

UNIVERSIDADE FEDERAL DO PARANÁ

JAILSON DOMINGOS DE OLIVEIRA

**ESTRATÉGIAS RELAX-AND-FIX APLICADA AO PROBLEMA DE ROTEAMENTO
EM ARCOS CAPACITADO E PERIÓDICO**

CURITIBA

2017

JAILSON DOMINGOS DE OLIVEIRA

**ESTRATÉGIAS RELAX-AND-FIX APLICADA AO PROBLEMA DE ROTEAMENTO
EM ARCOS CAPACITADO E PERIÓDICO**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Métodos Numéricos em Engenharia, no Curso de Pós-Graduação em Métodos Numérico em Engenharia, Setores de Tecnologia e de Ciências Exatas, da Universidade Federal do Paraná.

Orientador: Prof. Dr. Cassius Tadeu Scarpin

CURITIBA

2017

O48e

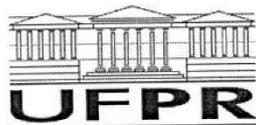
Oliveira, Jailson Domingos de
Estratégias relax-and-fix aplicada ao problema de roteamento em arcos
capacitado e periódico / Jailson Domingos de Oliveira. – Curitiba, 2017.
94 f. : il. color. ; 30 cm.

Dissertação - Universidade Federal do Paraná, Setor de Tecnologia e
Ciências Exatas, Programa de Pós-Graduação em Métodos Numérico em
Engenharia, 2017.

Orientador: Cassius Tadeu Scarpin .
Bibliografia: p. 86-94.

1. Métodos numérico em engenharia. I. Universidade Federal do Paraná.
II. Scarpin, Cassius Tadeu. III. Título.

CDD: 511.8



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
Setor TECNOLOGIA
Programa de Pós-Graduação MÉTODOS NUMÉRICOS EM ENGENHARIA

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em MÉTODOS NUMÉRICOS EM ENGENHARIA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **JAILSON DOMINGOS DE OLIVEIRA** intitulada: **ESTRATÉGIAS RELAX-AND-FIX APLICADA AO PROBLEMA DE ROTEAMENTO EM ARCOS CAPACITADO E PERIÓDICO**, após terem inquirido o aluno e realizado a avaliação do trabalho, são de parecer pela sua APROVAÇÃO.

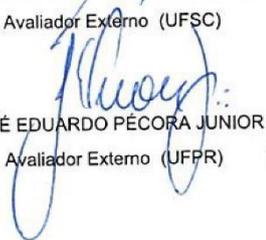
Curitiba, 10 de Fevereiro de 2017.


CASSIUS TADEU SCARPIN

Presidente da Banca Examinadora (UFPR)


LEANDRO MAGATAO
Avaliador Externo (UTFPR)


SERGIO FERNANDO MAYERLE
Avaliador Externo (UFSC)


JOSÉ EDUARDO PÉCORÁ JUNIOR
Avaliador Externo (UFPR)

Dedico este trabalho a todos aqueles que de alguma forma estiveram e estão próximos de mim. Em especial minha mãe Reni J. Domingos de Oliveira.

AGRADECIMENTOS

Diversas são as pessoas que contribuíram para a realização desse trabalho, quero deixar aqui meus agradecimentos:

Ao meu orientador Prof. Dr.Cassius Tadeu Scarpin, por toda atenção, incentivo e confiança disposta em mim.

Ao meu fiel amigo William Ferreira por estar presente em todos os desafios da minha vida.

A todos os membros professores, doutorandos, mestrandos e bolsistas do Grupo de Tecnologia Aplicada à Otimização (GTAO) pela amizade e constante apoio. Meus agradecimentos em especial aos colegas, Nathália Cristina Ortiz da Silva, Cleder Marcos Schenekemberg e Talita Mariana Pinho Schimidt.

À Guilherme Vinicyus Batista pela disponibilidade de material, pela troca de conhecimentos e por estar sempre disposto a me ajudar.

Aos meus familiares, pelo apoio e incentivo em todos os momentos de minha vida.

À CAPES REUNI, pelo apoio financeiro.

Meus sinceros agradecimentos a todos que colaboraram de algum modo, sejam de forma direta ou indireta, para o desenvolvimento desse trabalho.

Há um tempo em que é preciso abandonar as roupas usadas, que já tem a forma do nosso corpo, e esquecer os nossos caminhos, que nos levam sempre aos mesmos lugares. É o tempo da travessia: e, se não ousarmos fazê-la, teremos ficado, para sempre, à margem de nós mesmos.

(FERNANDO PESSOA)

RESUMO

Nesse trabalho, aplicou-se uma estratégia baseada na heurística *relax-and-fix* como método de solução para o Problema de Roteamento em Arcos Capacitado e Periódico (*Periodic Capacitated Arc Routing Problem* - PCARP). Considerou-se o caso especial em que os veículos não têm a necessidade de voltar ao depósito no final de um período e, ainda, têm a possibilidade de folgar em qualquer dia do horizonte de tempo. O PCARP é um problema pouco explorado na literatura. Configura-se como um problema *NP-hard*, sendo comumente aplicado em coleta de resíduos urbano, inspeção de linhas de força, despejo de sal em vias com neve, monitoramento de rodovias, inspeção de ferrovias, irrigação de árvores entre outros. Desenvolveu-se 5 estratégias diferentes para heurística *relax-and-fix* e uma variação denominada *enhanced relax-and-fix* avaliando-se seus desempenhos. Os testes computacionais realizados indicaram que as estratégias propostas para heurística são rápidas na determinação de soluções iniciais para o problema estudado. Destaca-se que das 23 instâncias testadas em nenhum caso se esgotou a memória do computador, fato que ocorre com frequência na tentativa de resolver o problema por métodos exatos.

Palavras-chave: Relax-and-Fix. Problema de Roteamento em Arcos Capacitado e Periódico. Heurística. Relaxation Induced Neighborhood Search.

ABSTRACT

On this research it was applied a strategic solution approach based on the heuristic relax-and-fix for the Periodic Capacitated Arc Routing Problem (PCARP). A special case was considered on which the vehicles do not need to return to a depot when finishing the route. In addition there is the possibility of some vehicles that do not work in any day during the time horizon. The PCARP is not so explored in the literature. It is a NP-Hard Problem, usually applied in urban waste collection, inspection of power lines, winter gritting, road monitoring, inspection of railroads and watering trees. To tackle the problem, it was developed five different strategies for the relax-and-fix heuristic and one variation named enhanced relax-and-fix. All these approaches had their performance evaluate and the computational results show that they are fast to find initial solutions. It is important to highlight that the solver, while running, did not stop by running out of memory, this fact frequently occurs when solving this problem by exact methods.

Key-words: Relax-and-Fix. Periodic Capacitated Arc Routing Problem. Heuristic. Relaxation Induced Neighborhood Search.

LISTA DE FIGURAS

FIGURA 1 - DESENHO DE EULER DAS PONTES DE KÖNIGSBERG	18
FIGURA 2 - TRANSFORMAÇÃO DAS SETE PONTES EM UM GRAFO POR EULER	19
FIGURA 3 - CLASSIFICAÇÃO DO PCARP	30
FIGURA 4 - EXEMPLO PCARPMC	49
FIGURA 5 – METODOLOGIA	52
FIGURA 6 – ESTRATÉGIA R&F-F: SUBPROBLEMA INICIAL	55
FIGURA 7 - ESTRATÉGIA R&F-F: SEGUNDO SUBPROBLEMA	55
FIGURA 8 - ESTRATÉGIA R&F-B: SUBPROBLEMA INICIAL.....	56
FIGURA 9 - ESTRATÉGIA R&F-B: SEGUNDO SUBPROBLEMA.....	57
FIGURA 10 - ESTRATÉGIA R&F-O: SUBPROBLEMA INICIAL	58
FIGURA 11 - ESTRATÉGIA R&F-O: SEGUNDO SUBPROBLEMA.....	58
FIGURA 12- ESTRATÉGIA R&F-O2: SUBPROBLEMA INICIAL	59
FIGURA 13 - ESTRATÉGIA R&F-O2: SEGUNDO SUBPROBLEMA.....	59
FIGURA 14 - GRAFO.....	61
FIGURA 15 – CONFIGURAÇÃO INICIAL R&F-NA.....	61
FIGURA 16 – SEGUNDA ITERAÇÃO R&F-NA.....	62
FIGURA 17 – TERCEIRA ITERAÇÃO R&F-NA	62
FIGURA 18 – CONFIGURAÇÃO DA SOLUÇÃO SUBPROBLEMA 10 DA INSTÂNCIA GDB19	72
FIGURA 19 - NÚMERO DE ITERAÇÕES POR NÓ RAIZ.....	76

