_i\sigma_{i+1}} + w_4 v_{\sigma_i\sigma_{i+1}}^2 \right) \quad (3.8)$$

$$Z_{PRP}(\mathbf{R}, \mathbf{v}) = \sum_{\sigma \in \mathbf{R}} \left(\omega_{FC} \sum_{i=1}^{|\sigma|-1} F_{\sigma_i\sigma_{i+1}}^F(v_{\sigma_i\sigma_{i+1}}) + \omega_{FD} t_{\sigma_{|\sigma|}} \right) \quad (3.9)$$

A função de consumo de combustível $F_{\sigma_i\sigma_{i+1}}^F(v_{\sigma_i\sigma_{i+1}})$ é uma função convexa, e o mínimo v_F^* desta função, i.e., a velocidade que minimiza o consumo de combustível, é dado pela Equação (3.10).

$$\frac{dF_{\sigma_i\sigma_{i+1}}^F(v_F^*)}{dv_{\sigma_i\sigma_{i+1}}} = 0 \Leftrightarrow v_F^* = \left(\frac{w_1}{2w_4} \right)^{1/3} \quad (3.10)$$

De maneira similar, para cada arco (σ_i, σ_{i+1}) , assumindo que não tem tempo de espera na rota depois de σ_{i+1} , o custo de viagem $F_{\sigma_i\sigma_{i+1}}^{FD}(v_{\sigma_i\sigma_{i+1}})$ incluindo o salário dos motoristas é dado pela Equação (3.11). A velocidade que minimiza os custos com combustível e motorista (em conjunto) é apresentada na Equação (3.12). Ambos os valores, v_F^* e v_{FD}^* , são independentes do arco em consideração.

$$F_{\sigma_i\sigma_{i+1}}^{FD}(v_{\sigma_i\sigma_{i+1}}) = \omega_{FC} d_{\sigma_i\sigma_{i+1}} \left(\frac{w_1}{v_{\sigma_i\sigma_{i+1}}} + w_2 + w_3 f_{\sigma_i\sigma_{i+1}} + w_4 v_{\sigma_i\sigma_{i+1}}^2 \right) + \omega_{FD} \frac{d_{\sigma_i\sigma_{i+1}}}{v_{\sigma_i\sigma_{i+1}}} \quad (3.11)$$

$$\frac{dF_{\sigma_i\sigma_{i+1}}^{FD}(v_{FD}^*)}{dv_{\sigma_i\sigma_{i+1}}} = 0 \Leftrightarrow v_{FD}^* = \left(\frac{\omega_{FD} + w_1}{2w_4 \omega_{FC}} \right)^{1/3} \quad (3.12)$$

Além do PRP, outros dois problemas serão tratados: o EMVRP e o FCVRP. Tais problemas, ao contrário do PRP, não levam em consideração as restrições de janela de tempo nem decisões relacionadas à velocidade dos veículos, sendo considerados casos particulares do PRP. Na prática, estes problemas podem ser encarados como PRVs com função objetivo orientada à questões ambientais.

A função objetivo do EMVRP corresponde a um modelo de emissões simplificado, que leva em consideração apenas a distância e o fator carga \times distância (i.e. *trabalho*) no cálculo do custo (Kara et al, 2007). A função objetivo do EMVRP consiste na minimização da Equação

Tabela 3.1: Dados utilizados no modelo do PRP

Notação	Descrição	Valores típicos
w	Peso do veículo sem carga (kg)	6350
ξ	Fuel-to-air mass ratio	1
k	Fator de fricção do motor ($kJ/rev/l$)	0,2
N	Velocidade da engrenagem (rev/s)	33
V	Cilindrada/Volume de deslocamento do motor (l)	5
g	Constante gravitacional (m/s^2)	9,81
C_d	Coefficiente de arrasto aerodinâmico	0,7
ρ	Densidade do ar (kg/m^3)	1,2041
A	Área da superfície frontal do veículo (m^2)	3,912
C_r	Coefficiente de resistência ao rolamento	0,01
η_{tf}	Eficiência da transmissão do veículo	0,4
η	Parâmetro de eficiência para motores a diesel	0,9
f_c	Custo de emissão de CO ₂ e combustível por litro (£)	1,4
f_d	Salário do motorista (£/s)	8,0/60/60 = 0,002222
κ	Valor de aquecimento de um típico combustível diesel (kJ/g)	44
ψ	Fator de conversão (g/s para l/s)	797
v_l	Limite inferior de velocidade (m/s)	5,5 (ou 20km/h)
v_u	Limite superior de velocidade (m/s)	25 (ou 90km/h)

FONTE: (Demir et al, 2012)

(3.13), onde ω representa o peso do veículo sem carga.

$$Z_{EMVRP}(\mathbf{R}, \mathbf{v}) = \sum_{\sigma \in \mathbf{R}} \sum_{i=1}^{|\sigma-1|} d_{\sigma_i \sigma_{i+1}} (\omega + f_{\sigma_i \sigma_{i+1}}) \quad (3.13)$$

Por fim, o FCVRP é baseado em uma função linear de consumo de combustível por distância, que considera o peso do veículo sem carga, a carga transportada e o custo fixo de utilização do veículo. A função objetivo do FCVRP consiste na minimização da Equação (3.14), onde h é o custo fixo do veículo e, ρ^* e ρ_0 são as taxas de consumo de combustível do veículo sem carga e completamente carregado, respectivamente.

$$Z_{FCVRP}(\mathbf{R}, \mathbf{v}) = \sum_{\sigma \in \mathbf{R}} \left(h + \omega_{FC} \sum_{i=1}^{|\sigma-1|} \left(d_{\sigma_i \sigma_{i+1}} \left(\rho_0 + \frac{\rho^* - \rho_0}{Q} f_{\sigma_i \sigma_{i+1}} \right) \right) \right) \quad (3.14)$$

Utilizando os mesmos valores de Xiao et al (2012), i.e, $h = 0$, $\omega_{FC} = 1$, $\rho^* = 2$, e $\rho_0 = 1$, a Equação (3.14) pode ser expressa como a Equação (3.15).

$$Z_{FCVRP}(\mathbf{R}, \mathbf{v}) = \sum_{\sigma \in \mathbf{R}} \sum_{i=1}^{|\sigma-1|} d_{\sigma_i \sigma_{i+1}} \left(1 + \frac{1}{Q} f_{\sigma_i \sigma_{i+1}} \right) \quad (3.15)$$

Nota-se que, como no caso do EMVRP, o FCVRP envolve uma combinação linear da distância e do fator carga \times distância.

4 PROCEDIMENTOS METODOLÓGICOS

Devido à complexidade dos problemas de roteamento de veículos estarem enquadrados na classe \mathcal{NP} -Difícil, exigindo tempos computacionais de ordem exponencial em relação ao tamanho do problema para sua resolução, e a inexistência de algoritmos exatos capazes de solucionar problemas de dimensões elevadas de maneira rápida, a utilização de métodos aproximativos, ou heurísticos, apresenta-se como a abordagem mais adequada para lidar com estes tipos de problemas.

A metodologia a ser empregada para resolução do *Pollution Routing Problem* consiste basicamente em duas etapas: (1) definição das rotas e (2) otimização das velocidades. Tais etapas são executadas iterativamente durante a execução do algoritmo proposto.

Sabe-se que os tempos de viagem entre clientes dependem das velocidades dos veículos que, por sua vez, tem um impacto direto nos custos. Porém, para definir as velocidades ótimas dos veículos, no contexto do PRP, é necessário o conhecimento *a priori* da sequência de clientes de uma dada rota. Deste modo, a etapa de definição das rotas será baseada em uma matriz dinâmica de velocidades \mathbf{v} que é inicializada em função da velocidade máxima permitida dada pela instância (i.e. $v_{ij} = v_{\text{MAX}}, \forall (i, j) \in \mathcal{A}$) e atualizada após a otimização das velocidades (etapa 2).

Ao solucionar o PRP desconsiderando decisões quanto às velocidades dos veículos, o problema se reduz a um VRPTW, porém objetivando a minimização da função de custo do PRP (Equação 3.9). Para tanto será utilizada a meta-heurística *Iterated Local Search* (ILS), proposta por Lourenço et al (2002), com busca local baseada no método de descida em vizinhança variável (*Variable Neighborhood Descent*, VND) proposto por Mladenović e Hansen (1997) com seleção

aleatória das estruturas de vizinhança, conforme abordagem apresentada em Penna et al (2013) e Subramanian (2012). Tal meta-heurística é chamada de ILS-RVND (*Iterated Local Search with Randomized Variable Neighborhood Descent*).

Uma vez definidas as rotas, utilizando a função objetivo do PRP e a matriz de duração de viagens, as velocidades dos veículos nos arcos da presente solução são otimizadas por um algoritmo recursivo (*Speed Optimization Algorithm, SOA*), o qual foi baseado nos algoritmos de Norstad et al (2011) e Hvattum et al (2013).

Considerando o exposto, a meta-heurística desenvolvida para solucionar o PRP será denominada de ILS-SOA (Algoritmo 1). A ILS-SOA é uma meta-heurística *multi-start* que se repete por n_R vezes e se caracteriza por efetuar diversas buscas locais em diferentes soluções de “partida”. Inicialmente, a matriz de velocidades é inicializada em função da velocidade máxima permitida pela instância (Linha 4). Baseado na matriz de velocidades v , constrói-se uma solução inicial (Linha 5) e efetua-se uma busca local RVND sobre ela (Linha 8). As velocidades dos arcos da solução obtida pela busca local RVND são otimizadas pelo SOA (Linha 9) e a matriz v é atualizada (Linha 10), substituindo-se as velocidades associadas aos arcos da solução corrente pelas velocidades obtidas pelo SOA. A solução encontrada após a busca local é então modificada (perturbada), sem objetivar melhora, com o intuito de escapar de um ótimo local, para, então, efetuar uma nova busca local. Este procedimento é repetido até que se tenha n_{ILS} repetições sem melhora. No entanto, a versão final do algoritmo trata-se de uma versão híbrida da meta-heurística ILS-SOA com uma abordagem exata para o Problema de Particionamento de Conjuntos (*Set Partitioning Problem - SPP*).

Os procedimentos de geração da solução inicial, de busca local RVND e de perturbação implementados são apresentados nas subseções posteriores com maiores detalhes. Inicialmente, na Seção 4.1, é exposta a maneira como as soluções são representadas. Nas Subseções 4.2 e 4.3 são apresentados como serão avaliados os custos das soluções e as estruturas de dados auxiliares implementadas. Na Subseção 4.4 o procedimento de construção da solução inicial é apresentado. A busca local RVND e as vizinhanças implementadas, bem como o procedimento de perturbação são descritos nas Subseções 4.5 e 4.6, respectivamente. O algoritmo de otimização das velocidades (SOA) é descrito na Subseção 4.7. Por fim, a versão final do algoritmo proposto para solucionar o PRP é apresentado na Subseção 4.8.

Algoritmo 1 ILS-SOA

```

1: Procedimento ILS-SOA( $n_R, n_{ILS}$ )
2:  $s^* \leftarrow \emptyset; f^* \leftarrow \infty;$ 
3: para  $i = 1, \dots, n_R$  faça
4:    $\mathbf{v} \leftarrow \text{InicializaMatrizVelocidades}(v_{MAX});$ 
5:    $s \leftarrow \text{GeraSolucaoInicial}(\mathbf{v});$ 
6:    $s' \leftarrow s; i_{ILS} \leftarrow 0;$ 
7:   enquanto  $i_{ILS} \leq n_{ILS}$  faça
8:      $s \leftarrow \text{RVND}(s);$ 
9:      $s \leftarrow \text{SOA}(s);$ 
10:     $\mathbf{v} \leftarrow \text{AtualizaMatrizVelocidades}(s);$ 
11:    se  $f(s) < f(s')$  então
12:       $s' \leftarrow s;$ 
13:       $i_{ILS} \leftarrow 0;$ 
14:       $s \leftarrow \text{Perturbacao}(s');$ 
15:       $i_{ILS} \leftarrow i_{ILS} + 1;$ 
16:    se  $f(s') < f^*$  então
17:       $s^* \leftarrow s';$ 
18:       $f^* \leftarrow f(s');$ 
19: retorne  $s^*;$ 
20: fim ILS-SOA.

```

4.1 REPRESENTAÇÃO DA SOLUÇÃO

Uma solução para o PRV consiste em um conjunto de clientes ordenados de acordo com a sequência em que os mesmos devem ser atendidos por um conjunto de veículos. Cada cliente pode ser representado por um número inteiro (não repetido). Para o PRP, além da sequência dos clientes, devem ser especificadas as velocidades que os veículos deverão trafegar entre cada par de clientes da sequência de atendimento.

As rotas (sequências de clientes) podem ser armazenadas/representadas por m vetores de tamanho $|\sigma_r|$, onde m é o número de veículos e $|\sigma_r| - 2$ é o número de clientes atendidos pelo veículo r . O primeiro e último elemento do vetor deve conter o depósito (representado por 0), dado que cada veículo deve iniciar e terminar o percurso no depósito.

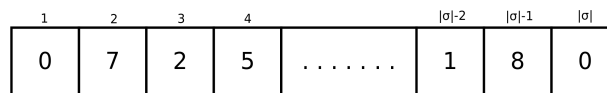


Figura 4.1: Representação de uma rota σ

As velocidades dos veículos também podem ser representadas por m vetores, porém de dimensão $|\sigma_r| - 1$, onde cada elemento representa a velocidade do veículo no percurso do cliente

de posição i ao cliente da posição posterior $i + 1$. Por exemplo, se o elemento de posição 3 do vetor de velocidades é igual a 23,71 e os elementos de posição 3 e 4 do vetor de sequência de clientes são 2 e 5, isso quer dizer que a velocidade do veículo no trajeto entre os clientes representados por 2 e 5 é de 23,71m/s.

1	2	3	4		$ \sigma -2$	$ \sigma -1$
15,33	15,33	23,71	23,71	20,97	20,97

Figura 4.2: Representação das velocidades de uma rota σ

De posse dos vetores com as sequências de atendimento dos clientes e das velocidades dos veículos é possível obter um conjunto informações que auxiliam na representação da solução, como o tempo de chegada em cada cliente, a carga atual do veículo em cada trecho da rota, o tempo de espera para o início do atendimento em cada cliente, a duração total da rota, bem como a distância total percorrida por cada veículo.

4.2 AVALIAÇÃO DO CUSTO COM PENALIZAÇÃO POR INVIABILIDADE

Ao lidar com o VRPTW o número de soluções inviáveis geradas durante a busca local devido ao não atendimento das janelas de tempo dos clientes é bastante elevado, dificultando a convergência para soluções de boa qualidade. Assim sendo, abordar o problema aceitando soluções inviáveis tem sido uma alternativa bastante utilizada para melhorar o desempenho do algoritmo durante a busca local (Vidal et al, 2012), facilitando a convergência para soluções de alta qualidade.

As principais formas de inviabilidade utilizadas na literatura consideram penalidade por antecipação e/ou atraso no atendimento (Taillard et al, 1997; Ibaraki et al, 2005; Hashimoto et al, 2008; Repoussis et al, 2009), e por “voltar no tempo” (Hashimoto e Yagiura, 2008; Nagata et al, 2010; Vidal et al, 2013).

Neste trabalho serão computadas penalidades por “voltar no tempo” para “viabilizar” a solução, comumente chamado na literatura de **uso do *time warp*** (Fig. 4.3). Não serão consideradas penalidades por antecipação na chegada aos clientes pois nestes casos os veículos podem esperar para iniciar o atendimento.

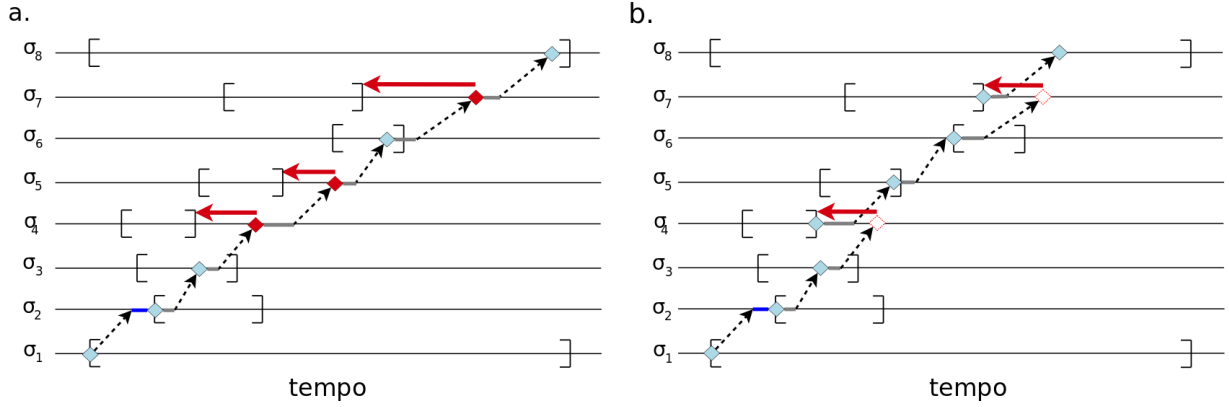


Figura 4.3: Penalidades por uso do *time warp*

4.3 ESTRUTURAS DE DADOS AUXILIARES

Durante a busca local é de extrema importância computar os custos das soluções vizinhas de maneira eficiente. Para avaliar as soluções em tempo constante serão utilizadas estruturas de dados auxiliares baseadas em Vidal et al (2013). Para tanto será necessário o pré-processamento de um conjunto de informações para todas as subsequências de cada rota da solução corrente (no sentido original e inverso) antes da busca local, efetuada em $\mathcal{O}(n^2)$. Deste modo, diz-se que a complexidade computacional necessária para computar o custo de cada solução durante a busca local é efetuada em $\mathcal{O}(1)$ amortizada.

Ao lidar com o VRPTW com penalidade por uso do *time warp*, as informações necessárias de serem computadas para cada subsequência (σ) são: a duração mínima para efetuar o atendimento $DU(\sigma)$, o uso mínimo do *time warp* $TW(\sigma)$ e o tempo de chegada mais cedo $E(\sigma)$ e mais tarde $L(\sigma)$ ao primeiro cliente da subsequência que geram a mínima duração e o mínimo uso do *time warp*. Além destas informações, para tratar com a função objetivo do PRP é necessário computar a carga de cada subsequência $Q(\sigma)$, a distância total $DI(\sigma)$, o tempo total de viagem $TT(\sigma)$, o produto carga \times distância $QD(\sigma)$ e o produto velocidade² \times distância $SSD(\sigma)$. Para todas as subsequências de um único cliente ($\bar{\sigma}$), essas informações são fáceis de serem obtidas, e podem ser calculadas conforme apresentados nas Equações (4.1)-(4.9).

$$DU(\bar{\sigma}) = \tau_i \quad (4.1)$$

$$TW(\bar{\sigma}) = 0 \quad (4.2)$$

$$E(\bar{\sigma}) = a_i \quad (4.3)$$

$$L(\bar{\sigma}) = b_i \quad (4.4)$$

$$Q(\bar{\sigma}) = q_i \quad (4.5)$$

$$DI(\bar{\sigma}) = 0 \quad (4.6)$$

$$TT(\bar{\sigma}) = 0 \quad (4.7)$$

$$QD(\bar{\sigma}) = 0 \quad (4.8)$$

$$SSD(\bar{\sigma}) = 0 \quad (4.9)$$

Considerando que o início das rotas devem ocorrer no instante de tempo 0 (i.e, não permitindo saídas atrasadas do depósito), os tempos de chegada mais cedo $E(\sigma_1)$ e mais tarde $L(\sigma_1)$ ao depósito (inicial) que geram a mínima duração e o mínimo uso do *time warp* devem ser: $E(\sigma_1) = 0$ e $L(\sigma_1) = 0$.

Para as demais subsequências tais informações são calculadas através da concatenação de pares de subsequências, fundamentado em Kindervater e Savelsbergh (1997), Irnich (2008) e Vidal et al (2013). Considerando que $\sigma = (\sigma_1, \dots, \sigma_{|\sigma|})$ e $\sigma' = (\sigma'_1, \dots, \sigma'_{|\sigma'|})$ representam duas subsequências distintas, \oplus representa o operador de concatenação e $\delta_{\sigma_{|\sigma|}\sigma'_1}$ a duração da viagem do último cliente da subsequência σ ($\sigma_{|\sigma|}$) ao primeiro cliente da subsequência σ' (σ'_1), tem-se:

$$\Delta = DU(\sigma) - TW(\sigma) + \delta_{\sigma_{|\sigma|}\sigma'_1} \quad (4.10)$$

$$\Delta WT = \max\{E(\sigma') - \Delta - L(\sigma), 0\} \quad (4.11)$$

$$\Delta TW = \max\{E(\sigma) + \Delta - L(\sigma'), 0\} \quad (4.12)$$

$$DU(\sigma \oplus \sigma') = DU(\sigma) + DU(\sigma') + \delta_{\sigma_{|\sigma|}\sigma'_1} + \Delta WT \quad (4.13)$$

$$TW(\sigma \oplus \sigma') = TW(\sigma) + TW(\sigma') + \Delta TW \quad (4.14)$$

$$E(\sigma \oplus \sigma') = \max\{E(\sigma') - \Delta, E(\sigma)\} - \Delta WT \quad (4.15)$$

$$L(\sigma \oplus \sigma') = \min\{L(\sigma') - \Delta, L(\sigma)\} + \Delta TW \quad (4.16)$$

$$Q(\sigma \oplus \sigma') = Q(\sigma) + Q(\sigma') \quad (4.17)$$

$$DI(\sigma \oplus \sigma') = DI(\sigma) + DI(\sigma') + d_{\sigma_{|\sigma|}\sigma'_1} \quad (4.18)$$

$$TT(\sigma \oplus \sigma') = TT(\sigma) + TT(\sigma') + \delta_{\sigma|\sigma|\sigma'_1} \quad (4.19)$$

$$QD(\sigma \oplus \sigma') = QD(\sigma) + QD(\sigma') + Q(\sigma')(DI(\sigma) + d_{\sigma|\sigma|\sigma'_1}) \quad (4.20)$$

$$SSD(\sigma \oplus \sigma') = SSD(\sigma) + SSD(\sigma') + v_{\sigma|\sigma|\sigma'_1}^2 d_{\sigma|\sigma|\sigma'_1} \quad (4.21)$$

Considerando $\sigma_{i,j}^r$ uma subsequência da rota r , iniciando no i -ésimo cliente e terminando no j -ésimo cliente, as informações apresentadas anteriormente, nas Equações (4.10)-(4.21), para cada subsequência de uma dada solução, percorrendo as rotas para frente e para trás, podem ser calculadas conforme apresentado nos algoritmos 2 e 3, respectivamente.

