

LUCAS DIETRICH SILVA BARBOSA

**HEURÍSTICAS PARA O ROTEAMENTO DE
FLUXOS NA COLETA E ENTREGA DE
RECURSOS ENTRE CANTEIROS DE OBRAS**

Belo Horizonte

17 de fevereiro de 2017

UNIVERSIDADE FEDERAL DE MINAS GERAIS
ESCOLA DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

**HEURÍSTICAS PARA O ROTEAMENTO DE
FLUXOS NA COLETA E ENTREGA DE
RECURSOS ENTRE CANTEIROS DE OBRAS**

Dissertação apresentada ao Curso de Pós-Graduação em Engenharia de Produção da Universidade Federal de Minas Gerais como requisito para a obtenção do grau de Mestre em Engenharia de Produção.

LUCAS DIETRICH SILVA BARBOSA

Belo Horizonte
17 de fevereiro de 2017



UNIVERSIDADE FEDERAL DE MINAS GERAIS

FOLHA DE APROVAÇÃO

Heurísticas para o Roteamento de Fluxos na Coleta e Entrega
de Recursos entre Canteiros de Obras

LUCAS DIETRICH SILVA BARBOSA

Dissertação defendida e aprovada pela banca examinadora constituída por:

Prof(a). MAURÍCIO CARDOSO DE SOUZA (DOUTOR) – Orientador
Universidade Federal de Minas Gerais

Prof(a). CARLOS ROBERTO VENÂNCIO DE CARVALHO (DOUTOR)
Universidade Federal de Minas Gerais

Prof(a). THIAGO AUGUSTO DE OLIVEIRA SILVA (DOUTOR)
Universidade Federal de Ouro Preto

Belo Horizonte, 17 de fevereiro de 2017

Resumo

Este trabalho lida com o problema MRCO (Movimentação de Recursos entre Canteiros de Obras) que é uma generalização do Problema de Roteamento de Veículos, cujo a motivação é tratar o fluxo de transporte em uma empresa do ramo da construção civil de Belo Horizonte. Este problema considera a existência de um conjunto de recursos que serão compartilhados por um conjunto de clientes (canteiros de obras) e que pertencem a um agrupamento em comum. A distribuição destes recursos é feita através de um conjunto de veículos capacitados. Existe um conjunto de depósitos (residência dos motoristas) e que abrigam exatamente um único veículo. O veículo deverá sair e retornar ao final de uma rota necessariamente ao seu respectivo depósito. Um recurso coletado em um canteiro de um agrupamento pode ser usado para atender qualquer outro canteiro no mesmo agrupamento. Cada canteiro pode ser visitado mais de uma vez. Os recursos podem ser discretizados em itens e assim tanto coleta e entrega podem ser divisíveis. O objetivo deste problema consiste em construir k rotas de veículos que atendam às requisições de alocação de recursos entre os canteiros de obras e que minimizem o somatório dos custos relativos ao transporte dos veículos. Nós propomos com o objetivo de tratar o problema em estudo duas heurísticas construtivas que atenderam a todas restrições presentes no problema. Tais heurísticas desenvolvidas são testados em instâncias próximas das reais e em instâncias de testes. Os resultados são comparados com a prática da empresa e com um método disponível na literatura do MRCO. Os resultados encontrados são superiores aos da empresa e são melhores, principalmente relativo ao tempo de execução computacional, comparado com o método da literatura.

Palavra-chaves: Movimentação de Recursos entre Canteiros de Obras, Problema de Roteamento de Veículos, Heurísticas Construtivas

Abstract

This paper deals with the problem MRCO (Movimentação de Recursos entre Canteiros de Obras) which is a generalization of the Vehicle Routing Problem, whose motivation is to treat the transport stream in a company in the business of construction of Belo Horizonte. This problem considers the existence of a set of resources that will be shared by a number of customers (construction sites) and belonging to a group in common. The distribution of these resources is via a set of trained vehicles. A number of depots (residence of drivers) and are home to just a single vehicle. The vehicle must leave and return at the end of a route necessarily to their respective deposit. A resource collected in a construction of a cluster can be used to meet any other site in the same grouping. Each site can be played more than once. Resources can be discretized in items and so both collection and delivery can be divisible. The objective of this problem is to build k vehicle routes that meet the resource allocation requests from the construction sites and minimize the sum of the costs of transport of vehicle. We propose with the objective of treating the problem two constructive heuristics that ensure all restrictions present in the problem. Such developed heuristics are tested in instances close to the real ones and in test instances. The results are compared with company practice and with the method available in the MRCO literature. The results found are superior to those of the company and are better, mainly related to the computational execution time, compared to the method of literature.

Keywords: Resource Movement between Construction Sites, Vehicle Routing Problem, Constructive Heuristics

*Em especial ao meu pai, minha mãe e meu irmão;
Dedico também a toda minha família e amigos;*

Agradecimentos

Primeiramente agradeço a Deus por tudo aquilo que Ele me proporciona. Ao meu pai Moacyr, minha mãe Marcsa e meu irmão Pedro pela atenção, carinho e amor. Aos meus amigos pela força e apoio em todos os momentos. Ao meu orientador prof. Dr. Maurício Cardoso de Souza e ao prof. Dr. Thiago Augusto de Oliveira Silva pela paciência, incentivo, confiança e apoio técnico em meu trabalho. Enfim, agradeço a todos familiares e amigos que de alguma forma contribuíram e torceram por esta vitória.

Sumário

1	Introdução	1
1.1	Contextualização do Problema	1
1.2	Objetivos	4
1.2.1	Objetivo Geral	4
1.2.2	Objetivos Específicos	4
1.3	Organização do Trabalho	4
2	Revisão da Literatura	5
2.1	Problema Geral de Coleta e Entrega	5
2.2	Variações do Problema Geral de Coleta e Entrega	7
2.3	Problemas com Restrições em Comum ao MRCO	10
2.4	Métodos de Resolução Propostos por Dutra	14
2.4.1	Método das Duas Fases	14
2.4.2	Modelo Integrado	18
2.4.3	Testes e Resultados	24
2.5	Considerações Finais do MRCO em Relação a Revisão da Literatura	24
3	Heurísticas para o MRCO	25
3.1	Parâmetros Comuns as Heurísticas	26
3.2	Heurística Baseada em Particionamento - HBP	26
3.3	Heurística Rota do Vizinho mais Próximo - RVP	32
4	Resultados Computacionais	37
4.1	Instâncias de Testes	37
4.2	Resultados	38
4.2.1	Instâncias Reais: Comparação com métodos de Dutra e da Empresa em estudo	38
4.2.2	Instâncias de Pequeno Porte - $ C = 10$	40
4.2.3	Instâncias de Porte Real - $ C = 29$	44

5	Considerações Finais	48
	Referências Bibliográficas	50

Lista de Figuras

1.1	Representação esquemática do problema.	2
3.1	Cálculo da variável max_r	26
3.2	Instância I1 do problema.	28
3.3	Sequências geradas a partir de \bar{C}	29
3.4	Cálculo do número de atribuições	29
3.5	Pequena instância PI1.	30
3.6	Pares origem-destino.	31
3.7	Instância I2 do problema.	32
3.8	Grafo gerado a partir dos pares canteiro-recurso.	32
3.9	Matriz de distâncias.	33
4.1	Agrupamento de um conjunto de obras. Fonte: Dutra(2015), p.28.	39

Lista de Tabelas

2.1	Restrições existentes no MRCO e seus devidos métodos de resolução: M.D: Múltiplos Depósitos; M.P: Múltiplos Produtos; M.V: Múltiplas Visitas; C.E.S: Coleta e Entrega Simultânea; D.C.E: Divisão de Coleta e Entrega. .	9
4.1	Resultados para 3 instâncias reais	40
4.2	Resultados para instâncias do tipo c com 10 canteiros de obras	41
4.3	Resultados para instâncias do tipo r com 10 canteiros de obras	41
4.4	Resultados para instâncias do tipo rc com 10 canteiros de obras	42
4.5	Porcentagens melhores instâncias - 10 canteiros	42
4.6	Heurística RVP modificada - 10 canteiros	43
4.7	Resultados para instâncias do tipo c com 29 canteiros de obras	44
4.8	Resultados para instâncias do tipo r com 29 canteiros de obras	45
4.9	Resultados para instâncias do tipo rc com 29 canteiros de obras	46
4.10	Porcentagens melhores instâncias - 29 canteiros	46
4.11	Heurística RVP modificada - 29 canteiros	47

Capítulo 1

Introdução

1.1 Contextualização do Problema

O Problema de Roteamento de Veículos (*Vehicle Routing Problem - VRP*) inicialmente estudado por [Dantzig e Ramser \(1959\)](#), consiste no atendimento de requisições de um conjunto de clientes através de uma frota de veículos. Cada veículo possui uma capacidade definida que deve ser respeitada. O ponto de origem dos veículos é denominado depósito, podendo existir um ou mais.

O objetivo deste problema varia de acordo com as características abordadas em cada estudo. Porém, existem alguns objetivos que são comuns, tais como: minimizar a distância total percorrida, minimizar o custo total do serviço, minimizar o tempo total de transporte, dentre outros.

Por ser uma generalização do Problema do Caixeiro Viajante (*Travelling Salesman Problem - TSP*) que pertence à classe de problemas NP-Difícil, o Problema de Roteamento de Veículos também é considerado como pertencente a esta classe de problemas, ver por exemplo [Garey e Johnson \(1979\)](#). Para estes problemas não são conhecidos algoritmos em tempo polinomial capazes de encontrarem soluções ótimas.

O problema objeto deste estudo denominado por Problema de Movimentação de Recursos em Canteiros de Obras (MRCO) é parte do grupo do problemas de roteamento de veículos. O MRCO apresenta diversas particularidades, tais como: múltiplos depósitos, múltiplos produtos, múltiplas visitas, coleta e entrega simultânea e divisões de coleta e entrega.

A situação prática motivadora deste estudo é tratar o fluxo de transporte em uma empresa do ramo da construção civil localizada na cidade de Belo Horizonte. Através de um planejamento diário, a empresa em questão identifica as eventuais demandas e ofertas de recursos de cada canteiro. Assim, poderá haver um conjunto de canteiros com demanda ou oferta ou demanda e oferta ao mesmo tempo de recursos.

A empresa em questão realiza um agrupamento de seus canteiros. Assim recursos que estão em canteiros de um agrupamento não podem ser transitados para canteiros não pertencentes a este agrupamento. Por exemplo, um agrupamento pode ser composto por canteiros e depósitos que estão localizados na cidade de Betim e outro agrupamento por canteiros e depósitos localizados na cidade de Contagem.

Um recurso coletado em um canteiro de um agrupamento pode ser usado para atender qualquer outro canteiro no mesmo agrupamento. Cada canteiro pode ser visitado mais de uma vez. Os recursos podem ser discretizados em itens e assim tanto coleta e entrega podem ser divisíveis. Por exemplo, se um canteiro demandar três sacos de cimento, estes podem se originar de canteiros diferentes.

A Figura 1.1 ilustra o problema descrito acima.

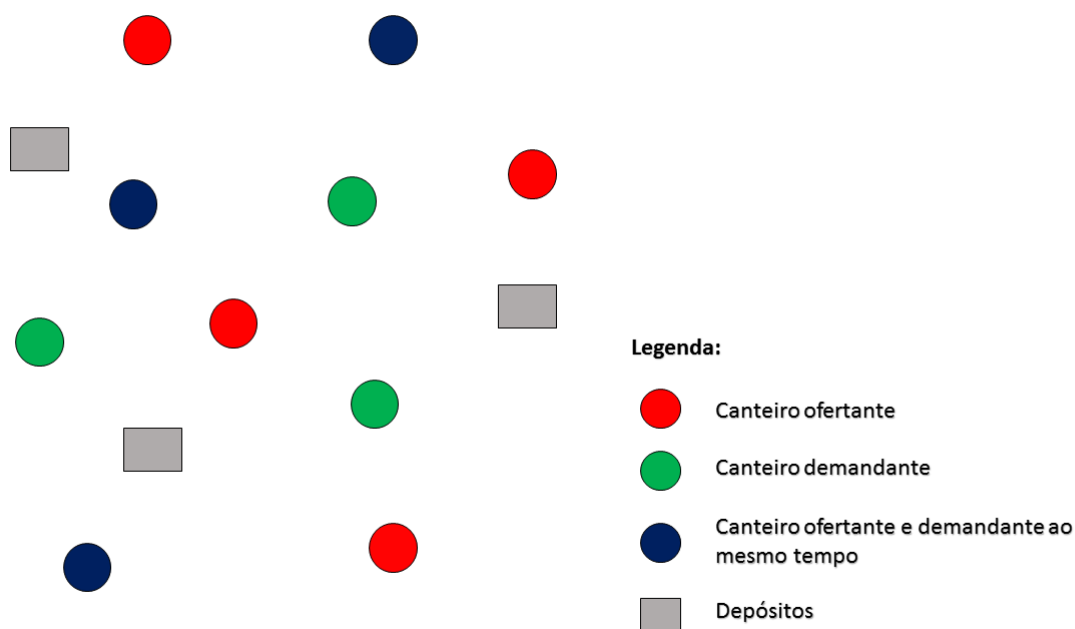


Figura 1.1: Representação esquemática do problema.

Quanto aos recursos eles podem ser divididos em:

- i- *Professional*: Os quais podem transitar entre as obras, até em um mesmo dia, caso acabem suas tarefas antes do fim da jornada de trabalho;
- ii - *Matéria Prima*: São compradas em grande escala e distribuídos aos locais demandantes, podendo ser: saco de cimento, areia, brita e etc;

- iii- *Ferramentas e Máquinas*: Normalmente recursos de custos elevados, havendo assim um compartilhamento entre os canteiros de obras, podendo ser: serra elétrica, betoneira, furadeira e etc;

Podemos definir o MRCO, detalhado acima, formalmente através de um grafo completo $G = (V_c \cup V_d, A)$, onde $V_c = \{1, \dots, n\}$ é o conjunto de canteiros de obras e $V_d = \{n+1, \dots, n+m\}$ é o conjunto m de depósitos. Seja $A = \{(v_i, v_j) : v_i, v_j \in V_c \cup V_d, i \neq j\}$ o conjunto de arcos em que cada arco representa a viagem direta entre v_i e v_j com uma distância ou custo sendo igual a c_{ij} . Temos um conjunto de recursos com diferentes tipos representado por $R = \{1, \dots, r\}$. A demanda por um recurso $r \in R$ pelo vértice $i \in V_c$, ou a oferta de um recurso $r \in R$ pelo vértice $i \in V_c$ é denotado respectivamente por d_{ir} e b_{ir} . Seja $K = \{1, \dots, k\}$ o conjunto de veículos e w_k sua respectiva capacidade.

No caso de estudo, a quantidade de depósitos m é igual a quantidade de veículos k , visto que cada depósito m coincide com a residência do motorista de cada veículo k . O depósito de origem é o mesmo do destino final para cada veículo.

O objetivo deste problema consiste em construir k rotas de veículos que atendam às requisições de alocação de recursos entre os canteiros de obras e que minimizem o somatório dos custos nos arcos das devidas rotas.

O custo de um arco c_{ij} inclui despesas como: relativos ao transporte, tais como, combustíveis e despesas geradas com a utilização do veículo e custos relativos ao tempo do veículo em rota, que se refere ao tempo despendido com motoristas em rota.

Dutra (2015) em sua dissertação de mestrado abordou essa nova variante do Problema de Roteamento de Veículos. Como método de resolução este autor propôs duas abordagens de tratamento para o MRCO: uma primeira abordagem através de um problema de programação linear inteiro misto e uma outra abordagem através do método denominado como Duas Fases, primeiro com um algoritmo alocando os recursos e depois através de um modelo de roteamento de veículos com coleta e entrega.

Espera-se com este estudo desenvolver heurísticas eficientes para a resolução do problema em questão e através disso poder contribuir com a literatura deste importante problema além de fornecer uma ferramenta poderosa de gestão para a empresa em estudo.

1.2 Objetivos

O presente trabalho possui os objetivos que serão descritos a seguir.