LISTA DE TABELAS

TABELA 1- CARACTERÍSTICAS DOS 23 PROBLEMAS	66
TABELA 2 – SOLUÇÕES ENCONTRADAS COM A APLICAÇÃO DAS ESTRATÉGIAS	68
TABELA 3 – TEMPO DE EXECUÇÃO DAS ESTRATÉGIAS <i>RELAX-AND-FIX</i>	69
TABELA 4 – DESVIO E TEMPOS PARA AS MELHORES ESTRATÉGIAS	71
TABELA 5 – RESULTADOS DA HEURÍSTICA <i>ENCHANCED RELAX-AND-FIX</i>	73
TABELA 6 - TEMPO DE EXECUÇÃO DAS ESTRATÉGIAS <i>ENCHANCED RELAX- AND-FIX</i>	75
TABELA 7 – MÉDIA DE ITERAÇÃO POR INSTÂNCIA	77
TABELA 8 - RESULTADOS HEURÍSTICA RINS	78
TABELA 9 - RESULTADOS E-R&F-NA APLICADO AO MODELO 2014.....	80

LISTA DE QUADROS

QUADRO 1 - RESUMO DA PROPOSTA TAXONÔMICA.....	35
QUADRO 2 - ABORDAGENS DE RESOLUÇÃO.....	37
QUADRO 3- ALGORITMO <i>RELAX-AND-FIX</i>	41
QUADRO 4 - ALGORITMO RINS	45
QUADRO 5 - ALGORITMO <i>ENHANCED RELAX-AND-FIX</i>	63

LISTA DE ABREVIATURAS E SIGLAS

ACO	- <i>Ant Colony Optimization</i>
ALNS	- <i>Adaptive Large Neighborhood Search</i>
AM	- Algoritmo Memético
ARP	- <i>Arc Routing Problems</i>
B&B	- <i>Branch-and-bound</i>
B&C	- <i>Branch-and-Cut</i>
BIH	- <i>Best Insertion Heuristic</i>
CARP	- Problema de Roteamento em Arcos Capacitado
DFNH	- <i>Decreasing Frequencies Nearest Insertion Heuristic</i>
EFR	- <i>Enhanced Fix-and-Relax</i>
IIBC	- <i>Insertion In Best Combination</i>
KBP	- <i>The Konigsber Bridges Problem</i>
K-PCC	- Problema dos K-Carteiros Chines
LBH	- <i>Lower Bound Heuristic</i>
MIP	- <i>Problema Inteiro Misto</i>
NDLB	- <i>Node Duplication Lower Bound</i>
PARPRP	- <i>Periodic Arc Routing With Refill Points</i>
PCARP	- Problema de Roteamento em Arcos Capacitado e Periódico
PCARPMC	- Problema de Roteamento Periódico em Arcos Capacitado com Movimento Contínuos
PCC	- Problema do Carteiro Chinês
PCCC	- Problema do Carteiro Chinês Capacitado
PCCD	- Problema do Carteiro Chinês Direcionado
PCCH	- Problema do Carteiro Chinês Não Hierárquico
PCCM	- Problema do Carteiro Chinês Misto
PCCND	- Problema do Carteiro Chinês Não Direcionado
PCCV	- Problema do Carteiro Chinês com Vento
PCR	- Problema do Carteiro Rural
PCRD	- Problema do Carteiro Direcionado
PCRM	- Problema do Carteiro Rural Misto
PCRMCP	- Problema do Carteiro Rural Misto com Conversões Penalizadas
PCRND	- Problema do Carteiro Rural Não Direcionado

PCRV	- Problema do Carteiro Rural com Vento
PI	- Programação Inteira
PMSC	- <i>Minimum Stacker Crane</i>
PRPV	- Problema de roteamento Periódico de Veículos
PRV	- Problema de Roteamento de Veículos
R&F	- <i>Relax-and-Fix</i>
R&F-B	- <i>Relax-and-Fix Backward</i>
R&F-F	- <i>Relax-and-Fix Forward</i>
RF-FPSP	- <i>Relax-and-Fix Forward</i> Orientado a Período com <i>Pool</i>
R&F-NA	- <i>Relax-and-Fix</i> Nós Adjacentes
R&F-O	- <i>Relax-and-Fix Overlapping</i>
R&F-O2	- <i>Relax-and-Fix Overlapping 2</i>
RM	- <i>Route-Merging</i>
RINS	- <i>Relaxation Induced Neighborhood Search</i>
SS	- <i>Scatter Search</i>

SUMÁRIO

1	INTRODUÇÃO	13
1.1	OBJETIVOS	15
1.1.1	Objetivo Geral.....	15
1.1.2	Objetivos Específicos	15
1.2	JUSTIFICATIVA.....	16
1.3	CONTRIBUIÇÕES E DELIMITAÇÕES.....	16
1.4	ESTRUTURA DO TRABALHO	17
2	REVISÃO DE LITERATURA.....	18
2.1	GRAFOS	18
2.2	PROBLEMA DE ROTEAMENTO DE VEÍCULOS.....	20
2.3	PROBLEMA DE ROTEAMENTO EM ARCOS	21
2.3.1	Problema do Carteiro Chinês	22
2.3.2	Problema do Carteiro Rural	24
2.3.3	Problema de Roteamento em Arcos Capacitado.....	25
2.3.4	Problema de Roteamento em Arcos Capacitado e Periódico.....	28
2.4	MÉTODOS DE SOLUÇÃO	38
2.4.1	Heurística <i>relax-and-fix</i>	38
2.4.2	Heurística <i>Relaxation Induced Neighborhood Search</i>	44
3	MODELO MATEMÁTICO.....	46
3.1	APLICAÇÃO	46
3.2	FORMULAÇÃO MATEMÁTICA.....	47
3.2.1	Modelo Batista (2014).....	47
3.2.2	Modelo Batista (2016).....	49
4	METODOLOGIA.....	52
4.1	HEURISTICA <i>RELAX-AND-FIX</i>	53
4.1.1	Decomposição por período.....	53
4.1.1.1	Estratégia <i>relax-and-fix forward</i> – R&F-F	54
4.1.1.2	Estratégia <i>relax-and-fix backward</i> – R&F-B	56
4.1.1.3	Estratégia <i>relax-and-fix overlapping</i> – R&F-O.....	57
4.1.1.4	Estratégia <i>relax-and-fix overlapping 2</i> – R&F-O2.....	58
4.1.2	Decomposição por nós adjacentes: estratégia <i>relax-and-fix</i> nós adjacentes – R&F-NA.....	60

4.2	HEURÍSTICA <i>ENHANCED RELAX-AND-FIX</i>	63
4.3	HEURÍSTICA RINS	64
5	TESTES COMPUTACIONAIS	65
5.1	CRITÉRIO DE PARADA.....	65
5.2	INSTÂNCIAS	66
5.3	<i>RELAX-AND-FIX</i> CLÁSSICO APLICADO AO MODELO BATISTA (2016) ..	67
5.4	<i>ENHANCED RELAX-AND-FIX</i> APLICADO AO MODELO BATISTA (2016) 73	
5.5	TESTES HEURÍSTICA RINS	78
5.6	TESTE APLICADO AO MODELO BATISTA (2014)	79
6	CONCLUSÕES E TRABALHOS FUTUROS	82
	REFERÊNCIAS	86

1 INTRODUÇÃO

Os grandes avanços tecnológicos, crescimento populacional e o aumento da oferta e demanda de produtos conduzem as empresas de transportes a revisar constantemente seus modelos de distribuição, pois estes fatores tornam cada vez mais complexas as redes de distribuição de bens e serviços. Deste modo, os processos logísticos são importantes nas operações gerenciais, fazendo com que essa área de atuação se consolide como elemento chave nas estratégias competitivas das empresas (VIEIRA, 2013).

A logística pode ser entendida como o processo que compreende o gerenciamento do fluxo de produtos, desde os pontos de fornecimento até os pontos de consumo, satisfazendo a demanda dos clientes de maneira eficiente e com menor custo.

Segundo a Fundação Dom Cabral¹ os resultados da pesquisa “Custo Logístico no Brasil em 2015”, mostram que 11,73% da receita das empresas pesquisadas é consumida pelo custo logístico. Estima-se que os custos de distribuição somam quase a metade dos custos totais, podendo chegar, em alguns casos, em 70% desse valor, como no caso das indústrias de alimentos e bebidas (PEREIRA, 2010). Ballou (2006) destaca que o custo com transporte é a atividade que absorve a maior percentagem dos custos logísticos, ficando normalmente entre um e dois terços do custo total.