Algoritmo 2 ComputeForwardSubsequences

- 1: Procedimento *ComputeForwardSubsequences(sol)*
 - 2: **para** $r = 1 \dots m$ **faça**
 - 3: **para** $i = 1 \dots |\sigma^r| - 1$ **faça**
 - 4: **para** $j = i + 1 \dots |\sigma^r|$ **faça**
 - 5: $\sigma_{i,j}^r = \sigma_{i,j-1}^r \oplus \sigma_{j,j}^r$
 - 6: **end** ComputeForwardSubsequences.
-

Algoritmo 3 ComputeBackwardSubsequences

- 1: Procedimento *ComputeBackwardSubsequences(sol)*
 - 2: **para** $r = 1 \dots m$ **faça**
 - 3: **para** $i = |\sigma^r| \dots 2$ **faça**
 - 4: **para** $j = i - 1 \dots 1$ **faça**
 - 5: $\sigma_{i,j}^r = \sigma_{i,j+1}^r \oplus \sigma_{j,j}^r$
 - 6: **end** ComputeBackwardSubsequences.
-

Utilizando as estruturas apresentadas nas Equações (4.1)-(4.21), o custo penalizado de uma rota σ pode ser calculado como mostrado na Equação (4.22), onde ω_{tw} é o custo unitário por uso do *time-warp*.

$$\begin{aligned} Z(\sigma) = & \omega_{FC} (w_1 TT(\sigma) + w_2 DI(\sigma) + w_3 QD(\sigma) + w_4 SSD(\sigma)) \\ & + \omega_{FD} DU(\sigma) + \omega_{TW} TW(\sigma) \end{aligned} \quad (4.22)$$

4.4 GERAÇÃO DA SOLUÇÃO INICIAL

O método de geração da solução inicial consiste na inserção iterativa de clientes às rotas. Inicialmente, insere-se um cliente em cada rota (associada à um veículo) de maneira aleatória. Os clientes restantes são, então, adicionados às rotas de acordo com a estratégia de inserção paralela e o critério de inserção mais barata, baseado em Penna et al (2013) e Subramanian (2012), porém com penalização nos custos de soluções inviáveis quanto ao atendimento das janelas de tempo.

Na estratégia de inserção paralela todas as possíveis posições de inserção de todas as rotas são avaliadas, de modo que cada cliente seja inserido na melhor posição possível, segundo um *critério* de avaliação, respeitando a capacidade dos veículos. O procedimento encerra quando todos os clientes estiverem em uma rota.

Os critérios de inserção servem para definir qual cliente será inserido e em qual posição será feita esta inserção. O critério de inserção mais barata calcula o menor custo de inserção de um cliente entre dois outros. Caso a inserção gere inviabilidade(s) por atraso no atendimento das janelas de tempo, o custo de inserção é penalizado, conforme apresentado nas subseções 4.2 e 4.3.

Considerando $g(i, k, p)$ o custo de inserção do cliente i na posição p de uma rota k , a construção da solução inicial pode ser executada de acordo com o Algoritmo 4.

Algoritmo 4 Gera Solução Inicial

- 1: Procedimento GeraSolucaoInicial(m)
 - 2: Inicialize a Lista de Clientes LC
 - 3: Seja $s = \{s^1, \dots, s^m\}$ o conjunto de m rotas vazias
 - 4: **para** $k = 1, \dots, m$ **faça**
 - 5: $s^k \leftarrow i \in LC$ selecionado aleatoriamente;
 - 6: Atualize LC ; $\{LC \leftarrow LC - \{i\}\}$
 - 7: **enquanto** $LC \neq \emptyset$ **faça**
 - 8: Avalie o custo de inserção $g(i, k, p)$;
 - 9: $g^{min} \leftarrow \min\{g(i, k, p)\}$
 - 10: $i' \leftarrow$ cliente i associado à g^{min}
 - 11: $k' \leftarrow$ rota k associada à g^{min}
 - 12: $p' \leftarrow$ posição p associada à g^{min}
 - 13: $s^{k'} \leftarrow s^{k'} \cup \{i'\}$
 - 14: Atualize LC ;
 - 15: **retorne** s ;
 - 16: **fim** GeraSolucaoInicial.
-

4.5 BUSCA LOCAL

A busca local implementada faz uso de um conjunto de estruturas de vizinhanças, selecionadas de maneira aleatória. Tais estruturas são separadas em duas listas de vizinhanças distintas: $LV1$ e $LV2$. A $LV1$ contém as vizinhanças/procedimentos que objetivam a melhoria de uma solução geral através de “trocas” sistemáticas entre clientes de **rotas distintas**. A $LV2$ contém as vizinhanças/procedimentos que objetivam a redução do custo de uma rota específica através de “trocas” sistemáticas entre clientes de uma **mesma rota**.

A busca local RVND, representada pelo Algoritmo 5, funciona da seguinte maneira: Inicializa-se $LV1$, inserindo todas as vizinhanças entre rotas à lista (Linha 2). Uma vizinhança entre-rotas é escolhida aleatoriamente (Linha 4) e o melhor vizinho é escolhido (Linha 5). Se a solução s' encontrada for melhor que a solução corrente s (Linha 6), s' passa a ser a nova solução corrente (Linha 7). Uma busca local intra rotas é efetuada nesta nova solução (Linha 8) e a $LV1$ é atualizada (Linha 9), i.e, a lista é reinicializada com todas as vizinhanças entre rotas. Caso a solução encontrada (Linha 5) não melhore a solução corrente, a vizinhança selecionada é removida da $LV1$ (Linha 11). Este procedimento é repetido enquanto a $LV1$ não estiver vazia (Linha 3), retornando a melhor solução local ao término do procedimento (Linha 12). Vale frisar que na busca local intra rotas (Linha 8) todas as vizinhanças de $LV2$ são chamadas uma única vez em uma sequencia aleatória.

Algoritmo 5 RVND

- 1: Procedimento RVND(s)
 - 2: Inicialize a Lista de Vizinhanças Entre-rotas $LV1$;
 - 3: **enquanto** $LV1 \neq 0$ **faça**
 - 4: Selecione uma vizinhança $N^{(\eta)} \in LV1$ aleatoriamente;
 - 5: Encontre o melhor vizinho s' de $s \in N^{(\eta)}$;
 - 6: **se** $f(s') < f(s)$ **então**
 - 7: $s \leftarrow s'$;
 - 8: $s \leftarrow$ BuscaIntraRota(s);
 - 9: Atualize $LV1$;
 - 10: **senão**
 - 11: Remova $N^{(\eta)}$ de $LV1$;
 - 12: **retorne** s ;
 - 13: **fim** RVND.
-

As estruturas de vizinhança implementadas consistem de 5 vizinhanças entre rotas e 5

vizinhanças intra rotas, todas já existentes na literatura.

4.5.1 Vizinhanças Entre Rotas

Das estruturas entre rotas implementadas, quatro se baseiam no esquema λ -*interchanges* (Osman, 1993), que consiste em trocar até λ clientes entre duas rotas, enquanto uma baseia-se no operador *Cross-exchange* (Taillard et al, 1997), que consiste em trocar dois segmentos de rotas diferentes. Para limitar o número de possibilidades, adotou-se $\lambda = 2$. Uma descrição de cada estrutura é apresentada a seguir.

1. Shift(1,0) — Consiste na transferência de um cliente de uma rota para outra rota.
2. Shift(2,0) — Consiste na transferência de dois clientes, k_1 e k_2 , de uma rota para outra. Neste caso é verificado tanto a transferência do arco (k_1, k_2) quanto a do arco (k_2, k_1) .
3. Swap(1,1) — Consiste na troca de um cliente de uma rota com um cliente de outra rota.
4. Swap(2,2) — Consiste na troca de dois clientes adjacentes, k_1 e k_2 , de uma rota com dois clientes adjacentes, k_3 e k_4 , de outra rota. Nesta situação, devem ser consideradas todas as 4 combinações possíveis de trocas entre os arcos formados pelos dois pares de clientes.
5. 2-opt* — Consiste na troca de um segmento de uma rota por um segmento de outra rota.

4.5.2 Vizinhanças Intra Rotas

O conjunto de vizinhanças intra rotas é composto pela Reinscrição, Or-opt (Or, 1976), Swap e 2-opt (Croes, 1958), descritas a seguir:

1. Reinscrição — Um cliente é removido da rota e inserido em outra posição.
2. Or-Opt2 — Dois clientes adjacentes são removidos da rota e inseridos em outra posição.
3. Or-Opt3 — Três clientes adjacentes são removidos da rota e inseridos em outra posição.
4. Swap — Troca entre dois clientes não adjacentes.
5. 2-opt — Remove dois arcos de uma rota e insere dois novos arcos à mesma rota.

foram considerados apenas os segmentos de rota terminando no depósito

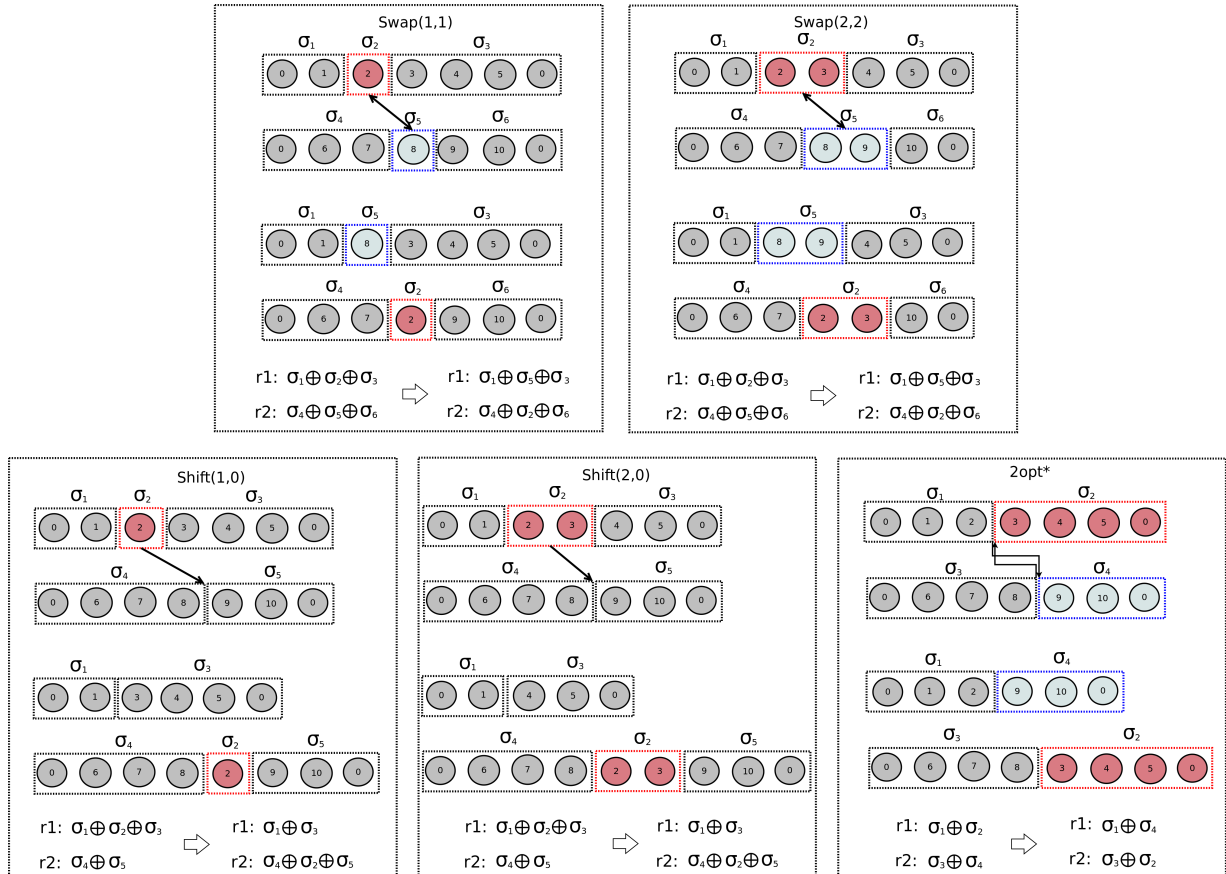


Figura 4.4: Vizinhanças entre rotas

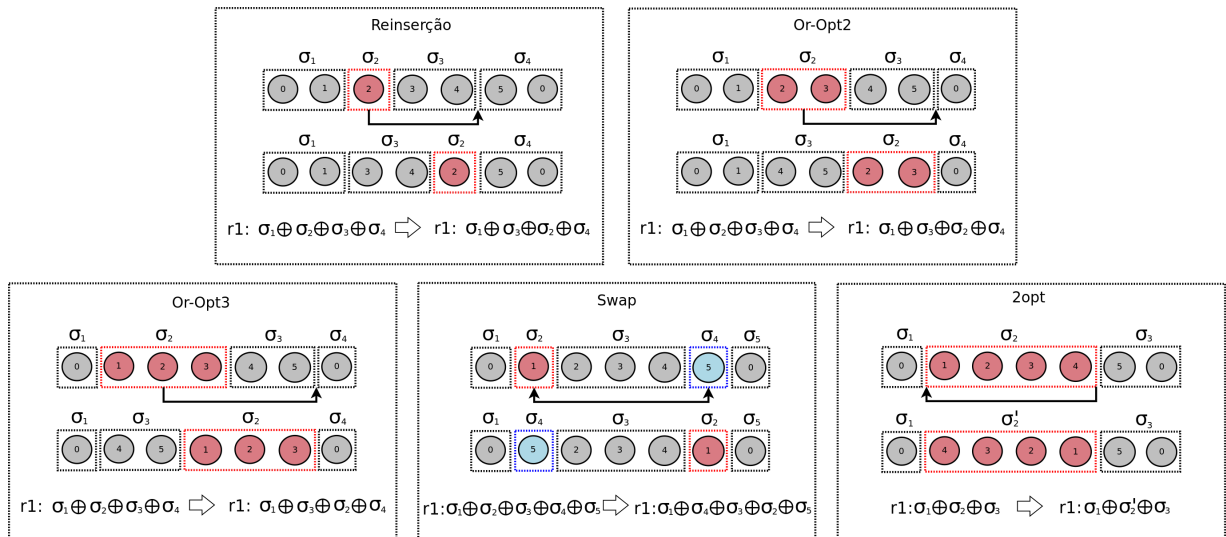


Figura 4.5: Vizinhanças intra rotas

4.6 PERTURBAÇÃO

Segundo Lourenço et al (2002), assim como os procedimentos de busca local, os mecanismos de perturbação são considerados aspectos chaves para o bom desempenho da meta-

heurística ILS. Ao contrário dos procedimentos de busca local, os mecanismos de perturbação não objetivam uma melhora na solução. Sua utilidade consiste em realizar modificações na solução corrente, de modo a escapar de ótimos locais.

Os mecanismos de perturbação podem seguir os princípios de intensificação e/ou diversificação. Segundo Souza (2007), a intensificação é obtida ao se aplicar “pequenas” perturbações à solução corrente, de modo a explorar a região local, enquanto que a diversificação das soluções é alcançada ao se aplicar “grandes” perturbações, deslocando-as para outras regiões do espaço de soluções.

Foram utilizados três mecanismos de perturbações, selecionados aleatoriamente, com diferentes probabilidades, relaxando as restrições de janela de tempo. Tais mecanismos são descritos a seguir:

1. *Multiple Swap(1,1)* — Aplicação de até dois movimentos Swap(1,1) consecutivos, selecionados aleatoriamente com distribuição uniforme.
2. *Reset Speed Matrix* — A matriz de velocidades é reinicializada com uma velocidade selecionada aleatoriamente, com distribuição uniforme, do conjunto SL , onde $SL = \{v_f^*, v_{fd}^*, v_{MAX}\}$, v_f^* é a velocidade ótima que minimiza o consumo de combustível, e v_{fd}^* é a velocidade ótima que minimiza os custos com combustível e motoristas.
3. *Merge Routes* — As duas rotas com menores cargas acumuladas são mescladas desde que não viole as restrições de capacidade. Dadas duas rotas $r_1 = (0, i, \dots, j, 0)$ e $r_2 = (0, k, \dots, l, 0)$, a nova rota mesclada r_3 será $r_3 = (0, i, \dots, j, k, \dots, l, 0)$.

A fase de perturbação funciona da seguinte maneira. Seja n o número de clientes da instância. Inicialmente a perturbação *Multiple Swap(1,1)* ou *Merge routes* é selecionada aleatoriamente, com diferentes probabilidades de seleção: 90% para a primeira e 10% para a segunda. Se o mecanismo *Multiple Swap(1,1)* for selecionado e após n tentativas não for possível encontrar uma solução viável quanto a capacidade dos veículos, o mecanismo *Merge routes* é chamado. Se a perturbação *Merge routes* também não conseguir gerar uma solução viável quanto a capacidade dos veículos, chama-se o mecanismo *Reset speed matrix*. A mesma idéia é aplicada se a perturbação *Merge routes* for selecionada primeiro. Tais percentuais foram selecionados após

experimentos preliminares. O fato da perturbação *Reset speed matrix* ser chamada apenas após o não sucesso das perturbações se justifica pelo fato desta se tratar de uma perturbação bastante agressiva e necessária para evitar ciclagem durante a resolução de algumas instâncias.

Algoritmo 6 Perturbação

```

1: Procedimento Perturbacao( $s'$ )
2:  $s \leftarrow$  solução inviável;  $trials \leftarrow 0$ ;
3:  $NrAleatorio \leftarrow \text{RandBetween}(0,10)$ ;
4: se  $NrAleatorio \leq 9$  então
5:   enquanto  $s$  for inviável e  $trials < n$  faça
6:      $s \leftarrow \text{MultipleSwap}(s')$ 
7:      $trials \leftarrow trials + 1$ 
8:   se  $s$  for inviável então
9:      $s \leftarrow \text{MergeRoutes}(s')$ 
10: senão
11:    $s \leftarrow \text{MergeRoutes}(s')$ 
12:   se  $s$  for inviável então
13:     enquanto  $s$  for inviável e  $trials < n$  faça
14:        $s \leftarrow \text{MultipleSwap}(s')$ 
15:        $trials \leftarrow trials + 1$ 
16:   se  $s$  for inviável então
17:      $s \leftarrow \text{ResetSpeedMatrix}(s')$ 
18: retorne  $s$ ;
19: fim Perturbacao.

```

4.7 ALGORITMO DE OTIMIZAÇÃO DAS VELOCIDADES — SOA

Uma vez definidas as rotas para o VRPTW, torna-se necessário solucionar o SOP de modo a definir as velocidades dos veículos em cada arco $(i, i + 1) \in \mathcal{A}'$ de uma dada rota que minimizem os custos associados à emissão de poluentes ao ambiente (proporcional ao consumo de combustível), bem como os custos associados ao salário dos motoristas, respeitando as restrições de janelas de tempo.

Com o intuito de simplificar a notação, seja $\sigma = (0, i, \dots, n + 1)$ uma rota iniciando e terminando no depósito; e as seguintes variáveis de decisão: $v_{i,i+1}$, associadas às velocidades do veículo no arco $(i, i + 1)$; e_i , associadas aos tempos de chegada no cliente i ; e w_i , associadas ao tempo de espera para o início do atendimento do cliente i . O modelo matemático para o SOP é apresentado em (4.23)-(4.30). A função objetivo (4.23) minimiza os custos com combustível e com salário dos motoristas, onde $F_{i,i+1}(v_{i,i+1})$ é a função de consumo de combustível, depen-

dente da variável $v_{i,i+1}$, e , f_c e f_d são os custos unitários por litro de combustível consumido e por tempo de trabalho dos motoristas, respectivamente.

As restrições (4.24) garantem que os tempos de chegada nos clientes $i + 1$ serão iguais ao tempo de chegada no cliente anterior i acrescido do tempo de espera w_i mais o tempo de atendimento t_i e o tempo de viagem $d_{i,i+1}/v_{i,i+1}$. As restrições (4.25) garantem que o atendimento dos clientes serão realizado durante a janela de tempo $[a_i, b_i]$ e as restrições (4.26) impõem que às velocidades dos veículos respeitem limites permitidos. Restrições (4.27)-(4.29) dizem respeito à natureza das variáveis e as restrições (4.30) definem os tempos de início, de atendimento e de espera no depósito, inicialmente.

$$\text{Minimize } \sum_{i=0}^n f_c F_{i,i+1}(v_{i,i+1}) + f_d e_{n+1} \quad (4.23)$$

$$\text{sujeito a } \quad e_{i+1} = e_i + w_i + t_i + d_{i,i+1}/v_{i,i+1} \quad i = 0, \dots, n \quad (4.24)$$

$$a_i \leq e_i + w_i \leq b_i \quad i = 1, \dots, n \quad (4.25)$$

$$v_{min} \leq v_{i,i+1} \leq v_{MAX} \quad i = 0, \dots, n \quad (4.26)$$

$$w_i \geq 0 \quad i = 1, \dots, n \quad (4.27)$$

$$e_i \geq 0 \quad i = 1, \dots, n + 1 \quad (4.28)$$

$$v_{i,i+1} \geq 0 \quad i = 0, \dots, n \quad (4.29)$$

$$e_0 = t_0 = w_0 = 0 \quad (4.30)$$

Percebe-se que o modelo de otimização das velocidades é não-linear devido o consumo de combustível ser uma função quadrática da velocidade (Equação 3.8; Figura 4.6).