1.2.1 Objetivo Geral

Desenvolver métodos heurísticos para realizar o roteamento de veículos que deverão fazer a movimentação de recursos entre canteiros de obras em uma empresa do ramo da construção civil. O desempenho de tais métodos serão apresentados e comparados entre si, com a metodologia utilizada pela empresa em estudo e com o método de Duas fases proposto por [Dutra \(2015\)](#).

1.2.2 Objetivos Específicos

- a- Sem a pretensão de ser exaustivo, realizar uma revisão literária de problemas que compartilhem restrições com o MRCO e estudar seus métodos de resolução;
- b- Implementar dois métodos heurísticos para roteamento dos veículos que farão a movimentação dos recursos entre os canteiros de obras, buscando reduzir a distância total percorrida pelos veículos;
- c- Implementar o método de Duas Fases, buscando evitar distorções na comparação dos resultados;
- d- Reproduzir a metodologia de distribuição dos recursos e roteamento dos veículos aplicada pela empresa;
- e- Comparar os resultados dos procedimentos propostos e implementados em instâncias de teste e instâncias reais;

1.3 Organização do Trabalho

O restante desta dissertação está organizado em quatro capítulos. O capítulo 2 apresenta uma revisão do problema geral de coleta e entrega, que faz parte do grupo de problema de roteamento de veículos, descrevendo suas principais variações em relação as restrições de nosso problema e seus devidos métodos e estratégias de resolução. Ainda neste capítulo é abordado os métodos de resolução utilizados por [Dutra \(2015\)](#) em sua dissertação de mestrado. No capítulo 3 é apresentado as heurísticas construtivas propostas. O capítulo 4 apresenta seus respectivos resultados computacionais. Finalmente, são apresentados os comentários finais e as oportunidades de pesquisas futuras.

Capítulo 2

Revisão da Literatura

Este capítulo é dedicado a um aprofundamento quanto as variações do Problema de Roteamento de Veículos que compartilhem características similares ao nosso problema de estudo MRCO. Para isto está dividido em quatro seções: Em 2.1 abordamos num sentido mais geral as principais definições e características do Problema Geral de Coleta e Entrega. Em 2.2 apresentamos as variações encontradas na literatura em relação as restrições de nosso problema de estudo. Em 2.3 é feita uma filtragem de problemas que possuem restrições quanto a múltiplos produtos, múltiplas visitas e divisão de coleta e entrega, relatando com maior detalhe suas principais características e métodos de resolução. Por fim, em 2.4 é apresentado os métodos propostos por [Dutra \(2015\)](#) em sua dissertação de mestrado para resolução do MRCO.

2.1 Problema Geral de Coleta e Entrega

O Problema Geral de Coleta e Entrega (General Pickup and Delivery Problem - GPDP) pode ser definido como um problema de construção de rotas em função da satisfação de requisições de pedidos de transportes feito por clientes. Estes pedidos possuem especificidades, tais como:

- 1- Tamanho e peso da carga.
- 2- Origem, lugar de coleta da carga.
- 3- Destino, lugar de entrega da carga.

Para tal atendimento o problema considera uma frota de veículos com disponibilidade para operação e que possuem uma determinada capacidade. Para cada veículo é considerado uma origem e destino. Neste problema de acordo com [Savelsbergh e Sol](#)

(1995) não há existências de locais de transbordo, assim sendo, os veículos devem sair de sua origem e irem diretamente para o destino especificado.

Dentro do mesmo contexto surge o *Dial-a-Ride Problem* (DARP), que considera o transporte de pessoas ao invés da requisição de pedidos de transportes para cargas. Dutra (2015) em sua dissertação de mestrado fornece uma rica revisão bibliográfica sobre o Problema Geral de Coleta e Entrega, e segundo este autor o DARP pode ser encontrado na literatura com problemas práticos de transportes de idosos e deficientes, transportes escolares de estudantes e serviços de ambulância.

Savelsbergh e Sol (1995) relatam duas situações que podem ocorrer ao Problema Geral de Coleta e Entrega em situações práticas, sendo que a segunda modifica algumas de suas características básicas:

1. *Requisições estáticas*: Na definição das rotas já se sabe todas as informações necessárias de todas as requisições.
2. *Requisições dinâmicas*: Ocorre atualização e inclusão de novas requisições após definição das rotas.

Dutra (2015) descreve a classificação referente as variações do GPDP em relação a função e utilização dos depósitos proposta por Parragh e Doerner (2008a) e Parragh e Doerner (2008b). A primeira classificação é relativa a problemas em que o depósito fornece ou recebe bens. Já a segunda classificação refere-se a problemas em que a utilização do depósito é apenas considerada no período em que o veículo encontra-se parado. Assim, considera-se para este caso, o depósito como um local de estacionamento para os veículos. A movimentação de bens ocorre apenas entre clientes, o que se assemelha ao nosso problema de estudo MRCO.

Um outra classificação foi a dos autores Berbeglia e Cordeau (2010) os quais dividem as variações do GPDP em relação a origem e destino dos bens. Abaixo temos as seguintes divisões feitas pelos autores:

- i- *Many-to-many*: Qualquer vértice de clientes pode ser origem ou destino de bens. Ocorre a troca de bens entre os clientes.
- ii- *One-to-many-to-one*: Consideram a existência de depósitos que podem ser a origem ou destino de bens.
- iii- *One-to-one*: As requisições já possuem origem e destino definidos. Por exemplo, os serviços de entrega dos correios.

Nesta última classificação o MRCO se enquadra na divisão *many-to-many*, pois em nosso caso de estudo os recursos são realocados entre os canteiros.

2.2 Variações do Problema Geral de Coleta e Entrega

Uma ampla quantidade de variações do Problema Geral de Coleta e Entrega podem surgir devido às características de cada problema em estudo. Apresentamos uma revisão da literatura de problemas que compartilham restrições com o nosso problema de estudo. Assim abordamos as seguintes restrições presentes no MRCO:

1. Múltiplos Depósitos;
2. Múltiplos Produtos;
3. Múltiplas Visitas;
4. Coleta e Entrega Simultânea;
5. *Splitting*;

Temos primeiramente o Problema de Roteamento de Veículos com Múltiplos Depósitos (*Multi-Depot Vehicle Routing Problems - MDVRP*) que considera um conjunto de depósitos. Laporte et al. (1988) propuseram como método de resolução para este problema um algoritmo *Branch and Bound*. Uma heurística Busca Tabu foi proposta por Renaud et al. (1996) e Cordeau et al. (1997). Thangiah e Salhi (2001) utilizaram uma heurística genética de clusterização. Dondo e Cerdá (2009) trabalharam uma heurística de refinamento com programação linear inteira mista.

Problemas que consideram a existência de um conjunto de produtos ou *commodities* são conhecidos na literatura como *Multi-Commodity Vehicle Routing Problem*. Como referência para este assunto podemos encontrar os trabalhos dos autores Hernández-Pérez e Salazar-González (2009) e Hernández-Pérez e Salazar-González (2014) que propuseram como método de resolução um modelo de programação linear inteira mista e um algoritmo *Branch and cut*. Uma heurística híbrida com três estágios de busca local sendo que um destes utiliza um modelo de programação inteira foi utilizada no trabalho de Hernández-Pérez et al. (2016).

O problema em estudo apresenta a particularidade em que cada cliente poderá ser visitado mais de uma vez por um mesmo veículo. Problemas com essa característica podem ser encontrados nos trabalhos de Mitra (2005) e Mitra (2008), Chemla et al. (2013), Erdoğan et al. (2010), Bordenave et al. (2009) e Psaraftis (2011).

Outra variação é o caso onde existem a coleta e entrega simultâneas (*Vehicle Routing Problem with Simultaneous Pick-up and Delivery - VRPSPD*), neste problema os

veículos não apenas entregam produtos aos clientes, mas também coletam simultaneamente sem a restrição de precedência. Em geral entregas de produtos originam de um depósito e as coletas devem ser transportadas a este mesmo depósito. Uma Busca Tabu foi proposta por [Bianchessi e Righini \(2007\)](#). [Gajpal e Abad \(2009\)](#) trabalharam neste problema com uma heurística baseada em probabilidade conhecida como Colônia de Formigas. [Subramanian et al. \(2010\)](#) propuseram uma heurística paralela que foi composta pelas heurísticas VND (*Variable Neighborhood Descent*) e ILS (*Iterated Local Search*). [Goksal et al. \(2013\)](#) trabalharam com uma heurística baseada em PSO (*Particle Swarm Optimization*) e a heurística VND (*Variable Neighborhood Descent*).

O problema com Divisão de Entrega (*Split Delivery Vehicle Routing Problem – SDVRP*), é definido como um problema no qual a entrega pode ser dividida. Consideraremos para fins deste estudo este problema como sendo tanto divisão da entrega como de coleta. Uma revisão da literatura sobre este tipo de problema mostrando suas principais propriedades e métodos de resolução pode ser encontrado no trabalho de [Archetti e Speranza \(2012\)](#). [Mitra \(2005\)](#) e [Mitra \(2008\)](#) utilizou em ambos os trabalhos modelos de programação linear inteira mista resolvidos heurísticamente. [Archetti et al. \(2006\)](#) propuseram uma heurística Busca Tabu com três fases. [Archetti et al. \(2008\)](#) apresentaram uma heurística híbrida que também utiliza a heurística Busca Tabu mas combinada com uma formulação de programação inteira.

Nesta seção foram apresentadas de maneira mais geral as variações do GPDP relacionadas com as restrições de nosso problema de estudo. Na seção seguinte serão apresentados apenas os problemas que compartilhem uma ou mais restrições referentes a múltiplos produtos, múltiplas visitas e divisão de coleta e entrega. Serão abordados os seguintes trabalhos de seus respectivos autores: [Hernández-Pérez et al. \(2016\)](#), [Chemla et al. \(2013\)](#), [Chen et al. \(2014\)](#), [Gulczynski et al. \(2011\)](#), [Erdoğan et al. \(2010\)](#), [Anily e Hassin \(1992\)](#), [Bordenave et al. \(2009\)](#) e [Psaraftis \(2011\)](#).

Por fim, a Tabela [2.1](#) resume os problemas apresentados neste capítulo que compartilham restrições com o MRCO. Para melhor entendimento do leitor apresentamos cada estudo encontrado na literatura e suas devidas restrições bem como os métodos que foram utilizados para sua devida resolução.

Tabela 2.1: Restrições existentes no MRCO e seus devidos métodos de resolução: M.D: Múltiplos Depósitos; M.P: Múltiplos Produtos; M.V: Múltiplas Visitas; C.E.S: Coleta e Entrega Simultânea; D.C.E: Divisão de Coleta e Entrega.

Artigo	M.D	M.P	M.V	C.E.S	D.C.E	Método de Resolução
Laporte et al. (1988)	*					<i>Branch and Bound</i>
Cordeau et al. (1997)	*					Heurística <i>Busca Tabu</i>
Thangiah e Salhi (2001)	*					Heurística Genética de Clusterização
Dondo e Cerdá (2009)	*					Heurística de Busca Local e PLIM
Bianchessi e Righini (2007)				*		Heurística Busca Tabu
Gajpal e Abad (2009)				*		Heurística de Colônia de Formigas
Subramanian et al. (2010)				*		Heurística VND e ILS
Goksal et al. (2013)				*		Heurística Baseada em PSO e VND
Hernández-Pérez e Salazar-González (2009)		*				<i>Branch and cut</i>
Hernández-Pérez e Salazar-González (2014)		*				<i>Branch and cut</i>
Mitra (2005)			*		*	PLIM e Heurística
Mitra (2008)			*		*	PLI e Heurística
Archetti et al. (2006)					*	Heurística Busca Tabu
Archetti et al. (2008)					*	Heurística Busca Tabu e PI
Hernández-Pérez et al. (2016)		*		*		Heurística VND e PD
Chemla et al. (2013)			*			<i>Branch and cut</i> e Busca Tabu
Chen et al. (2014)		*		*	*	PLIM e Heurística VND
Andersson et al. (2011)		*			*	Geração de Colunas
Gulczynski et al. (2011)	*				*	Heurística Baseada em PI
Erdoğan et al. (2010)			*	*		<i>Branch and cut</i>
Bordenave et al. (2009)			*	*		<i>Branch and cut</i>
Psaraftis (2011)		*	*			PD
Dutra (2015)	*	*	*	*	*	PLIM

2.3 Problemas com Restrições em Comum ao MRCO

Este problema, trabalhado por [Hernández-Pérez et al. \(2016\)](#), conhecido como *Multi-Commodity Pickup-and-Delivery Traveling Salesman Problem* (m-PDTSP), é considerado uma generalização do problema do caixeiro viajante em que um veículo capacitado com base em um depósito deve visitar um conjunto de clientes. Assim, cada local, depósito e clientes, devem ser visitados exatamente uma única vez pelo veículo. Não existem relações de precedência entre locais de coleta e entrega de um produto na rota.

O custo entre os pares de vértices não são necessariamente simétricos. Cada cliente demanda e/ou fornece uma quantidade de produtos. Um produto recolhido em um cliente pode ser fornecido a qualquer outro que demande deste mesmo produto. O veículo deve começar e terminar o percurso em um depósito. Como objetivo, este problema deseja encontrar uma rota Hamiltoniana para o veículo satisfazendo sua devida capacidade e suas requisições de coleta e entrega.

Como método de resolução os autores propuseram um algoritmo heurístico híbrido composto de três estágios, além de fazerem uso de técnicas de programação matemática para executar a busca local. Os três estágios consistem basicamente em:

- 1- Primeiro estágio: É a fase de inicialização que consiste de um processo que gera um conjunto de soluções iniciais, que não necessariamente serão viáveis, através de um procedimento aleatório ganancioso de escolha a cada iteração do vizinho mais próximo. O procedimento é executado um número específico de vezes para se obter conjunto de soluções iniciais;
- 2- Segundo estágio: Com o objetivo de melhorar as soluções obtidas no estágio anterior, é utilizado a heurística VND (*Variable Neighborhood Descent*) que é aplicada a cada uma das soluções selecionadas na primeira fase. Seis operadores derivados dos movimentos 2-opt e 3-opt são propostos;
- 3- Terceiro Estágio: Neste estágio acontece o refinamento e perturbação de cada mínimo local através da aplicação de uma combinação de três procedimentos. Primeira realiza um movimento aleatório 3-opt, onde seleciona três arcos não consecutivos e reconstrói uma nova turnê. O segundo procedimento executa movimentos aleatórios 4-opt mantendo a orientação dos quatro caminhos. Já o terceiro procedimento utiliza de uma perturbação que utiliza um modelo programação inteira mista;

Para testar o algoritmo proposto [Hernández-Pérez et al. \(2016\)](#) geraram quatro grupos de instâncias com: conjunto de produtos entre três e cinco, conjunto de vértices entre

cem e quatrocentos. As três estratégias de perturbação propostas no terceiro estágio obtiveram resultados bem sucedidos. Os resultados computacionais obtidos com a implementação da primeira estratégia foram considerados bastantes robustos. Quanto ao tempo computacional eles aumentam a medida que o número de vértices e produtos aumentam e a capacidade do veículo diminui. Com relação ao tamanho das instâncias o algoritmo foi capaz de resolver eficazmente instâncias com até quatrocentos vértices e com cinco produtos.

O m-PDST estudado por [Hernández-Pérez et al. \(2016\)](#) tem como uma das aplicações o serviço de contratação de bicicletas, ultimamente utilizado em cidades pelo mundo, que envolve coleta e entrega em determinados pontos conhecidos. Um único veículo capacitado visa fazer a redistribuição das bicicletas ao fim do dia entre os pontos de parada, com o intuito de restaurar a configuração do sistema. Cada ponto de parada, ou vértices, pode ser visitado mais de uma vez e também pode armazenar um produto, no caso bicicleta, para uma coleta posterior.

[Chemla et al. \(2013\)](#) estudaram o *Single Vehicle One-commodity Capacitated Pickup and Delivery Problem (SVOCPP)* como um problema de reequilíbrio com um único veículo e, no caso estático, pois os usuários não interferem durante o processo. Como característica deste problema cada vértice pode ser visitado mais de uma vez e ele pode ser origem ou destino dos recursos. Neste trabalho pode ser encontrado como métodos de resolução um modelo matemático que devido sua intratabilidade sofre uma relaxação, e posteriormente é resolvido através de um algoritmo *branch-and-cut* e uma busca tabu puramente combinatória fornecendo limites superiores para o problema.