A procura por minimização desses custos acarretou no aumento de pesquisas e artigos publicados. Segundo Doulabi e Seife (2013), uma parte dos pesquisadores têm estudado três grandes problemas logísticos: gerenciamento de armazéns, localização de facilidades e roteamento de veículos. Nesse último, em especial, a literatura atual sugere dois objetivos explorados: a redução do número de veículos utilizados e/ou a redução da distância total percorrida.

O termo roteirização surge como forma equivalente do inglês “*Routing*” e é utilizado para determinar uma ou mais rotas ou sequências de paradas a serem realizadas por um conjunto de veículos, com o objetivo de visitar um conjunto de

¹<http://www.fdc.org.br/blogespacodialogo/Lists/Postagens/Post.aspx?List=95696fb1-15d4-444c-9a1e-506231d17962&ID=482&Web=e06f7d1a-c7ed-49ae-95f1-a5c1408f0875>

pontos ou clientes que necessitam de atendimento, sujeitos a restrições operacionais adicionais (CUNHA, 2000).

Problemas de roteamento de veículos podem ser classificados em duas classes: Problema de Roteamento em Nós e Problemas de Roteamento em Arcos. O primeiro consiste em determinar uma ou mais rotas que passe(m) por todos os nós ou vértices de um grafo, enquanto o roteamento em arcos tem por objetivo determinar uma ou mais rotas que atenda(m) todos os arcos ou arestas de um grafo (MONROY; AMAYA; LANGEVIN, 2013).

Um dos primeiros problemas de roteirização em arcos a ser estudado foi o Problema do Carteiro Chinês – PCC (*Chinese Postman Problem*) proposto por Mei Gu Guan em 1962, que consiste em encontrar o roteiro ou sequência de ruas a serem percorridas, no mínimo uma vez cada rua, por um carteiro de forma a minimizar a distância total percorrida. Desde então, novas restrições vêm sendo incorporadas ao problema, como: um conjunto de arestas que devem ser obrigatoriamente percorrido é um subconjunto das arestas do grafo, um conjunto de k carteiros ($k > 1$) trabalhando em conjunto entre outras, de modo a modelar as diferentes aplicações reais que requerem roteirização dando origem, assim, há diversos novos problemas, a saber, por exemplo: o Problema do K-Carterio Chinês (GOLDBARG, 2005), Problema do Carteiro Chinês com Vento (MINIEKA, 1979), Problema do Carteiro Chinês Hierárquico (DROR *et al.*, 1987) entre outros.

Seguindo esse processo de evolução, destaca-se o Problema de Roteamento em Arcos Capacitado e Periódico (*Periodic Capacitated Arc Routing Problem - PCARP*) uma extensão do Problema de Roteamento em Arcos Capacitado - CARP. Nesse problema, ao invés de ser abordado para a tomada de decisões apenas de nível operacional, tem-se decisões de níveis tático e operacionais, para horizontes de tempo maiores do que um período e com restrições de frequência (LACOMME; PRINS; RAMDANE-CHÉRIF, 2002a).

Lacome, Prins e Ramdame-Chérif (2005) afirmam que pesquisas sobre o PCARP são ainda embrionárias e muito recentes. Todavia, o interesse e a demanda pela aplicação de modelos de roteirização em arcos para problemas reais fazem com que pesquisas sobre o PCARP tenham aumentado gradativamente na última década. O PCARP tem sido amplamente aplicado em: coleta de resíduos urbanos (Lacome, Prins & Ramdame-Chérif, 2002a, 2005; Chu, Labadi & Prins, 2006), monitoramento de redes rodoviárias (Moroy, Amaya e Langevin, 2013; Marzolf,

Trépanier e Langevin, 2006), supressão de poeira em estradas (Riquelme-Rodríguez, Langevin e Gamache, 2014, 2016, não publicado), manutenção preditiva e preventiva de linhas férreas (Batista, 2014; Batista e Scarpin, 2015), irrigação de árvores (Huang e Lin, 2014) entre outras.

O CARP é considerado *NP-hard* por envolver restrições operacionais que limitam a capacidade de atendimento. Sua resolução de maneira exata é considerada difícil (GOLDEN; WONG, 1981). Portanto, sob a ótica da otimização, os problemas de roteamento em arcos capacitado e periódico pertencem à categoria *NP-hard*, pois inclui o CARP como caso particular. Consequentemente, métodos exatos e o uso de *softwares* comerciais para a resolução do PCARP tornam-se, em muitos casos, não operacionais para problemas reais. Problemas classificados como *NP-hard* encontram nos métodos heurísticos uma boa estratégia de solução, pois as heurísticas são capazes de determinar soluções aceitáveis em tempo computacional curto, na qual as características do problema são exploradas de forma direcionada, fornecendo uma solução viável (CUNHA, 1997).

Neste contexto, considerou-se o problema de roteamento em arcos, mais especificamente, o Problema de Roteamento em Arcos Capacitado e Periódico, baseando-se nas formulações matemáticas propostas por Batista (2014, 2016, não publicado). Como abordagem de solução foi proposta uma heurística do tipo *relax-and-fix* (R&F), descrita em Wolsey (1998), que decompõe o modelo original em submodelos que podem ser resolvidos de modo exato sem, entretanto, garantir a solução ótima do problema original.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Propor o uso da heurística *relax-and-fix* como método de resolução para o Problema de Roteamento Periódico em Arcos Capacitados (PCARP), com múltiplos veículos de capacidade restrita e restrições de periodicidades no atendimento dos arcos.

1.1.2 Objetivos Específicos

O objetivo geral desdobra-se nos seguintes objetivos específicos:

- Implementar a heurística *relax-and-fix*;
- Propor 5 estratégias de decomposição para a heurística *relax-and-fix*; Destas, 4 são baseadas nas estratégias clássicas e 1 estratégia inédita desenvolvida;
- Realizar testes com 23 instâncias para o PCARP propostas por Batista (2014);
- Aplicar a heurística *Relaxation Induced Neighborhood Search* – RINS como método de melhoria da solução encontrada pelo *relax-and-fix*.

1.2 JUSTIFICATIVA

Especificamente no caso do PCARP, ainda que verificado aumento na última década, essas pesquisas são consideradas embrionárias quando comparadas com outros problemas como, por exemplo, o problema de roteamento em nós.

A complexidade matemática que envolve o PCARP justifica o constante interesse em busca de novas estratégias de solução para o problema. Na maioria das vezes essas estratégias são métodos heurísticos.

Apesar dos métodos heurísticos não assegurarem a obtenção da solução ótima, esses têm demonstrados bons resultados de acordo com literatura. Segundo Chu, Labadi e Prins (2005) uma solução exata com o uso de software comercial é impraticável para problemas reais.

Como cada variante do problema PCARP apresenta suas particularidades, então, justifica-se que métodos heurísticos sejam desenvolvidos e adaptados especificamente para cada tipo de problema. Nesse sentido, o trabalho desenvolvido por Batista (2014), apresenta limitações devido a carência de soluções de maior qualidade, com tempo de solução razoável. Dessa forma, as proposições heurísticas podem contribuir para suprir estas necessidades.

1.3 CONTRIBUIÇÕES E DELIMITAÇÕES

As principais contribuições do trabalho são:

- A adaptação da heurística *relax-and-fix* como método de solução para o PCARP;

- O desenvolvimento de uma nova estratégia de decomposição baseado em adjacência em nós para a heurística;
- Uma forma de realizar a exploração dessa nova estratégia desenvolvida;
- A abordagem heurística desenvolvida para o modelo é capaz de determinar boas soluções (não, necessariamente, a ótima) em um tempo de processamento menor do que o tempo de processamento da solução exata.

O trabalho apresenta as seguintes delimitações:

- Os modelos estudados são aplicáveis em redes representadas por grafos não orientados;
- Não é desenvolvido um estudo a fim de determinar limites inferiores para os modelos;
- Não foi desenvolvida uma heurística de melhoria para ser aplicada após a heurística *relax-and-fix*.

1.4 ESTRUTURA DO TRABALHO

Estruturalmente, o trabalho está dividido em seis capítulos. O capítulo dois é composto por revisão bibliográfica e fundamentação teórica, onde são abordados temas e/ou definições importantes e onde são analisados os estudos de problemas e estratégias semelhantes.

No capítulo três o PCARP é descrito com maior detalhamento, apresentando os modelos matemáticos e suas formulações. No capítulo quatro são apresentadas as abordagens heurísticas de solução. No capítulo cinco são descritos e discutidos os resultados obtidos e as discussões sobre o problema, heurísticas e seus desempenhos. E, por fim, no capítulo seis são apresentadas as conclusões e sugestões de trabalhos futuros.