Deste modo, para otimizar as velocidades de uma dada rota, um algoritmo recursivo é proposto (Algoritmo 7), o qual baseia-se em Norstad et al (2011) e Hvattum et al (2013). Tal algoritmo possui complexidade $\mathcal{O}(n^2)$ no pior caso. Diferentemente dos algoritmos apresentados por Norstad et al (2011) e Hvattum et al (2013), o Algoritmo (7) é capaz de otimizar as velocidades em situações em que o tempo de chegada no último cliente não é definido, além de considerar o custo com motorista e aceitar esperas antes do atendimento dos clientes.

O Algoritmo 7 é aplicado sobre um rota completa, definindo $s = 1$ e $e = |\sigma|$. Considerando t'_{σ_i} o tempo de chegada ao cliente σ_i sem aceitar atrasos e esperas, ou seja, o tempo de

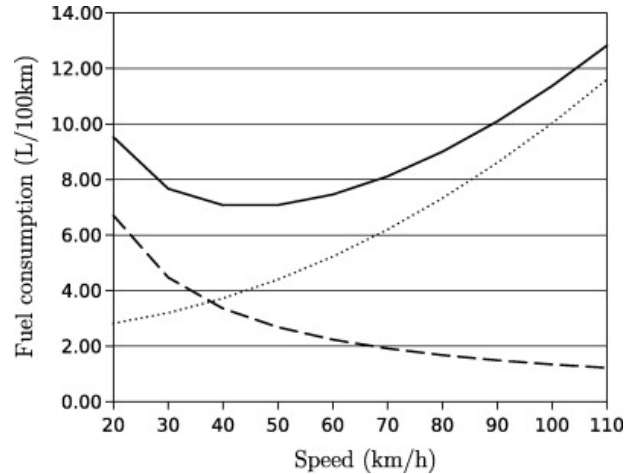


Figura 4.6: Consumo de combustível em função da velocidade
 Fonte: Bektaş e Laporte (2011)

início do atendimento do cliente σ_i e, ainda, $t'_{\sigma_1} = 0$, o algoritmo de otimização das velocidades (Algoritmo 7) funciona da seguinte maneira. Inicialmente, calcula-se o tempo de chegada $t'_{\sigma_{|\sigma|}}$ ao último cliente utilizando a velocidade v_{FD}^* (minimizando custos com combustível e motorista). Se $t'_{\sigma_{|\sigma|}}$ for maior ou menor que os limites de janela de tempo, o tempo é atualizado para o limite de janela de tempo mais próximo (Linha 6). Em seguida, a velocidade necessária v_{REF} para chegar em t'_{σ_e} é calculada (Linha 7), e o algoritmo encontra o cliente σ_p com maior violação de janela de tempo ao trafegar com a velocidade v_{REF} (Linhas 8-13). Se nenhuma violação for encontrada, retorna-se a solução. Caso contrário, o tempo de chegada t'_{σ_p} é atualizado para o limite de janela de tempo mais próximo (Linha 15) e o SOA é chamado recursivamente para dois subproblemas: de s à p (Linha 16), e de p à e (Linha ??).

Uma vez calculado os tempos $t'_{\sigma_i}, \forall i = 1, \dots, |\sigma|$, as velocidades associadas à cada arco são revisadas de modo que todas as velocidades inferiores à velocidade ótima v_F^* que minimiza o consumo de combustível seja substituída por v_F^* (Linha 21). Após o cálculo das velocidades ótimas em cada arco, o tempo de chegada em cada cliente é calculado de modo a aceitar esperas para o atendimento (Linha 22).

A Figura 4.7 ilustra um exemplo de execução do Algoritmo 7 em um problema com oito paradas. O eixo horizontal representa o tempo e o eixo vertical representa a sequência de clientes (de baixo para cima). Os colchetes correspondem aos limites das janelas de tempo e os pontos denotam os tempos de chegada aos clientes. Pontos pretos estão relacionados à tempos de chegada viáveis; pontos cinza indicam tempos de chegada antecipada ou atrasada; e os pontos

Algoritmo 7 Speed Optimization Algorithm — SOA

```

1: Procedimento  $SOA(\sigma, s, e)$ 
2:  $maxViolation \leftarrow 0$ 
3:  $D \leftarrow \sum_{i=s}^{e-1} d_{\sigma_i, \sigma_{i+1}}$ 
4:  $T \leftarrow \sum_{i=s}^{e-1} \tau_{\sigma_i}$ 
5: se  $e = |\sigma|$  então
6:    $t'_{\sigma_e} = \min\{\max\{a_{\sigma_e}, t'_{\sigma_s} + D/v_{FD}^* + T\}, b_{\sigma_e}\}$ 
7:  $v_{REF} \leftarrow D/(t'_{\sigma_e} - t'_{\sigma_s} - T)$ 
8: para  $i = s + 1 \dots e$  faça
9:    $t'_{\sigma_i} = t'_{\sigma_{i-1}} + \tau_{\sigma_{i-1}} + d_{\sigma_{i-1}, \sigma_i}/v_{REF}$ 
10:   $violation = \max\{0, t'_{\sigma_i} - b_{\sigma_i}, a_{\sigma_i} - t'_{\sigma_i}\}$ 
11:  se  $violation > maxViolation$  então
12:     $maxViolation = violation$ 
13:     $p = i$ 
14: se  $maxViolation > 0$  então
15:    $t'_{\sigma_p} = \min\{\max\{a_{\sigma_p}, t'_{\sigma_p}\}, b_{\sigma_p}\}$ 
16:    $SOA(\sigma, s, p)$ 
17:    $SOA(\sigma, p, e)$ 
18: se  $s = 1$  e  $e = |\sigma|$  então
19:    $t_{\sigma_1} = 0$ 
20:   para  $i = 2 \dots |\sigma|$  faça
21:      $v_{\sigma_{i-1}, \sigma_i} = \max\{d_{\sigma_{i-1}, \sigma_i}/(t'_{\sigma_i} - t'_{\sigma_{i-1}} - \tau_{\sigma_{i-1}}), v_F^*\}$ 
22:      $t_{\sigma_i} = \max\{a_{\sigma_{i-1}}, t_{\sigma_{i-1}}\} + \tau_{\sigma_{i-1}} + d_{\sigma_{i-1}, \sigma_i}/v_{\sigma_{i-1}, \sigma_i}$ 

```

quadrados indicam os tempos após correção das inviabilidades. Por simplicidade, os tempos de atendimento dos clientes foram desconsiderados.

A Figura 4.7a ilustra os tempos de chegada em cada cliente partindo de $\sigma_1 = 0$ no instante de tempo $t'_{\sigma_1} = a_{\sigma_1}$ utilizando a velocidade ótima (v_{FD}^*). Verificam-se duas violações, e σ_7 é o cliente com maior violação ($p = 7$). O tempo de chegada em σ_7 é, então, ajustado para $t'_{\sigma_7} = b_{\sigma_7}$ (Figura 4.7b) e dois subproblemas, P1 e P2, são resolvidos. P1 considera a subsequência de σ_1 até σ_7 , enquanto P2 considera a subsequência de σ_7 até $\sigma_{|\sigma|}$. Em P1, a velocidade v_{REF} é calculada de modo que o tempo de chegada em σ_7 seja t'_{σ_7} . Uma nova violação é observada no cliente σ_4 ($p = 4$). Essa violação é corrigida (Figura 4.7c) e dois novos subproblemas, de σ_1 à σ_4 (P1.1), e de σ_4 à σ_7 (P1.2), são resolvidos. Durante a resolução de P1.1 e P1.2, nenhuma violação é identificada (Figura 4.7c-4.7d) e, então, nenhuma recursão é aplicada e o Subproblema P1 está resolvido. O Subproblema P2 é resolvido utilizando o mesmo raciocínio de P1 (ver Figura 4.7e), porém calculando o tempo de chegada aos clientes utilizando a velocidade ótima v_{FD}^* (ver Linha 6 do Algoritmo 7). Novamente, nenhuma violação é encontrada, e o problema original está resolvido. Por fim, as velocidades são revisadas de modo a inserir tempos de espera nos casos em que as velocidades são inferiores às v_F^* (Linhas 18-22 do Algoritmo 7).

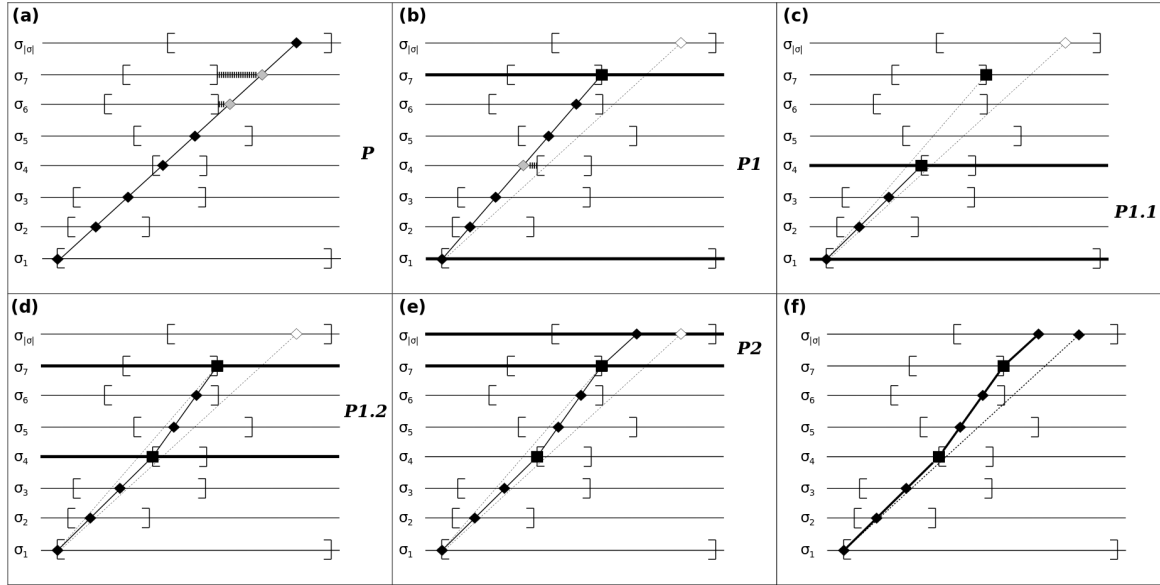


Figura 4.7: Cálculo dos tempos de chegada com o SOA

4.8 ABORDAGEM HÍBRIDA

Para aumentar a qualidade das soluções geradas, um modelo exato para o problema de particionamento de conjuntos foi incorporado ao ILS-SOA, baseado em Subramanian et al (2013). A abordagem híbrida de meta-heurísticas com métodos exatos é comumente denominada de *mat-heurística* (Maniezzo et al, 2009). Logo, a *mat-heurística* aqui apresentada será denominada de ILS-SP-SOA.

Sejam \mathcal{R} o conjunto de todas as possíveis rotas de todos os tipos de veículos, $\mathcal{R}_i \subseteq \mathcal{R}$ o subconjunto de rotas que contém o cliente $i \in \mathcal{V} - \{0\}$ e y_j uma variável binária associada à rota $j \in \mathcal{R}$ de custo c_j . A formulação do SPP é apresentada em (4.31) - (4.34).

$$\text{Minimize } \sum_{j \in \mathcal{R}} c_j y_j \quad (4.31)$$

$$\text{sujeito a } \sum_{j \in \mathcal{R}_i} y_j = 1 \quad \forall i \in \mathcal{V} - \{0\} \quad (4.32)$$

$$\sum_{j \in \mathcal{R}} y_j = |\mathbf{R}| \quad (4.33)$$

$$y_j \in \{0, 1\} \quad \forall j \in \mathcal{R}. \quad (4.34)$$

A função objetivo (4.31) minimiza a soma dos custos das rotas. As restrições (4.32)

garantem que uma única rota do subconjunto \mathcal{R}_i visite o cliente $i \in \mathcal{V}$. Dado que a enumeração do conjunto \mathcal{R} é uma tarefa impraticável, o ILS-SP-SOA considera apenas um subconjunto deste. Seja $|\mathbf{R}|$ o número de veículos. As restrições (4.33) garantem que a quantidade utilizada de veículos será igual ao número de veículos utilizados na melhor solução encontrada pela heurística.

O pseudocódigo do procedimento *Set Partitioning* (SP) é apresentado no Algoritmo 8. O parâmetro de entrada T_{MIP} define o limite de tempo imposto ao resolvidor do modelo de Programação Inteira Mista (PIM). Assume-se que o resolvidor PIM usa um procedimento *branch-and-bound* ou *branch-and-cut*. Inicialmente, o modelo SP é criado (Linha 2) e o valor *Cutoff* é inicializado (Linha 3) com o melhor custo encontrado pela meta-heurística. O modelo SP é dado como parâmetro de entrada do resolvidor de PIM (Linha 4) que chama o ILS-SOA sempre que uma nova solução incumbente é encontrada (procedimento *IncumbentCallback*). Se a solução s^* é melhorada no *IncumbentCallback*, o valor *Cutoff* é atualizado (Linhas 10-12), mas s^* não retorna ao resolvidor dado que pode conter rotas que não pertençam ao subconjunto \mathcal{R} . O resolvidor interrompe sua execução se: (i) a solução ótima é encontrada; (ii) *Lower Bound* $>$ *Cutoff*; (iii) ou T_{MIP} é excedido.

Algoritmo 8 SP

```

1: Procedimento SP( $s^*$ ,  $PoolDeRotas$ ,  $T_{\text{MIP}}$ );
2:  $Modelo\_SP \leftarrow \text{CriaModeloProbCoberturaConjuntos}(PoolDeRotas, s^*)$ ;
3:  $Cutoff \leftarrow f(s^*)$ ;
4:  $s^* \leftarrow \text{ResolvidorMIP}(Modelo\_SP, s^*, Cutoff, T_{\text{MIP}}, \text{IncumbentCallback}(s^*))$ ;
5: retorne  $s^*$ ;
6: fim SP.
7: Procedimento IncumbentCallback( $s^*$ )
8:  $s \leftarrow$  Solução Incumbente
9:  $s \leftarrow \text{ILS-SOA}(1, n_{\text{ILS}}, s)$ 
10: se  $f(s) < f(s^*)$  então
11:    $s^* \leftarrow s$ 
12:    $Cutoff \leftarrow f(s)$ 
13: fim IncumbentCallback

```

Encontrar um equilíbrio entre a tratabilidade computacional e o potencial de melhoramento em um modelo de PIM é um dos desafios no desenvolvimento de uma abordagem híbrida unificada. Por exemplo, um modelo SP que exceda o limite de tempo apenas para resolver a relaxação linear (e.g. devido ao número elevado de rotas) não é um mecanismo adequado de melhoramento. Por outro lado, um modelo SP que contenha poucas rotas é facilmente solucionável mas raramente melhora as soluções. Assim, é necessário que o modelo SP gerado encontre

um equilíbrio entre a tratabilidade computacional e o potencial de melhoramento (Subramanian, 2012).

Tal equilíbrio pode ser encontrado, entre diversas formas, em função do número de clientes da instância. Para tanto, duas estratégias foram desenvolvidas. Na primeira, executada quando o número de clientes (n) é menor ou igual ao parâmetro n_{SP} , o procedimento SP é executado apenas uma vez, após a heurística ILS-SOA. A segunda, executada quando $n > n_{SP}$, o procedimento SP é executado após cada iteração da heurística ILS-SOA.

4.8.1 Versão Final do Algoritmo — ILS-SP-SOA

Ao incorporar o SPP ao ILS-SOA torna-se necessária algumas modificações à estrutura do algoritmo de modo a armazenar rotas durante sua execução, conforme apresentado no Algoritmo 9.

O pseudocódigo da mat-heurística ILS-SP-SOA é apresentado no Algoritmo 9. O algoritmo é reinicializado n_R vezes (Lines 3-23) combinando os procedimentos ILS, RVND, SOA e SP. A cada reinicialização, a matriz de velocidades \mathbf{v} é inicializada com a velocidade máxima, $v_{ij} = v_{MAX}, \forall (i, j) \in \mathcal{A}$. Uma solução S é obtida ao aplicar o procedimento RVND e SOA em uma solução inicial (Linha 8). As velocidades dos arcos associadas à S são, então, atualizadas na matriz \mathbf{v} .

O procedimento ILS é executado por n_{ILS} iterações consecutivas sem melhora. A cada iteração (Linhas 10-18) a solução ótima local é modificada por um dos três mecanismos de perturbação conforme descrito na Subseção 4.6. A solução modificada é possivelmente melhorada ao se aplicar o procedimento RVND e o SOA (Linha 12), e a matriz de velocidades é atualizada (Linha 13). Um *pool* temporário de rotas é atualizado durante o *loop* do ILS adicionando rotas associadas às soluções ótimas locais (Linha 14). Por fim, a matriz de velocidades \mathbf{v} é reinicializada quando o número de iteração do ILS exceder $n_{ILS}/2$ (Linhas 17-18).

Se o tamanho da instancia é menor n_{SP} , então o procedimento SP é chamado uma única vez após a última fase de reinicialização; caso contrário, o procedimento SP é chamado após cada fase de reinicialização. O procedimento SP procura uma nova solução formada por rotas de um *pool* temporário de rotas P_{TEMP} derivados de ótimos locais obtidos pelo RVND, que é

esvaziado a cada n_{POOL} reinicializações (Linhas 24-25), e de um *pool* permanente de rotas P_{PERM} que contém rotas associadas às melhores soluções S_{BEST} de cada fase de reinicialização (Linha 23). Os SPPs são resolvidos utilizando um resolvidor de PIM que chama o ILS-SOA toda vez que uma nova solução incumbente é encontrada (Linha 20). Essa abordagem colaborativa, descrita em Subramanian et al (2013), não só reduz o tempo de execução do resolvidor, como também contribui para encontrar soluções melhores que a melhor combinação possível de rotas dos *pools* de rotas. Se isto ocorrer, o *pool* P_{TEMP} é atualizado. O resolvidor PIM é executado diversas vezes até que nenhuma melhora seja encontrada sobre S_{BEST} , e um tempo limite T_{MIP} é imposto à cada execução do resolvidor PIM para evitar elevados tempos de processamento. Por fim, o algoritmo retorna a melhor solução encontrado dentre todas as fases (Linhas 26).

Algoritmo 9 ILS-SP-SOA($n_{\text{R}}, n_{\text{ILS}}, n_{\text{SP}}, n_{\text{POOL}}, T_{\text{MIP}}, \text{seed}$)

```

1:  $S_{\text{BEST-ALL}} \leftarrow \emptyset; f(S_{\text{BEST-ALL}}) \leftarrow \infty; \{S_{\text{BEST-ALL}} \text{ is the overall best solution}\}$ 
2:  $P_{\text{PERM}} \leftarrow \emptyset; i_{\text{R}} = 0; \{P_{\text{PERM}} \text{ is the pool of permanent routes in the SP}\}$ 
3: enquanto  $i_{\text{R}} < n_{\text{R}}$  faça
4:    $i_{\text{R}} \leftarrow i_{\text{R}} + 1;$ 
5:    $S_{\text{BEST}} \leftarrow \emptyset; f(S_{\text{BEST}}) \leftarrow \infty; \{S_{\text{BEST}} \text{ is the best solution of the restart phase}\}$ 
6:    $P_{\text{TEMP}} \leftarrow \emptyset; i_{\text{ILS}} \leftarrow 0; \{P_{\text{TEMP}} \text{ is the pool of temporary routes in the SP}\}$ 
7:    $\mathbf{v} \leftarrow \text{InitializeSpeedMatrix}(v_{\text{MAX}});$ 
8:    $S \leftarrow \text{SpeedOptimization}(\text{LocalSearch}(\text{GenInitSol}(\text{seed})));$ 
9:    $\mathbf{v} \leftarrow \text{UpdateSpeedMatrix}(S);$ 
10:  enquanto  $i_{\text{ILS}} < n_{\text{ILS}}$  faça
11:     $i_{\text{ILS}} \leftarrow i_{\text{ILS}} + 1;$ 
12:     $S \leftarrow \text{SpeedOptimization}(\text{LocalSearch}(\text{Perturbation}(S_{\text{BEST}}, \text{seed})));$ 
13:     $\mathbf{v} \leftarrow \text{updateSpeedMatrix}(S);$ 
14:     $P_{\text{TEMP}} \leftarrow P_{\text{TEMP}} \cup (\text{Feasible routes of } S);$ 
15:    se  $f(S) < f(S_{\text{BEST}})$  então
16:       $S_{\text{BEST}} \leftarrow S; i_{\text{ILS}} \leftarrow 0;$ 
17:    se  $i_{\text{ILS}} = n_{\text{ILS}}/2$  então
18:       $\mathbf{v} \leftarrow \text{ReinitializeSpeedMatrix}(v_{\text{MAX}}, S_{\text{BEST}});$ 
19:    se  $(n \leq n_{\text{SP}} \text{ e } i_{\text{R}} = n_{\text{R}} - 1)$  ou  $n > n_{\text{SP}}$  então
20:       $S_{\text{BEST}} \leftarrow \text{MIPSolver}(S_{\text{BEST}}, P_{\text{TEMP}}, P_{\text{PERM}}, n_{\text{ILS}}, T_{\text{MIP}});$ 
21:    se  $f(S_{\text{BEST}}) < f(S_{\text{BEST-ALL}})$  então
22:       $S_{\text{BEST-ALL}} \leftarrow S_{\text{BEST}};$ 
23:     $P_{\text{PERM}} \leftarrow P_{\text{PERM}} \cup (\text{Feasible routes of } S_{\text{BEST}});$ 
24:    se  $\text{number of consecutive restarts} \geq n_{\text{POOL}}$  então
25:       $P_{\text{TEMP}} \leftarrow \emptyset;$ 
26:  retorne  $S_{\text{BEST-ALL}}$ 

```

5 RESULTADOS COMPUTACIONAIS

A mat-heurística ILS-SP-SOA foi implementada em linguagem de programação C++. A fim de verificar a eficiência do algoritmo proposto foram resolvidas algumas instâncias para o PRP e outras variantes, como o VRPTW, o FCVRP e o EMVRP. Inicialmente, na Seção 5.1 são descritos os conjuntos de instâncias considerados para cada problema. Nas Seções 5.2, 5.3, 5.4 e 5.5 são apresentados os resultados obtidos para o PRP, o VRPTW, o FCVRP e o EMVRP, respectivamente.