O Problema de Coleta e Entrega não Pareado com Divisão de Entregas (*Pickup and Delivery Vehicle Routing Problem with Split Loads - UPDPSL*), trabalhado por [Chen et al. \(2014\)](#), tem como motivação o planejamento da realocação de mercadorias realizado pelas cadeias de varejo, mais especificamente em um departamento de logística de uma grande fabricante de tabaco na China central, que possui cinco unidades de produção dispersas em diferentes localidades.

Esta empresa realiza mensalmente um planejamento do tipo e quantidade de produto que cada planta deverá produzir. Através deste planejamento é possível identificar quais plantas demandarão e/ou ofertarão mercadorias. Cada planta possui uma frota de veículos homogênea para transporte de mercadorias entre os armazéns e está autorizada a ser visitada por mais do que um veículo. Cada veículo deve estar em um depósito no início e fim de cada jornada. O problema também considera a existência de várias mercadorias. Portanto, este problema tem como objetivo minimizar o custo total de viagens dentro desta rede de transporte atendendo aos requisitos.

Foi proposto primeiramente como método de resolução um modelo matemático não linear, mas que foi adaptado para um programa linear inteira mista, de modo que pode

ser resolvido pelo software CPLEX. [Chen et al. \(2014\)](#) também propuseram a utilização da heurística *Variable Neighborhood Search* (VNS).

As instâncias geradas para os devidos testes deste trabalho tiveram entre cinco e dez vértices e entre dois e seis a quantidade de materiais. Verificou-se a ineficiência do modelo matemático proposto. Em virtude disso foi adicionado um conjunto de restrições ao modelo matemático com o objetivo de se obter um intervalo entre os limites inferior e superior mais justo. Através dos testes verificou-se que solução ótima em várias instâncias. Quanto ao tempo computacional verificou-se que ele aumenta a medida que o as instâncias testadas são de grande escala e também com o aumento da demanda. O algoritmo heurístico proposto, VNS, forneceu soluções viáveis bem próximas do ótimo. Quanto ao tempo de execução, o VNS possui superioridade para instâncias de grande escala.

Um problema similar ao citado anteriormente encontrado na literatura é o *The Maritime Pickup and Delivery Problem with Time Windows and Split Loads*, trabalhado por [Andersson et al. \(2011\)](#) onde as companhias de navegação lidam com transportes de cargas a granel através de um conjunto de navios e um conjunto de portos. Para este problema em específico as cargas podem ser divididas em navios diferentes, com a introdução da ideia de cargas parciais. Os autores utilizaram como método de resolução a técnica de geração de colunas.

Trabalhado por [Gulczynski et al. \(2011\)](#) o Problema de Roteamento de Veículos com Múltiplos Depósitos e divisão de Entrega (*Multi-Depot Split Delivery Vehicle Routing Problem - MDSDVRP*), consiste no despacho de veículos de vários depósitos para atender as demandas dos clientes, sendo que neste caso, estas demandas podem ser divididas entre veículos de diferentes rotas e depósitos. MDSDVRP tem como objetivo minimizar a distância total que a frota de veículos de todos os depósitos irá percorrer, com a permissão que mais de um veículo atenda ao pedido de um cliente.

Como método de resolução os autores propuseram uma heurística baseada em programação inteira, que pode ser sintetizada nos seguintes passos:

- 1- Atribuição dos clientes aos depósitos;
- 2- Resolve nesta fase o problema de divisão de entregas em cada depósito separadamente;
 - a Uma solução inicial é gerada para o PRV sem divisões de entrega;
 - b Através de um programa inteiro misto, busca-se maximizar a economia proporcionada pela divisão de entrega e realocação de clientes em novas rotas;
 - c Algoritmo heurístico é proposto para reduzir a distância total percorrida pela frota;

- 3- Formulação do MDSDVRP como um programa inteiro misto, para através da divisão de entrega entre depósitos melhorar a solução gerada nos processos anteriores;
- 4- É proposto um algoritmo com o objetivo de melhorar as rotas geradas;

Para testar a eficiência da heurística proposta foram adaptadas instâncias cujo a quantidade de depósitos e clientes esteve respectivamente nos intervalos entre dois e cinco e entre cinquenta e duzentos e quarenta e nove. A partir dos resultados obtidos, verificou-se que as soluções encontradas foram consideradas eficientes pelos autores.

Outro trabalho presente na literatura que compartilha características com o MRCO é o *Non-Preemptive Capacitated Swapping Problem* (NPCSP), que é uma variante do Swapping Problem (SP) que pode ser visto no trabalho de [Anily e Hassin \(1992\)](#). Segundo [Erdoğan et al. \(2010\)](#) o NPCSP pode ser representado por um grafo completo direcionado onde existe: Um conjunto de vértices, sendo um vértice considerado o depósito e os demais sendo clientes. Um conjunto de arcos que contém o custo da viagem do vértice i ao vértice j . Cada vértice, exceto o depósito, pode fornecer ou demandar apenas um produto ou ambos. Neste problema, diferentemente do nosso problema de estudo, existe os chamados vértices de transbordo ou vértices intermediários. Os vértices podem ser visitados mais de uma vez. Cada produto possui um peso. A oferta e demanda total de um produto deverão ser iguais. Um veículo capacitado é utilizado para satisfazer a demanda entre os clientes. Este veículo deverá iniciar e terminar uma jornada no seu devido depósito. Objetivo é elaborar uma rota que tenha um custo mínimo de transporte.

Como método de resolução para o problema descrito acima os autores desenvolveram um algoritmo *branch and cut* através de três formulações matemáticas tomando como base um grafo transformado. Desigualdades válidas foram geradas e adicionadas ao algoritmo proposto. As instâncias utilizadas para o teste computacional tiveram entre vinte e trinta vértices e entre quatro e oito produtos.

Dentro do mesmo contexto do problema descrito acima, mas como uma particularidade, onde tanto a capacidade do veículo quanto demanda dos clientes são iguais a um. [Bordenave et al. \(2009\)](#) desenvolveram também um algoritmo *branch-and-cut* que foi testado para instâncias que tiveram no máximo duzentos vértices e oito produtos.

O Problema Generalizado de Roteamento de Veículos Capacitado com Coleta e Entrega (*General version of the Vehicle Routing Pickup and Delivery Problem - VRPPD-G*) trabalhado por [Psaraftis \(2011\)](#) considera vários produtos e o uso de apenas um ou dois veículos. Este problema possui como objetivo encontrar um planejamento de rotas que minimizem o custo total de cada viagem. Cada veículo deverá iniciar sua rota em um vértice considerado o depósito e posteriormente visitar outros vértices,

onde deverá coletar a quantidade necessária de produtos nos vértices ofertantes para posterior entregar aos vértices que demandam. Assim como no MRCO cada vértice pode ser visitado mais de uma vez. A matriz de origem destino é completa, mas não necessariamente simétrica. Uma outra característica deste problema é considerar que cargas viagem em sentidos opostos nos pares de vértices, o que sugere neste caso o estudo da utilização de dois veículos. Como método de resolução, o autor propôs a utilização de uma técnica exata, que foi a programação dinâmica.

2.4 Métodos de Resolução Propostos por Dutra

Nesta seção será apresentado os métodos de resolução para o MRCO propostos por Dutra (2015) em sua dissertação de mestrado.

2.4.1 Método das Duas Fases

Este método, denominado como método de duas fases, consiste basicamente em:

- i- Primeira Fase: É feita uma associação entre pares origem-destino (oferta-demanda), criando-se vértices para cada par. Obtém-se ao final uma matriz de distâncias.
- ii- Segunda Fase: Resolução dos dados obtidos anteriormente através de um modelo matemático do problema de roteamento de veículos capacitado com coleta e entrega.

Primeira Fase: Transforma num Problema de Coleta-Entrega

Esta fase tem o objetivo de transformar o MRCO num problema de coleta e entrega. Para isso são criados pares origem-destino entre os canteiros, onde um canteiro que ofereça um recurso será associado a outro canteiro que demande este mesmo recurso.

Em síntese a heurística propõem que para cada recurso disponível ou solicitado por um canteiro de obra cria-se um vértice, denominado como canteiro-recurso. Assim, partindo de um grafo vazio, a cada etapa será criado e adicionado um novo vértice. As arestas deste grafo são as distâncias originais entre os canteiros.

Uma vez criado o grafo com vértices de oferta e demanda, o objetivo passa a ser associar os pares origem-destino. Para atribuição destes vértices o autor propôs utilizar como critério de alocação dos recursos um método guloso baseado na distância mínima entre as arestas deste grafo.

Os vértices e arestas deste grafo que forem associados são retirados e adicionados a um grafo bipartido onde de um lado tem-se os vértices de oferta e do outro os vértices

de demanda. As arestas deste grafo bipartido são formadas pelas distâncias originais entre os canteiros.

Abaixo temos o algoritmo 1 do procedimento descrito acima.

Algoritmo 1: Criar pares origem-destino, proposto por (Dutra, 2015)

```

1 Início (G)
2   Ordenar em um vetor  $v$ , em ordem crescente, as arestas de  $G$  pela distância.
3   Enquanto  $v$  não estiver vazio Faça
4     Selecionar a primeira aresta de  $v$ .
5     Para cada par de nós  $(m, n) \in G_n$  um de oferta e um de demanda Faça
6       Se  $m = i$  e  $n = j$  e ambos os nós contenham o mesmo recurso Então
7         vincule  $(m, n) \in G_n$  como origem destino
8         retire o novo par origem-destino  $(m, n)$  de  $G_n$ 
9       Fim Se
10    Fim Para
11    retire a aresta selecionada de  $v$ 
12  Fim Enquanto
13 Fim

```

Ao fim desta primeira fase de atribuição é possível se obter uma matriz de distâncias, que pode ser calculada através das arestas do grafo bipartido obtido com as associações dos pares.

Segunda Fase: Problema de Coleta e Entrega Capacitado

Esta fase tem como objetivo resolver o problema de roteamento de veículos capacitados com coleta e entrega. Para isto, a primeira fase forneceu o insumo necessário para poder ser resolvido como um problema próximo do problema de roteamento de veículos com coleta e entrega e múltiplos depósitos proposto por Ropke e Pisinger (2006). O trabalho destes autores também considera janelas de tempo, porém essa característica não se encontra no nosso problema de estudo.

Estes autores propuseram como método de resolução uma heurística adaptativa que utiliza a heurística LNS (*Large Neighborhood Search*) que por sua vez é composta por várias heurísticas que tem o objetivo de avaliar a performance dos resultados. Inicialmente para representar o problema em estudo os autores apresentaram um modelo matemático, o qual foi utilizado nesta fase.

Assim, Dutra (2015) implementou o modelo matemático citado acima cuja função objetivo busca a minimização da soma ponderada das distâncias percorridas e do tempo total de viagens do veículos. Após a elaboração do modelo, o autor propôs a utilização da técnica de decomposição Benders (1962).

Abaixo temos o modelo na íntegra desenvolvido por Dutra (2015) :

Parâmetros:

S_r : Conjunto de Ofertas do recurso r .

t_{ij} : tempo de viagem entre os vértices $(i,j) \in A$.

l_j : peso acrescido ao veículo no vértice j , onde $l_i = -l_{i-n} \forall i \in D$.

s_i : tempo de serviço no vértice i .

Variáveis:

$s_{ik} \geq 0$: data em que o veículo $k \in K$ começa o serviço no vértice i .

$l_{ik} \geq 0$: peso do veículo $k \in K$ quando ele sai do vértice i .

$$x_{ijk} = \begin{cases} 1 & \text{se o veículo } k \in K \text{ se desloca pelo arco } (i,j) \in A \\ 0 & \text{caso contrário.} \end{cases}$$

$$\text{minimizar } \alpha \sum_{k \in K} \sum_{(i,j) \in A} d_{ij} x_{ijk} + \beta \sum_{k \in K} s_{\tau'_k} \quad (2.1)$$

sujeito à

$$\sum_{k \in K} \sum_{j \in N} x_{ijk} = 1 \quad \forall i \in P \quad (2.2)$$

$$\sum_{j \in NU\{\tau_k\}} x_{jik} - \sum_{j \in NU\{\tau'_k\}} x_{n+i,j,k} = 0 \quad \forall k \in K, i \in P \quad (2.3)$$

$$\sum_{j \in PU\{\tau'_k\}} x_{\tau_k j k} = 1 \quad \forall k \in K \quad (2.4)$$

$$\sum_{i \in DU\{\tau_k\}} x_{i\tau'_k} = 1 \quad \forall k \in K \quad (2.5)$$

$$\sum_{j \in NU\{\tau_k\}} x_{jik} + x_{\tau_k, \tau'_k, k} \leq 1 \quad \forall k \in K, i \in P \quad (2.6)$$

$$\sum_{i \in NU\{\tau_k\}} x_{ijk} - \sum_{i \in NU\{\tau'_k\}} x_{jik} = 0 \quad \forall k \in K, j \in N \quad (2.7)$$

$$s_{ik} + s_i + t_{ij} - M(1 - x_{ijk}) \leq s_{jk} \quad \forall k \in K, (i,j) \in A \quad (2.8)$$

$$x_{ijk} + x_{jik} \leq 1 \quad \forall k \in K, \forall (i,j) \in A \quad (2.9)$$

$$s_{n+i,k} \geq s_{ik} \quad \forall k \in K, i \in P \quad (2.10)$$

$$l_{ik} + l_j - M(1 - x_{ijk}) \leq l_{jk} \quad \forall k \in K, (i,j) \in A \quad (2.11)$$

$$l_{ik} \leq C_k \quad \forall i \in V, k \in K \quad (2.12)$$

$$l_{\tau_k, k} = l_{\tau'_k, k} = 0 \quad \forall k \in K \quad (2.13)$$

$$x_{ijk} \in \{0, 1\} \quad \forall (i,j) \in A, k \in K \quad (2.14)$$

$$s_{ik} \geq 0 \quad \forall i \in V, k \in K \quad (2.15)$$

$$l_{ik} \geq 0 \quad \forall i \in V, k \in K \quad (2.16)$$

A função objetivo do problema, **2.1**, minimiza a soma ponderada das distâncias percorridas e do tempo total de viagem de cada veículo. Em relação às restrições temos o seguinte:

Restrição **2.2**: Garante que todo o local de oferta $i \in P$ seja visitado.

Restrição **2.3**: Um veículo $k \in K$ deve sair de τ_k ou de um canteiro qualquer $j \in N$ e prosseguir até um local de oferta $i \in P$, e que ao sair de sua respectiva demanda $i + n$ deve prosseguir até τ'_k ou a um canteiro qualquer $j \in N$.

Restrições **2.4** e **2.5**: Asseguram que um veículo $k \in K$ sairá de seu respectivo depósito τ_k e retornará obrigatoriamente a ele τ'_k após sua rota.

Restrição **2.6**: Asseguram que se um veículo $k \in K$ for diretamente de τ_k para τ'_k ele não poderá fazer outra rota. Neste caso se diz que o veículo não foi utilizado.

Restrição **2.7**: Asseguram a existência de caminhos consecutivos entre τ_k e τ'_k para cada veículo $k \in K$.

Restrições **2.8** e **2.9**: Fazem com que se eliminem sub-rotas e que s_{ik} seja definido corretamente ao longo do percurso.

Restrição **2.10**: Restringe que a coleta seja feita antes da entrega.

Restrições **2.11**, **2.12** e **2.13**: Garantem que as variáveis de carga l_{ik} sejam definidas corretamente e que a capacidade dos veículos seja obedecida.

Restrições **2.14**, **2.15** e **2.16**: Definem o domínio das variáveis de decisão.

2.4.2 Modelo Integrado

Uma outra abordagem para tratar o MRCO de maneira integrada foi proposta. Para isto o problema foi modelado como um problema de programação linear inteira mista.