2 REVISÃO DE LITERATURA

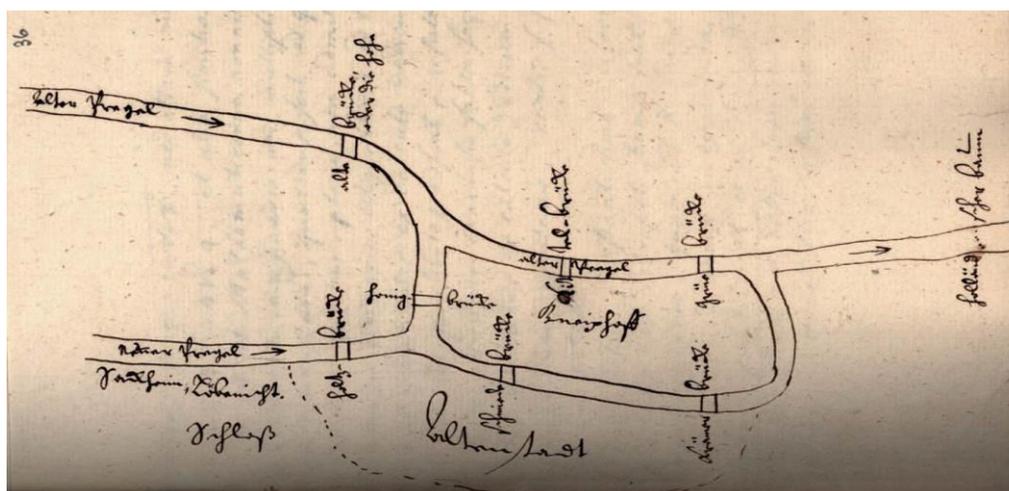
2.1 GRAFOS

Ao contrário de diversos ramos da matemática, que possuem suas origens puramente teóricas, a teoria dos grafos tem sua origem em problemas práticos relacionados a diversas áreas e na constatação da existência de estruturas e propriedades comuns a esses problemas (BOAVENTURA NETTO; OSWALDO, 1979).

O primeiro problema cuja solução exigiu conceitos, que posteriormente vieram a ser denominado de teoria dos grafos, foi o problema das pontes de Königsberg. Königsberg é uma cidade que pertencia à Alemanha, e, hoje, pertence à Rússia; chamada atualmente de Kaliningrad; esse problema foi resolvido e publicado pelo famoso matemático suíço Euler, em 1736 (ARENALES *et al.*, 2007).

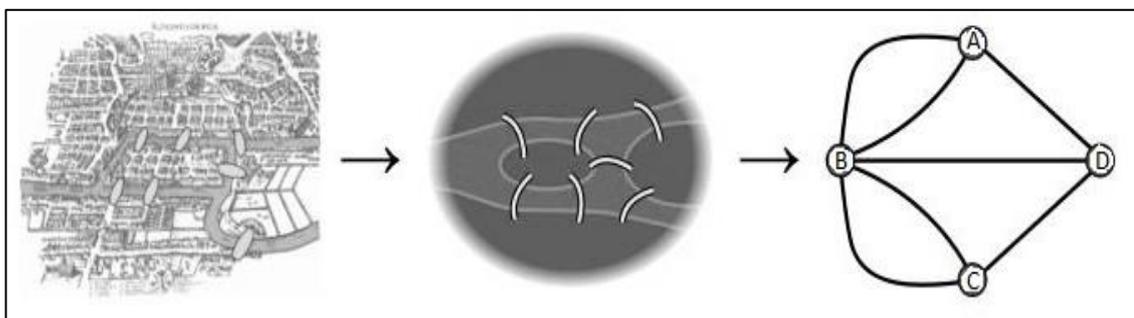
O problema das pontes de Königsberg (The Königsberg Bridges Problem – KBP) consiste em: É possível que um pedestre caminhe em todas as sete pontes de Königsberg sem cruzar nenhuma ponte duas vezes? (GRÖTSCHEL; YUAN, 2012). A FIGURA 1 é uma ilustração realizada por Euler, e a FIGURA 2, o grafo associado ao problema.

FIGURA 1 - DESENHO DE EULER DAS PONTES DE KÖNIGSBERG



FONTE: Grötschel e Yaun (2012)

FIGURA 2 - TRANSFORMAÇÃO DAS SETE PONTES EM UM GRAFO POR EULER



FONTE: Negreiros *et al.* (2009), modificado pelo autor.

Euler mostrou que, dada a disposição das pontes, era impossível traçar um caminho sem percorrer mais de uma vez a mesma ponte (ANDRADE, 1980).

Segundo Grötschel e Yaun (2012), Euler introduz os símbolos $a, b, c...$ para as pontes (arestas) e os símbolos $A, B, C...$ para as áreas de Königsberg ligadas por pontes (nós ou vértices do grafo). Também denotou uma aresta por um par de nós, tais como $a = AB$ introduz a notação de um caminho para ligar os nós A e C por meio da sequência de arestas AB e BC . Euler denominou o grau de um vértice (nó) de um grafo, definindo como o número de caminhos (arestas) que tocam o vértice (ANDRADE, 1980). Os termos nós, vértices e arestas não existiam ainda nesse período, mas observa-se que as notações para a teoria dos grafos não mudaram muito desde 1736.

Após Euler ter demonstrado que o problema das pontes era impossível de ser resolvido, ele descreve por meio de um teorema as condições necessárias para realizar um caminho fechado em um grafo onde cada aresta seja atravessada uma única vez, denominado posteriormente como ciclo Euleriano (ANDRADE, 1980).

Teorema: um grafo conexo possui um ciclo euleriano, se e somente se, todos os vértices do grafo possuem grau par.

A forma como se desenvolveu a teoria dos grafos ao longo dos anos gerou diversos problemas de nomenclaturas e de notações, as quais diferem de um autor para outro; logo, a criação de um único sistema se tornou difícil, pois o mesmo símbolo pode representar conceitos diferentes, quando utilizado por diferentes autores. Nesse trabalho, optou-se pelas nomenclaturas e definições utilizadas em Arenales *et al.* (2007) e Boaventura Netto (2001).

O par $G = (X, E)$ é denominado grafo, onde X é um conjunto finito, cujos elementos são chamados de nós (ou vértices) e E um conjunto de pares de nós, cujos elementos (i, j) são chamados arestas ou arcos (ARENALES *et al.*, 2007).

Em um grafo $G = (X, E)$ tem-se o conjunto $X = \{x_1, x_2, \dots, x_n\}$ formado por n nós e o conjunto $E = \{e_1, e_2, \dots, e_m\}$ formado por m arestas ou arcos, significa que $|X| = n$ e $|E| = m$, isto é, a cardinalidade de X é n e a cardinalidade de E é m .

Um grafo $G = (X, E)$ pode ser representado graficamente da seguinte forma: cada nó/vértice x_i é representado por um círculo, e cada aresta/arco e por uma linha, conectando dois nós x_i e x_j . Arcos ou arestas são associados a um par de vértices representados por x_{ij} quando formados pelos pontos (x_i, x_j) ou simplesmente (i, j) . O que difere arcos e arestas é a ordenação dos nós que o compõem.

Quando se trata de uma aresta, a ordem dos nós, na notação (x_i, x_j) , é irrelevante, isto é, $(x_i, x_j) = (x_j, x_i)$. Porém, no caso de um arco, a ordem é do nó inicial para o final (direcionado), pois em alguns casos a ordem inversa pode não existir ou ter custos associados diferentes; normalmente um grafo direcionado é representado por $G = (X, A)$. Um grafo no qual os arcos são pares ordenados (subconjunto de $N \times N$) é denominado grafo orientado ou dígrafo (ARENALES *et al.*, 2007).

Para Boaventura Netto (1979), a teoria dos grafos trata basicamente das relações existentes entre elementos de um ou mais conjuntos; a teoria pode tratar de relações não quantificáveis e suas premissas são pouco exigentes. O autor destaca que a estrutura de grafo tem uma definição muito mais simples do que, por exemplo, a de espaço vetorial. E sua abrangência é muito maior, seja em química orgânica, eletricidade, organização, transporte, etc. Um conjunto de modelos e aplicações detalhadas da teoria de grafos é descrito no trabalho de Gross e Yellen (2006).

2.2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS

Para Franco e Oliveira (2012), entende-se roteamento de veículos como um conjunto de problemas que tem por objetivo determinar as melhores rotas para uma frota de veículos atender um conjunto de consumidores (arcos ou nós). Algumas de suas aplicações são: distribuições de jornais, alimentos, bebidas, produtos diversos,

transporte escolar, recolhimento de lixo, entrega de correspondência, serviço de emergência, entre outros (GOLDBARG, 2005).