5.1 Instâncias

O primeiro conjunto de instâncias considerado neste trabalho foram as da biblioteca PRPLIB, propostas por Demir et al (2012). As instâncias da PRPLIB estão agrupadas em nove diferentes grupos, variando de 10 a 200 clientes, cada um contendo 20 diferentes instâncias.

Entretanto, as instâncias propostas em Demir et al (2012) possuem tamanhos de janela de tempo muito grandes, de modo que visitar os clientes dentro do sua respectiva janela de tempo não é algo tão difícil. Deste modo, dois novos conjuntos de instâncias, com tamanhos de janela de tempo mais apertados, são propostos. Foram adotados os mesmos dados das instâncias da PRPLIB, porém com diferentes janelas de tempo. O horizonte de tempo foi definido como sendo igual a 32400 segundos. O tamanho das janelas de tempo do primeiro conjunto de instâncias (conjunto *B*) foram geradas aleatoriamente, com probabilidades uniformemente distribuídas, entre o intervalo 2000 e 5000 segundos, enquanto no segundo (conjunto *C*) foram selecionadas entre 2000 e 15000 segundos.

Uma vez definido o tamanho, o limite inferior da janela de tempo é selecionado aleatoriamente, garantindo que os clientes poderão ser servidos dentro do intervalo e, também, que os veículos poderão retornar ao depósito antes do término do horizonte de tempo. Desta forma, as janelas de tempo de cada cliente i foram geradas conforme os seguintes passos:

1. Calcular a duração mínima do depósito ao cliente i , denotado por $\delta_{0i_{min}}$. Mais precisamente, $\delta_{0i_{min}} = d_{0i}/v_{max}$;
2. Calcular a duração mínima do cliente i ao depósito, denotado por $\delta_{i0_{min}}$. Mais precisamente, $\delta_{i0_{min}} = d_{i0}/v_{max}$;
3. Calcular o tamanho da janela de tempo: $width_i = 2000 + \text{RandBetween}[0, 3000]$ (para o Conjunto B) ou $width_i = 2000 + \text{RandBetween}[0, 13000]$ (para o Conjunto C);
4. Calcular o limite inferior da janela de tempo: $a_i = \text{RandBetween}[\lceil \delta_{0i_{min}} \rceil, b_0 - \lceil \delta_{i0_{min}} \rceil - \tau_i - width_i]$;
5. Calcular o limite superior da janela de tempo: $b_i = a_i + width_i$;

Para a resolução das instâncias do PRP, os mesmos parâmetros adotados Demir et al (2012) foram utilizados, ou seja:

$$w_1 = 1,01763908 \times 10^{-3}$$

$$w_2 = 5,33605218 \times 10^{-5}$$

$$w_3 = 8,40323178 \times 10^{-9}$$

$$w_4 = 1,41223439 \times 10^{-7}$$

$$\omega_{FC} = 1,4\text{£}/l$$

$$\omega_{FD} = 2,2222222 \times 10^{-3}\text{£}/s.$$

Para o VRPTW foram resolvidas as instâncias de Solomon (1987) de 100 clientes, objetivando a redução da distância total percorrida. Já para o FCVRP e o EMVRP foram resolvidas as instâncias propostas por Christofides et al (1979) e Golden et al (1998) para o CVRP. Para o EMVRP, o valor adotado para o parâmetro ω foi o mesmo utilizado por Kara et al (2007), i.e., $\omega = 0.15Q$.

5.2 Resultados para o PRP

Os testes realizados nas instâncias do PRP foram executadas em um *cluster* de computadores com processador Xeon 3.07 GHz com 16 GB de memória RAM e sistema operacional Oracle Linux Server 6.4. O resolvidor CPLEX 12.4 foi utilizado para resolver os Problemas de Particionamento de Conjuntos (SPP). Uma única *thread* foi considerada nos testes e cada instância foi resolvida 10 vezes.

Os seguintes valores foram adotados como parâmetros: $n_R = 20$, $n_{ILS} = n + 5r$, $n_{SP} = 150$, $n_{POOL} = 2$, $T_{MIP} = 360$ segundos, e $\omega_{TW} = 10^8$. Os quatro primeiros parâmetros foram calibrados de acordo com Subramanian et al (2012) e Subramanian et al (2013), enquanto o último se trata de um número grande para evitar soluções inviáveis.

Além do algoritmo proposto no Capítulo 4, aqui denotado como ILS-SP-SOA-Dyn, onde a matriz \mathbf{v} é modificada dinamicamente durante a execução do algoritmo, uma versão estática (i.e. \mathbf{v} não muda durante a execução do algoritmo), denominada de ILS-SP-SOA-Stat, também foi implementada, para efeitos de comparação.

Nas tabelas apresentadas a seguir, *Instances* denotam o nome do problema, *Cost* indica o valor da função objetivo, $|\mathbf{R}|$ é o número de veículos da solução, *CPU T (s)* representa o tempo de processamento em segundos, *Gap (%)* corresponde à diferença percentual entre o custo da solução e o custo obtido pelo CPLEX (para instâncias de 10 clientes) ou entre o custo da melhor solução obtida dentre todos os experimentos, denotados nas tabelas por *BKS*. Vale frisar que as soluções do CPLEX foram obtidas a partir da formulação de PIM proposta por Demir et al (2012), as quais se tratam de soluções ótimas aproximadas devido a discretização das variáveis contínuas de velocidades. De modo a se obter uma boa aproximação, os resultados do CPLEX apresentados foram obtidos a partir de uma discretização das velocidades em 500 níveis, variando de v_F^* à v_{FD}^* .

A Tabela 5.1 mostra os resultados obtidos para as instâncias originais da PRPLIB. Os resultados são comparados com os apresentados por Demir et al (2012). Verifica-se que a qualidade das soluções obtidas por ambas as versões (estática e dinâmica) do ILS-SP-SOA são bem melhores que os alcançados pela heurística ALNS de Demir et al (2012). Comparando as versões estáticas e dinâmicas do ILS-SP-SOA, verifica-se que as duas possuem tempo de execução

similares, mas a versão dinâmica é um pouco melhor.

Os resultados obtidos para as instâncias dos Conjuntos B e C são apresentados nas Tabelas 5.2 e 5.3, respectivamente. Devido ao fato destas instâncias terem sido propostas neste trabalho de dissertação, os resultados são comparados apenas entre as versões estática e dinâmica do ILS-SP-SOA. A versão estática foi implementada para, além de verificar o efeito da utilização de matrizes dinâmicas de velocidades, comparar com uma abordagem similar à proposta por Demir et al (2012) para resolver as instâncias da PRPLIB.

Para as instâncias do Conjunto B e C (Tabelas 5.2 e 5.3), pode-se verificar uma ligeira vantagem do ILS-SP-SOA-Dyn ao se comparar a média dos *gaps* em relação às BKS. Verifica-se porém, que o tempo de processamento do ILS-SP-SOA-Dyn foi mais elevado, especialmente nas instâncias de 100 e 200 clientes. Tal diferença pode ser justificada por um número mais elevado de rotas no *pool* e/ou pela existência de rotas similares/parecidas com uma diferença de custo muito pequena entre si, dificultando a convergência do resolvidor do (sub)problema de particionamento de conjuntos.

No geral, pode-se concluir que ambas as versões do ILS-SP-SOA são bastante similares ao se comparar os *gaps* para as BKS, com uma certa vantagem para a versão dinâmica.

Quanto aos novos conjuntos de instâncias gerados, ao se observar os tempos médios de processamento ($CPU\ T(s)$) e a média dos *gaps* das soluções médias em relação as BKS, pode-se concluir que as novas instâncias são mais difíceis de serem resolvidas.

Com o intuito de ilustrar o impacto das janelas de tempo nas velocidades dos veículos e nas soluções, as Figuras 5.1-5.6 mostram as melhores soluções obtidas para as instâncias UK10_01 e UK10_08 dos conjuntos de instância da PRPLIB e para os conjuntos B e C propostos. As linhas vermelhas indicam que a velocidade v_F^* foi utilizada, as linhas pretas indicam que a velocidade v_{FD}^* foi utilizada e as linhas tracejadas indicam a utilização de outro nível de velocidade (diferente de v_F^* e v_{FD}^*). Verifica-se que nas instâncias da PRPLIB as janelas de tempo são maiores e a velocidade ótima v_{FD}^* é mais utilizada.

Tabela 5.1: Resultados para as instâncias da PRPLIB

Instance	ALNS				Avg. ILS-SP-SOA-Dyn				Avg. ILS-SP-SOA-Stat				BKS	
	Cost	R	CPU T(s)*	Gap (%)	Cost	R	CPU T(s)	Gap (%)	Cost	R	CPU T(s)	Gap (%)	Cost	R
UK10_01	170.64	2	2.1	0.00	170.64	2.0	0.04	0.00	170.64	2.0	0.05	0.00	170.64	2
UK10_02	204.88	2	2.3	0.00	204.88	2.0	0.05	0.00	204.88	2.0	0.05	0.00	204.88	2
UK10_03	200.42	3	2.0	0.04	200.60	3.0	0.04	0.13	200.62	3.0	0.04	0.14	200.34	3
UK10_04	189.99	2	2.2	0.05	189.88	2.0	0.05	-0.01	189.88	2.0	0.05	-0.01	189.89	2
UK10_05	175.59	2	2.3	0.00	175.59	2.0	0.05	0.00	175.59	2.0	0.06	0.00	175.59	2
UK10_06	214.48	2	2.2	0.00	214.48	2.0	0.06	0.00	214.48	2.0	0.05	0.00	214.48	2
UK10_07	190.14	2	2.9	0.00	190.14	2.0	0.05	0.00	190.14	2.0	0.05	0.00	190.14	2
UK10_08	222.17	2	2.1	0.00	222.17	2.0	0.04	0.00	222.17	2.0	0.05	0.00	222.17	2
UK10_09	174.54	2	2.2	0.00	174.54	2.0	0.05	0.00	174.54	2.0	0.05	0.00	174.54	2
UK10_10	190.04	2	2.6	0.12	189.82	2.0	0.06	0.00	189.82	2.0	0.06	0.00	189.82	2
UK10_11	262.08	2	2.2	0.00	262.08	2.0	0.04	0.00	262.08	2.0	0.05	0.00	262.08	2
UK10_12	183.19	2	2.2	0.00	183.19	2.0	0.05	0.00	183.19	2.0	0.05	0.00	183.19	2
UK10_13	195.97	2	2.2	0.00	195.97	2.0	0.05	0.00	195.97	2.0	0.05	0.00	195.97	2
UK10_14	163.28	2	2.4	0.07	163.17	2.0	0.02	0.00	163.17	2.0	0.02	0.00	163.17	2
UK10_15	127.24	2	2.4	0.11	127.10	2.0	0.02	0.00	127.10	2.0	0.01	0.00	127.10	2
UK10_16	186.73	2	1.9	0.05	186.63	2.0	0.05	0.00	186.63	2.0	0.05	0.00	186.63	2
UK10_17	159.03	2	2.3	0.00	159.03	2.0	0.05	0.00	159.03	2.0	0.05	0.00	159.03	2
UK10_18	162.09	2	2.2	0.00	162.09	2.0	0.05	0.00	162.09	2.0	0.05	0.00	162.09	2
UK10_19	169.59	2	4.1	0.08	169.46	2.0	0.02	0.00	169.46	2.0	0.02	0.00	169.46	2
UK10_20	168.80	2	2.0	0.00	168.80	2.0	0.04	0.00	168.80	2.0	0.05	0.00	168.80	2
Avg.			2.34	0.03			0.04	0.01			0.05	0.01		
UK100_01	1240.79	14	92.1	2.62	1209.65	14.0	28.65	0.04	1210.15	14.0	30.46	0.09	1209.11	14
UK100_02	1168.17	13	98.2	1.86	1147.04	13.0	27.67	0.02	1147.51	13.0	27.52	0.06	1146.79	13
UK100_03	1092.73	13	207.9	1.30	1078.92	13.0	29.59	0.02	1079.15	13.0	30.53	0.04	1078.75	13
UK100_04	1106.48	14	149.7	2.90	1075.40	14.0	59.84	0.01	1076.12	14.0	74.19	0.08	1075.29	14
UK100_05	1043.41	14	159.0	1.41	1028.86	14.0	28.10	0.00	1029.24	14.0	27.54	0.04	1028.86	14
UK100_06	1213.61	14	133.8	1.70	1193.38	14.0	40.31	0.00	1193.38	14.0	42.24	0.00	1193.38	14
UK100_07	1060.08	12	102.6	1.44	1045.14	12.0	24.40	0.01	1045.39	12.0	24.17	0.04	1045.02	12
UK100_08	1106.78	13	209.5	1.55	1090.47	12.7	25.81	0.06	1090.61	12.9	23.38	0.07	1089.84	12
UK100_09	1015.46	13	154.0	2.74	988.58	13.0	20.42	0.02	989.83	13.0	24.03	0.14	988.41	13
UK100_10	1076.56	12	199.0	1.57	1059.95	12.0	36.34	0.00	1059.95	12.0	35.36	0.00	1059.95	12
UK100_11	1210.25	15	107.1	1.15	1196.52	14.0	63.68	0.00	1196.60	14.0	46.34	0.01	1196.50	14
UK100_12	1053.02	12	206.4	2.50	1027.80	12.0	49.20	0.04	1028.05	12.0	41.59	0.07	1027.38	12
UK100_13	1154.83	13	87.9	2.01	1132.19	13.0	24.25	0.01	1132.16	13.0	24.24	0.01	1132.03	13
UK100_14	1264.50	14	91.8	1.76	1242.68	14.0	21.32	0.00	1242.68	14.0	20.96	0.00	1242.68	14
UK100_15	1315.50	15	110.9	1.18	1301.25	15.0	32.59	0.09	1301.03	15.0	31.72	0.07	1300.13	15
UK100_16	1005.03	12	254.7	2.36	981.88	12.0	41.10	0.00	982.32	12.0	31.05	0.05	981.86	12
UK100_17	1284.81	15	152.8	2.12	1258.33	15.0	33.54	0.01	1258.67	15.0	36.66	0.04	1258.16	15
UK100_18	1106.00	13	92.6	3.04	1078.80	12.8	33.95	0.50	1079.49	12.8	29.13	0.57	1073.38	12
UK100_19	1044.71	13	91.0	2.83	1016.54	13.0	43.42	0.06	1017.05	13.0	42.86	0.11	1015.95	13
UK100_20	1263.06	14	204.4	1.86	1240.34	14.0	23.49	0.03	1240.37	14.0	24.04	0.03	1240.00	14
Avg.			145.27	1.99			34.38	0.05			33.40	0.07		
UK200_01	2111.70	28	724.4	4.86	2021.12	27.6	329.59	0.36	2024.00	27.7	262.83	0.50	2013.84	28
UK200_02	1988.64	24	1020.9	4.31	1909.00	24.0	415.31	0.13	1909.00	24.0	384.75	0.13	1906.46	24
UK200_03	2017.63	27	404.1	3.55	1950.10	27.0	149.10	0.08	1950.00	27.0	131.02	0.08	1948.50	27
UK200_04	1934.13	26	411.7	4.37	1857.66	26.0	329.57	0.24	1857.00	26.0	244.61	0.21	1853.16	26
UK200_05	2182.91	27	926.4	3.88	2103.62	27.0	145.65	0.11	2105.00	27.0	153.68	0.17	2101.34	27
UK200_06	1883.22	27	450.3	3.94	1813.74	26.9	155.66	0.10	1815.00	26.7	160.20	0.17	1811.85	27
UK200_07	2021.95	27	943.4	4.96	1928.35	27.0	165.87	0.10	1930.00	27.0	227.12	0.18	1926.47	27
UK200_08	2116.76	27	430.6	4.02	2037.44	27.0	199.19	0.13	2039.00	27.0	301.91	0.20	2034.87	27
UK200_09	1894.18	25	553.1	6.01	1788.87	25.0	161.49	0.11	1788.00	25.0	172.92	0.07	1786.83	25
UK200_10	2199.95	28	500.0	3.55	2127.34	27.0	239.05	0.14	2128.00	27.0	188.78	0.17	2124.44	27
UK200_11	1941.19	27	842.8	4.75	1854.82	27.0	144.87	0.09	1856.00	27.0	151.99	0.16	1853.12	27
UK200_12	2105.14	25	711.0	2.41	2056.93	25.0	201.33	0.07	2058.00	25.0	172.53	0.12	2055.55	25
UK200_13	2141.26	25	444.6	4.42	2055.93	25.0	684.89	0.26	2058.00	25.0	591.50	0.36	2050.70	25
UK200_14	2011.35	27	450.6	3.57	1945.11	27.0	160.29	0.15	1946.00	27.0	175.03	0.20	1942.10	27
UK200_15	2110.86	25	542.0	5.21	2013.25	25.2	216.47	0.34	2012.00	25.2	358.38	0.28	2006.42	25
UK200_16	2075.83	27	455.6	4.76	1983.77	27.0	161.90	0.11	1984.00	27.0	153.60	0.12	1981.57	27
UK200_17	2218.28	26	409.2	4.55	2122.45	26.0	277.15	0.03	2123.00	26.0	269.87	0.06	2121.77	26
UK200_18	2004.68	27	788.9	3.12	1955.38	26.7	178.04	0.58	1954.00	26.7	162.75	0.51	1944.03	26
UK200_19	1844.90	25	973.9	4.70	1763.94	25.0	142.42	0.10	1764.00	25.0	176.61	0.11	1762.14	25
UK200_20	2150.57	27	531.1	3.71	2077.46	26.1	447.91	0.18	2079.00	26.0	408.78	0.26	2073.68	26
Avg.			625.73	4.23			245.29	0.17			242.44	0.20		