Abaixo temos o modelo na íntegra desenvolvido por [Dutra \(2015\)](#) :

Parâmetros:

R : conjunto de recursos/produtos.

K : conjunto de veículos.

B : conjunto de canteiros de obras (nós).

τ : conjunto de depósitos.

A : conjunto de arestas $(i, j) \mid i \in \{B \cup \tau\}, j \in \{B \cup \tau\}, i \neq j$.

$V = \{B \cup \tau\}$: conjunto de vértices do grafo.

τ'_k : depósito de retorno do veículo $k \in K \mid \tau'_k \subset \tau$.

τ_k : depósito de saída do veículo $k \in K \mid \tau_k \subset \tau$.

d_{ir} : demanda do nó $i \in B$ do recurso $r \in R$.

o_{ir} : oferta do nó $i \in B$ do recurso $r \in R$. Obs.: Depósitos não têm demanda ou oferta.

s_{ir} : tempo de serviço necessário para carregar/descarragar uma unidade do recurso $r \in R$ no nó $i \in B$.

t_{ij} : tempo de viagem do nó $i \in V$ até o nó $j \in V$.

c_k : capacidade do veículo $k \in K$.

p_{ir} : penalidade em unidades monetárias por não entregar o recurso $r \in R$ em $i \in B$.

ct : custo vinculado ao tempo de utilização dos veículos.

c_{ij} : custo de transporte devido a utilização da aresta $(i, j) \in A$.

t_{max} : quantidade de viagens máxima que um veículo pode realizar.

l_r : peso do recurso $r \in R$.

$T = \{t \in \mathbb{Z}_+ | t \leq t_{max}\}$: conjunto de viagens.

Variáveis:

$x_{ikr}^t \geq 0$: quantidade do recurso $r \in R$ coletado pelo veículo $k \in K$ no nó $i \in B$ na viagem $t \in T$.

$y_{ikr}^t \geq 0$: quantidade do recurso $r \in R$ entregue pelo veículo $k \in K$ no nó $i \in B$ na viagem $t \in T$.

$s_{ik}^t \geq 0$: data de chegada do veículo $k \in K$ em sua viagem $t \in T$ no nó $i \in V$.

$l_{ik}^t \geq 0$: peso do veículo k em sua viagem $t \in T$ ao chegar no nó $i \in V$.

$w_{ir} \geq 0$: quantidade do recurso $r \in R$ não entregue em $i \in B$.

$$z_{ijk}^t = \begin{cases} 1 & \text{se veículo } k \in K \text{ se desloca pelo arco } (i, j) \in A \text{ na viagem } t \in T \\ 0 & \text{caso contrário.} \end{cases}$$

$$\text{minimizar } ct \sum_{t \in T} \sum_{k \in K} s_{\tau'_k}^t + \sum_{i \in B} \sum_{r \in R} p_{ir} \times w_{ir} + \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} \sum_{t \in T} c_{ij} \times z_{ijk}^t \quad (2.17)$$

sujeito à

$$\sum_{t \in T} \sum_{k \in K} x_{ikr}^t \leq o_{ir} \quad \forall r \in R, i \in B \quad (2.18)$$

$$\sum_{t \in T} \sum_{k \in K} y_{ikr}^t + w_{ir} = d_{ir} \quad \forall r \in R, i \in B \quad (2.19)$$

$$\sum_{t_1 \in T} \sum_{i \in B} x_{ikr}^{t_1} - \sum_{t_2 \in T} \sum_{j \in B} y_{jkr}^{t_2} = 0 \quad \forall k \in K, r \in R \quad (2.20)$$

$$y_{jkr}^{t_2} + \sum_{i \in B} \sum_{\substack{t_1: t_1 < t_2 \\ t_1 \in T}} (y_{ikr}^{t_1} - x_{ikr}^{t_1}) \leq 0 \quad \forall j \in B, r \in R, k \in K, t_2 \in T \quad (2.21)$$

$$x_{jkr}^t - \sum_{i \in \{B \cup \tau_k\}} z_{ijk}^t \times c_k \leq 0 \quad \forall k \in K, j \in B, t \in T, r \in R \quad (2.22)$$

$$y_{jkr}^t - \sum_{i \in \{B \cup \tau_k\}} z_{ijk}^t \times c_k \leq 0 \quad \forall k \in K, j \in B, t \in T, r \in R \quad (2.23)$$

$$\sum_{i \in \{B \cup \tau_k\}} z_{ijk}^t - \sum_{p \in \{B \cup \tau'_k\}} z_{jpk}^{t+1} = 0 \quad \forall k \in K, t \in T \mid t < t_{max}, j \in B \quad (2.24)$$

$$z_{i\tau_k k}^t = 0 \quad \forall k \in K, i \in V, t \in T \quad (2.25)$$

$$z_{\tau'_k ik}^t = 0 \quad \forall k \in K, i \in V, t \in T \quad (2.26)$$

$$\sum_{i \in V} \sum_{j \in V} z_{ijk}^t \leq 1 \quad \forall k \in K, t \in T \quad (2.27)$$

$$\sum_{i \in V} \sum_{j \in V} z_{ijk}^{t+1} - \sum_{i \in V} \sum_{j \in V} z_{ijk}^t \leq 0 \quad \forall k \in K, t \in T \mid t < t_{max} \quad (2.28)$$

$$\sum_{j \in \{B \cup \tau'_k\}} z_{\tau_k jk}^1 - z_{ipk}^t \geq 0 \quad \forall k \in K, i \in B, p \in \{B \cup \tau'_k\}, t \in T \mid t > 1 \quad (2.29)$$

$$\sum_{\substack{t_1 \in T: \\ t_1 > t_2}} \sum_{i \in B} z_{i\tau'_k k}^{t_1} - z_{jpk}^{t_2} \geq 0 \quad \forall k \in K, p \in B, j \in \{B \cup \tau_k\}, t_2 \in T \quad (2.30)$$

$$\sum_{t \in T} \sum_{i \in \{B \cup \tau_k\}} z_{i\tau'_k k}^t \leq 1 \quad \forall k \in K \quad (2.31)$$

$$z_{i\tau'_k k_2}^t = 0 \quad \forall i \in V, t \in T, k \in K, k_2 \in K \mid k \neq k_2 \quad (2.32)$$

$$z_{\tau_k ik_2}^t = 0 \quad \forall i \in V, t \in T, k \in K, k_2 \in K \mid k \neq k_2 \quad (2.33)$$

$$s_{jk}^t - (s_{ik}^{t-1} + \sum_{r \in R} (s_{ir} \times l_r(x_{ikr}^{t-1} + y_{ikr}^{t-1})) + t_{ij}) + M(1 - z_{ijk}^t) \geq 0 \quad \forall i \in B, j \in \{B \cup \tau'_k\}, k \in K, t \in T \mid t \geq 2 \quad (2.34)$$

$$s_{ik}^1 - t_{\tau_k i} \times z_{\tau_k ik}^1 = 0 \quad \forall k \in K, i \in V \quad (2.35)$$

$$l_{jk}^t - (l_{ik}^{t-1} + \sum_{r \in R} (l_r(x_{ikr}^{t-1} - y_{ikr}^{t-1}))) + M(1 - z_{ijk}^t) \geq 0 \quad \forall i \in B, j \in \{B \cup \tau'_k\}, k \in K, t \in T \mid t \geq 2 \quad (2.36)$$

$$l_{ik}^1 = 0 \quad \forall i \in V, k \in K \quad (2.37)$$

$$l_{\tau'_k k}^t = 0 \quad \forall t \in T, k \in K \quad (2.38)$$

$$l_{jk}^t - c_k \leq 0 \quad \forall j \in V, k \in K, t \in T \quad (2.39)$$

$$z_{\tau_k j k}^t = 0 \quad \forall t \in T | t > 1, k \in K, j \in V \quad (2.40)$$

$$s_{jk}^t - M \sum_{i \in \{B \cup \tau_k\}} z_{ijk}^t \leq 0 \quad \forall j \in \{B \cup \tau_k'\}, k \in K, t \in T \quad (2.41)$$

$$l_{jk}^t - M \sum_{i \in \{B \cup \tau_k\}} z_{ijk}^t \leq 0 \quad \forall j \in \{B \cup \tau_k'\}, k \in K, t \in T \quad (2.42)$$

$$x_{ikr}^t \geq 0 \quad \forall i \in V, k \in K, r \in R, t \in T \quad (2.43)$$

$$y_{ikr}^t \geq 0 \quad \forall i \in V, k \in K, r \in R, t \in T \quad (2.44)$$

$$s_{ik}^t \geq 0 \quad \forall i \in V, k \in K, t \in T \quad (2.45)$$

$$l_{ik}^t \geq 0 \quad \forall i \in V, k \in K, t \in T \quad (2.46)$$

$$z_{ijk}^t \in \{0, 1\} \quad \forall i \in V, j \in V, k \in K, t \in T \quad (2.47)$$

$$w_{ir} \geq 0 \quad \forall i \in V, r \in R \quad (2.48)$$

A função objetivo do problema, [2.17](#), minimiza os custos relativos ao tempo de utilização dos veículos, os custos devidos à não entrega de recursos em algum dos nós demandantes e os custos de transporte dos veículos. Em relação às restrições temos o seguinte:

Restrição 2.18: O somatório de recursos do tipo $r \in R$ coletados no canteiro de obra $i \in B$ por todos os veículos em todo o horizonte de tempo deve ser no máximo igual à oferta deste recurso r no nó i .

Restrição 2.19: O somatório de recursos do tipo $r \in R$ entregues no canteiro de obra $i \in B$ por todos os veículos em todo o horizonte de tempo acrescido do montante deste recurso r que não foi entregue a i deve ser igual a demanda do recurso r pelo nó i .

Restrição 2.20: Essa restrição é a conservação do fluxo. Para cada carro $k \in K$, o somatório de recursos do tipo $r \in R$ coletados pelo veículo deve ser totalmente entregues aos canteiros de obras, de forma que o veículo retorne ao depósito vazio.

Restrição 2.21: A quantidade entregue do recurso $r \in R$ no nó $j \in B$ pelo veículo $k \in K$ na viagem $t_2 \in T$ deve ser no máximo igual a quantidade deste recurso r coletado anteriormente nas viagens $t_1 \in T | t_1 < t_2$ nos canteiros de obras menos a quantidade deste recurso r entregue anteriormente nas viagens $t_1 \in T | t_1 \leq t_2$ nos canteiros de obras.

Restrição 2.22: Trata-se de uma restrição de ativação da variável x_{jkr}^t . Na viagem $t \in T$ do veículo $k \in K$, somente haver à coleta na obra $j \in J$ de algum recurso $r \in R$

se existe uma viagem deste carro no tempo t vindo de uma obra $i \in B$. Um limite superior para a ativação pode ser a capacidade do veículo k .

Restrição 2.23: Trata-se de uma restrição de ativação da variável y_{jkr}^t . Na viagem $t \in T$ do veículo $k \in K$, somente haver à entrega na obra $j \in J$ de algum recurso $r \in R$ se existe uma viagem deste carro no tempo t vindo de uma obra $i \in B$. Um limite superior para a ativação pode ser a capacidade do veículo k .

Restrição 2.24: O veículo $k \in K$ que chega em $j \in B$ vindo de algum nó $i \in V \setminus \{\tau_k'\}$ na viagem $t \in T | t < t_{max}$, deve sair na viagem $t + 1 \in T$ para uma obra $p \in V \setminus \{\tau_k\}$, de forma que ele não fique parado em j .

Restrição 2.25: Como os depósitos são as residências dos motoristas, fizemos uma duplicação dos vértices dos depósitos num grafo bipartido para representar nós do depósito com função de saída de veículo e nós do depósito com função de retorno de veículo. Neste sentido, nenhum carro $k \in K$ em nenhuma viagem $t \in T$ pode retornar ao depósito de saída τ_k , independente de qual obra $i \in B$ tal viagem teve início.

Restrição 2.26: Por motivos similares expostos acima, nenhum carro $k \in K$ em nenhuma viagem $t \in T$ pode originar seu trajeto no depósito de retorno τ_k' , independente de qual obra $i \in B$ tal viagem iria visitar primeiramente.

Restrição 2.27: Cada veículo $k \in K$, para cada viagem $t \in T$, pode realizar somente um transcurso numa aresta $(i, j) \in A$.

Restrição 2.28: Somente pode existir uma viagem $t + 1 \in T$ numa aresta $(i, j) \in A$ com o veículo $k \in K$ se o mesmo veículo k viajou em $t \in T | t < t_{max}$ numa aresta $(i, j) \in A$.

Restrição 2.29: O veículo $k \in K$ somente pode percorrer uma aresta $(i, p) \in A$ numa viagem $t \in T | t > 1$ se o veículo k sai de seu depósito τ_k na primeira viagem.

Restrição 2.30: Se o veículo $k \in K$ utilizar uma aresta $(j, p) \in A$ numa viagem $t_2 \in T$ significa que ele o veículo saiu do depósito. Logo, ele deve retornar ao depósito em alguma viagem $t_1 \in T | t_1 > t_2$ vindo de qualquer $i \in B$.

Restrição 2.31: O veículo $k \in K$ só pode voltar ao seu depósito τ_k' uma única vez.

Restrição **2.32**: Nenhum veículo $k_2 \in K$ pode retornar a um depósito τ'_k se $k_2 \neq k$ em qualquer viagem $t \in T$ vindo de qualquer nó $i \in V$.

Restrição **2.33**: Nenhum veículo $k_2 \in K$ pode sair de um depósito τ_k se $k_2 \neq k$ em qualquer viagem $t \in T$ indo para qualquer nó $i \in V$.

Restrição **2.34**: Essa restrição refere-se ao acoplamento sequencial entre as datas de chegada de um veículo $k \in K$ quando k se desloca de $i \in V$ a $j \in B$.

Restrição **2.35**: A data de chegada do veículo $k \in K$, em sua primeira viagem, em uma obra $i \in B$ deve ser igual ao tempo de viagem de seu depósito τ_k à esta obra i , caso o veículo tenha visitado i logo após sair do depósito. Caso os veículos cheguem em seu respectivo depósito de retorno em sua primeira viagem, temos que $t_{\tau_k \tau'_k} = 0$, representando a permanência do carro no depósito. Mesma ideia se aplica quando $i = \tau_k$.

Restrição **2.36**: Essa restrição refere-se ao acoplamento sequencial entre os pesos de um veículo $k \in K$ em rota.

Restrição **2.37**: O peso que todo veículo $k \in K$ transporta em sua primeira viagem é nulo.

Restrição **2.38**: O peso que todo veículo $k \in K$ transporta ao retornar o depósito é nulo.

Restrição **2.39**: O peso que todo veículo $k \in K$ transporta em qualquer viagem $t \in T$ ao chegar numa obra $i \in B$ deve ser menor à sua capacidade.

Restrição **2.40**: Nenhum veículo pode sair do depósito em uma viagem $t > 1$

Restrição **2.41** e **2.42**: Nenhum veículo k pode chegar em uma obra ou em um depósito de retorno j em uma viagem t se ele não tiver vindo de uma obra ou depósito i na mesma viagem. Com isso, o peso do veículo segue a mesma lógica.

Restrição **2.43** a **2.48**: Delimitam o domínio das variáveis do problema.

2.4.3 Testes e Resultados

Em relação as instâncias utilizadas, o número máximo de canteiros de obra considerado foi de trinta vértices, devido a empresa estudada ter menos que o valor estipulado. A quantidade de veículos, bem como o número de depósito esteve entre três e cinco. Os recursos e seus devidos pesos tiveram respectivamente entre dez e vinte, e cinco e trinta. A demanda de cada obra foi gerada aleatoriamente entre o valor zero e no máximo a quantidade recursos que foi gerada.

O autor concluiu que o método de duas fases utilizando a resolução padrão do *Cplex* quanto a decomposição de *Benders* precisava de muito tempo computacional para resolver o problema. Assim este método demonstrou ser ineficiente para resolver o MRCO.

O Modelo Integrado demonstrou ser ineficiente para instâncias grandes, mas aparenta ser viável para instâncias pequenas. Contudo, este método também é considerado para fins da organização estudada improdutivo como assim conclui o autor.