Para Goldberg (2005), os problemas de roteamento de veículos estão entre os mais complexos da área de otimização combinatória. Isto se deve pelo grande número de variáveis, diversidade de restrição e objetivos. Segundo Cunha (2000), os problemas de roteirização de veículos pertencem à categoria de problemas *NP-hard*.

Uma cuidadosa taxonomia pode ser encontrada em Goldberg (2005), em que os Problemas de Roteamento de Veículos são classificados segundo diversos critérios, como: número de domicílios (um domicílio ou mais de um), tamanho/tipo da frota de veículos, natureza da demanda, localização da demanda, grafo, restrição na capacidade do veículo, tempo de roteamento, custos, operação, objetivo e restrição na capacidade do arco.

Segundo Goldberg (2005), problemas de roteamento podem ser classificados em: problema de roteamento em nós (associado aos ciclos Hamiltonianos) e problemas de roteamento em arcos (associados aos ciclos Eulerianos).

Problemas de roteamento em nós consistem em determinar uma ou mais rotas que passem por todos os nós ou vértices de um grafo, enquanto que o roteamento em arcos tem por objetivo determinar uma ou mais rotas que atendam todos os arcos ou arestas de um grafo (MONROY; AMAYA; LANGEVIN, 2013).

2.3 PROBLEMA DE ROTEAMENTO EM ARCOS

O Problema de Roteamento em Arcos (*Arc Routing Problems* - ARP) é um dos mais antigos problemas relacionados a grafos. A primeira referência do ARP vem do problema das pontes de Königsberg de 1736, descrito anteriormente, mas o termo percurso em arco foi estabelecido mais claramente por Eisel *et al.* em 1995 (NEGREIROS *et al.*, 2009).

O problema de roteamento em arcos refere-se aos problemas de roteamento onde a atividade principal é cobrir os arcos e suas demandas. Enquanto que nos problemas de roteamento em nós os arcos são de interesse apenas como caminhos que conectam os nós (EISEL *et al.*, 1995). Segundo Eisel *et al.* (1995a), o principal objetivo dos ARP's é determinar o menor custo para atravessar um conjunto de arcos de um grafo, com ou sem restrições.

De acordo com Corberán e Prins (2010), os ARPs têm sido menos estudados do que os problemas de roteamento em nós, mas têm ganhado destaque nas últimas décadas, e desenvolvimentos impressionantes foram alcançados. Existe um grande número de situações reais que podem ser modelados por esse tipo de problema, tais como: a coleta de lixo urbano, a inspeção de linhas de forças, o despejo de sal em vias com neves, entrega de correspondências, entre outros (CORBERÁN; PRINS, 2010).

Diversos são os problemas da literatura que se encaixam nessa classificação de ARPs, entre esses destacam-se: Problema do Carteiro Chinês, o Problema do Carteiro Rural, Problema de Roteamento em Arcos Capacitado e o Problema de Roteamento em Arcos Capacitado e Periódico.

2.3.1 Problema do Carteiro Chinês

O Problema do Carteiro Chinês - PCC (*Chinese Postman Problem - CPP*) foi proposto pelo matemático chinês Mei Gu Guan em 1962. Guan definiu esse problema da seguinte maneira: Um carteiro tem que entregar correspondências para um determinado bairro. Ele precisa caminhar através de todas as ruas do bairro e retornar ao posto de correio. Como ele poderia traçar seu percurso, de modo que ele ande a mínima distância possível? (GRÖTSCHEL; YUAN, 2012).

Eisel *et al.* (1995a) divide o Problema do Carteiro Chinês em três subproblemas, sendo:

1. Problema do Carteiro Chinês Não Direcionado - PCCND trata-se do caso do PCC com arestas não direcionadas;
2. Problema do Carteiro Chinês Direcionado - PCCD trata-se do caso do PCC com arestas direcionadas;
3. Problema do Carteiro Chinês Misto - PCCM trata-se do caso do PCC com arestas direcionadas e não direcionadas.

Para descrever melhor cada caso pode-se pensar na distribuição de correspondências em uma cidade que possui somente ruas de mão dupla para um PCCND, uma cidade que contém apenas ruas de uma mão única PCCD e uma cidade que possui ruas de mão duplas e mão única PCCM.

Em Araújo (2003), podemos encontrar a seguinte formulação matemática para PCCND, partindo de um grafo $G = (X, A)$:

$$\min z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (1)$$

Sujeito a:

$$\sum_{j=1}^n x_{ji} - \sum_{j=1}^n x_{ij} = 0 \quad i = 1, \dots, n. \quad (2)$$

$$x_{ij} + x_{ji} \geq 1 \quad \forall (i, j) \in A \quad (3)$$

$$x_{ij} \geq 0 \text{ e inteiro} \quad (4)$$

Onde:

x_{ij} representam o número de vezes que a aresta (i, j) é percorrida;

c_{ij} representam o coeficiente de custo associado ao atravessar a aresta (i, j) .

No modelo matemático acima, a equação (1) descreve a função objetivo, que consiste em minimizar os custos associados à travessia das arestas. A restrição (2) garante a continuidade do percurso, a restrição (3) é a garantia de que todas as arestas serão percorridas. A substituição da restrição (3) por $x_{ij} \geq 1 \forall (i, j) \in A$, transforma o modelo PCCND em um PCCD.

O PCCM é aplicado em um grafo misto, contém tanto arcos orientados como arcos não orientados. O grafo deve ser fortemente conexo, ter todos os seus nós com grau par e simétricos, para que o PCC tenha solução trivial.

Além dos três subproblemas citados acima, outras variantes do PCC são referidas na literatura, como: o Problema dos K-Carteiros Chinês (k-PCC), o Problema k-Carteiros Chinês Min-Max (MM-k-PCC) (GOLDBARG, 2005), o Problema do Carteiro Chinês com Vento (*Windy Postman* - PCCV) (MINIEKA, 1979) e o Problema do Carteiro Chinês Hierárquico (PCCH) (DROR *et al.*, 1987).

O k-PCC, trata-se do caso com K carteiros ($k > 1$) trabalhando em conjunto, com o objetivo de minimizar o comprimento dos k ciclos formados, o problema MM-k-PCC trata do caso do K-Carteiro Chinês encontrar K rotas em que a maior aresta é minimizada (GOLDBARG, 2005).

O PCCV foi introduzido inicialmente por Minieka (1979). Trata-se do caso do PCC em que a matriz de custos de G é assimétrica. O sentido de percurso mais caro é denominado: “contra o vento”; e, o mais barato: “a favor do vento”.

Esta variação tem ganhado atenção de alguns pesquisadores, porque os três casos PCCND, PCCD e PCCM podem ser derivados dessa variação. Por exemplo, o

PCCM pode ser formulado como PCCV; para isso, basta fixar custos iguais para as arestas, em ambos os sentidos, e para os arcos, considerar o custo normal no sentido da orientação do arco, e custo igual a $+\infty$ no sentido oposto (KONOWALENKO, 2012).

O PCCH foi introduzido por Dror *et al.* (1987). Nessa variante, o carteiro chinês faz sua rota dentro de grupamentos de arcos, obedecendo à relação de procedência entre eles. Na prática, o PCCH descreve situações em que o atendimento de alguns arcos deve ser priorizado (ARAUJO, 2003). O modelo possui aplicações na limpeza de neve (GHIANI; IMPROTA, 2000).

Outra variante do PCC é o Problema do Carteiro Chinês com Janelas de Tempo. Nesta variante, são adicionadas restrições que interferem em quão cedo e quão tarde um arco será atendido. A partir do Problema do Carteiro Chinês, as pesquisas crescem, em geral, motivadas em aplicações reais (BATISTA, 2014).

2.3.2 Problema do Carteiro Rural

O Problema do Carteiro Rural (PCR), descrito por Orloff (1974), é uma variante do PCC, em que o conjunto de arestas que deve ser obrigatoriamente percorrido é um subconjunto das arestas de um grafo G . O problema tem por objetivo descobrir o caminho de menor custo em G que percorra o subconjunto de arestas.

Eiselt *et al.* (1995b) descrevem algumas aplicações para PCR, por exemplo: casos em que as ruas ou estradas têm de ser percorrida para manutenção e/ou limpeza, entrega de correspondência, coleta de lixo, rota de ônibus escolares, entrega de leite, leitura de medidor de eletricidades e inspeção de linhas elétricas e de redes de gás. O autor apresenta duas variações do PCR. O Problema do Carteiro Rural não Direcionado (PCRND) e o Problema do Carteiro Rural Direcionado (PCRD). Além disso, apresenta heurísticas para solução de tais problemas.