Tabela 5.2: Resultados para as instâncias propostas para o PRP — Conjunto B

Instance	Avg. Sol. ILS-SP-SOA-Dyn				Avg. Sol. ILS-SP-SOA-Stat				BKS	
	Cost	R	CPU T (s)	Gap (%)	Cost	R	CPU T (s)	Gap (%)	Cost	R
UK10_01-B	246.45	2.0	0.04	0.00	246.45	2.0	0.05	0.00	246.45	2
UK10_02-B	303.73	2.0	0.06	0.00	303.73	2.0	0.06	0.00	303.73	2
UK10_03-B	301.89	3.0	0.04	0.00	301.89	3.0	0.04	0.00	301.89	3
UK10_04-B	273.90	2.0	0.05	0.00	273.90	2.0	0.05	0.00	273.91	2
UK10_05-B	255.07	2.0	0.07	0.00	255.07	2.0	0.06	0.00	255.08	2
UK10_06-B	332.34	3.0	0.04	0.00	332.34	3.0	0.05	0.00	332.34	3
UK10_07-B	314.64	3.0	0.04	0.00	314.64	3.0	0.05	0.00	314.64	3
UK10_08-B	339.36	2.0	0.05	0.00	339.36	2.0	0.06	0.00	339.37	2
UK10_09-B	261.10	2.0	0.04	0.00	261.10	2.0	0.06	0.00	261.10	2
UK10_10-B	285.20	2.0	0.05	0.00	285.20	2.0	0.06	0.00	285.20	2
UK10_11-B	410.66	3.0	0.04	0.52	410.66	3.0	0.05	0.52	408.55	3
UK10_12-B	252.18	2.0	0.05	0.21	252.18	2.0	0.06	0.21	251.65	2
UK10_13-B	274.07	3.0	0.05	0.00	274.07	3.0	0.05	0.00	274.08	3
UK10_14-B	267.92	2.0	0.05	0.00	267.92	2.0	0.06	0.00	267.92	2
UK10_15-B	197.50	2.0	0.02	0.00	197.50	2.0	0.02	0.00	197.50	2
UK10_16-B	245.76	2.0	0.04	0.00	245.76	2.0	0.05	0.00	245.76	2
UK10_17-B	283.83	2.0	0.05	0.00	283.83	2.0	0.06	0.00	283.83	2
UK10_18-B	245.00	2.0	0.05	1.44	245.00	2.0	0.05	1.44	241.53	2
UK10_19-B	330.38	3.0	0.04	0.00	330.38	3.0	0.04	0.00	330.38	3
UK10_20-B	208.06	2.0	0.05	0.00	208.06	2.0	0.05	0.00	208.06	2
Avg.			0.05	0.11			0.05	0.11		
UK100_01-B	1593.53	14.0	105.34	0.15	1594.28	14.0	42.66	0.19	1591.20	14
UK100_02-B	1599.88	13.0	161.71	0.02	1601.96	13.0	52.08	0.15	1599.56	13
UK100_03-B	1504.15	13.0	219.58	0.25	1505.99	13.0	126.29	0.37	1500.40	13
UK100_04-B	1472.73	14.0	319.04	0.00	1473.00	14.0	80.93	0.02	1472.73	14
UK100_05-B	1490.50	14.1	235.23	0.07	1490.73	14.0	87.32	0.08	1489.51	15
UK100_06-B	1648.28	14.0	83.07	0.15	1649.00	14.0	33.93	0.20	1645.77	14
UK100_07-B	1508.41	12.0	84.86	0.02	1508.76	12.0	34.90	0.05	1508.05	12
UK100_08-B	1467.80	12.0	290.95	0.08	1473.74	12.3	50.62	0.49	1466.62	12
UK100_09-B	1379.53	13.0	68.92	0.14	1379.58	13.0	35.45	0.14	1377.64	13
UK100_10-B	1478.28	12.0	89.08	0.07	1479.68	12.0	34.43	0.16	1477.25	13
UK100_11-B	1620.01	14.0	69.89	0.07	1619.93	14.0	43.11	0.06	1618.94	14
UK100_12-B	1362.30	12.0	183.65	0.01	1362.39	12.0	41.56	0.02	1362.14	12
UK100_13-B	1606.08	13.0	103.51	0.01	1606.42	13.0	45.49	0.03	1605.99	13
UK100_14-B	1690.25	14.0	115.89	0.00	1690.25	14.0	44.74	0.00	1690.25	14
UK100_15-B	1734.92	15.0	58.09	0.00	1735.17	15.0	34.00	0.01	1734.92	15
UK100_16-B	1381.56	12.0	95.03	0.04	1383.84	12.0	40.45	0.20	1381.03	13
UK100_17-B	1678.21	15.0	181.89	0.01	1678.32	15.0	39.51	0.02	1678.04	15
UK100_18-B	1506.16	12.8	212.83	0.38	1507.65	12.8	59.65	0.48	1500.48	13
UK100_19-B	1400.58	13.0	118.21	0.02	1400.68	13.0	40.22	0.03	1400.28	13
UK100_20-B	1628.93	14.0	44.93	0.00	1629.00	14.0	25.95	0.01	1628.89	14
Avg.			142.09	0.07			49.66	0.14		
UK200_01-B	2732.98	27.9	1850.81	0.18	2736.96	28.0	938.06	0.33	2728.01	28
UK200_02-B	2578.30	24.0	1202.26	0.29	2583.30	24.0	665.27	0.48	2570.95	24
UK200_03-B	2760.63	27.0	2857.83	0.30	2758.15	27.0	1161.90	0.21	2752.24	28
UK200_04-B	2589.18	26.0	2770.41	0.43	2590.58	26.0	1593.64	0.49	2578.03	27
UK200_05-B	2810.10	26.9	2046.84	0.31	2818.30	27.0	906.80	0.61	2801.33	27
UK200_06-B	2579.76	26.0	2277.52	0.40	2576.78	26.0	664.12	0.28	2569.57	27
UK200_07-B	2660.93	27.0	2130.63	0.17	2662.59	27.0	1114.23	0.23	2656.45	27
UK200_08-B	2769.52	27.0	3067.32	0.55	2769.80	27.0	1432.06	0.56	2754.46	27
UK200_09-B	2494.22	25.0	1338.48	0.20	2494.68	25.0	691.68	0.22	2489.19	25
UK200_10-B	2889.83	27.0	1701.55	0.22	2890.13	27.0	879.63	0.23	2883.48	28
UK200_11-B	2591.21	27.0	1968.04	0.29	2597.51	27.0	1007.32	0.53	2583.74	27
UK200_12-B	2762.25	25.0	3054.28	0.17	2762.54	25.0	1679.25	0.18	2757.61	25
UK200_13-B	2737.41	25.0	1593.24	0.27	2734.75	25.0	1166.04	0.17	2730.06	26
UK200_14-B	2691.48	27.0	1266.70	0.25	2690.92	27.0	826.07	0.23	2684.68	27
UK200_15-B	2778.54	25.1	1821.23	0.46	2779.84	25.3	1028.80	0.51	2765.68	26
UK200_16-B	2741.18	27.0	2349.99	0.65	2736.27	26.9	882.35	0.47	2723.54	27
UK200_17-B	2822.11	26.0	3357.44	0.25	2820.06	26.0	865.92	0.17	2815.16	26
UK200_18-B	2722.89	26.5	1979.02	0.53	2727.18	26.5	1404.65	0.69	2708.62	27
UK200_19-B	2449.30	25.0	1448.64	0.13	2449.69	25.0	736.44	0.14	2446.17	25
UK200_20-B	2817.58	26.0	2540.67	0.41	2818.53	26.0	1651.50	0.45	2806.02	26
Avg.			2131.15	0.32			1064.79	0.36		

Tabela 5.3: Resultados para as instâncias propostas para o PRP — Conjunto C

Instance	Avg. Sol. ILS-SP-SOA-Dyn				Avg. Sol. ILS-SP-SOA-Stat				BKS	
	Cost	R	CPU T (s)	Gap (%)	Cost	R	CPU T (s)	Gap (%)	Cost	R
UK10_01-C	210.18	2.0	0.04	0.00	210.18	2.0	0.05	0.00	210.18	2
UK10_02-C	271.93	2.0	0.06	0.00	271.93	2.0	0.06	0.00	271.93	2
UK10_03-C	229.18	2.0	0.04	0.00	229.18	2.0	0.04	0.00	229.18	2
UK10_04-C	230.58	2.0	0.05	0.00	230.58	2.0	0.06	0.00	230.59	2
UK10_05-C	205.49	2.0	0.05	0.00	205.49	2.0	0.05	0.00	205.49	2
UK10_06-C	255.82	2.0	0.05	0.00	255.82	2.0	0.05	0.00	255.82	2
UK10_07-C	217.79	2.0	0.06	0.00	217.79	2.0	0.05	0.00	217.79	2
UK10_08-C	251.29	2.0	0.05	0.00	251.29	2.0	0.05	0.00	251.29	2
UK10_09-C	186.04	2.0	0.05	0.00	186.04	2.0	0.05	0.00	186.04	2
UK10_10-C	231.62	2.0	0.07	0.00	231.62	2.0	0.06	0.00	231.62	2
UK10_11-C	298.20	2.0	0.05	0.00	298.20	2.0	0.05	0.00	298.20	2
UK10_12-C	206.82	2.0	0.05	0.12	208.43	2.0	0.05	0.90	206.58	2
UK10_13-C	211.75	2.0	0.05	0.00	211.75	2.0	0.06	0.00	211.75	2
UK10_14-C	209.07	2.0	0.05	0.00	209.07	2.0	0.06	0.00	209.07	2
UK10_15-C	176.56	2.0	0.02	0.00	176.56	2.0	0.01	0.00	176.56	2
UK10_16-C	229.15	2.0	0.05	0.00	229.15	2.0	0.06	0.00	229.15	2
UK10_17-C	219.20	2.0	0.05	0.00	219.20	2.0	0.05	0.00	219.20	2
UK10_18-C	195.04	2.0	0.05	0.00	195.72	2.0	0.05	0.35	195.04	2
UK10_19-C	218.19	2.0	0.06	0.00	218.19	2.0	0.05	0.00	218.19	2
UK10_20-C	189.56	2.0	0.06	0.00	189.56	2.0	0.05	0.00	189.56	2
Avg.			0.05	0.01			0.05	0.06		
UK100_01-C	1487.36	14.0	59.54	0.07	1489.69	14.0	40.64	0.23	1486.34	14
UK100_02-C	1431.69	13.0	69.38	0.01	1432.47	13.0	42.32	0.06	1431.55	13
UK100_03-C	1323.69	13.0	122.78	0.06	1324.24	13.0	72.00	0.10	1322.93	13
UK100_04-C	1378.00	14.0	142.05	0.02	1378.48	14.0	83.20	0.05	1377.73	14
UK100_05-C	1307.11	14.0	67.19	0.08	1307.52	14.0	52.33	0.11	1306.04	14
UK100_06-C	1486.08	14.0	73.48	0.01	1486.36	14.0	40.76	0.02	1485.99	14
UK100_07-C	1332.08	12.0	255.17	0.03	1333.25	12.0	73.38	0.12	1331.67	12
UK100_08-C	1378.52	12.4	114.46	0.39	1376.59	12.2	50.98	0.25	1373.20	13
UK100_09-C	1271.83	13.0	51.10	0.08	1273.20	13.0	45.88	0.19	1270.84	13
UK100_10-C	1332.00	12.0	102.06	0.12	1333.43	12.0	50.87	0.23	1330.41	12
UK100_11-C	1499.15	14.0	46.01	0.00	1499.15	14.0	32.76	0.00	1499.15	14
UK100_12-C	1234.80	12.0	110.75	0.12	1236.30	12.0	58.32	0.24	1233.28	12
UK100_13-C	1442.65	13.0	105.85	0.00	1442.67	13.0	42.03	0.00	1442.65	13
UK100_14-C	1553.78	14.0	58.10	0.05	1553.82	14.0	36.16	0.06	1552.94	14
UK100_15-C	1626.53	15.0	56.39	0.02	1627.29	15.0	40.04	0.07	1626.22	15
UK100_16-C	1216.84	12.0	65.67	0.00	1218.12	12.0	47.79	0.11	1216.84	12
UK100_17-C	1556.27	15.0	168.14	0.18	1557.92	15.0	105.12	0.28	1553.50	15
UK100_18-C	1325.73	12.1	128.23	0.34	1330.81	12.5	78.81	0.73	1321.19	13
UK100_19-C	1273.75	13.0	185.51	0.06	1273.40	13.0	60.10	0.03	1272.96	13
UK100_20-C	1538.07	14.0	57.78	0.06	1540.13	14.0	37.33	0.20	1537.13	14
Avg.			101.98	0.09			54.54	0.15		
UK200_01-C	2553.76	27.5	1348.48	0.49	2551.49	27.3	1025.24	0.40	2541.36	28
UK200_02-C	2417.46	24.0	1458.51	0.38	2416.57	24.0	1108.56	0.34	2408.27	24
UK200_03-C	2492.57	27.0	627.78	0.17	2491.50	27.0	544.86	0.13	2488.24	27
UK200_04-C	2376.92	26.0	1160.76	0.29	2376.45	26.0	1022.63	0.27	2369.99	26
UK200_05-C	2588.32	26.7	990.15	0.56	2592.61	26.6	668.79	0.73	2573.91	27
UK200_06-C	2335.46	26.1	833.10	0.22	2338.19	26.1	289.31	0.34	2330.26	27
UK200_07-C	2449.21	27.0	1516.93	0.36	2452.54	27.0	1010.45	0.50	2440.32	27
UK200_08-C	2505.57	27.0	1174.60	0.29	2507.12	27.0	1741.41	0.35	2498.29	27
UK200_09-C	2307.32	25.0	1205.85	0.39	2307.66	25.0	625.49	0.40	2298.45	25
UK200_10-C	2579.25	27.0	490.84	0.18	2580.51	27.0	323.09	0.23	2574.70	27
UK200_11-C	2397.80	27.0	759.66	0.29	2398.85	27.0	1092.66	0.33	2390.89	27
UK200_12-C	2547.48	25.0	764.05	0.12	2547.51	25.0	950.21	0.12	2544.44	25
UK200_13-C	2501.05	25.0	2003.87	0.42	2502.14	25.0	1230.16	0.47	2490.52	25
UK200_14-C	2482.37	27.0	1118.29	0.38	2481.76	27.0	857.82	0.36	2472.98	27
UK200_15-C	2524.31	25.2	1847.11	0.36	2529.27	25.5	838.48	0.56	2515.23	26
UK200_16-C	2516.38	26.8	869.18	0.69	2516.23	26.9	983.05	0.69	2499.09	27
UK200_17-C	2587.92	26.0	1123.71	0.25	2585.69	26.0	1135.63	0.16	2581.55	26
UK200_18-C	2437.42	26.0	1331.59	0.40	2443.40	26.4	471.64	0.64	2427.76	27
UK200_19-C	2261.51	25.0	862.35	0.20	2267.25	25.0	395.16	0.46	2256.93	25
UK200_20-C	2537.52	26.0	960.11	0.29	2538.64	26.0	980.09	0.34	2530.16	26
Avg.			1122.35	0.34			864.74	0.39		