2.5 Considerações Finais do MRCO em Relação a Revisão da Literatura

Este capítulo, motivado pelo problema de estudo que surgiu de uma situação prática de organização de fluxo de transporte de uma empresa do ramo da construção civil, buscou contextualizar o Problema Geral de Coleta e Entrega bem como suas diversas variações pelo incremento de restrições presentes na literatura.

Com o objetivo de exaurir ao máximo as possibilidades de variações, foi utilizado como critério, a busca de trabalhos presentes na literatura que compartilhassem uma ou mais restrições com o MRCO.

Por fim, verifica-se que além do trabalho proposto por [Dutra \(2015\)](#), não foi possível encontrar trabalhos, no campo de buscas na literatura feitas pelos autores, que possuam todas as restrições concomitantemente assim como no problema objeto deste presente trabalho.

Para tanto, os próximos capítulos tem o objetivo de fornecer métodos de resolução deste problema e respectivos testes para provar sua eficácia em relação ao que foi proposto.

Capítulo 3

Heurísticas para o MRCO

Neste capítulo são apresentados os procedimentos heurísticos construtivos para o MRCO, que tem por objetivo gerar um conjunto de rotas que atendam as necessidades dos clientes, no caso canteiros de obras, e que respeitem a capacidade do veículo.

O problema em estudo, como já mencionado no primeiro capítulo deste trabalho, é considerado pertencente à classe de problemas NP-Difícil, por ser uma generalização do Problema de Roteamento de Veículos, que por sua vez é uma generalização do Problema do Caixeiro Viajante que pertence a esta classe de problemas, ver por exemplo [Garey e Johnson \(1979\)](#). Para estes problemas não são conhecidos algoritmos em tempo polinomial capazes de encontrarem soluções ótimas.

O MRCO possui uma quantidade de restrições concomitantemente considerável, o que o torna mais complexo. Outra questão é quanto à inviabilidade, em fornecer solução em tempo computacional aceitável com o uso de pacotes de otimização, dos modelos matemáticos propostos para este problema.

O uso de técnicas heurísticas tem se mostrado extremamente eficaz, fornecendo boas soluções, não necessariamente a ótima, em tempo computacional satisfatório.

Diante destas considerações espera-se obter boas soluções em tempo computacional consideravelmente pequeno através de métodos heurísticos.

Este capítulo foi estruturado da seguinte forma: Primeiro abordaremos parâmetros comuns ao problema e utilizados em ambas heurísticas. As seções seguintes tratam de apresentar os procedimentos heurísticos propostos.

3.1 Parâmetros Comuns as Heurísticas

O problema estudado neste trabalho, cujo o objetivo é fazer o roteamento dos veículos saindo de seus respectivos depósitos, passando pelos canteiros de obras para coleta e/ou entrega e retornando a seus depósito de origem considera os seguintes parâmetros, apresentados a seguir.

Seja L o conjunto dos depósitos. Neste problema em estudo cada depósito possui exatamente um único veículo. C é o conjunto de canteiros de obras. Existe um conjunto R de recursos a serem demandados e ofertados pelas obras. Seja S o conjunto de oferta e B o conjunto de demanda. Seja w_l a capacidade do veículo pertencente ao depósito $l \in L$. Seja p_r o peso do recurso $r \in R$. Seja c_{ij} a distância entre canteiros de obras ou entre depósitos e canteiros de obras. Por fim, seja T_l a devida rota gerada para cada $l \in L$ depósitos.

No problema em questão pode ocorrer que um recurso possua mais unidades ofertadas do que demandadas ou vice-versa. Para isso, utilizamos a variável max_r para poder controlar esta questão. Assim tal variável é definida como: $max_r = \min(S, B)$. A Figura 3.1 descreve exatamente esta situação.

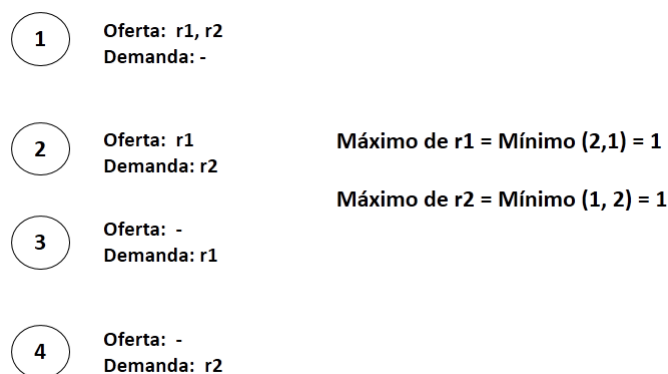


Figura 3.1: Cálculo da variável max_r .

3.2 Heurística Baseada em Particionamento - HBP

Nesta Heurística, para cada $l \in L$ desde que se tenha demanda de um recurso $r \in R$ que possua oferta, faremos um particionamento no conjunto C de canteiros criando um subconjunto \tilde{C} de elementos aleatórios de C . Esse subconjunto será sequenciado de maneira gulosa, ou seja pela menor distância entre canteiros. A sequência gerada será viável, não excedendo a capacidade do veículo e deverá atender um número máximo de demandas por recursos. Cada melhor sequência gerada será atribuída a rota de T_l .

Ao final desse procedimento, se ainda existir demanda que possua oferta, estas serão inseridas em um das rotas T_l de maneira gulosa.

Para o algoritmo 2, que apresenta a primeira parte da heurística HBP, seja μ o número máximo de atribuições possíveis para uma dada instância. Esta variável μ é o somatório de todos max_r , conf. em 3.1, para cada $r \in R$. Essa variável faz com que se tenha um valor exato de atribuições, pois pode ocorrer de em alguns casos a oferta por um recurso ser maior que a demanda ou vice-versa. Pela Figura 3.1 teríamos $\mu = 2$. Entende-se como atribuição at para este problema, quando a demanda de uma unidade de um recurso por um canteiro é atendida pela oferta de uma unidade deste mesmo recurso por outro canteiro, e que ao incluir este recurso, cujo peso é p_r , no veículo, não exceda a sua capacidade w_l .

Para cada $l \in L$, desde que $\mu \neq 0$, existem atribuições a serem feitas. Monta-se um subconjunto $\bar{C} \subseteq C$ de canteiros de obras selecionados aleatoriamente de C , cujo o tamanho deve ser proporcional a cada instância. Neste caso, o tamanho de \bar{C} é definido por θ que é o resultado da divisão da quantidade de canteiros da instância pela variável γ , que é um valor informado ao algoritmo.

A partir desse subconjunto \bar{C} é feito um percurso sequenciado de seus elementos a partir de um procedimento guloso com características semelhantes a heurística do vizinho mais próximo. Repetimos esse procedimento, fornecendo a cada iteração, uma origem diferente, que neste caso será cada elemento de \bar{C} . Essa repetição faz com que tenhamos um conjunto de sequências $S_{\bar{c}}$, para todo $\bar{c} \in \bar{C}$. Em cada sequência $S_{\bar{c}}$ gerada a sua respectiva variável $at_{\bar{c}}$ guarda a quantidade de atribuições possíveis de se realizar.

Existe a possibilidade de em cada sequência $S_{\bar{c}}$ gerada conter canteiros de obras que não estejam nem ofertando e nem demandando alguma unidade de um recurso $r \in R$. Logo, este canteiro deverá ser excluído de sua respectiva sequência.

Para exemplificar a situação descrita anteriormente tomemos as informações contidas na Figura 3.1. Uma possível sequência de canteiros gerada a partir dos passos anteriores poderia ser: $S_1 : \{1 \rightarrow 4 \rightarrow 3 \rightarrow 2\}$. Nesta sequência temos a coleta dos recursos r_1 e r_2 no canteiro 1. No canteiro 4 é entregue o recurso r_2 e não é coletado nenhum recurso. No canteiro 3 é entregue o recurso r_1 e não é coletado nenhum recurso. Ao sair do canteiro 3 o veículo está vazio. Ao chegar no canteiro 2, o último da sequência, o veículo que se encontra vazio não atenderá à sua respectiva demanda. Por ser o último da sequência tal canteiro não poderá ofertar nenhum recurso. Logo, este canteiro deverá ser excluído da sequência, pois não oferta e nem recebe recurso. Assim tem-se a nova sequência: $S_1 : \{1 \rightarrow 4 \rightarrow 3\}$.

Esse procedimento de escolha de \bar{C} e seus respectivos sequenciamentos podem ser repetidos β , valor este fornecido ao algoritmo. Por fim, deve-se escolher, dentre todas as

sequências geradas para este depósito $l \in L$ a que possua o maior número de atribuições. Para isso utilizam-se as variáveis B que guarda a melhor sequência e atg que guarda o número de atribuições desta sequência. Essa sequência é atribuída a T_l , que representa a rota associada ao depósito $l \in L$.

A quantidade de atribuições atg desta sequência escolhida e atribuída a rota T_l é subtraída de μ . Assim, caso $\mu \neq 0$, passa para o próximo depósito $l \in L$, caso contrário finaliza-se o algoritmo, pois isso garante que todas as demandas foram atendidas pelas respectivas ofertas disponíveis. Abaixo temos o algoritmo 2 descrito acima.

Algoritmo 2: Gera Rota Viável

```

1   $\beta$ :  $n^\circ$  de iterações
2   $\theta$ : tamanho de  $\bar{C}$ 
3  Para cada ( $l \in L : \mu \neq 0$ ) Faça
4     $atg \leftarrow 0$ 
5    Para ( $z \leftarrow 1$  até  $\beta$ ) Faça
6      Preencha  $\bar{C}$  com elementos escolhidos aleatoriamente  $c \in C$ 
7      Para cada ( $\bar{c} \in \bar{C}$ ) Faça
8         $at_{\bar{c}}, S_{\bar{c}} \leftarrow$  Vizinho mais próximo Adaptado
9        Se ( $at_{\bar{c}} > atg$ )
10          $B \leftarrow$  Melhor sequenciamento gerado
11          $atg \leftarrow$  Número de atribuições de  $B$ 
12       Fim se
13     Fim para
14   Fim para
15    $\mu \leftarrow \mu - atg$ 
16    $T_l \leftarrow T_l \cup B$ 
17 Fim para

```

Seja a instância do problema em estudo representada pela Figura 3.2.

Canteiro	Oferta	Demanda
c1	2r3	r4
c2	r1, r2	r3
c3	r4	r1, r3
c4	-	r2
c5	-	r1
c6	-	r3, r4
c7	-	r3
c8	-	r4
c9	-	r2
c10	-	r1

- C = 10 canteiros
- L = 2 depósitos
- R = 4 tipos de recursos
- $\mu = 5 \rightarrow$ atribuições
- $\beta = 3 \rightarrow$ repetições
- $\gamma = 2 \rightarrow$ proporção de divisão de C
- $\theta = 5 \rightarrow$ tamanho do subconjunto

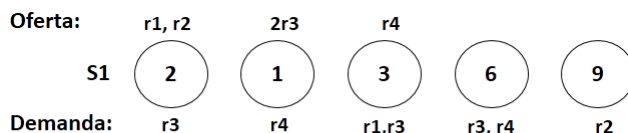
Figura 3.2: Instância I1 do problema.

Desde que $\mu \neq 0$, cada um dos dois depósitos terá uma respectiva rota a ele associada. Para a instância da Figura 3.2, iniciamos o procedimento com o primeiro depósito $l = 1$. Para este depósito o próximo passo é gerar o subconjunto \bar{C} , cujo o tamanho é definido por $\theta = 5$, que será realizada através da escolha aleatória de canteiros $c \in C$. Aleatoriamente podemos ter $\bar{C} = \{2, 3, 9, 6, 1\}$. Para o conjunto \bar{C} aplica-se o procedimento guloso semelhante a heurística do vizinho mais próximo. Para este exemplo podemos ter a seguinte situação com cinco sequências geradas, assim como na Figura 3.3:



Figura 3.3: Sequências geradas a partir de \bar{C} .

A escolha da melhor sequência é feita pela sequência que possuir o maior número de atribuições efetivadas. Tomemos como exemplo a primeira sequência gerada S_1 . A Figura 3.4 define como é calculado o número de atribuições de cada sequência.



Neste caso o número de atribuições é dado por:

- ✓ Canteiro 2 oferta r1 para Canteiro 3

O recurso 1 inserido no veículo excede a capacidade?
Sim: Não efetuar atribuição
Não: Atribuição efetivada

- ✓ Canteiro 2 oferta r2 para Canteiro 9
- ✓ Canteiro 1 oferta r3 para Canteiro 3
- ✓ Canteiro 1 oferta r3 para Canteiro 6
- ✓ Canteiro 3 oferta r4 para Canteiro 6

Figura 3.4: Cálculo do número de atribuições

Na Figura 3.4, vemos que dado um canteiro que ofereça um recurso que possua demanda em um canteiro à sua frente no percurso, deve-se sempre verificar se tal atribuição não excede a capacidade do veículo. Neste exemplo temos que S_1 possui 5 atribuições efetivadas.

Pela Figura 3.3 pode-se observar que dentre as sequências geradas, S_1 possui o maior número de atribuições, que é o de 5 atribuições. Logo, S_1 para esta primeira iteração é a melhor sequência. Assim B recebe S_1 . Como $\beta = 3$ para esta instância, outros dois subconjuntos serão ainda gerados e novas sequências $S_{\bar{c}}$ serão geradas em próximas iterações, a este mesmo depósito. Ao final de cada iteração o algoritmo verifica se encontra alguma sequência melhor que B . A melhor sequência B , escolhida dentre todas as iterações, é atribuída ao depósito l .

Para este exemplo, como μ é o número de atribuições realizadas na melhor sequência é de 5, o algoritmo irá finalizar gerando apenas uma rota para o primeiro depósito. Caso contrário passaria ao próximo depósito e todo o procedimento seria repetido.

No algoritmo 2 existe a possibilidade, embora pequena, de ao percorrer todos os $l \in L$ depósitos, ainda sim existir atribuições a serem feitas. Neste caso $\mu \neq 0$. Para garantir que todas as demandas que possuam ofertas sejam atendidas o objetivo passa a ser percorrer todos os $c \in C$ canteiros e verificar, para dado um recurso $r \in R$, quais deles possuem ofertas. O próximo passo é verificar se existe algum canteiro que demande este mesmo recurso r . Caso o recurso r não possua mais oferta ou demanda tal recurso não está mais disponível e passa-se a avaliar o próximo recurso.

Na Figura 3.5 temos um exemplo desta situação. Ao final do algoritmo 2 verificou-se que ainda existia demanda que possuía oferta disponível. Para fins didáticos tomemos este exemplo:

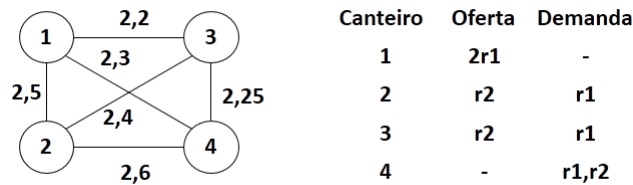


Figura 3.5: Pequena instância PI1.

A próxima etapa passa a ser a de criar pares origem-destino. Esta atribuição é realizada em função da menor distância entre origem e destino. Tais pares devem ser viáveis, ou seja, o peso dos recursos inseridos nesses pares não deve exceder a capacidade do veículo. Assim, na Figura 3.6 temos um exemplo de tal aplicação. O canteiro 1 oferta duas unidades do recurso 1 e existe 3 canteiros que demandam este

mesmo recurso. Tais pares serão criados avaliando a menor distância origem-destino. No caso do recurso 2 existem dois canteiros ofertando e apenas uma demanda. Tal atribuição seguirá o mesmo método descrito anteriormente.