Outras variações do PCR são apresentadas por SHERAFAT (2004):

- Problema do Carteiro Rural Misto - PCRM: é o caso mais genérico do PCR, as variações PCRND e PCRD assim como as varrições do Problema do Carteiro Chinês podem ser formulados como caso particular do PRM;

- Problema do Carteiro Rural Misto com Conversões Penalizadas - PCRMCP: incorpora restrições de conversões nos vértices, com essa restrição é possível levar em conta conversões proibidas no trânsito urbano;

- Problema do Carteiro Rural com Vento - PCRV: as arestas, requeridas ou não, podem ser percorridas em ambos os sentidos com custos diferentes.

O PCR é um problema *NP-hard*, possui uma formulação mais genérica e realista para os problemas de distribuição e tem ganhado destaque pela importância para os problemas reais de roteamento (SHERAFAT, 2004).

2.3.3 Problema de Roteamento em Arcos Capacitado

O Problema de Roteamento em Arcos Capacitado (*Capacitated Arc Routing Problem* – CARP) foi proposto por Golden e Wong (1981) e envolve a capacidade dos veículos. Essas capacidades, muitas vezes, podem estar associadas à carga máxima, a um tempo máximo de trabalho ou, também, a uma distância percorrida. Os ARPs capacitados são mais difíceis de serem resolvidos e seu estudo é mais recente (CORBERÁN; PRINS, 2010).

Grande parte dos trabalhos sobre CARP considera um grafo não direcionado $G = (V, E)$, onde os nós de V incluem um depósito v_0 , com k veículos idênticos de capacidades W . Um subconjunto R , de arestas necessárias de E , que devem ser servidos por um veículo, em que cada aresta possui uma demanda q_e . O objetivo é determinar um conjunto de rotas de custo mínimo, onde as arestas com demandas devem ser percorridas ao menos uma vez.

Dror (2000) propõe a seguinte formulação para o CARP, partindo de um grafo $G = (V, A)$, com um subconjunto R de arcos requeridos de A :

$$\min z = \sum_{(i,j) \in A} \sum_{k=1}^k C_{ij} x_{ijk} \quad (5)$$

Sujeito à:

$$\sum_{p \in A} x_{pik} = \sum_{p \in A} x_{ipk} \quad \forall i \in V, \forall k = 1, 2, \dots, k \quad (6)$$

$$\sum_{k=1}^k y_{ijk} = 1 \quad \forall (i, j) \in R \quad (7)$$

$$x_{ijk} \geq y_{ijk} \quad \forall (i, j) \in R, k = 1, 2, \dots, k \quad (8)$$

$$\sum_{(i,j) \in R} q_{ij} y_{ijk} \leq W \quad \forall k = 1, 2, \dots, K \quad (9)$$

$$M \sum_{i \notin S, j \in S} x_{ijk} \geq \sum_{(j,p) \in A[S] \cap R} x_{jpk} \quad \forall S \subseteq N, 1 \notin S, A[S] \cap R \neq \emptyset, k = 1, 2, \dots, k \quad (10)$$

$$y_{ijk} \in \{0, 1\}, \forall (i, j) \in R, k = 1, 2, \dots, k \quad (11)$$

$$x_{ijk} \in Z^+, \forall (i, j) \in A, k = 1, 2, \dots, k \quad (12)$$

Onde:

K , número de veículos;

W , capacidade do veículo K ;

q_{ij} , demanda do arco $(i, j) \in R \subseteq A$;

C_{ij} , o custo de travessia do arco $(i, j) \in A$.

Variáveis:

x_{ijk} , é o número de vezes que o arco (i, j) é atravessado pelo veículo k ;

$y_{ijk} \begin{cases} 1, & \text{se o veículo } k \text{ descarrega ou coleta ao longo do arco } (i, j); \\ 0, & \text{caso contrário} \end{cases}$

No modelo matemático apresentado, tem-se: (5) é a função objetivo que minimiza o custo da distância total percorrida pelos veículos; a restrição (6) garante a conservação do fluxo dos veículos; a restrição (7) faz com que cada arco requerido seja atendido uma vez por um único veículo durante todo o trajeto; a restrição (8) assegura que cada arco $(i, j) \in R$ possa ser atendido por um veículo k somente se este atravessar tal trecho. A equação (9) assegura a não violação da capacidade W de cada veículo. A restrição (10) evita a formação de subciclos e ainda garante que o depósito representado pelo nó 1 esteja incluído na rota. Em (11), a variável binária define o atendimento ou não de um arco. E por fim em (12), a variável x_{ijk} assume um valor inteiro que representa o número de vezes que um arco foi atravessado.

Golden e Wong (1981) mostram que o Problema do Carteiro Chinês, Problema do Carteiro Chinês Capacitado - PCCC e o Problema do Carteiro Rural podem ser vistos como casos especiais do CARP, pois se tiver apenas um veículo k de capacidade $W \geq \sum_i \sum_j q_{ij}$, tem-se o PCC; quando a demanda $q_{ij} > 0$ para todo $(i, j) \in E$, o CARP se reduz ao PCCC, no caso em que apenas um veículo k de capacidade $W \geq \sum_i \sum_j q_{ij}$ e $q_{ij} > 0$ para todo $(i, j) \in R \subset E$, tem-se um PCR.

Como o CARP envolve restrições operacionais que limitam a capacidade de atendimento, sua resolução de maneira exata é considerada difícil. Golden e Wong (1981) demonstram que a complexidade do CARP é *NP-Hard*. Devido sua complexidade, diversos métodos heurísticos foram propostos na literatura para a resolução do CARP. Eiselt *et al.* (1995b) classificam esses métodos em três categorias: os construtivos simples; os construtivos de duas fases; e os de melhoria.

Na primeira categoria, as rotas são formadas de maneira sequencial e, posteriormente, essas rotas podem ser agrupadas segundo um critério de ganho. Dentro dessa classificação, destacamos o algoritmo *Construct-Strik* desenvolvido por Christofides em 1973 (PEARN, 1989), *Path-Scanning* (GOLDEN *et al.*, 1983) *Augment-Merge* (GOLDEN *et al.*, 1981) e *Augment-Insert* (PEARN, 1991).

Os métodos construtivos de duas fases consistem em dividir o processo de solução em: fase de particionamento, que determina os segmentos que serão designados a cada unidade de trabalho, e fase de roteirização, que determina a sequência em que esses segmentos devem ser percorridos. Tais métodos são compostos por duas estratégias básicas de solução: agrupar primeiro – roteirizar depois (*cluster first – route second*) e roteirizar primeiro – agrupar depois (*route first – cluster second*).

A terceira categoria consiste em métodos de melhoria. São métodos aplicados sobre uma solução gerada em um dos métodos anteriores. Seu objetivo é produzir soluções melhores e mais próximas possíveis do limite inferior que caracteriza o problema. Nessa categoria, podem ser empregados métodos baseados na troca de segmentos entre ciclos adjacentes, técnicas como 2-opt, 3-opt e k-opt adaptados. Helsgaun (2000) apresenta a implementação e os aspectos conceituais dessas técnicas.

O processo de melhoria pode envolver ainda a utilização de *meta-heurísticas*, tais como: *Busca Tabu*, *Simulated Annealing*, Algoritmo Genético, *Scatter Search*, Colônia de Formigas, além do uso de estratégias híbridas que envolvem o uso combinado de duas ou mais dessas estratégias (ARAÚJO, 2003). O uso de algumas dessas meta-heurísticas pode ser visto em Eglese (1994) - *Simulated Annealing*; Hertz, Laporte e Mittaz (2000) - *Busca Tabu*; Beullens *et al.* (2003) - *Guided Local Search*; Lacomme e Prins e Sevaux (2006) - *Algoritmo Genético*.

Além dessas três categorias de métodos de resolução do CARP, apresentadas por Eiselt *et al.* (1995b), outros métodos têm sido desenvolvidos.

Lacomme, Prins e Ramdane-Chérif (2002b) propõem um algoritmo de planos de corte; Wang e Wen (2002) apresentam um método de resolução para o CARP, com janelas de tempo, utilizando a lógica *fuzzy*, e Tagmouti, Gendrau e Polvin (2007) propõe uma transformação do CARP em um problema de roteamento em nós, solucionando através de um esquema de geração de colunas.

2.3.4 Problema de Roteamento em Arcos Capacitado e Periódico

Lacomme, Prins e Ramdane-Chérif (2002a) ressaltam que o CARP é útil para modelagem de problemas que exigem tomadas de decisão em nível operacional, mas muitas aplicações requerem tomadas de decisões em níveis táticos, onde o planejamento seja realizado em um horizonte de multiperíodos, sujeito a restrições de frequência.