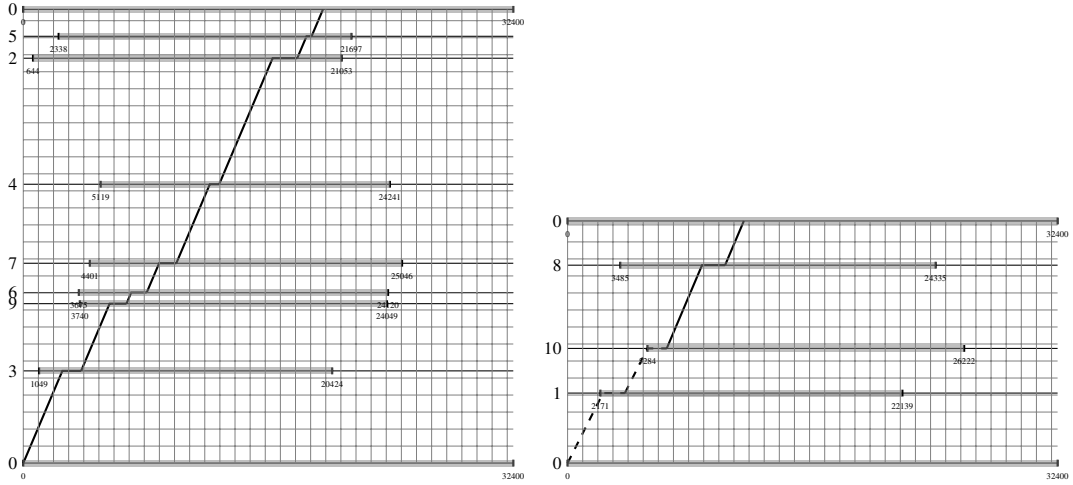


Figura 5.1: Solução da instância *UK10_01-PRPLIB*

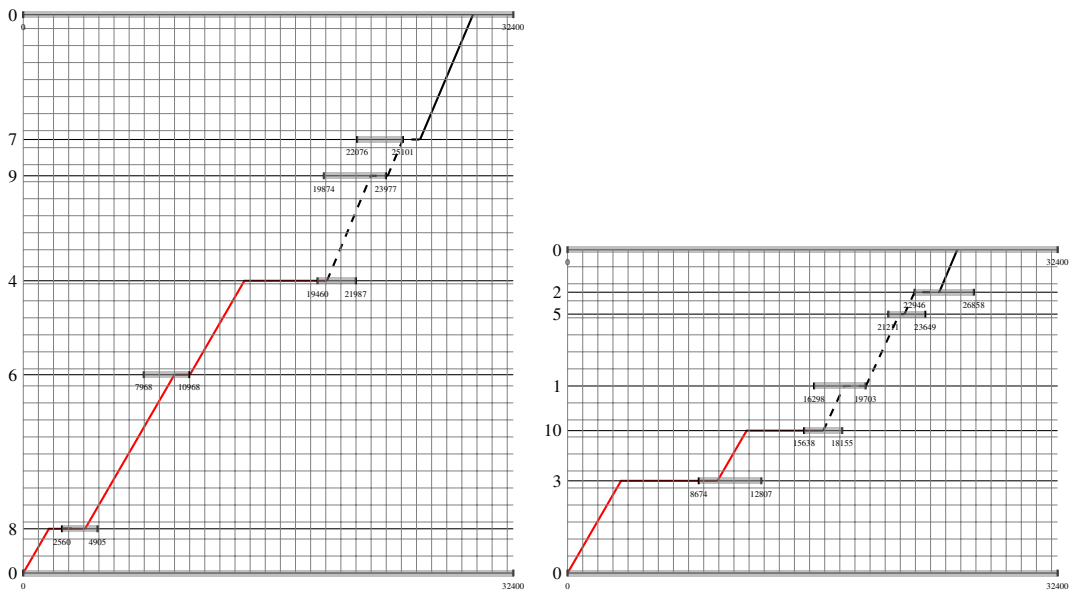


Figura 5.2: Solução da instância *UK10_01-B*

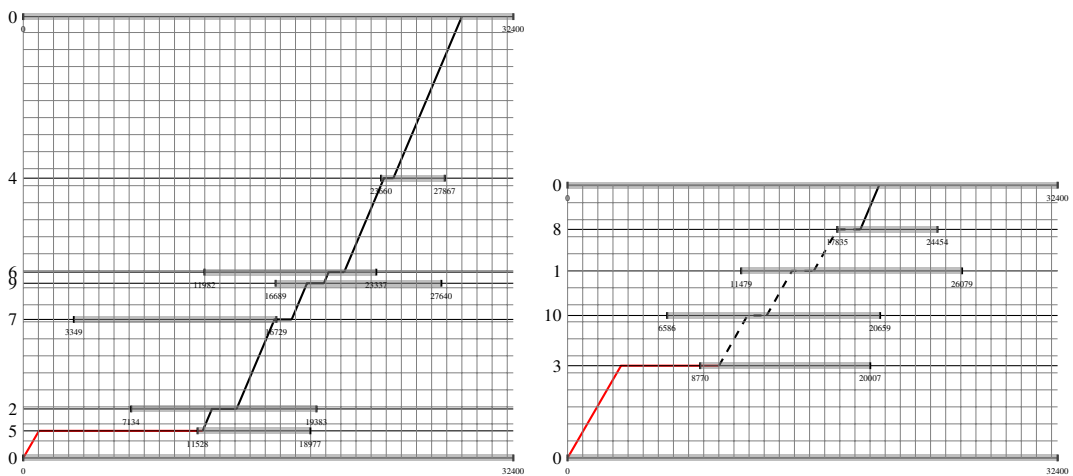


Figura 5.3: Solução da instância *UK10_01-C*

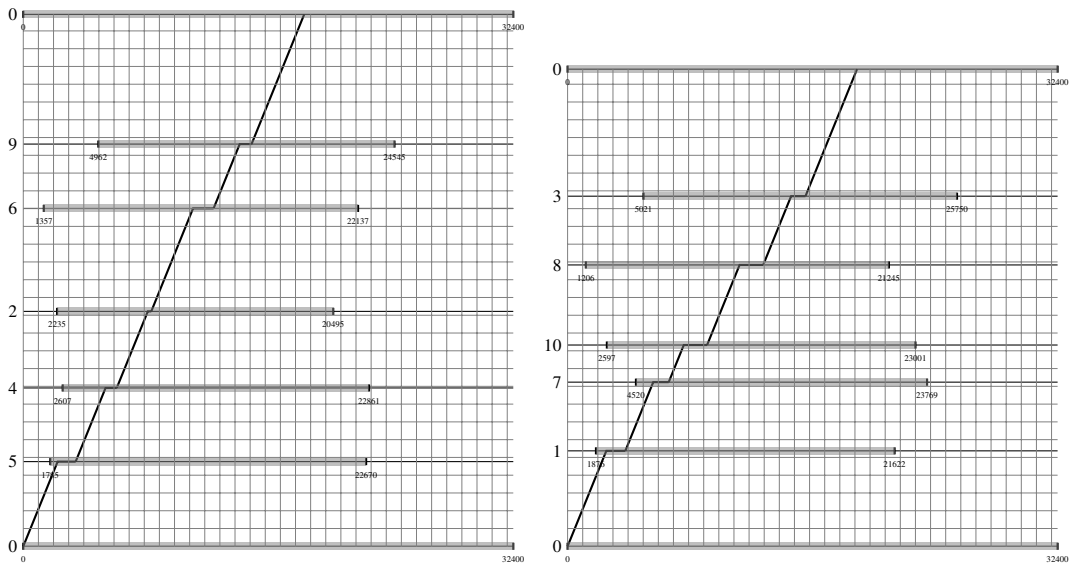


Figura 5.4: Solução da instância *UK10_08-PRPLIB*

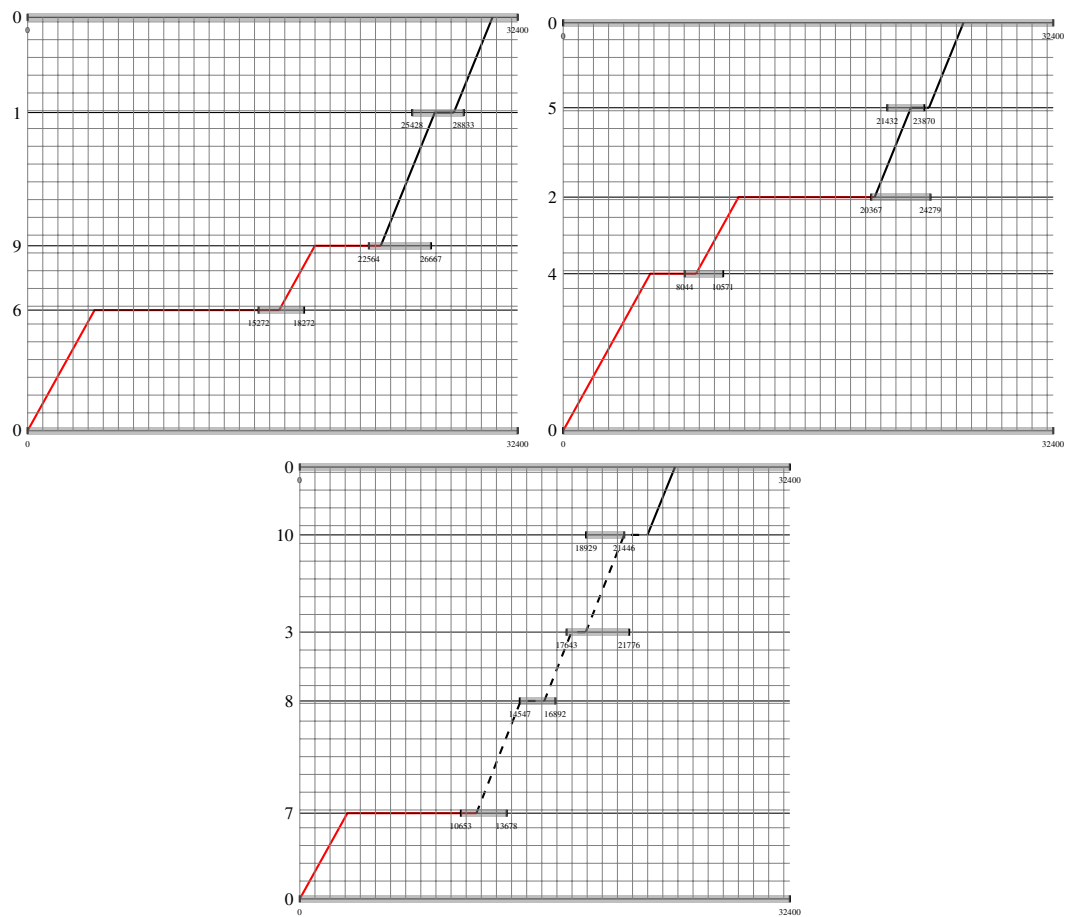


Figura 5.5: Solução da instância *UK10_08-B*

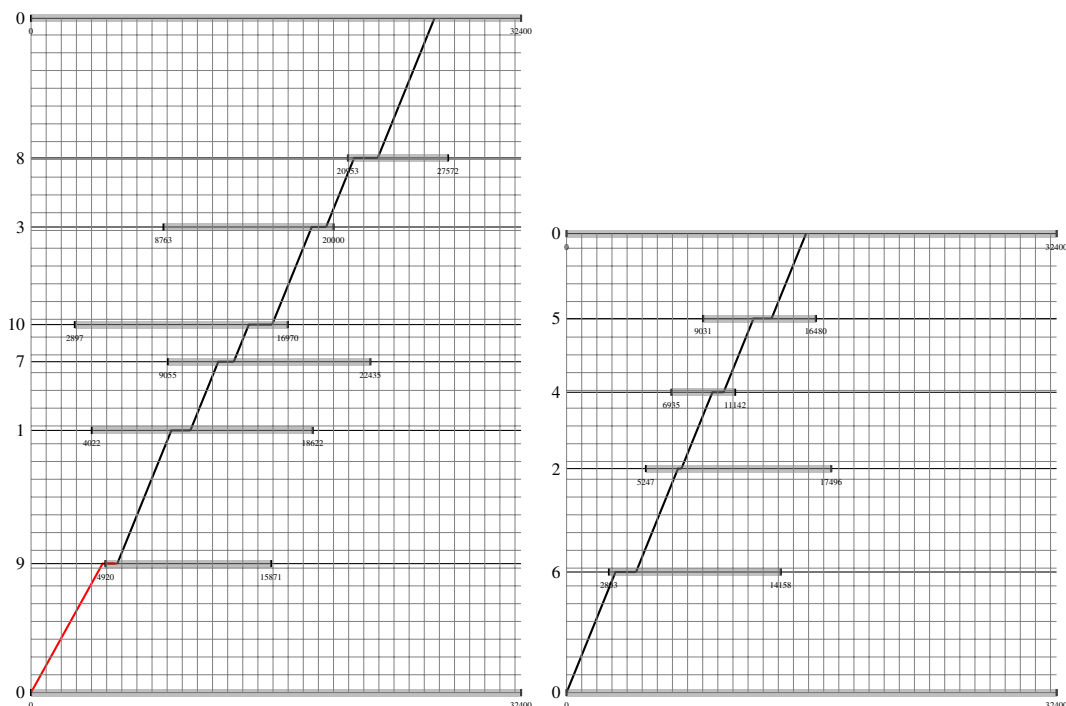


Figura 5.6: Solução da instância *UK10_08-C*

5.3 Resultados para o VRPTW

Para o VRPTW foram resolvidas as instâncias de Solomon de 100 clientes, objetivando a redução da distância total percorrida. Os resultados estão expostos nas Tabelas 5.4, 5.5 e 5.6, onde são comparados os resultados obtidos com as melhores soluções disponíveis na literatura, encontrados em Pisinger e Ropke (2005), Milthens (2009) e Demir et al (2012).

Os testes foram executados em um PC com processador Intel Core i7-2600 3.4 GHz com 16 GB de memória RAM e sistema operacional Linux Mint Release 13. Uma única *thread* foi considerada nos testes e cada instância foi resolvida 10 vezes.

Após experimentos preliminares, o parâmetro n_R foi adotado como sendo igual a 10 para todas as instâncias. Já o parâmetro n_{ILS} foi configurado em função das características da instância como sendo $n_{ILS} = n/5 + 5r$.

Para cada uma das 56 instâncias resolvidas, foi possível encontrar o melhor resultado da literatura ao menos uma vez dentre as 10 execuções, com exceção da instância *rc106*. Contudo, em um teste adicional (com os mesmos parâmetros), o melhor valor da literatura para a instância *rc106* foi alcançado após a 60ª execução.

Tabela 5.4: Resultados do ILS-SP-SOA para as instâncias tipo c de Solomon (1987)

Instance	Best Sol. ALNS			Avg. Sol. ILS-SP-SOA				Best Sol. ILS-SP-SOA				BKS
	Cost	CPU time (s)	Gap (%)	Cost	R	CPU time (s)	Gap (%)	Cost	R	CPU time (s)	Gap (%)	Cost
c101	827.3	41	0.00	827.30	10.00	10.66	0.00	827.30	10	10.75	0.00	827.3
c102	827.3	40	0.00	827.30	10.00	12.70	0.00	827.30	10	12.76	0.00	827.3
c103	826.3	41	0.00	826.30	10.00	13.09	0.00	826.30	10	13.07	0.00	826.3
c104	822.9	40	0.00	822.90	10.00	14.38	0.00	822.90	10	15.86	0.00	822.9
c105	827.3	39	0.00	827.30	10.00	14.57	0.00	827.30	10	14.22	0.00	827.3
c106	827.3	41	0.00	827.30	10.00	15.11	0.00	827.30	10	14.85	0.00	827.3
c107	827.3	41	0.00	827.30	10.00	17.66	0.00	827.30	10	16.98	0.00	827.3
c108	827.3	42	0.00	827.30	10.00	17.67	0.00	827.30	10	17.72	0.00	827.3
c109	827.3	40	0.00	827.30	10.00	14.68	0.00	827.30	10	14.85	0.00	827.3
c201	589.1	82	0.00	589.10	3.00	27.18	0.00	589.10	3	27.11	0.00	589.1
c202	589.1	85	0.00	589.10	3.00	29.30	0.00	589.10	3	29.31	0.00	589.1
c203	588.7	92	0.00	588.70	3.00	31.13	0.00	588.70	3	28.67	0.00	588.7
c204	588.1	91	0.00	588.10	3.00	41.26	0.00	588.10	3	41.70	0.00	588.1
c205	586.4	86	0.00	586.40	3.00	28.05	0.00	586.40	3	28.16	0.00	586.4
c206	586	81	0.00	586.00	3.00	26.84	0.00	586.00	3	26.31	0.00	586
c207	585.8	86	0.00	585.80	3.00	26.05	0.00	585.80	3	26.90	0.00	585.8
c208	585.8	88	0.00	585.80	3.00	25.24	0.00	585.80	3	24.99	0.00	585.8
Avg.		62.12	0.00			21.50	0.00			21.42	0.00	

Tabela 5.5: Resultados do ILS-SP-SOA para as instâncias tipo r de Solomon (1987)

Instance	Best Sol. ALNS			Avg. Sol. ILS-SP-SOA				Best Sol. ILS-SP-SOA				BKS
	Cost	CPU time (s)	Gap (%)	Cost	R	CPU time (s)	Gap (%)	Cost	R	CPU time (s)	Gap (%)	Cost
r101	1637.7	48	0.00	1637.70	20.00	20.55	0.00	1637.70	20	18.82	0.00	1637.7
r102	1466.6	46	0.00	1466.60	18.00	21.79	0.00	1466.60	18	21.37	0.00	1466.6
r103	1208.7	46	0.00	1208.70	14.00	27.72	0.00	1208.70	14	24.65	0.00	1208.7
r104	971.5	45	0.00	971.97	11.00	32.54	0.05	971.50	11	28.21	0.00	971.5
r105	1355.3	47	0.00	1355.30	15.00	26.56	0.00	1355.30	15	27.92	0.00	1355.3
r106	1234.6	46	0.00	1234.60	13.00	34.04	0.00	1234.60	13	30.19	0.00	1234.6
r107	1064.6	42	0.00	1064.60	11.00	35.53	0.00	1064.60	11	39.26	0.00	1064.6
r108	936.1	44	0.43	933.57	10.00	33.47	0.16	932.10	10	31.72	0.00	932.1
r109	1146.9	46	0.00	1146.90	13.00	30.83	0.00	1146.90	13	29.42	0.00	1146.9
r110	1073.9	49	0.55	1068.00	12.00	33.37	0.00	1068.00	12	33.38	0.00	1068
r111	1049.9	42	0.11	1048.70	12.00	33.46	0.00	1048.70	12	32.60	0.00	1048.7
r112	948.6	40	0.00	949.32	10.00	30.10	0.08	948.60	10	26.82	0.00	948.6
r201	1143.2	71	0.00	1143.20	8.00	34.04	0.00	1143.20	8	34.47	0.00	1143.2
r202	1032.2	72	0.25	1029.71	7.90	39.96	0.01	1029.60	8	39.23	0.00	1029.6
r203	873.3	76	0.29	870.80	6.00	39.48	0.00	870.80	6	42.45	0.00	870.8
r204	731.3	75	0.00	731.47	4.90	42.69	0.02	731.30	5	35.41	0.00	731.3
r205	950.4	71	0.06	952.59	5.20	41.90	0.29	949.80	5	48.88	0.00	949.8
r206	881	76	0.58	875.90	5.00	43.83	0.00	875.90	5	42.03	0.00	875.9
r207	794	85	0.00	795.42	4.00	37.70	0.18	794.00	4	40.02	0.00	794
r208	702.9	88	0.24	701.24	3.90	45.60	0.01	701.00	4	41.81	0.00	701.0
r209	854.8	74	0.00	854.80	5.00	41.26	0.00	854.80	5	39.21	0.00	854.8
r210	906.3	70	0.64	906.10	6.00	36.64	0.62	900.50	6	32.11	0.00	900.5
r211	751.6	74	0.66	752.98	4.30	39.44	0.84	746.70	4	41.89	0.00	746.7
Avg.		64.41	0.22			37.61	0.13			37.10	0.00	

Tabela 5.6: Resultados do ILS-SP-SOA para as instâncias tipo *rc* de Solomon (1987)

Instance	Best Sol. ALNS			Avg. Sol. ILS-SP-SOA				Best Sol. ILS-SP-SOA				BKS
	Cost	CPU time (s)	Gap (%)	Cost	R	CPU time (s)	Gap (%)	Cost	R	CPU time (s)	Gap (%)	Cost
rc101	1619.80	44	0.00	1625.10	15.60	22.73	0.33	1619.80	15	20.98	0.00	1619.8
rc102	1463.50	42	0.42	1459.02	14.00	24.96	0.11	1457.40	14	24.07	0.00	1457.4
rc103	1267.10	43	0.72	1259.71	11.10	33.76	0.14	1258.00	11	36.24	0.00	1258
rc104	1133.10	42	0.07	1132.33	10.00	32.71	0.00	1132.30	10	33.14	0.00	1132.3
rc105	1513.70	43	0.00	1513.75	15.10	27.33	0.00	1513.70	15	29.43	0.00	1513.7
rc106	1372.70	41	0.00	1373.50	13.00	32.44	0.06	1372.70*	13	31.89	0.00	1372.7
rc107	1209.30	40	0.12	1208.70	12.00	31.34	0.07	1207.80	12	30.63	0.00	1207.8
rc108	1114.20	43	0.00	1114.20	11.00	25.38	0.00	1114.20	11	24.70	0.00	1114.2
rc201	1262.70	74	0.07	1262.41	8.90	29.90	0.05	1261.80	9	29.43	0.00	1261.8
rc202	1095.80	71	0.32	1093.62	7.60	31.01	0.12	1092.30	8	30.14	0.00	1092.3
rc203	923.70	73	0.00	931.19	5.00	45.58	0.81	923.70	5	45.47	0.00	923.7
rc204	783.80	76	0.04	783.50	4.00	37.26	0.00	783.50	4	35.05	0.00	783.5
rc205	1154.00	64	0.00	1154.74	7.20	29.62	0.06	1154.00	7	29.72	0.00	1154
rc206	1051.10	68	0.00	1051.21	6.90	38.25	0.01	1051.10	7	41.92	0.00	1051.1
rc207	966.60	64	0.38	963.04	6.00	35.65	0.01	962.90	6	39.59	0.00	962.9
rc208	777.30	72	0.00	779.74	4.80	33.76	0.31	776.10	4	32.86	0.00	776.1
Avg.		56.73	0.14			32.31	0.13			32.49	0.00	

*Resultado obtido após a 60^a execução do algoritmo

5.4 Resultados para o FCVRP

Os testes computacionais para o FCVRP foram executados em um PC com processador Intel Core i7 3.40 GHz com 16 GB de memória RAM e sistema operacional Linux Mint Release 13. Os valores dos parâmetros adotados foram: $n_R = 10$, $n_{ILS} = n/5 + 5r$, $n_{SP} = 150$, $n_{POOL} = 2$, $T_{MIP} = 60$ segundos. O resolvidor CPLEX 12.4 foi utilizado para resolver os Problemas de Particionamento de Conjuntos (SPP). Uma única *thread* foi considerada nos testes e cada instância foi resolvida 10 vezes.

Os resultados foram comparados com os obtidos por Xiao et al (2012), como pode ser observado na Tabela 5.7. Todos os valores apresentados na coluna *BKS* foram obtidos pelo ILS-SP-SOA (com diferentes parametrizações) com exceção do resultado da instância *C11* que foi obtido por Xiao et al (2012). Os valores sublinhados e destacados em negrito indicam que os valores obtidos pelo ILS-SP-SOA melhoraram os melhores resultados disponíveis na literatura, e os destacados apenas em negrito indicam que os resultados foram iguais aos melhores da literatura. Além disto, pode-se observar que sete instâncias, *C6-C10* e *C13-C14*, ainda não tinham sido resolvidas no contexto do FCVRP. Tais instâncias são as iguais às outras sete (*C1-C5* e *C11-C12*), porém com requisitos adicionais de tempo para atendimento dos clientes e duração máxima de cada rota. Comparando os custos entre os pares de instâncias, *C1-C6*, *C2-C7*, *C3-C8*, *C4-C9*, *C5-C10*, *C11-C13* e *C12-C14*, verifica-se que estas restrições adicionais possuem

grande influência no custo total das soluções. Percebe-se, ainda, que os valores médios das soluções apresentam uma diferença média de 0,16% e 0,31%, em relação às melhores soluções conhecidas, para as instâncias de Christofides et al (1979) e Golden et al (1998), respectivamente, indicando que o ILS-SP-SOA é bastante robusto ao lidar com o FCVRP.

Tabela 5.7: Resultados para o FCVRP

Inst.	n	Avg. Sol. SMSAH			Avg. Sol. ILS-SP-SOA					BKS	
		Cost	Dist.	Gap (%)	Cost	Dist.	R	CPU time (s)	Gap (%)	Cost	R
C1	50	751,43	534,14	0,04	752,93	537,84	5,00	0,49	0,24	751,11	5
C2	75	1188,62	862,60	1,36	1178,47	856,56	10,00	2,72	0,50	1172,62	10
C3	100	1153,56	849,45	0,50	1148,51	850,03	8,00	4,46	0,06	1147,83	8
C4	150	1461,69	1066,11	1,04	1448,16	1061,83	12,00	13,04	0,10	1446,64	12
C5	199	1865,30	1348,88	1,55	1842,81	1352,03	17,00	38,23	0,32	1836,86	17
C6	50	-	-	-	1255,39	565,41	6,00	0,69	0,00	1255,39	6
C7	75	-	-	-	1987,28	925,11	11,00	3,21	0,02	1986,91	11
C8	100	-	-	-	2168,77	883,72	9,00	6,78	0,04	2168,01	9
C9	150	-	-	-	3051,07	1182,87	14,00	44,74	0,11	3047,77	14
C10	199	-	-	-	3907,15	1433,91	18,00	131,86	0,11	3902,99	18
C11	120	1516,42	1054,06	0,19	1523,40	1054,22	7,00	9,00	0,66	1513,48 ¹	-
C12	100	1175,59	827,98	0,13	1174,02	827,05	10,00	2,74	0,00	1174,02	10
C13	120	-	-	-	7979,57	1559,94	11,00	130,46	0,08	7973,52	11
C14	100	-	-	-	10213,40	868,68	11,00	5,41	0,00	10213,40	11
Avg				2,20				28,13	0,16		
G1	240	7714,29	5706,19	0,70	7666,47	5861,77	10,00	143,79	0,08	7660,64	10
G2	320	11195,02	8490,24	0,39	11154,66	8499,27	10,00	468,22	0,03	11151,20	10
G3	400	14566,73	11097,24	0,58	14504,64	11098,67	10,00	906,14	0,15	14482,60	10
G4	480	18605,37	13848,72	2,18	18218,20	14021,93	11,70	1078,72	0,05	18209,30	12
G5	200	8576,91	6471,57	0,18	8561,53	6460,98	5,00	55,90	0,00	8561,53	5
G6	280	11121,04	8431,39	0,42	11074,44	8583,11	8,00	197,31	0,00	11074,40	8
G7	360	13477,07	10249,70	0,63	13393,53	10242,70	9,00	557,62	0,00	13392,90	9
G8	440	16098,60	11888,02	3,92	15566,23	11878,90	10,90	1265,40	0,48	15491,30	11
G9	255	858,34	612,79	2,47	842,33	609,73	14,00	78,98	0,56	837,63	14
G10	323	1090,85	777,10	2,37	1071,27	775,77	16,00	181,59	0,53	1065,62	16
G11	399	1360,20	971,48	2,85	1328,99	963,44	18,00	362,31	0,49	1322,54	18
G12	483	1661,07	1182,80	3,73	1609,99	1161,38	19,00	741,51	0,54	1601,35	19
G13	252	1269,37	904,67	2,70	1241,51	897,19	26,90	43,91	0,44	1236,03	27
G14	320	1604,83	1141,03	2,52	1570,75	1129,12	30,00	95,48	0,34	1565,44	30
G15	396	1987,76	1417,55	2,65	1940,30	1397,30	33,80	166,61	0,20	1936,45	34
G16	480	2408,72	1714,48	2,89	2349,75	1694,47	37,10	410,99	0,37	2341,08	37
G17	240	1033,88	725,33	1,56	1021,26	718,00	22,00	40,74	0,32	1018,02	22
G18	300	1469,97	1032,93	1,67	1452,19	1031,25	27,90	113,69	0,44	1445,88	28
G19	360	2014,26	1420,35	2,09	1982,49	1406,14	33,00	186,37	0,48	1972,97	33
G20	420	2699,29	1903,12	2,46	2651,22	1890,50	39,00	304,80	0,64	2634,42	39
Avg				1,95				370,00	0,31		

¹ Valor obtido por Xiao et al (2012)

5.5 Resultados para o EMVRP

Os testes para o EMVRP foram executados no mesmo ambiente computacional utilizado para o FCVRP e os parâmetros do algoritmo foram os mesmos utilizados para o PRP, com exceção do parâmetro ω_{TW} que não é necessário, dado que tanto o EMVRP como o FCVRP não consideram restrições de janelas de tempo. O resolvidor CPLEX 12.4 foi utilizado para resolver os Problemas de Particionamento de Conjuntos (SPP). Uma única *thread* foi considerada nos

testes e cada instância foi resolvida 10 vezes.

De acordo com as pesquisas realizadas, as instâncias de Christofides et al (1979) e Golden et al (1998) ainda não tinham sido resolvidas no contexto do EMVRP. Deste modo, os resultados apresentados na Tabela 5.8 são analisados apenas em termos da diferença entre as médias das soluções e o melhor valor encontrado dentre as 10 execuções de cada instância. Ao contrário do ocorrido nos resultados do FCVRP, verifica-se que as instâncias com restrições adicionais não possuem grande impacto nos custos das soluções, quando comparadas com as mesmas desconsiderando tempos para atendimento dos clientes e duração máxima de cada rota. Porém, pode-se observar que os tempos de processamento ($CPUT(s)$) para as instâncias C6-C10 e C13-C14 foram mais elevados. No geral, o *gap* médio das médias das soluções em relação ao melhores resultados, assim como para o FCVRP, foram relativamente baixos: 0,09% e 0,08%, para as instâncias de Christofides et al (1979) e Golden et al (1998), respectivamente, indicando que o ILS-SP-SOA também é bastante robusto ao lidar com o EMVRP.