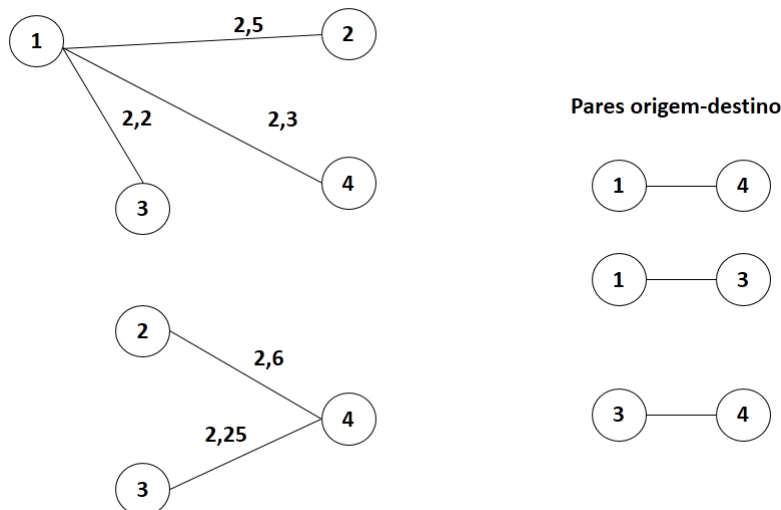


Figura 3.6: Pares origem-destino.

Com o conjunto de pares origem-destino o objetivo passa a ser criar pequenas rotas que possuam a mesma origem em comum. Ainda nesta etapa deve-se verificar se tal pequena rota excede a capacidade do veículo. Assim para Figura 3.6 temos: $PequenaRota_1 : \{1 \rightarrow 4 \rightarrow 3\}$, $PequenaRota_2 : \{3 \rightarrow 4\}$.

Das duas pequenas rotas, hipoteticamente a primeira excede a capacidade do veículo. Verifica-se em qual canteiro a rota extrapola a capacidade do veículo. Para este exemplo, verifica-se que os recursos que deverão ser entregues no canteiro 3 são os que extrapolam a capacidade do veículo. Assim tais recursos não serão carregados no veículo. O veículo deverá retornar uma segunda vez no canteiro 1 para coleta-los. A pequena rota passa a ser: $PequenaRota_1 : \{1 \rightarrow 4 \rightarrow 1 \rightarrow 3\}$.

Por fim, tem-se um conjunto de pequenas rotas viáveis. Cada pequena rota é atribuída após o último canteiro de cada uma da(s) T_l rota(s) geradas no algoritmo 2. Para cada pequena rota criada tal escolha de atribuição se dá pela menor distância entre o seu primeiro canteiro e o último canteiro de cada rota T_l . Com isso, todo o procedimento é finalizado, e é garantido que se tem T_l rota(s) viáveis(s).

3.3 Heurística Rota do Vizinho mais Próximo - RVP

Nesta Heurística cada unidade de um recurso $r \in R$ e seu respectivo canteiro $c \in C$ de oferta ou demanda é transformado em um vértice de um grafo canteiro-recurso. Assim teremos vértices de oferta e de demanda. A distância entre eles é a própria distância original dos canteiros e depósitos. Esses serão atribuídos a cada depósito (veículo) de maneira gulosa, pela menor distância, desde que o peso do recurso inserido na rota não exceda a capacidade do veículo. Ao fim, todas as atribuições serão garantidas.

Inicialmente este procedimento transforma cada canteiro e seus respectivos recursos de oferta e demanda em um grafo em que cada vértice é um par canteiro-recurso. As distâncias entre cada vértice deste grafo é a própria distância entre cada canteiro pertencente ao vértice par canteiro-recurso. Tal procedimento é exemplificado dado a instância fornecida pela Figura 3.7 abaixo.

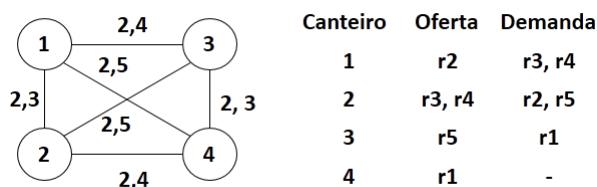


Figura 3.7: Instância I2 do problema.

Como já mencionado, cada vértice irá conter o par canteiro-recurso. Assim teremos para este problema vértices de oferta e vértices de demanda. Abaixo temos a representação através da Figura 3.8.

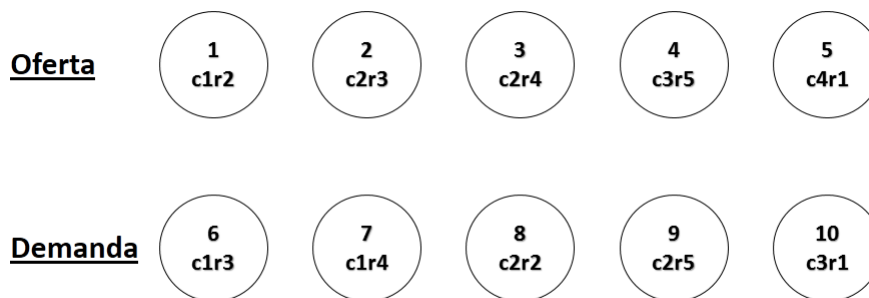


Figura 3.8: Grafo gerado a partir dos pares canteiro-recurso.

As arestas deste novo grafo é composta pelas distâncias originais entre os canteiros. Para este exemplo, vamos supor que exista apenas um único depósito, sendo que nele está a distância dos 4 canteiros na seguinte ordem: 2 de C1, 4 de C2, 5 de C3 e 3 de

C4. Tal instância seria composta pela matriz de distâncias representada pela Figura 3.9, sendo D1 o depósito e os demais vértices do grafo:

	1	2	3	4	5	6	7	8	9	10	D1
1	0,0	2,3	2,3	2,4	2,5	0,0	0,0	2,3	2,3	2,4	2,2
2		0,0	0,0	2,5	2,4	2,3	2,3	0,0	0,0	2,5	2,4
3			0,0	2,5	2,4	2,3	2,3	0,0	0,0	2,5	2,4
4				0,0	2,3	2,4	2,4	2,5	2,5	0,0	2,5
5					0,0	2,5	2,5	2,4	2,4	2,3	2,3
6						0,0	0,0	2,3	2,3	2,4	2,2
7							0,0	2,3	2,3	2,4	2,2
8								0,0	0,0	2,5	2,4
9									0,0	2,5	2,4
10										0,0	2,5

Figura 3.9: Matriz de distâncias.

Uma vez explicado o procedimento anterior, passaremos a descrever o algoritmo 3 que apresenta o método de roteamento dos veículos aos canteiros proposto para atender às características do problema estudado neste trabalho. O algoritmo considera os seguintes parâmetros:

Parâmetros:

R : Conjunto de Recursos.

S_r : Conjunto de Vértices de Oferta do recurso r .

B_r : Conjunto de Vértices de Demanda do recurso r .

$S = \cup_{r \in R} S_r$: Conjunto de Vértices de Oferta.

$B = \cup_{r \in R} B_r$: Conjunto de Vértices de Demanda.

L : Conjunto dos Depósitos.

i_l : Último vértice inserido na rota l .

N_l : Conjunto de candidatos a visitar a partir de i_l .

T_l : Rota do depósito l .

max_r : Máximo de atribuições a ser realizada por cada recurso $r \in R$.

q_{r_l} : Quantidade do recurso r na rota l .

$c_{i_l, v}$: Custo no arco entre $i_l \in L$ até $v \in N_l$.

w_l : Capacidade do veículo pertencente a rota l .

p_l : Peso acumulado dos recursos na rota l .

p_v : Peso do recurso do vértice $v \in N_{i_l}$.

$p_v > 0$: se $v \in S$.

$p_v < 0$: se $v \in B$.

Algoritmo 3: Gera Rota pelo Menor Custo

```

1  Para todo  $(l \in L)$  Faça
2       $i_l = l$ 
3       $T_l = \{i_l\}$ 
4       $N_l = S$ 
5       $p_l = 0$ 
6  Fim para
7  Para todo  $(r \in R)$  Faça
8       $max_r = \min(|S_r|, |B_r|)$ 
9      Para todo  $(l \in L)$  Faça  $q_{rl} = 0$ 
10 Fim para
11 Enquanto  $(\exists l \in L : N_l \neq \emptyset)$  Faça
12      $\Delta = \infty$ 
13     Para  $l \in L : N_l \neq \emptyset$  Faça
14          $j_l = \arg \min_{v \in N_l : p_l + p_v \leq w_l} \{c_{il,v}\}$ 
15         Se  $(c_{il,v} < \Delta)$ 
16              $\bar{l} = l$ 
17              $j = j_l$ 
18              $\Delta = c_{il,v}$ 
19         Fim Se
20     Fim para
21     Para  $l \in L$  faça  $N_l = N_l - \{j\}$ 
22      $p_{\bar{l}} = p_{\bar{l}} + p_j$ 
23      $i_{\bar{l}} = j$ 
24      $T_{\bar{l}} = T_{\bar{l}} \cup \{i_{\bar{l}}\}$ 
25     Se  $(j \in S)$ 
26         Seja  $r_j$  o recurso ofertado pelo nó  $j$ 
27          $q_{r\bar{l}} ++$ 
28          $max_r --$ 
29         Se  $(max_r = 0)$  então para  $l \in L$  faça  $N_l = N_l - S_r$ 
30          $N_l = N_l \cup B_r$ 
31     Fim Se
32     Senão
33         Seja  $r_j$  o recurso demandado pelo nó  $j$ 
34          $q_{r\bar{l}} --$ 
35         Se  $(q_{r\bar{l}} = 0)$  então  $N_{\bar{l}} = N_{\bar{l}} - B_r$ 
36          $B_r = B_r - \{j\}$ 
37     Fim senão
38 Fim

```

Na fase de inicialização, a variável i_l , que armazena a informação do último vértice inserido na rota, recebe o seu respectivo depósito de origem. Para cada depósito $l \in L$, seu respectivo conjunto rota T_l recebe a variável i_l . Cada depósito possui um conjunto N_l de candidatos a inserir em sua respectiva rota. Este conjunto inicializa recebendo o conjunto de ofertas S , pois ao sair do depósito o veículo está vazio, sendo necessário passar por um vértice de oferta. Vale ressaltar que ao passar por um vértice de oferta, este veículo terá adicionado em seu conjunto de candidatos as respectivas demandas do recurso ofertado. Tal atribuição é realizada na linha 30 do algoritmo 3.

Tomando como exemplo a instância representada pelas Figuras 3.7 e 3.8 que possuem apenas um depósito. Teríamos o conjunto de candidatos do depósito inicialmente formado por $N_{D1} = \{1, 2, 3, 4, 5\}$ que são os vértices de oferta. Pela matriz de distâncias da Figura 3.9, vemos que a distância do depósito até o vértice 1 é a menor. Neste caso, o vértice 1 será retirado de N_l e deverá ser acrescentado os vértices que demandam do mesmo recurso oferecido pelo vértice 1. Assim teríamos $N_{D1} = \{2, 3, 4, 5, 8\}$. Tal comando é repetido ao longo do algoritmo até que para todo $l \in L : N_l = \emptyset$.

A variável p_l representa o peso acumulado na rota do depósito $l \in L$. Inicialmente esta rota não possui nenhum recurso inserido, logo seu peso é zero. A variável p_v representa o peso do recurso presente no vértice v . Se este for um vértice de oferta $p_v > 0$, caso for um vértice de demanda $p_v < 0$.

No problema em questão existe a possibilidade de ocorrer uma oferta maior que a demanda e vice-versa. Para que seja atendido o número exato de atribuições possíveis, as variáveis q_{r_l} e max_r serão utilizadas. A variável max_r foi melhor explicado na seção 3.1.

A variável q_{r_l} , que representa a quantidade de um recurso $r \in R$ na rota do depósito $l \in L$, iniciará para cada rota com zero unidades. Sempre que um recurso for alocado em sua respectiva rota $l \in L$ será adicionado a ela uma unidade. Este recurso ao ser consumido por um vértice demanda desta mesma rota $l \in L$, será subtraído uma unidade.

A variável max_r representa a quantidade máxima de atribuição de cada recurso. Por exemplo, se um recurso r_1 possui três ofertas e duas demandas, o máximo de atribuições max_r deste recurso será o mínimo entre a oferta S_r e demanda B_r . Neste caso $max_{r_1} = 2$.

Enquanto existir candidatos pelo menos em um dos últimos vértices de cada rota, ($\exists i_l \in L : N_l \neq \emptyset$), os próximos passos serão executados. Para cada último vértice i_l , desde que seu respectivo conjunto de candidatos N_l não esteja vazio, procura-se a menor distância entre um vértice pertencente a N_l e o último vértice da rota i_l .

Outra condição verificada é a viabilidade da rota, ou seja, o peso do recurso pertencente ao vértice escolhido adicionado ao peso residual da rota não deve exceder a capacidade do veículo. Ao final desta etapa é selecionado um vértice j para entrar em umas das rotas escolhida, podendo ser ele de oferta ou demanda.

O vértice j escolhido na etapa anterior deverá ser retirado do conjunto N_l para todo $l \in L$. O peso $p_{\bar{l}}$ acumulado desta rota será atualizado. A variável $i_{\bar{l}}$ que representa o último vértice inserido nesta rota recebe o vértice j . A rota $T_{\bar{l}}$ recebe esta variável $i_{\bar{l}}$.

Posteriormente verificando que o vértice j seja de oferta $j \in S$ a unidade coletada de seu respectivo recurso é adicionada para esta rota a variável $q_{r_{\bar{l}}}$. A variável max_r será subtraída em uma unidade, o que informa que aquela unidade do recurso foi coletada. Depois verifica se max_r é igual a zero, e se sim, todas as ofertas deste recurso deverão ser excluídas do conjunto de candidatos para todo N_l em que $l \in L$. Esta informação diz que, todas as possíveis atribuições deste recurso já foram efetuadas. O próximo passo é liberar os canteiros que demandem o recurso coletado no canteiro j , adicionando-os ao conjunto $N_{\bar{l}}$.

Caso o vértice seja de demanda $j \in B$ a unidade consumida de seu respectivo recurso é subtraída da variável $q_{r_{\bar{l}}}$ o que informa que aquela unidade do recurso foi entregue na respectiva rota. Depois verifica se $q_{r_{\bar{l}}}$ é igual a zero, e se sim, todas as demandas deste recurso deverão ser excluídas do conjunto de candidatos para todo N_l em que $l \in L$. Essa informação diz que este recurso não encontra-se disponível mais na rota, assim os elementos que possuírem demandas do recurso do candidato escolhido deverão ser excluídas do conjunto N_{i_l} . Por fim o vértice de demanda escolhido é retirado do conjunto de demandas B_r deste recurso.

Ao final de todas estas etapas o algoritmo fornece as devidas rotas T_l atribuídas as seus respectivos depósitos e que sejam viáveis ao problema em estudo.

Capítulo 4

Resultados Computacionais

Este capítulo é dedicado a apresentação dos conjuntos de instâncias utilizados, bem como os testes computacionais e análises realizados por meio dos procedimentos heurísticos propostos, que foram desenvolvidos neste trabalho. Assim, o objetivo deste capítulo passa ser a comparação entre os métodos heurísticos propostos e método de terceiros.

4.1 Instâncias de Testes

O conjunto de instâncias foi gerado semelhante ao proposto por Dutra (2015), as quais foram construídas a partir de adequações de instâncias *benchmark* de Solomon e que estão disponíveis na seguinte página da internet: <http://www.sintef.no/Projectweb/TOP/VRPTW/Solomon-benchmark/100-customers/>. Estas instâncias são divididas em três conjuntos:

- c- Em que clientes são divididos aleatoriamente sobre uma região.
- r- Em que clientes são divididos por agrupamentos.
- rc- Mesclam características das duas anteriores.

As instâncias do tipo *r* serão consideradas para fins de teste como as instâncias que mais se assemelham a realidade encontrada do problema em estudo.

Cada instância contem as seguintes características:

1. Conjunto de Vértices: Foram considerados conjuntos de instâncias com $|C| = 10$ e $|C| = 29$. Sendo $|C|$ a quantidade de canteiros de obras (vértices).
2. Conjunto de Depósitos: Variam aleatoriamente entre 3 e 5 depósitos.

3. Conjunto de Recursos: Variam aleatoriamente entre 10 a 20 tipos de recursos.
4. Peso dos recursos: Variam aleatoriamente entre 5 a 30 unidades de peso.
5. Conjunto de Veículos: Para o problema de estudo o número de depósito foi considerado como igual ao número de veículos.
6. Capacidade dos Veículos: 100 unidades de peso.
7. Quantidade de Demanda e Oferta: As demandas e ofertas de cada canteiro foram geradas aleatoriamente entre 0 e o número máximo de recursos gerados.