Os autores apresentam esse novo problema, o qual foi denominado *Periodic CARP* ou Problema de Roteamento em Arcos Capacitado e Periódico (*Periodic Capacitated Arc Routing Problem - PCARP*). O PCARP é capaz de abordar os níveis táticos e operacionais em paralelo. Para a resolução do PCARP Lacomme, Prins e Ramdane-Chérif (2002a) propõem a utilização de um Algoritmo Genético proposto por Lacomme, Prins e Ramdane-Chérif (2001), o qual era considerado o método mais eficiente de solução para o CARP, o modelo é aplicado na coleta de resíduos urbanos na cidade de Troyes na França.

Chu, Labadi e Prins (2005) definem o PCARP em um grafo não direcionado $G = (X, E)$ e um horizonte de tempo H de p períodos (dias), onde o nó 1 é considerado o depósito, que contém V veículos idênticos de capacidade limitada W . Cada aresta (i, j) esta associada a um custo C_{ij} e uma demanda q_{ij} , um subconjunto de arestas (tarefas) requeridas R , que devem ser atendidas e cada tarefa (i, j) tem uma frequência f_{ij} número de serviços necessários em H , que devem respeitar um espaçamento mínimo e máximo entre dois tratamentos sucessivos para uma tarefa (i, j) .

O objetivo do PCARP é determinar uma rota de custo mínimo, onde todas as viagens comecem e terminem no depósito e as tarefas (i, j) sejam atendidas f_{ij} vezes ao longo de H respeitando o espaçamento e a capacidade dos veículos v . O PCARP é um problema *NP-hard* uma vez que inclui o CARP para um único período como caso particular. Até mesmo, em casos que o número de períodos e o número

de veículos seja igual a 2 ($np = nv = 2$), o custo para todas as arestas é nulo $c_{ij} = 0$ e a periodicidade das tarefas $f_{ij} = 1$, o problema residual ainda é *NP-hard* (LACOMME; PRINS; RAMDANE-CHÉRIF, 2002c).

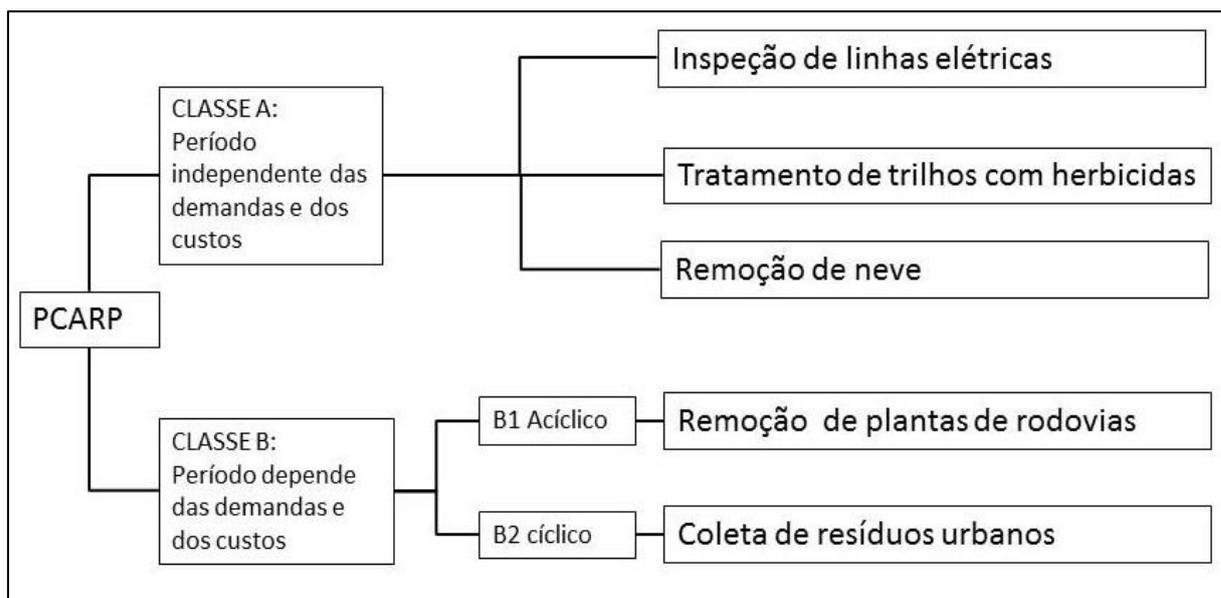
Lacomme, Prins e Ramdane-Chérif (2002a) fazem uma analogia do PCARP com o gerenciamento da produção. A fase de atribuição do tratamento das ruas corresponde a decisões de níveis táticos, enquanto que, a solução do CARP que define uma rota para cada dia corresponde à decisão operacional. Portanto, resolver PCARP implica a determinação simultânea de decisões táticas: determinando para cada rua um número de tratamento de acordo com sua frequência; e operacionais: determinando uma rota, de forma que, para cada período, as ruas sejam atendidas. O custo total depende da combinação das decisões de designação e roteamento (LACOMME; PRINS; RAMDANECHÉRIF, 2005).

Em algumas aplicações como a coleta de resíduos urbanos a aplicação do PCARP pode apresentar algumas complicações, por apresentar demandas flutuantes que podem variar de acordo com os dias de serviço e/ou apresentarem restrições de espaçamento entre dias de serviço (LACOMME; PRINS; RAMDANE-CHÉRIF, 2002a). Observa-se que, os custos em servir os arcos derivam das demandas e as demandas muitas vezes dependem do período. Baseado nisso Lacomme, Prins e Ramadane-Chérif (2005) propõem uma classificação para o PCARP, FIGURA 3.

A *classe A* é formada pelo conjunto de problemas em que a demanda não depende de um acúmulo diário, portanto, a demanda de cada tarefa e seu custo será idêntica para todos os dias de serviço. Por exemplo, o tratamento de trilhos com herbicidas; nesse caso, a quantidade aplicada e o tempo de pulverização por metro são fixos.

A *classe B* é composta por problemas em que a demanda aumenta com o passar do tempo, por exemplo, coleta de resíduos urbanos. Dois casos são destacados, um em que a produção diária é constante e outro em que se tem uma tendência global na rede, como um pico de resíduos produzidos em um determinado período (dia), por exemplo, finais de semana e feriados. Além disso, problemas da *classe B*, podem ser subdivididos em duas subclasses: acíclicos (*subclasse B1*) e cíclicos (*subclasse B2*).

FIGURA 3 - CLASSIFICAÇÃO DO PCARP



FONTE: Lacomme, Prins e Ramdane-Chérif (2005).

Na *subclasse B1*, o horizonte de tempo é linear e as demandas são flutuantes ao longo do tempo. É o caso da remoção de plantas em rodovia, em que, dependendo da estação, o crescimento é menor. Na *subclasse B2*, o intervalo de tempo entre os serviços se repetem continuamente, por exemplo, a coleta de lixo. Problemas dessa classe são difíceis de serem resolvidos por algoritmos que constroem a solução dia a dia (LACOMME; PRINS; RAMDANE-CHÉRIF, 2002a).

Observa-se que, em algumas aplicações, têm-se diferentes frequências para algumas tarefas ao longo do horizonte de tempo. Pensando nisso, Moroy, Amaya e Langevin (2013) propõem uma subdivisão dos serviços, classificando-os em regulares ou irregulares. Por exemplo, o monitoramento da rede rodoviária é realizado periodicamente, pois as estradas principais podem exigir maior monitoramento em determinados períodos devido ao aumento do tráfego nos finais de semana e feriados. Como durante a semana o tráfego apresenta menor frequência, uma subdivisão de todo horizonte de tempo é necessária e as frequências de servir as tarefas variam de acordo com estas subdivisões (MOROY; AMAYA; LANGEVIN, 2013).

Moroy, Amaya e Langevin (2013) propõem uma heurística de duas fases, agrupar primeiro – roteirizar depois (*cluster first – route second*). Na primeira fase, os arcos são atribuídos às rotas sem violar a capacidade dos veículos; na segunda fase, um problema de roteamento em arco é resolvido para otimizar cada rota. A

abordagem heurística foi testada em cinco conjuntos de instâncias, mostrando que o pior desvio do limite superior foi de 24% com tempos de execução inferiores a 32 minutos para todas as instâncias.

Batista (2014) ressalta que os primeiros conceitos sobre o PCARP começam a ser tratados no trabalho de Eglese (1994) com o roteamento de veículos responsáveis por despejo de sal em rodovias durante as madrugadas de inverno. Apesar de não ser nomeado de PCARP, o problema possui suas características, pois leva em conta a capacidade dos veículos, possui múltiplos depósitos, mas não permite rotas que comecem e terminem em diferentes depósitos, e o planejamento é realizado para um dia com intervalos para o atendimento das vias.