Tabela 5.8: Resultados para o EMVRP

Inst.	n	Avg. Sol. ILS-SP-SOA (10 runs)					Best Sol. ILS-SP-SOA (10 runs)				
		Cost	Dist.	R	CPU T(s)	Gap (%)	Cost	Dist.	R	CPU T(s)	
C1	50	46383.08	564.04	5.00	0.91	0.37	46210.40	564.75	5	0.97	
C2	75	60575.77	885.88	10.00	3.71	0.02	60565.00	883.89	10	3.68	
C3	100	83490.40	883.17	8.00	8.03	0.00	83490.40	883.17	8	8.76	
C4	10	105869.60	1121.89	12.00	22.88	0.03	105842.00	1123.03	12	23.49	
C5	199	135143.10	1441.77	17.00	48.14	0.01	135124.00	1454.19	17	42.99	
C6	50	42403.20	604.90	6.00	1.33	0.00	42403.20	604.90	6	1.54	
C7	75	61164.50	965.81	11.00	5.67	0.00	61164.50	965.81	11	5.89	
C8	100	80247.20	944.25	9.00	14.00	0.00	80247.20	944.25	9	14.32	
C9	150	105618.80	1235.21	14.00	95.29	0.32	105282.00	1235.05	14	71.35	
C10	199	136390.20	1513.39	18.00	119.93	0.42	135819.00	1524.32	18	132.36	
C11	120	123430.80	1098.82	7.00	27.81	0.07	123348.00	1100.81	7	26.45	
C12	100	93521.30	859.54	10.00	5.80	0.00	93521.30	859.54	10	5.06	
C13	120	128356.30	1603.45	11.00	76.65	0.01	128346.00	1602.06	11	111.48	
C14	100	93827.20	922.69	11.00	10.68	0.00	93827.20	922.69	11	10.67	
Avg.					31.49	0.09					
G1	240	1451440.00	6000.85	10.00	281.01	0.00	1451440.00	6000.85	10	282.16	
G2	320	2736277.00	8599.93	10.00	1288.77	0.01	2736090.00	8601.13	10	1228.78	
G3	400	4527650.00	11189.40	10.00	2007.98	0.00	4527650.00	11189.40	10	2101.69	
G4	480	6150984.00	14459.29	12.00	2395.43	0.08	6146280.00	14452.60	12	2430.96	
G5	200	2742480.00	6536.42	5.00	100.62	0.00	2742480.00	6536.42	5	102.71	
G6	280	3352706.00	8793.30	8.00	380.48	0.01	3352280.00	8794.36	8	339.86	
G7	360	4205440.00	10336.80	9.00	1305.71	0.00	4205440.00	10336.80	9	1363.80	
G8	440	4830982.00	11971.71	11.00	2397.67	0.04	4828830.00	11970.00	11	2417.57	
G9	255	302320.90	707.76	14.00	162.71	0.06	302128.00	705.51	14	146.63	
G10	323	384217.20	908.24	16.00	325.81	0.04	384069.00	917.59	16	347.21	
G11	399	476933.10	1133.93	18.00	646.64	0.04	476744.00	1128.05	18	644.06	
G12	483	579327.30	1327.30	19.00	1401.90	0.48	576583.00	1326.01	19	930.49	
G13	252	448193.00	1007.59	27.00	89.00	0.07	447899.00	1004.09	27	81.55	
G14	320	567382.90	1273.73	30.00	186.63	0.10	566803.00	1277.20	30	135.04	
G15	396	700357.70	1596.06	34.00	394.35	0.11	699609.00	1596.33	34	460.52	
G16	480	846679.50	1952.54	38.00	675.41	0.08	845985.00	1947.01	38	584.14	
G17	240	80393.12	772.07	22.00	82.65	0.09	80319.30	768.26	22	77.46	
G18	300	111393.30	1132.17	28.00	182.85	0.07	111318.00	1131.81	28	175.26	
G19	360	151155.10	1559.64	33.00	319.42	0.21	150845.00	1557.55	33	312.78	
G20	420	199052.20	2160.16	40.60	502.77	0.04	198977.00	2176.22	41	459.86	
Avg.					756.39	0.08					

6 CONCLUSÃO

Esta dissertação teve seu foco na resolução do *Pollution-Routing Problem* (PRP), um problema de roteamento de veículos (PRV) que, diferentemente das versões tradicionais, considera aspectos ambientais em sua formulação. Além do PRP, outras variantes (casos particulares do PRP) foram consideradas, como o PRV com Janelas de Tempo (VRPTW), o PRV com Minimização de Energia (EMVRP) e o PRV considerando Consumo de Combustível (FCVRP).

Foi proposto um algoritmo híbrido (*matheurística*), denominado ILS-SP-SOA, que utiliza métodos aproximativos e exatos em seu escopo. A *matheurística* ILS-SP-SOA combina as meta-heurísticas ILS e RVND com um algoritmo recursivo para otimizar as velocidades, juntamente com um procedimento exato para resolver problemas de particionamento de conjuntos.

O algoritmo proposto foi testado em diferentes instâncias da literatura, comparando-os com os melhores resultados conhecidos e com os obtidos pelo otimizador CPLEX. Novas instâncias foram propostas para o PRP, de modo a dificultar as restrições de janela de tempo. Para as instâncias do PRP, uma versão alternativa do ILS-SP-SOA foi testada (similar à abordagem adotada por Xiao et al (2012)), de modo a verificar a importância da utilização de uma matriz dinâmica de velocidades (versão original do ILS-SP-SOA). Observou-se que o algoritmo proposto teve um bom desempenho quando comparado com os resultados da literatura, tanto em termos de robustez quanto em qualidade das soluções.

No entanto, sabe-se que o controle das velocidades dos veículos é uma tarefa difícil de ser posta em prática, sendo afetada por diversos fatores como, por exemplo, condições meteorológicas, normas de regulamentação do trânsito e congestionamento das ruas. Como trabalhos futuros, o algoritmo ILS-SP-SOA pode ser adaptado para considerar o efeito do congestionamento.

mento das ruas nas emissões de CO₂, bem como o efeito do uso de uma frota de veículos com características distintas, que emitem diferentes quantidades de poluentes ao ambiente. Dado que a função de custo do PRP é formada por duas parcelas (custo com combustível e custo com motoristas) de objetivos conflitantes, o algoritmo ILS-SP-SOA ainda pode ser adaptado para lidar com a versão bi-objetivo do PRP.

Além da adaptação do algoritmo proposto para outras variantes do PRP, trabalhos futuros podem ser elaborados na melhoria do algoritmo ILS-SP-SOA através do desenvolvimento de novos mecanismos para calcular as velocidades ótimas de uma rota em tempo constante durante a busca local.

Apêndices

Apêndice A

Exemplo da utilização das estruturas de dados auxiliares apresentadas na Seção 4.3

Sejam duas rotas distintas, $r1$ e $r2$, conforme apresentado na Figura A.1. Supondo um movimento da vizinhança $Shift(1,0)$ da rota $r1$ para a rota $r2$ e considerando os dados apresentados nas Tabelas A.1 e A.2, um exemplo de cálculo das estruturas auxiliares apresentadas na Seção 4.3 do Capítulo 4 é apresentado a seguir.

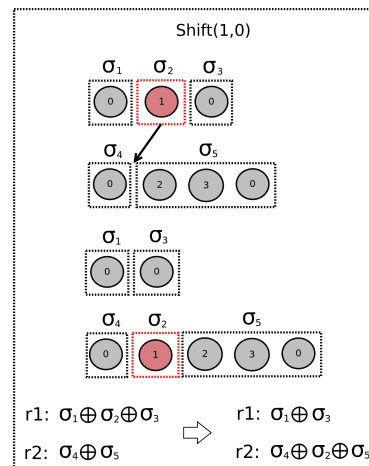


Figura A.1: Exemplo $Shift(1,0)$

Tabela A.1: Características dos clientes

Cliente	Janela de tempo		Tempo de serviço	Demanda
i	a_i	b_i	τ_i	q_i
0	0	50	0	0
1	8	18	2	7
2	3	25	2	11
3	28	38	3	5

Tabela A.2: Matriz de distâncias entre clientes

	0	1	2	3
0	0	6	9	10
1	10	0	7	9
2	7	9	0	6
3	6	8	7	0

Conforme apresentado nas equações 4.1-4.21 o custo da rota $r2$ após a inserção do cliente 1 entre o depósito inicial 0 e o cliente 2 pode ser calculado através da concatenação das subsequências σ_4 , σ_2 e σ_5 .

Considerando uma matriz de velocidades com todos os elementos iguais à 1, levando a uma matriz de tempos de viagens igual à matriz de distâncias (Tabela A.2), temos:

Para as subsequências de um único nó (cliente ou depósito), tais informações podem ser facilmente calculadas. Utilizando as Equações 4.1-4.9, os valores obtidos para os dados do exemplo proposto estão apresentados na Tabela A.3.

Tabela A.3: Valores das estruturas auxiliares para subsequências de um nó (exemplo)

Estrutura	Depósito Inicial	Cliente	Cliente	Cliente	Depósito Final
	0	1	2	3	0'
DU	0	2	2	3	0
TW	0	0	0	0	0
E	0	8	3	28	0
L	0	18	25	38	50
Q	0	7	11	5	0
DI	0	0	0	0	0
TT	0	0	0	0	0
QD	0	0	0	0	0
SSD	0	0	0	0	0

Considerando apenas as subsequências envolvidas no movimento $Shif(1,0)$ da Figura A.1, torna-se necessário calcular as informações relativas à subsequência σ_5 que consiste na concatenação dos clientes 2, 3 e do depósito final 0'. Tal subsequência pode ser obtida da concatenação dos clientes 2 e 3, gerando uma subsequência σ' , e posterior concatenação com o depósito final 0', de modo que $\sigma_5 = \sigma' \oplus 0'$.

Concatenando o cliente 2 com o cliente 3 ($2 \oplus 3 = \sigma'$) temos:

$$\Delta = 2 - 0 + 6 = 8 \quad (\text{A.1})$$

$$\Delta WT = \max\{28 - 8 - 25, 0\} = 0 \quad (\text{A.2})$$

$$\Delta TW = \max\{3 + 8 - 38, 0\} = 0 \quad (\text{A.3})$$

$$DU(2 \oplus 3) = 2 + 3 + 6 + 0 = 11 \quad (\text{A.4})$$

$$TW(2 \oplus 3) = 0 + 0 + 0 = 0 \quad (\text{A.5})$$

$$E(2 \oplus 3) = \max\{28 - 8, 3\} - 0 = 20 \quad (\text{A.6})$$

$$L(2 \oplus 3) = \min\{38 - 8, 25\} + 0 = 25 \quad (\text{A.7})$$

$$Q(2 \oplus 3) = 11 + 5 = 16 \quad (\text{A.8})$$

$$DI(2 \oplus 3) = 0 + 0 + 6 = 6 \quad (\text{A.9})$$

$$TT(2 \oplus 3) = 0 + 0 + 6 = 6 \quad (\text{A.10})$$

$$QD(2 \oplus 3) = 0 + 0 + 5(0 + 6) = 30 \quad (\text{A.11})$$

$$SSD(2 \oplus 3) = 0 + 0 + (1^2 \times 6) = 6 \quad (\text{A.12})$$

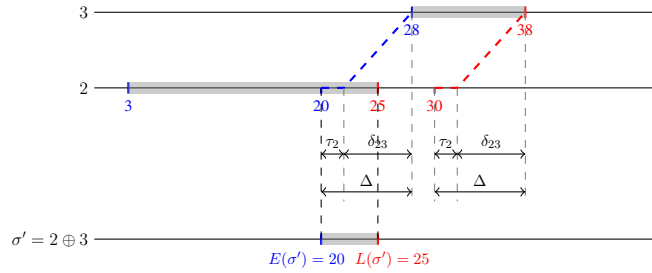


Figura A.2: Concatenação dos clientes 2 e 3 ($\sigma' = 2 \oplus 3$)

Concatenando os clientes 2 e 3 (σ') com o depósito final $0'$ ($2 \oplus 3 \oplus 0' = \sigma' \oplus 0' = \sigma_5$),

temos:

$$\Delta = 11 - 0 + 6 = 17 \quad (\text{A.13})$$

$$\Delta WT = \max\{0 - 17 - 25, 0\} = 0 \quad (\text{A.14})$$

$$\Delta TW = \max\{20 + 17 - 50, 0\} = 0 \quad (\text{A.15})$$

$$DU(\sigma' \oplus 0') = 11 + 0 + 6 + 0 = 17 \quad (\text{A.16})$$

$$TW(\sigma' \oplus 0') = 0 + 0 + 0 = 0 \quad (\text{A.17})$$

$$E(\sigma' \oplus 0') = \max\{0 - 17, 20\} - 0 = 20 \quad (\text{A.18})$$

$$L(\sigma' \oplus 0') = \min\{50 - 17, 25\} + 0 = 25 \quad (\text{A.19})$$

$$Q(\sigma' \oplus 0') = 16 + 0 = 16 \quad (\text{A.20})$$

$$DI(\sigma' \oplus 0') = 6 + 0 + 6 = 12 \quad (\text{A.21})$$

$$TT(\sigma' \oplus 0') = 6 + 0 + 6 = 12 \quad (\text{A.22})$$

$$QD(\sigma' \oplus 0') = 30 + 0 + 0(6 + 6) = 30 \quad (\text{A.23})$$

$$SSD(\sigma' \oplus 0') = 6 + 0 + (1^2 \times 6) = 12 \quad (\text{A.24})$$

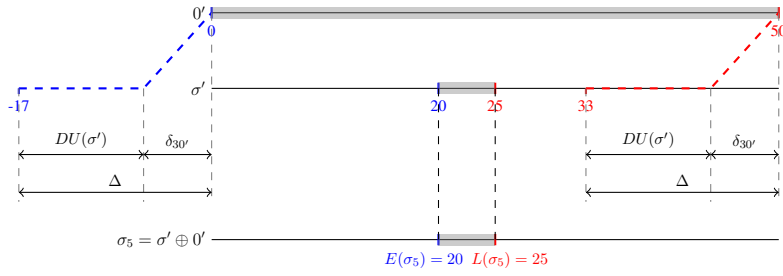


Figura A.3: Concatenação da subsequência σ' com o depósito final $0'$ ($\sigma_5 = \sigma' \oplus 0'$)

Concatenando os clientes 1, 2 e 3 com o depósito final $0'$ ($1 \oplus 2 \oplus 3 \oplus 0' = \sigma_2 \oplus \sigma_5 = \sigma''$),

temos:

$$\Delta = 2 - 0 + 7 = 9 \quad (\text{A.25})$$

$$\Delta WT = \max\{20 - 9 - 18, 0\} = 0 \quad (\text{A.26})$$

$$\Delta TW = \max\{8 + 9 - 25, 0\} = 0 \quad (\text{A.27})$$

$$DU(\sigma_2 \oplus \sigma_5) = 2 + 17 + 7 + 0 = 26 \quad (\text{A.28})$$

$$TW(\sigma_2 \oplus \sigma_5) = 0 + 0 + 0 = 0 \quad (\text{A.29})$$

$$E(\sigma_2 \oplus \sigma_5) = \max\{20 - 9, 8\} - 0 = 11 \quad (\text{A.30})$$

$$L(\sigma_2 \oplus \sigma_5) = \min\{25 - 9, 18\} + 0 = 16 \quad (\text{A.31})$$

$$Q(\sigma_2 \oplus \sigma_5) = 7 + 16 = 23 \quad (\text{A.32})$$

$$DI(\sigma_2 \oplus \sigma_5) = 0 + 12 + 7 = 19 \quad (\text{A.33})$$

$$TT(\sigma_2 \oplus \sigma_5) = 0 + 12 + 7 = 19 \quad (\text{A.34})$$

$$QD(\sigma_2 \oplus \sigma_5) = 0 + 30 + 16(0 + 7) = 142 \quad (\text{A.35})$$

$$SSD(\sigma_2 \oplus \sigma_5) = 0 + 12 + (1^2 \times 7) = 19 \quad (\text{A.36})$$

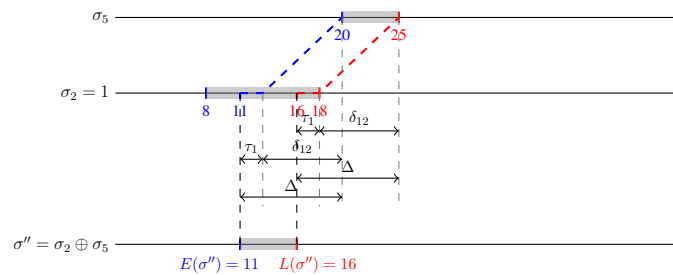


Figura A.4: Concatenação das subsequências σ_2 e σ_5 ($\sigma'' = \sigma_2 \oplus \sigma_5$)

Por fim, concatenando o depósito inicial 0 com os clientes 1, 2 e 3 e com o depósito final $0'$ ($0 \oplus 1 \oplus 2 \oplus 3 \oplus 0' = \sigma_4 \oplus \sigma_2 \oplus \sigma_5 = \sigma_4 \oplus \sigma'' = r_2$), temos:

$$\Delta = 0 - 0 + 6 = 6 \quad (\text{A.37})$$

$$\Delta WT = \max\{11 - 6 - 0, 0\} = 5 \quad (\text{A.38})$$

$$\Delta TW = \max\{0 + 6 - 16, 0\} = 0 \quad (\text{A.39})$$

$$DU(\sigma_4 \oplus \sigma'') = 0 + 26 + 6 + 5 = 37 \quad (\text{A.40})$$

$$TW(\sigma_4 \oplus \sigma'') = 0 + 0 + 0 = 0 \quad (\text{A.41})$$

$$E(\sigma_4 \oplus \sigma'') = \max\{11 - 6, 0\} - 5 = 0 \quad (\text{A.42})$$

$$L(\sigma_4 \oplus \sigma'') = \min\{16 - 6, 0\} + 0 = 0 \quad (\text{A.43})$$

$$Q(\sigma_4 \oplus \sigma'') = 0 + 23 = 23 \quad (\text{A.44})$$

$$DI(\sigma_4 \oplus \sigma'') = 0 + 19 + 6 = 25 \quad (\text{A.45})$$

$$TT(\sigma_4 \oplus \sigma'') = 0 + 19 + 6 = 25 \quad (\text{A.46})$$

$$QD(\sigma_4 \oplus \sigma'') = 0 + 142 + 23(0 + 6) = 280 \quad (\text{A.47})$$

$$SSD(\sigma_4 \oplus \sigma'') = 0 + 19 + (1^2 \times 6) = 25 \quad (\text{A.48})$$

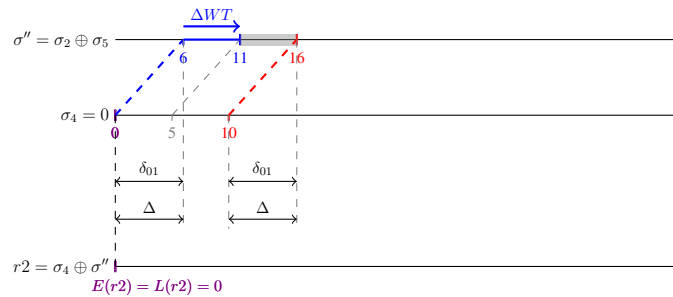


Figura A.5: Concatenação das subsequências σ_4 e σ'' ($r_2 = \sigma_4 \oplus \sigma''$)

De posse das informações apresentadas em (A.37)-(A.48), utilizando os mesmos valores apresentados na Seção 5.1 para w_1 , w_2 , w_3 , w_4 , ω_{FC} e ω_{FD} , o novo custo da rota r_2 após o movimento $Shift(1,0)$ pode ser calculado conforme apresentado na Equação A.49.

$$\begin{aligned} Z(r_2) &= 1,4(1,01763908 \times 10^{-3} \times 25) \\ &+ 1,4(5,33605218 \times 10^{-5} \times 25) \\ &+ 1,4(8,40323178 \times 10^{-9} \times 280) \\ &+ 1,4(1,41223439 \times 10^{-7} \times 25) \\ &+ 2,2222222 \times 10^{-3}(37) \\ &+ \omega_{\text{TW}}(0) = 119,715445 \times 10^{-3} \end{aligned} \tag{A.49}$$

Apêndice B

Concatenação com *time-warp*

Considerando os dados constantes nas Tabelas B.1 e B.2, pode-se observar na Figura B.1 que ao sair do cliente 1 para o cliente 2, iniciando o atendimento de 1 no instante mais cedo permitido (i.e., $a_1 = 12$), não é possível chegar no cliente 2 dentro da sua respectiva janela de tempo, sendo necessário um uso mínimo de *time-warp* de 6 unidades de tempo, como pode ser verificado nas Equações (B.3)-(B.5).