4.2 Resultados

Antes de apresentarmos os resultados, objeto este desta seção, falaremos primeiro a respeito dos parâmetros utilizados na heurística Gera Rota para cada Veículo-GRV e da função objetivo, comum as duas heurísticas.

Após os testes verificou-se que para $\gamma = 1.2$ e $\beta = 10$ a heurística GRV apresentou melhores desempenhos, no que diz respeito ao custo total menor.

A função objetivo, é o somatório total dos custos das arestas do canteiros presentes na(s) devida(s) rota(s) proposta(s) pelos procedimentos, ou seja, é a distância total percorrida pelo(s) veículo(s).

Nossas análises serão baseadas nas comparações de desempenho das heurísticas propostas com os métodos aplicados na Empresa e o método de Duas Fases proposto por [Dutra \(2015\)](#) em Instâncias Reais. Faremos análises de comparação entre as heurísticas propostas e em uma heurística modificada, que será descrita posteriormente, sobre as instâncias de testes, que serão divididas em: Instâncias de Pequeno Porte e Instâncias de Porte Real.

4.2.1 Instâncias Reais: Comparação com métodos de Dutra e da Empresa em estudo

Para atingir ao objetivo desta seção, foi selecionado o método de Duas Fases proposto por ([Dutra, 2015](#)), o qual foi abordado no capítulo 2. Este procedimento foi implementado pelos autores deste estudo com o objetivo de evitar distorções nos resultados. A segunda fase deste procedimento foi resolvida através do solver *Cplex* em seu default.

Para fins de comparação, o método de Duas Fases, o método aplicado pela Empresa e as heurísticas construtivas propostas pelos autores deste estudo levaram em consideração na elaboração da função objetivo os mesmo parâmetros. Assim a função objetivo destes é composta pela soma das distâncias percorridas dos veículos em rota.

Nesta seção o objetivo é a comparação dos desempenhos dos métodos dos autores e de terceiros aplicados à situação real encontrada diariamente pela empresa. Para isso, em virtude da indisponibilidade de todos os dados, restrição esta imposta pela Empresa, os autores construíram instâncias aproximadas das reais através de algumas informações concedidas.

Assim como na Figura 4.1 que representa o agrupamento de obras da região de Contagem e Betim, os autores consideraram aproximadamente 30 canteiros de obras e 4 depósitos neste agrupamento. As coordenadas de cada canteiro de obra e depósito foi encontrada e a partir delas foi possível, assim como realizado pelo decisor na empresa, elaborar uma matriz de distâncias entre cada canteiro e depósito.

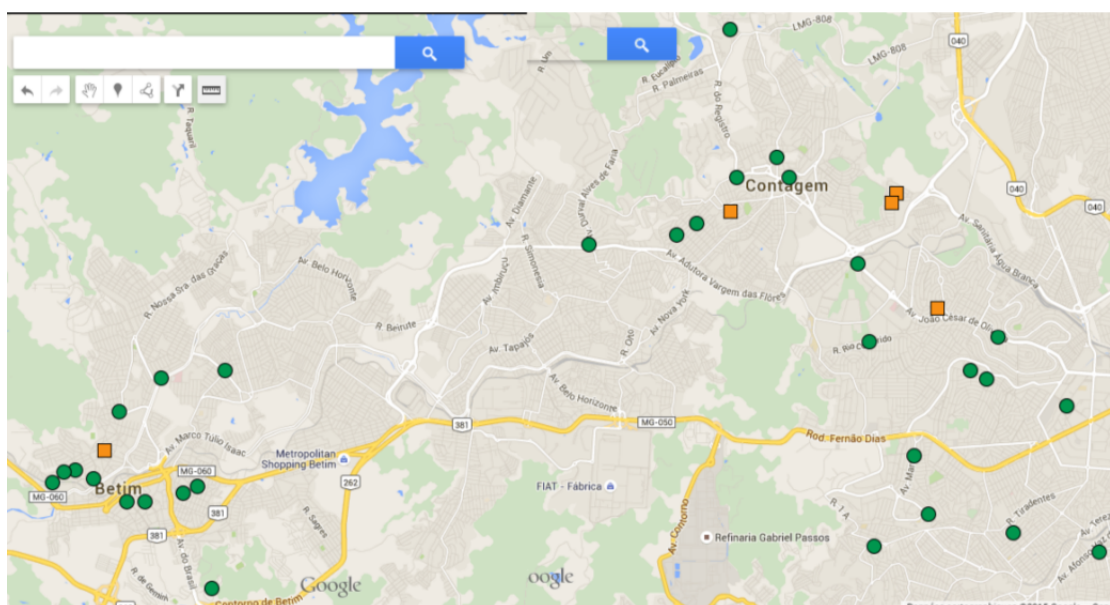


Figura 4.1: Agrupamento de um conjunto de obras. Fonte: Dutra(2015), p.28.

Para a coleta da quantidade de recursos, demandados e ofertados, identificou-se uma média ao longo de três dias, assim como pode ser visto em Dutra (2015). O número de recursos envolvidos está entre 4 e 8 tipos de recursos, com demandas e ofertas variando entre 0 e 6 unidades.

Com o fim de simular a situação real de roteamento dos veículos enfrentada na organização, os autores utilizaram da mesma metodologia aplicada pelos decisores na empresa. Dada a matriz de distâncias entre os canteiros e depósitos, o empregado responsável pela decisão deverá fazer primeiramente a alocação dos recursos entre as obras, isso em relação a menor distância entre elas. Finalizado as alocações de recursos, o decisor elabora as rotas em função destas alocações entre obras através da ferramenta *Maps* do Google.

Desta forma, os autores aplicaram tal procedimento acima, manualmente, para os

três dias de trabalho propostos e obtiveram uma aproximação dos custos da empresa em relação a distância total percorrida pelos veículos diariamente.

Com estes resultados foi possível fazer uma comparação aproximada da situação real enfrentada pela organização com os métodos propostos pelos autores e o método de Duas fases proposto por (Dutra, 2015). Tais resultados podem ser vistos na Tabela 4.1.

Tabela 4.1: Resultados para 3 instâncias reais

I	Real	Duas Fases		HBP		RVP	
		U.B	T[s]	F.O	T[s]	F.O	T[s]
dia01	766,12	726,71	1800,00	499,99	0,06	392,92	0,01
dia02	828,05	626,61	1800,00	567,64	0,07	500,99	0,01
dia03	957,72	-	1800,00	587,88	0,10	617,44	0,01

Na Tabela 4.1 pode-se verificar que o método de duas fases encontrou em duas de três instâncias uma solução inteira, a qual identificamos como *Upper Bound* U.B, para a segunda fase do problema. Porém o tempo gasto para encontrar tal solução é bem superior aos tempos encontrados nas heurísticas. Cogita-se de que este método tenha encontrado uma solução inteira, para a segunda fase do problema, devido a quantidade de recursos e suas respectivas demandas e ofertas não serem relativamente grandes. Com isso o número de vértices pares canteiro-recurso utilizado por este método e gerado em sua primeira fase é relativamente pequeno fazendo com que o modelo consiga encontrar estas soluções.

Ambas heurísticas apresentam resultados relativamente melhores comparado ao método de Duas Fases, principalmente em relação ao tempo de execução. A heurística RVP apresenta resultados melhores em duas das três instâncias em relação à heurística HBP. Ainda assim não podemos dizer que a heurística RVP será sempre superior à heurística HBP. Pode se concluir que a utilização de tais heurísticas em relação à metodologia aplicada pela empresa apresentou em todas instâncias reais melhores resultados em relação a distância total percorrida e em relação aos tempos computacionais que foram relativamente baixos.

4.2.2 Instâncias de Pequeno Porte - $|C| = 10$

As Tabelas 4.2, 4.3 e 4.4 apresentam os resultados obtidos em cada instância para cada procedimento, bem como seus tempos de execuções. Já a Tabela 4.5 informa para cada conjunto de instâncias qual a porcentagem de resultados melhores cada procedimento obteve.

Com já mencionado, nesta sub-seção o objetivo é verificar o comportamento de cada procedimento para conjunto de instâncias de pequeno porte, contendo 10 canteiros.

Tabela 4.2: Resultados para instâncias do tipo *c* com 10 canteiros de obras

Instâncias	Heurística HBP		Heurística RVP	
	Função Objetivo	Tempo [s]	Função Objetivo	Tempo [s]
c101	280,17	0,0128	103,03	0,0046
c102	543,01	0,0076	147,30	0,0108
c103	383,90	0,0036	127,27	0,0025
c104	469,55	0,0056	101,84	0,0018
c105	590,85	0,0112	152,90	0,0045
c106	360,08	0,0042	158,90	0,0030
c107	525,22	0,0068	160,02	0,0065
c108	364,12	0,0025	112,86	0,0024
c109	308,01	0,0020	73,81	0,0024

A Tabela 4.2 fornece os resultados do tipo C, nos quais os clientes são divididos aleatoriamente em uma região. Pode-se verificar que para todas as instâncias a heurística RVP obtém os melhores resultados. Além disso verifica-se uma proximidade nos tempos de execuções de cada procedimento.

Tabela 4.3: Resultados para instâncias do tipo *r* com 10 canteiros de obras

Instâncias	Heurística HBP		Heurística RVP	
	Função Objetivo	Tempo [s]	Função Objetivo	Tempo [s]
r101	476,70	0,0023	603,82	0,0012
r102	930,81	0,0051	692,35	0,0049
r103	396,08	0,0016	407,06	0,0016
r104	554,31	0,0021	630,82	0,0017
r105	643,39	0,0028	470,26	0,0020
r106	695,72	0,0024	676,18	0,0015
r107	350,71	0,0014	341,83	0,0009
r108	518,71	0,0020	461,46	0,0012
r109	530,78	0,0028	398,87	0,0015

A Tabela 4.3 fornece os resultados do tipo r, nos quais clientes são divididos por agrupamentos. Este tipo de instância se assemelha a característica do nosso problema, mas aqui temos em uma dimensão menor, por se tratar de apenas 10 canteiros.

Pode-se verificar que a heurística HBP encontra 3 soluções, de um total de 9 instâncias, com custo menor do que a heurística RVP que obteve 6 soluções com custo menores. Com isso, pode-se cogitar que para este tipo de instância a heurística HBP torna-se viável em alguns casos.

Tabela 4.4: Resultados para instâncias do tipo *rc* com 10 canteiros de obras

Instâncias	Heurística HBP		Heurística RVP	
	Função Objetivo	Tempo [s]	Função Objetivo	Tempo [s]
rc101	643,68	0,0036	348,52	0,0038
rc102	538,73	0,0031	444,89	0,0017
rc103	566,33	0,0033	260,17	0,0024
rc104	426,35	0,0035	364,86	0,0030
rc105	654,41	0,0046	323,95	0,0031
rc106	691,30	0,0055	456,13	0,0048
rc107	516,07	0,0022	468,38	0,0020
rc108	554,93	0,0036	334,15	0,0032

A Tabela 4.4 fornece os resultados do tipo *rc*, que mesclam características das duas anteriores. Para este conjunto de instâncias pode-se verificar, que em todas as instâncias a heurística RVP obtém os melhores resultados. Verifica-se, novamente, uma proximidade quanto aos tempos de execuções de cada procedimento.

Tabela 4.5: Porcentagens melhores instâncias - 10 canteiros

Grupo	#instâncias	Heurística HBP	Heurística RVP
c	9	0,0	100,0
r	9	33,3	66,7
rc	8	0,0	100,0
Total	26	11,1	88,9

Por fim, com o auxílio da Tabela 4.5 pode-se verificar que para as instâncias de pequeno porte a heurística RVP em aproximadamente 89% foi mais eficiente do que a heurística HBP.

Cogita-se com isso, que pelo fato da heurística HBP possuir a característica de gerar rotas com uma quantidade fixada, inicialmente, de viagens para cada depósito, isso faz com que praticamente para cada depósito seja criada uma rota, o que para este caso demonstra não ser eficiente.

Com a finalidade de verificar a possibilidade de melhora dos resultados da Heurística RVP, tal procedimento foi adaptado recebendo um caráter aleatório. Vale ressaltar que a Heurística HBP já possui em sua estrutura um caráter aleatório, por isso não foi considerada nesta análise.

Para tanto, no algoritmo 3 que representa a Heurística RVP, as linhas de 13 a 20 foram substituídas por:

13 - **escolha** um depósito $l \in L : N_l \neq \emptyset$ **aleatoriamente**

14 - **procure** um vértice $j_l = \arg \min_{v \in N_l : p_l + p_v \leq w_l} \{c_{il,v}\}$

A Tabela 4.6 apresenta a comparação da Heurística RVP modificada, descrita anteriormente, com a Heurística RVP. Cada instância teve 100, 500 e 1000 execuções para a Heurística RVP modificada. Esta Tabela fornece as informações respectivas à melhor função objetivo de cada execução e seu devido tempo total de execução. A melhor função objetivo dentre todas as execuções de cada instância é destacada por um asterisco. O resultado obtido pela Heurística RVP é fornecido nas últimas colunas da Tabela para critério de comparação. Nesta análise foram escolhidas aleatoriamente duas instâncias de cada tipo.

Tabela 4.6: Heurística RVP modificada - 10 canteiros

Instâncias	Heurística RVPm			Heurística RVP	
	Execuções	F.O*	T[s]	F.O	T[s]
c106	100 vezes	436,75	0,65		
	500 vezes	433,25	3.08	158,90	0,003
	1000 vezes	306.75*	6.09		
c107	100 vezes	489.97	1.02		
	500 vezes	387.38	4.97	160,02	0,007
	1000 vezes	383.74*	9.79		
r101	100 vezes	643,57*	0,12		
	500 vezes	665,03	0,99	603,82	0,001
	1000 vezes	648,13	1,96		
r104	100 vezes	736,78	0,29		
	500 vezes	663,76*	1,38	630,82	0,002
	1000 vezes	709,41	2,68		
rc101	100 vezes	510,48	0.71		
	500 vezes	491,55	3,41	348,52	0,004
	1000 vezes	488,45*	6,77		
rc102	100 vezes	528.43	0,37		
	500 vezes	435,97	1,79	444,89	0,002
	1000 vezes	425,89*	3,53		

Analisando o efeito da Heurística RVP modificada, observa-se que em apenas uma instância este método foi capaz de melhorar a função objetivo da Heurística original RVP. Tal situação sugere que este caráter aleatório, para instâncias de pequeno porte demonstra ser ineficiente. O desempenho pior da Heurística HBP que possui caráter aleatório, para com a RVP, como pode ser visto na Tabela 4.5, corrobora com esta análise.

4.2.3 Instâncias de Porte Real - $|C| = 29$

Nesta sub-seção trataremos as instâncias de porte real, as quais em todos os casos possuem 29 canteiros, sendo esta a situação encontrada na prática. Assim, as Tabelas 4.7, 4.8 e 4.9 apresentam os resultados obtidos em cada instância para cada procedimento, bem como seus tempos de execuções. Já a Tabela 4.10 informa para cada conjunto de instâncias qual a porcentagem de resultados melhores cada procedimento obteve.

Tabela 4.7: Resultados para instâncias do tipo c com 29 canteiros de obras

Instâncias	Heurística HBP		Heurística RVP	
	Função Objetivo	Tempo [s]	Função Objetivo	Tempo [s]
c101	923,07	0,2664	861,76	0,1005
c102	1180,64	0,4190	755,77	0,1675
c103	1298,79	0,2686	1021,47	0,0957
c104	792,69	0,1945	432,27	0,0424
c105	1593,78	0,4223	1260,65	0,2308
c106	1202,07	0,3038	1067,83	0,1254
c107	1238,69	0,3786	1099,65	0,1155
c108	829,54	0,1819	730,33	0,0577
c109	763,82	0,2000	619,80	0,0709
c201	1334,35	0,1729	1494,62	0,0559
c202	1320,93	0,1586	1141,20	0,0267
c203	1422,54	0,0949	1033,41	0,0185
c204	1043,21	0,1298	798,37	0,0298
c205	1036,52	0,1360	1124,56	0,0292
c206	1792,58	0,3969	1414,58	0,1327
c207	962,44	0,1552	851,46	0,0385
c208	1164,21	0,1796	923,21	0,0399

A Tabela 4.7 fornece os resultados do tipo C , nos quais os clientes são divididos aleatoriamente em uma região. Nesta situação, verifica-se que em 15 das 17 instâncias a heurística RVP obtém os melhores resultados. Percebe-se com isso que ambas heurísticas não mudam substancialmente para este conjunto de instâncias C com a variação da quantidade de canteiros.