Tabela B.1: Características dos clientes

Cliente	Janela de tempo		Tempo de serviço	Demanda
i	a_i	b_i	τ_i	q_i
1	12	22	2	4
2	10	15	2	7
3	27	32	3	5

Tabela B.2: Matriz de distâncias entre clientes

	1	2	3
1	0	7	9
2	9	0	5
3	8	7	0

Considerando uma matriz de velocidades com todos os elementos iguais à 1, levando a uma matriz de tempos de viagens igual à matriz de distâncias, ao concatenar o cliente 1 com o cliente 2 ($\sigma' = 1 \oplus 2$), temos:

$$\Delta = 2 - 0 + 7 = 9 \quad (\text{B.1})$$

$$\Delta WT = \max\{10 - 9 - 22, 0\} = 0 \quad (\text{B.2})$$

$$\Delta TW = \max\{12 + 9 - 15, 0\} = 6 \quad (\text{B.3})$$

$$DU(1 \oplus 2) = 2 + 2 + 7 + 0 = 11 \quad (\text{B.4})$$

$$TW(1 \oplus 2) = 0 + 0 + 6 = 6 \quad (\text{B.5})$$

$$E(1 \oplus 2) = \max\{10 - 9, 12\} - 0 = 12 \quad (\text{B.6})$$

$$L(1 \oplus 2) = \min\{15 - 9, 22\} + 6 = 12 \quad (\text{B.7})$$

$$Q(1 \oplus 2) = 4 + 7 = 11 \quad (\text{B.8})$$

$$DI(1 \oplus 2) = 0 + 0 + 7 = 7 \quad (\text{B.9})$$

$$TT(1 \oplus 2) = 0 + 0 + 7 = 7 \quad (\text{B.10})$$

$$QD(1 \oplus 2) = 0 + 0 + 7(0 + 7) = 49 \quad (\text{B.11})$$

$$SSD(1 \oplus 2) = 0 + 0 + (1^2 \times 7) = 7 \quad (\text{B.12})$$

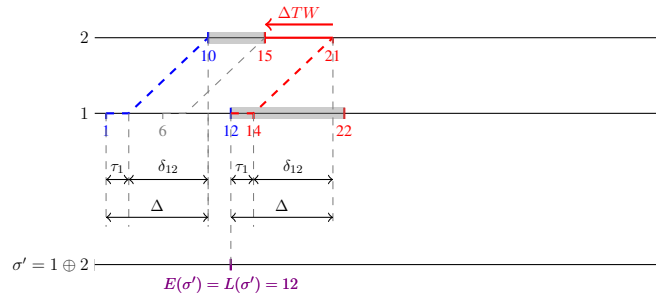


Figura B.1: Concatenação dos clientes 1 e 2 ($\sigma' = 1 \oplus 2$)

Posteriormente, ao concatenar a subsequência σ' com o cliente 3, verifica-se um tempo de espera para o início do atendimento do cliente 3 de 5 unidades de tempo, resultando em um tempo total de duração de 24 unidades de tempo.

$$\Delta = 11 - 6 + 5 = 10 \quad (\text{B.13})$$

$$\Delta WT = \max\{27 - 10 - 12, 0\} = 5 \quad (\text{B.14})$$

$$\Delta TW = \max\{12 + 10 - 32, 0\} = 0 \quad (\text{B.15})$$

$$DU(1 \oplus 2) = 11 + 3 + 5 + 5 = 24 \quad (\text{B.16})$$

$$TW(1 \oplus 2) = 6 + 0 + 0 = 6 \quad (\text{B.17})$$

$$E(1 \oplus 2) = \max\{27 - 10, 12\} - 5 = 12 \quad (\text{B.18})$$

$$L(1 \oplus 2) = \min\{32 - 10, 12\} + 6 = 12 \quad (\text{B.19})$$

$$Q(1 \oplus 2) = 11 + 5 = 16 \quad (\text{B.20})$$

$$DI(1 \oplus 2) = 7 + 0 + 5 = 12 \quad (\text{B.21})$$

$$TT(1 \oplus 2) = 7 + 0 + 5 = 12 \quad (\text{B.22})$$

$$QD(1 \oplus 2) = 49 + 0 + 5(7 + 5) = 109 \quad (\text{B.23})$$

$$SSD(1 \oplus 2) = 7 + 0 + (1^2 \times 5) = 12 \quad (\text{B.24})$$

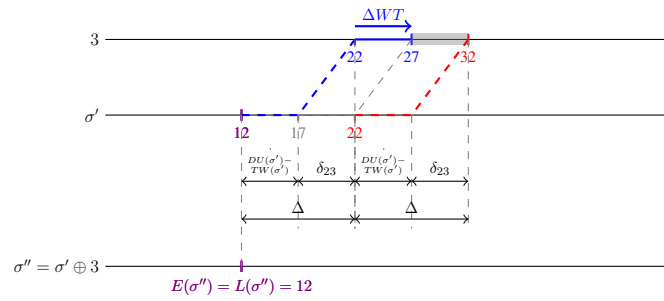


Figura B.2: Concatenação da subsequência σ' e 3 ($\sigma'' = \sigma' \oplus 3$)

REFERÊNCIAS BIBLIOGRÁFICAS

- BAKER, E. K. *The Vehicle Routing Problem*, Capítulo Evolution of Microcomputer-Based Vehicle Routing Software: Case Studies in the United States, p. 353–361. Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, 2002.
- BARTH, M.; YOUNGLOVE, T.; SCORA, G. Development of a heavy-duty diesel modal emissions and fuel consumption model. Relatório técnico, California Partners for Advanced Transit and Highways (PATH), UC Berkeley, 2005.
- BARTH, M.; BORIBOONSOMSIN, K. Real-world carbon dioxide impacts of traffic congestion. *Transportation Research Record*, v. 1, n. 2058, p. 163–171, 2008.
- BEKTAŞ, T.; LAPORTE, G. The pollution-routing problem. *Transportation Research Part B: Methodological*, v. 45, n. 8, p. 1232–1250, 2011.
- BJÖRKLUND, M. Influence from the business environment on environmental purchasing — drivers and hinders of purchasing green transportation services. *Journal of Purchasing and Supply Management*, v. 17, n. 1, p. 11–22, 2011.
- BLOEMHOF-RUWAARD, J. M.; VAN BEEK, P.; HORDIJK, L.; WASSENHOVE, L. N. V. Interactions between operational research and environmental management. *European Journal of Operational Research*, v. 85, n. 2, p. 229 – 243, 1995.
- BLUM, C.; ROLI, A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, v. 35, n. 3, p. 268–308, 2003.
- BODEN, T. A.; MARLAND, G.; ANDRES, R. Global, regional, and national fossil-fuel CO₂ emissions. Carbon Dioxide Information Analysis Center, Oak Ridge National Laboratory, U.S. Department of Energy, Oak Ridge, Tenn., U.S.A, 2012.
- CHRISTOFIDES, N.; MINGOZZI, A.; TOTH, P. *Combinatorial Optimization*, Capítulo The Vehicle Routing Problem, p. 315–338. Wiley, Chinchester, UK, 1979.
- CLARKE, G.; WRIGHT, J. W. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, v. 12, p. 568–581, 1964.

- CORDEAU, J.-F.; LAPORTE, G.; SAVELSBERGH, M. W. P.; VIGO, D. *Transportation, Handbooks in Operations Research and Management Science*, volume 14, Capítulo Vehicle Routing, p. 367–428. Elsevier, Amsterdam, 2007.
- CROES, G. A. A method for solving traveling salesman problems. *Operations Research*, v. 6, p. 791–812, 1958.
- DANIEL, S.; DIAKOULAKI, D.; PAPPIS, C. Operations research and environmental planning. *European Journal of Operational Research*, v. 102, n. 2, p. 248 – 263, 1997.
- DANTZIG, G. B.; RAMSER, J. H. The truck dispatching problem. *Management Science*, v. 6, n. 1, p. 80–91, 1959.
- DEKKER, R.; BLOEMHOF, J.; MALLIDIS, I. Operations research for green logistics — an overview of aspects, issues, contributions and challenges. *European Journal of Operational Research*, v. 219, n. 3, p. 671–679, 2012.
- DEMIR, E. *Models and Algorithms for the Pollution-Routing Problem and Its Variations*. PhD thesis, University of Southampton, School of Management, United Kingdom, 2012.
- DEMIR, E.; BEKTAŞ, T.; LAPORTE, G. The bi-objective pollution-routing problem. *European Journal of Operational Research*, v. 232, n. 3, p. 464 – 478, 2014a.
- DEMIR, E.; BEKTAŞ, T.; LAPORTE, G. A review of recent research on green road freight transportation. *European Journal of Operational Research*. ISSN 0377-2217. In Press, Corrected Proof, 2014b.
- DEMIR, E.; BEKTAS, T.; LAPORTE, G. An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, v. 223, n. 2, p. 346–359, 2012.
- EGLESE, R.; MADEN, W.; SLATER, A. A road timetable to aid vehicle routing and scheduling. *Computers & Operations Research*, v. 33, n. 12, p. 3508 – 3519, 2006.
- ERDOĞAN, S.; MILLER-HOOKS, E. A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, v. 48, n. 1, p. 100 – 114, 2012.

- FAGERHOLT, K.; LAPORTE, G.; NORSTAD, I. Reducing fuel emissions by optimizing speed on shipping routes. *Journal of the Operational Research Society*, v. 61, n. 3, p. 523–529, 2010.
- FIGLIOZZI, M. Vehicle Routing Problem for Emissions Minimization. *Transportation Research Record*, , n. 2197, p. 1–7, 2010.
- FLOOD, M. M. The traveling-salesman problem. *Annals of Operations Research*, v. 4, n. 1, p. 61–75, 1956.
- FRANCESCHETTI, A.; HONHON, D.; VAN WOENSEL, T.; BEKTAŞ, T.; LAPORTE, G. The time-dependent pollution-routing problem. *Transportation Research Part B: Methodological*, v. 56, p. 265 – 293, 2013.
- GENDREAU, M.; POTVIN, J. Y.; BRÄYSY, O.; HASLE, G.; LØKKETANGEN, A. Metaheuristics for the vehicle routing problem and its extensions: a categorized bibliography. GOLDEN, B.; RAGHAVAN, S.; WASIL, E. (Eds.), *The Vehicle Routing Problem - Latest Advances and New Challenges*. Springer Verlag, Heidelberg, 2008.
- GENDREAU, M.; POTVIN, J.-Y. *Handbook of Metaheuristics*. Springer Publishing Company, Incorporated, 2nd edição, 2010.
- GOLDEN, B.; RAGHAVAN, S.; WASIL, E. *The Vehicle Routing Problem: Latest Advances and New Challenges*. Springer, New York, 2008.
- GOLDEN, B. L.; ASSAD, A. A.; WASIL, E. A. *The Vehicle Routing Problem*, Capítulo Routing vehicles in the real world: applications in the solid waste, beverage, food, dairy and newspaper industries, p. 245–286. Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, 2002.
- GOLDEN, B. L.; WASIL, E. A.; KELLY, J. P.; CHAO, I.-M. *Fleet Management and Logistics*, Capítulo Metaheuristics in Vehicle Routing, p. 33–56. Kluwer, Boston, 1998.
- HASHIMOTO, H.; YAGIURA, M. A path relinking approach with an adaptive mechanism to control parameters for the vehicle routing problem with time windows. VANHEMERT, J.;

- COTTA, C. (Ed.), *8th European Conference on Evolutionary Computation in Combinatorial Optimization*, volume 4972 of *Lecture Notes in Computer Science*, p. 254–265. Naples, Italy, 2008.
- HASHIMOTO, H.; YAGIURA, M.; IBARAKI, T. An iterated local search algorithm for the time-dependent vehicle routing problem with time windows. *Discrete Optimization*, v. 5, n. 2, p. 434–456, 2008.
- HUNG LAI, K.; WONG, C. W. Green logistics management and performance: Some empirical evidence from chinese manufacturing exporters. *Omega*, v. 40, n. 3, p. 267 – 282, 2012.
- HVATTUM, L. M.; NORSTAD, I.; FAGERHOLT, K.; LAPORTE, G. Analysis of an exact algorithm for the vessel speed optimization problem. *Networks*, v. 62, n. 2, p. 132–135, 2013.
- IBARAKI, T.; IMAHORI, S.; KUBO, M.; MASUDA, T.; UNO, T.; YAGIURA, M. Effective local search algorithms for routing and scheduling problems with general time window constraints. *Transportation Science*, v. 39, p. 206–232, 2005.
- IEA. CO₂ emissions from fuel combustion — highlights. IEA Statistics. International Energy Agency — 2012 Edition, 2012.
- IRNICH, S. A unified modeling and solution framework for vehicle routing and local search-based metaheuristics. *INFORMS Journal on Computing*, v. 20, n. 2, p. 270–287, 2008.
- JABALI, O.; VAN WOENSEL, T.; DE KOK, A. Analysis of travel times and CO₂ emissions in time-dependent vehicle routing. *Production and Operations Management*, v. 21, n. 6, p. 1060–1074, 2012.
- KARA, I.; KARA, B. Y.; YETIS, M. K. Energy minimizing vehicle routing problem. DRESS, A.; XU, Y.; ZHU, B. (Eds.), *Combinatorial Optimization and Applications, Proceedings*, volume 4616 of *Lecture Notes in Computer Science*, p. 62–71, 2007.
- KARA, I.; KARA, B. Y.; YETIS, M. K. *Vehicle Routing Problem*, Capítulo Cumulative Vehicle Routing Problems. InTech, 2008.

- KINDERVATER, G. A. P.; SAVELSBERGH, M. W. P. Vehicle routing: handling edge exchanges. AARTS, E. H. L.; LENSTRA, J. K. (Eds.), *Local search in combinatorial optimization*, p. 337–360. Princeton University Press, 1997.
- KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by simulated annealing. *Science*, v. 220, n. 4598, p. 671–680, 1983.
- KOPFER, H.; SCHÖNBERGER, J.; KOPFER, H. Reducing greenhouse gas emissions of a heterogeneous vehicle fleet. *Flexible Services and Manufacturing Journal*, p. 1–28, 2013.
- KUO, Y. Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. *Computers & Industrial Engineering*, v. 59, n. 1, p. 157–165, 2010.
- KUO, Y.; WANG, C.-C. Optimizing the VRP by minimizing fuel consumption. *Management of Environmental Quality: An International Journal*, v. 22, n. 4, p. 440–450, 2011.
- LAPORTE, G.; SEMET, F. *The Vehicle Routing Problem*, Capítulo Classical heuristics for the Capacitated VRP, p. 109–128. Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, 2002.
- LIN, C.; CHOY, K.; HO, G.; CHUNG, S.; LAM, H. Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications*, v. 41, n. 4, Part 1, p. 1118 – 1138, 2014.
- LOURENÇO, H. R.; MARTIN, O. C.; STÜTZLE, T. *Handbook of Metaheuristics*, Capítulo Iterated Local Search, p. 321–353. Kluwer Academic Publishers, Norwell, MA, 2002.
- MANIEZZO, V.; STÜTZLE, T.; VOSS, S. *The Vehicle Routing Problem*, volume 10 of *Annals of Information Systems*. Springer, New York, 2009.
- MASON, S. Backward progress: Turning the negative perception of reverse logistics into happy returns. IEE Solutions, 2002.
- MILTHERS, N. P. M. Solving VRP using voronoi diagrams and adaptive large neighborhood search. Master's thesis, University of Copenhagen, Denmark, 2009.

- MLADENović, N.; HANSEN, P. Variable neighborhood search. *Computers & Operations Research*, v. 24, n. 11, p. 1097–1100, 1997.
- NAGATA, Y.; BRAYSY, O.; DULLAERT, W. A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows. *Computers & Operations Research*, v. 37, n. 4, p. 724–737, 2010.
- NORSTAD, I.; FAGERHOLT, K.; LAPORTE, G. Tramp ship routing and scheduling with speed optimization. *Transportation Research Part C - Emerging Technologies*, v. 19, n. 5, SI, p. 853–865, 2011.
- OR, I. *Traveling salesman-type combinatorial problems and their relation to the logistics of blood banking*. PhD thesis, Northwestern University, USA, 1976.
- OSMAN, I. H.; KELLY, J. P. (Eds.). *Meta-Heuristics: Theory and Applications*. Kluwer Academic Publishers, Norwell, MA, USA, 1996.
- OSMAN, I. H. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, v. 41, n. 1-4, p. 421–451. ISSN 0254-5330, 1993.
- OSMAN, I. H.; LAPORTE, G. Metaheuristics: A bibliography. *Annals of Operations Research*, v. 63, p. 513–623, 1996.
- PALMER, A. *The Development of an Integrated Routing and Carbon Dioxide Emissions Model for Goods Vehicles*. PhD thesis, Cranfield University. School of Management, Reino Unido, 2007.
- PENG, Y.; WANG, X. Research on a vehicle routing schedule to reduce fuel consumption. *Proceedings of the 2009 International Conference on Measuring Technology and Mechatronics Automation - Volume 03*, ICMTMA '09, p. 825–827, Washington, DC, USA. IEEE Computer Society, 2009.
- PENNA, P. H. V.; SUBRAMANIAN, A.; OCHI, L. S. An iterated local search heuristic for the heterogeneous fleet vehicle routing problem. *Journal of Heuristics*, v. 19, p. 201–232, 2013.

- PISINGER, D.; ROPKE, S. A general heuristic for vehicle routing problems. Relatório técnico, DIKU - Department of Computer Science, University of Copenhagen, 2005.
- PWC. World in 2050. The BRICs and beyond: prospects, challenges and opportunities. PricewaterhouseCoopers International Limited (PwCIL), 2013.
- QI, X.; SONG, D.-P. Minimizing fuel emissions by optimizing vessel schedules in liner shipping with uncertain port times. *Transportation Research Part E: Logistics and Transportation Review*, v. 48, n. 4, p. 863–880, 2012.
- QIAN, J.; EGGLESE, R. Finding least fuel emission paths in a network with time-varying speeds. *Networks*, v. 63, n. 1, p. 96–106, 2014.
- REPOUSSIS, P. P.; TARANTILIS, C. D.; IOANNOU, G. Arc-Guided Evolutionary Algorithm for the Vehicle Routing Problem with Time Windows. *IEEE Transactions on Evolutionary Computation*, v. 13, n. 3, p. 624–647, 2009.
- RODRIGUE, J.-P.; SLACK, B.; COMTOIS, C. *Handbook of Logistics and Supply-Chain Management. (Handbooks in Transport, 2)*, Capítulo Green Logistics (The Paradoxes of), p. 339–351. Pergamon/Elsevier, London, 2001.
- SABERI, M.; VERBAS, I. O. Continuous Approximation Model for the Vehicle Routing Problem for Emissions Minimization at the Strategic Level. *Journal of Transportation Engineering-ASCE*, v. 138, n. 11, p. 1368–1376, 2012.
- SALIMIFARD, K.; SHAHBANDARZADEH, H.; RAEESI, R. Green transportation and the role of operation research. *Proceedings of 2012 International Conference on Traffic and Transportation Engineering*, volume 26, Singapore. IACSIT Press, 2012.
- SBIHI, A.; EGGLESE, R. W. Combinatorial optimization and Green Logistics. *Annals of operations research*, v. 175, n. 1, p. 159–175, 2010.
- SCORA, M.; BARTH, M. Comprehensive modal emission model (CMEM), version 3.01, user's guide. Relatório técnico, Center for Environmental Research and Technology, University of California, Riverside, 2006.

- SCOTT, C.; URQUHART, N.; HART, E. Influence of topology and payload on CO₂ optimised vehicle routing. CHIO, C.; BRABAZON, A.; CARO, G.; EBNER, M.; FAROOQ, M.; FINK, A.; GRAHL, J.; GREENFIELD, G.; MACHADO, P.; O'NEILL, M.; TARANTINO, E.; URQUHART, N. (Eds.), *Applications of Evolutionary Computation*, volume 6025 of *Lecture Notes in Computer Science*, p. 141–150. Springer Berlin Heidelberg, 2010.
- SOLOMON, M. Algorithms for the Vehicle-Routing and Scheduling Problems with Time Window Constraints. *Operations Research*, v. 35, n. 2, p. 254–265, 1987.
- SOUZA, M. J. F. Inteligência computacional para otimização. Notas de Aula. Universidade Federal de Ouro Preto, 2007.
- SUBRAMANIAN, A. *Heuristic, Exact and Hybrid Approaches for Vehicle Routing Problems*. PhD thesis, Universidade Federal Fluminense, Computing Graduate Program, Niterói, 2012.
- SUBRAMANIAN, A.; PENNA, P. H. V.; UCHOA, E.; OCHI, L. S. A hybrid algorithm for the heterogeneous fleet vehicle routing problem. *European Journal of Operational Research*, v. 221, n. 2, p. 285 – 295, 2012.
- SUBRAMANIAN, A.; UCHOA, E.; OCHI, L. S. A hybrid algorithm for a class of vehicle routing problems. *Computers & Operations Research*, v. 40, n. 10, p. 2519–2531, 2013.
- TAILLARD, É.; BADEAU, P.; GENDREAU, M.; GUERTIN, F.; POTVIN, J.-Y. A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*, v. 31, p. 170–186, 1997.
- TOTH, P.; VIGO, D. *The Vehicle Routing Problem*. Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, 2002.
- UBEDA, S.; ARCELUS, F.; FAULIN, J. Green logistics at eroski: A case study. *International Journal of Production Economics*, v. 131, n. 1, p. 44 – 51, 2011.
- URQUHART, N.; SCOTT, C.; HART, E. Using an evolutionary algorithm to discover low CO₂ tours within a travelling salesman problem. CHIO, C.; BRABAZON, A.; CARO, G.; EBNER, M.; FAROOQ, M.; FINK, A.; GRAHL, J.; GREENFIELD, G.; MACHADO,

- P.; O'NEILL, M.; TARANTINO, E.; URQUHART, N. (Eds.), *Applications of Evolutionary Computation*, volume 6025 of *Lecture Notes in Computer Science*, p. 421–430. Springer Berlin Heidelberg, 2010.
- VIDAL, T.; CRAINIC, T. G.; GENDREAU, M.; PRINS, C. An empirical analysis of time-window relaxations in vehicle routing heuristics. *Anais do XVI CLAIO e XLIV SBPO*, Rio de Janeiro. SOBRAPO, 2012.
- VIDAL, T.; CRAINIC, T. G.; GENDREAU, M.; PRINS, C. A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers & Operations Research*, v. 40, n. 1, p. 475 – 489, 2013.
- XIAO, Y.; ZHAO, Q.; KAKU, I.; XU, Y. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers & Operations Research*, v. 39, n. 7, p. 1419 – 1431, 2012.
- ÖZCEYLAN, E.; KIRAN, M. S.; ATASAGUN, Y. A new hybrid heuristic approach for solving green traveling salesman problem. 41st International Conference on Computers & Industrial Engineering, 2011.