Tabela 4.8: Resultados para instâncias do tipo r com 29 canteiros de obras

Instâncias	Heurística HBP		Heurística RVP	
	Função Objetivo	Tempo [s]	Função Objetivo	Tempo [s]
r101	1222,96	0,1454	1458,90	0,0395
r102	2600,51	0,3624	2745,84	0,1321
r103	1537,28	0,1320	1719,15	0,0227
r104	1496,38	0,2050	1572,74	0,0559
r105	1796,55	0,2058	2102,50	0,0622
r106	1613,63	0,1594	1899,25	0,0353
r107	1367,17	0,2381	1708,55	0,0650
r108	1041,08	0,1522	1198,43	0,0277
r109	1959,94	0,1572	2018,10	0,0444
r110	1581,98	0,1502	1988,70	0,0360
r111	1288,72	0,1438	1590,13	0,0339
r112	1182,95	0,1619	1498,18	0,0293
r201	2048,16	0,3206	2318,30	0,1289
r202	1259,40	0,0902	1208,57	0,0217
r203	1995,21	0,2383	2058,39	0,0603
r204	1598,89	0,1475	1604,31	0,0390
r205	1971,96	0,2175	1823,20	0,0627
r206	1984,11	0,2248	2153,41	0,0871
r207	1449,30	0,2487	1746,66	0,0772
r208	1751,92	0,2213	2047,42	0,0529
r209	2086,73	0,2316	2310,66	0,0751
r210	1143,29	0,1252	1256,99	0,0318
r211	2462,55	0,3985	2717,07	0,1512

A Tabela 4.8 fornece os resultados do tipo r , nos quais clientes são divididos por agrupamentos. Este tipo de instância se assemelha a característica do nosso problema, inclusive na quantidade de canteiro de obras.

Podemos verificar, que a heurística HBP para este caso obtêm em 21 das 23 instâncias menores custos. Podemos dizer, que na situação com características mais próximas da situação real, esta heurística se mostra mais eficiente. Cogitamos neste caso este resultado, pelo fato do algoritmo fazer diversas combinações de rotas e encontrar através das buscas entre todas rotas geradas a melhor.

Tabela 4.9: Resultados para instâncias do tipo *rc* com 29 canteiros de obras

Instâncias	Heurística HBP		Heurística RVP	
	Função Objetivo	Tempo [s]	Função Objetivo	Tempo [s]
rc101	1797,46	0,2030	2062,32	0,0733
rc102	2544,96	0,1765	2001,86	0,0636
rc103	1287,05	0,1790	1581,69	0,0469
rc104	1565,96	0,2302	1642,95	0,0726
rc105	2704,98	0,2232	2127,01	0,0687
rc106	2363,11	0,2731	1761,95	0,0980
rc107	1195,42	0,1470	1495,12	0,0309
rc108	1285,07	0,1995	1654,52	0,0448
rc201	2339,45	0,3612	2568,63	0,1187
rc202	1925,29	0,1451	1826,50	0,0396
rc203	1606,15	0,1419	1587,26	0,0221
rc204	2225,63	0,2262	2059,29	0,0959
rc205	1555,81	0,1692	1755,53	0,0271
rc206	957,42	0,1857	1513,71	0,0427
rc207	2438,95	0,1860	1864,61	0,0837
rc208	1667,20	0,2297	1835,47	0,0326

A Tabela 4.9 fornece os resultados do tipo *rc*, que mesclam características das duas anteriores. Para este conjunto de instâncias pode-se verificar que em 9 de 16 instâncias com custos menores foram obtidas pela heurística HBP. Não obstante, a heurística RVP também se mostrou eficiente para este conjunto de instâncias, onde obteve 7 de 16 instâncias com custos menores.

Tabela 4.10: Porcentagens melhores instâncias - 29 canteiros

Grupo	#instâncias	Heurística HBP	Heurística RVP
c	17	11,7	88,3
r	23	91,3	8,7
rc	16	56,3	43,7
Total	56	53,1	46,9

Através das informações contidas na Tabela 4.10 verifica-se que para a situação real a heurística HBP se mostrou mais eficiente que a heurística RVP em aproximadamente 53% das instâncias testadas.

Cogita-se, como já mencionado anteriormente, que pelo fato da heurística HBP gerar diversas combinações de rotas e posteriormente entre todas as rotas buscar a melhor entre elas, isso faz com que ela encontre na situação real resultados melhores.

Com a finalidade de verificar a possibilidade de melhora dos resultados da Heurística RVP, tal procedimento foi adaptado recebendo um caráter aleatório. Esta modificação foi descrita ao final da subseção anterior. A Tabela 4.11 apresenta a comparação da Heurística RVP modificada com a RVP. Cada instância teve 100, 500 e 1000 execuções para a Heurística RVP modificada.

A Tabela 4.11 fornece as informações respectivas à melhor função objetivo de cada execução e seu devido tempo total de execução. A melhor função objetivo dentre todas as execuções de cada instância é destacada por um asterisco. O resultado obtido pela Heurística RVP é fornecido nas últimas colunas da Tabela, para critério de comparação. Nesta análise foram escolhidas aleatoriamente duas instâncias de cada tipo.

Tabela 4.11: Heurística RVP modificada - 29 canteiros

Instâncias	Heurística RVPm			Heurística RVP	
	Execuções	F.O*	T[s]	F.O	T[s]
c201	100 vezes	1325,75	13,79	1494,62	0,06
	500 vezes	1289,07*	68,59		
	1000 vezes	1331,35	138,43		
c205	100 vezes	1109,12	6,23	1124,56	0,03
	500 vezes	1100,40	30,94		
	1000 vezes	1045,59*	63,09		
r101	100 vezes	1495,93	7,39	1458,90	0,04
	500 vezes	1397,74	36,53		
	1000 vezes	1389,93*	73,64		
r111	100 vezes	1627,64	5,87	1590,13	0,03
	500 vezes	1298,93	28,49		
	1000 vezes	1288,36*	57,88		
rc101	100 vezes	1971,81	15,72	2062,32	0,07
	500 vezes	1754,58	78,00		
	1000 vezes	1710,03*	158,08		
rc107	100 vezes	1260,40	6,12	1495,12	0,03
	500 vezes	1221,26	30,39		
	1000 vezes	1149,11*	60,78		

Analisando os resultados, verifica-se que o método Heurístico RVP modificado foi capaz de encontrar melhores resultados, comparados ao método Heurístico RVP para todas as seis instâncias testadas, em alguns casos bastando 100 execuções. Para os casos de 1000 execuções o tempo máximo de execução foi de 158,08 segundos, tempo este ainda aceitável para um horizonte de planejamento operacional da empresa.

Como análise, sugere-se que o caráter aleatório demonstra ser eficaz para instâncias de grande porte, assim como demonstrado pela Heurística HBP que possui tal característica, como pode ser visto na Tabela 4.10. Tais situações corroboram com esta análise.

Capítulo 5

Considerações Finais

Neste trabalho, heurísticas construtivas foram propostas para atender o problema de Movimentação de Recursos entre Canteiros de Obras - MRCO. Tal problema é considerado como uma generalização do Problema de Roteamento de Veículos, mas que possui múltiplos depósitos, múltiplos produtos, frota homogênea, coleta e entrega simultânea com divisão de entregas e coletas.

Duas heurísticas construtivas foram propostas. A primeira, a Heurística Baseada em Particionamento - HBP, considera a instância do problema original e a partir disso gera para cada veículo, desde que possível, uma rota a partir de um procedimento parcialmente aleatório e posteriormente guloso, caso necessário. A segunda, a Heurística Rota do Vizinho mais Próximo - RVP, transforma cada canteiro de obra e seus respectivos recursos em um grafo onde cada vértice é formado por um par canteiro-recurso. A distância deste grafo é obtida através das distâncias originais entre canteiros e depósitos. Posteriormente este procedimento busca fazer o roteamento através de um procedimento guloso.

Os testes computacionais foram realizados em instâncias aproximadas das reais e em instâncias de testes. Os resultados obtidos demonstram a eficiência das heurísticas propostas em relação à metodologia aplicada pela empresa. Outro ponto a se dizer é que tais procedimentos se mostraram eficientes comparados ao método de Duas Fases, principalmente em relação aos tempos computacionais.

Pode-se dizer que os objetivos deste presente trabalho foram alcançados, dado a complexidade de tal problema e também pelo fato de não ter sido possível encontrar trabalhos, além do proposto por [Dutra \(2015\)](#), que possuam todas as restrições concomitantemente presentes no MRCO, dado o campo de buscas na literatura feitas pelos autores. Tais métodos heurísticos sugerem a aplicabilidade na situação prática encontrada pela empresa por reduzir substancialmente a distância percorrida pelos veículos diariamente.

Desta forma, para continuidade dos estudos sugere-se o desenvolvimento de novas heurísticas construtivas que explorem diferentes características, tais como tratar o presente problema através de uma árvore geradora mínima capacitada; a construção de estruturas de vizinhanças que visem refinar as soluções obtidas; a utilização de metaheurísticas para integrar tais estruturas e por permitir escapar dos ótimos locais, percorrendo assim um espaço de soluções maior.

Referências Bibliográficas

- Andersson, H.; Christiansen, M. e Fagerholt, K. (2011). An optimization-based heuristic for the split delivery vehicle routing problem. *Infor*, 49:79–91.
- Anily, S. e Hassin, R. (1992). The swapping problem. *Networks*, 22:419–433.
- Archetti, C. e Speranza, M. G. (2012). Vehicle routing problems with split deliveries. *International Transactions in Operational Research*, 19:3–22.
- Archetti, C.; Speranza, M. G. e Hertz, A. (2006). A tabu search algorithm for the split delivery. *Transportation Science*, 40:64–73.
- Archetti, C.; Speranza, M. G. e Savelsbergh, M. W. P. (2008). An optimization-based heuristic for the split delivery vehicle routing problem. *Transportation Science*, 42:22–31.
- Benders, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik*, 1:238–252.
- Berbeglia, G. e Cordeau, J. L. G. (2010). Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202:8–15.
- Bianchessi, N. e Righini, G. (2007). Heuristic algorithms for the vehicle routing problem with simultaneous pick-up and delivery. *Computers and Operations Research*, 34:578–594.
- Bordenave, C.; Gendreau, M. e Laporte, G. (2009). A branch-and-cut algorithm for the non-preemptive capacitated swapping problem. *Naval Research Logistics*, 56:478–486.
- Chemla, D.; Meunier, R. e Wolfer-Calvo, R. W. (2013). Bike sharing systems: Solving the static rebalancing problem. *Discrete Optimization*, 10:120–146.
- Chen, Q.; Li, K. e Z.Liu (2014). Model and algorithm for an unpaired pickup and delivery vehicle routing problem with split loads. *Transportation Research Part E: Logistics and Transportation Review*, 69:218–235.

- Coene, S. e Arnout, A. S. F. (2010). The periodic vehicle routing problem: a case study. *Journal of the Operational Research Society*, 61:1719–1728.
- Cordeau, J. F.; Gendreau, M. e Laporte, G. (1997). A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks*, 30:105–119.
- Cormen, T.; Leiserson, C.; Rivest, R. e Stein, C. (2002). *Algoritmos – teoria e prática*. Editora Campus.
- Dantzig, G. B. e Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, 6:80–91.
- Dondo, R. G. e Cerdá, J. (2009). A hybrid local improvement algorithm for large-scale multi-depot vehicle routing problems with time windows. *Computers and Chemical Engineering*, 33:513–530.
- Dutra, M. D. S. (2015). Roteamento de veículos na movimentação de recursos entre canteiros de obras. Dissertação de mestrado, Programa de Pós-Graduação em Engenharia de Produção - UFMG.
- Erdoğan, G.; Cordeau, J. F. e Laporte, G. (2010). A branch-and-cut algorithm for solving the non-preemptive capacitated swapping problem. *Discrete Applied Mathematics*, 158:1599–1614.
- Gajpal, Y. e Abad, P. (2009). An ant colony system (acs) for vehicle routing problem with simultaneous delivery and pickup. *Computers and Operations Research*, 36:3215–23.
- Garey, M. e Johnson, D. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.
- Goksal, F. P.; Karaoglan, I. e Altiparmak, F. (2013). A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. *Computers and Industrial Engineering*, 65:39–53.
- Gulczynski, D.; Golden, B. e Wasil, E. (2011). The multi-depot split delivery vehicle routing problem: An integer programming-based heuristic, new test problems, and computational results. *Computers and Industrial Engineering*, 61:794–804.
- Hernández-Pérez, H.; Rodríguez-Martín, I.; Juan-José e Salazar-González (2016). A hybrid heuristic approach for the multi-commodity pickup-and-delivery traveling salesman problem. *European Journal of Operational Research*, 251:44–52.

- Hernández-Pérez, H. e Salazar-González, J. J. (2009). The multi-commodity one-to-one pickup-and-delivery traveling salesman problem. *European Journal of Operational Research*, 196:987–995.
- Hernández-Pérez, H. e Salazar-González, J. J. (2014). The multi-commodity pickup-and-delivery traveling salesman problem. *Networks*, 63:46–59.
- Laporte, G.; Nobert, Y. e Taillefer, S. (1988). Solving a family of multi-depot vehicle routing and location-routing problem. *Transportation Science*, 22:161–172.
- Mitra, S. (2005). An algorithm for the generalized vehicle routing problem with backhauling. *Asia-Pacific Journal of Operational Research*, 22:153–169.
- Mitra, S. (2008). A parallel clustering technique for the vehicle routing problem with split deliveries and pickups. *Journal of the Operational Research Society*, 59:1532–1546.
- Mladenovic, N. e Hansen, P. (1997). Variable neighborhood search. *Computers and Operations Research*, 24:1097 – 1100.
- Parragh, S. N. e Doerner, K. F. H. R. F. (2008a). A survey on pickup and delivery problems: Part i: Transportation between customers and depot. *Journal für Betriebswirtschaft*, 58:21–51.
- Parragh, S. N. e Doerner, K. F. H. R. F. (2008b). A survey on pickup and delivery problems: Part ii: Transportation between customers and depot. *Journal für Betriebswirtschaft*, 58:81–117.
- Psaraftis, H. N. (2011). A multi-commodity, capacitated pickup and delivery problem: The single and two-vehicle cases. *European Journal of Operational Research*, 215:572–580.
- Renaud, J.; Laporte, G. e Boctor, F. F. (1996). A tabu search heuristic for the multi-depot vehicle routing problem. *Computers and Operations Research*, 23:229–235.
- Resende, M. e Ribeiro, C. (2013). *GRASP: Greedy Randomized Adaptive Search Procedure*. Search Methodologies (E.K. Burke e G. Kendall, editores), Capítulo 11, Springer, 2a edição, 287-312.
- Ropke, S. e Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time. *Transportation Science*, 40:455–472.
- Savelsbergh, M. W. P. e Sol, M. (1995). The general pickup and delivery problem. *Transportation Science*, 29:17–29.

- Subramanian, A.; Drummond, L. M. A.; Bentes, C.; Ochi, L. S. e Farias, R. (2010). A parallel heuristic for the vehicle routing problem with simultaneous pick-up and delivery. *Computers and Operations Research*, 37:1899–1911.
- Talbi, E. (2009). *Metaheuristics: From design to implementation*. New York: Wiley and Sons.
- Thangiah, S. R. e Salhi, S. (2001). Genetic clustering: An adaptive heuristic for the multidepot vehicle routing problem. *Applied Artificial Intelligence*, 15:361–383.