O despejo de sal nas vias é realizado durante o início da manhã, quando a formação de gelo torna as estradas perigosas. Eglese (1994) destaca que, se o despejo de sal fosse realizado muito cedo, o vento dispersaria todo o sal, e se fosse feito muito tarde, poderia tornar-se inviável devido ao aumento do tráfego pela manhã. Portanto, tem-se uma restrição de tempo ligada ao tratamento das vias, divididas em duas categorias de acordo com a frequência de atendimento: a primeira categoria é formada por vias atendidas a cada 2 horas; a segunda, por vias atendidas a cada 4 horas

O objetivo do problema proposto por Eglese (1994) era minimizar os custos totais anuais com a operação, minimizando o custo fixo com veículos (número de veículos), reduzir a distância total percorrida e o número de dias no ano em que esses veículos eram requeridos. Como método de solução, o problema é dividido em duas etapas. Na primeira etapa, o autor considera apenas as vias de categoria 1, resultando assim em um Problema do Carteiro Chinês o qual foi resolvido pelo uso de métodos heurísticos. A segunda etapa consiste no uso da metaheurística *Simulated Annealing* para melhoria da solução (etapa anterior), especificamente usado para minimizar o número de rotas.

Chu, Labadi e Prins (2004) apresentam um modelo mais geral para o PCARP, não sendo necessário uma combinação de dias em que se pode executar o serviço em determinada tarefa, levando em consideração apenas os espaçamentos mínimos e máximos entre dois tratamentos sucessivos. Os autores sugerem limites inferiores para o PCARP, baseando-se na transformação do grafo do PCARP em um grafo especial para o CARP, provando que os limites inferiores para o problema transformado são também para o PCARP. Como método de solução, tais autores

propõem a aplicação da heurística *Scatter Search* (SS) para 23 instâncias adaptadas de Golden, Dearmon e Baker (1983), nas quais o desvio médio em relação aos limites inferiores foi considerado satisfatório.

Em algumas aplicações, faz-se necessário uma combinação de dias em que uma tarefa pode ser atendida, como ocorre, por exemplo, na coleta de resíduos urbanos, podendo ser necessário designar os dias da semana em que uma via deve ser atendida: diariamente ou em uma combinação de dias [segunda, sexta], [terça, quinta], [segunda, quarta, sexta], etc.

Chu, Labadi e Prins (2005) apresentam um modelo de programação binária que leva em consideração essas combinações de dias. Como método de solução, os autores desenvolvem e avaliam o desempenho de três heurísticas. A primeira, uma heurística denominada *Decreasing Frequencies Nearest Insertion Heuristic* (DFNH), na qual as tarefas são organizadas em uma lista de frequência em ordem decrescente e inseridas em rotas preestabelecidas, obedecendo a ordem em que aparecem na lista. A segunda, denominada *Best Insertion Heuristic* (BIH), a qual não possui uma lista de prioridades, pois em cada interação todas as tarefas restantes são candidatas à inserção. Por último, uma heurística de duas fases, a *Lower Bound Heuristic* (LBH). Na primeira fase, utiliza um limite inferior de modo a preparar uma lista de possíveis tarefas para cada dia, isto é, um *cluster*. As rotas são construídas na segunda, solucionando um CARP para cada *cluster*. Segundo os autores, a heurística de duas fases supera as outras duas, mas esta requer um tempo maior de execução.

Lacomme, Prins e Ramdame-Chérif (2005) destacam que pesquisas em problemas de roteamento periódicos em arcos são ainda muito recentes, sendo necessário aproximar os modelos acadêmicos de aplicações reais complexas. Para isso, os autores consideram o roteamento em arco baseado em redes mais realistas. Neste caso, o PCARP foi proposto para grafos mistos, com voltas proibidas e custos distintos para passagem e tratamento de um mesmo arco, mas não apresentam um modelo matemático. O método de solução proposto foi um Algoritmo Memético (AM), que envolve um Algoritmo Genético com um procedimento de busca local. No entanto, testes computacionais destacam a falta de um limite inferior para proporcionar uma avaliação mais precisa do GA híbrido.

Chu, Labadi e Prins (2006) consideram o PCARP como uma extensão natural do CARP com multiperíodos. A maioria das empresas dão prioridade ao tamanho da

frota (custo de investimento). Partindo desse pressuposto, os autores propõem uma função objetivo com um objetivo primário e um secundário para o PCARP:

$$F(S) = M \cdot nvu(S) + cost(S) \quad (13)$$

onde $nvu(S)$ representa o tamanho da frota, $cost(S)$ representa o custo total das rotas (distância percorrida ou o tempo gasto) e M é uma constante positiva sempre maior que o $cost(S)$ de modo a garantir uma prioridade no número de veículos. Observa-se que se a frota já foi adquirida ($M = 0$) a função objetivo restringe-se a minimizar os custos totais das rotas, objetivo clássico do Problema de Roteamento Periódico de Veículos (PRPV) (CHU; LABADI; PRINS, 2006). Como método de solução, os autores aplicam dois algoritmos, uma heurística gulosa de inserção *Best Insertion Heuristic* (BIH) e um algoritmo baseado em busca local *Scatter Search* (SS). Dois conjuntos de instâncias são avaliadas para o PCARP, mostrando que o SS apresenta desempenho superior à heurística BIH.

Com o objetivo de propor um método de solução para o PCARP mais realista, Kansou e Yassine (2009) consideraram um grafo com as mesmas características que Lacomme, Prins e Ramdame-Chérif (2005). Como método de solução, desenvolveram um método híbrido, combinando a metaheurística Colônia de Formigas (*Ant Colony Optimization - ACO*) com a heurística de inserção *Insertion In Best Combination* (IIBC). A metaheurística ACO é utilizada para otimizar a ordem das tarefas para a inserção, e a heurística IIBC determina a combinação e a posição adequada para a inserção. Os resultados apresentados mostraram-se robustos e rápidos. O método híbrido proposto encontrou treze soluções melhores, quando comparados com resultados apresentadas por Lacomme, Prins e Ramdame-Chérif (2005).

Mei, tang e Yao (2011) propuseram um modelo matemático onde a função objetivo é composta por dois objetivos, primário e secundário, assim como proposto por Chu, Labadi e Prins (2006). No primário, a finalidade é minimizar o número de veículos e o secundário engloba o custo total. Os autores destacam que o objetivo primário dificilmente pode ser melhorado com os operadores tradicionais de pesquisa existentes para o CARP. Para atender a esses objetivos, os autores propõem um Algoritmo Memético (MA) e um procedimento de fusão de rotas *Route Merging* (RM), este último é concebido e incorporado ao MA para atender o objetivo primário, dando origem ao algoritmo denominado MARM. Assim, uma solução é otimizada primeiramente para o objetivo primário por meio da RM e, posteriormente,

melhorado para o secundário. O MARM foi comparado com outras metaheurísticas existentes para dois conjuntos de instâncias do PCARP, um adaptado de Golden, Dearmon e Wong (1983) e outro de Benavent *et al.* (1992), apresentando soluções melhores do que os algoritmos em comparação e com ganho em tempo computacional.

Em aplicações do PCARP como coleta de lixo, o material é acumulado ao longo do tempo e, em seguida, removido. Riquelme-Rodríguez, Langevin e Gamache (2014, 2016, não publicado) consideram um problema em que o material é entregue por uma frota de veículos e, em seguida, é consumido com o passar do tempo. São exemplos dessa aplicação a supressão de poeira em estradas e a irrigação de plantas ao longo das calçadas. O principal objetivo do modelo apresentado pelos autores é reduzir o custo associado à penalidade por não irrigar a estrada e ao mesmo tempo reduzir os custos de roteamento.

Tal modelo foi chamado PCARP com restrição de inventário. Como método de solução para o modelo proposto, foi aplicado a heurística *Adaptive Large Neighborhood Search* (ALNS), que inicia sua busca a partir de uma solução inicial viável e trabalha com diferentes operadores que destroem e reparam a solução inicial, permitindo alcançar uma grande vizinhança. Desenvolveram-se oito operadores de destruir/reparar para solução inicial, os quais são escolhidos aleatoriamente para serem aplicados. Testes computacionais mostraram que a utilização de operadores de diversificação obtiveram desempenho superior aos operadores de intensificação para todas as instâncias testadas.

Batista (2014) apresentou um modelo com características diferenciadas das já propostas na literatura, como: não necessitar voltar a um depósito ao final de um dia; a possibilidade de um veículo folgar em um determinado dia; uma penalidade caso a periodicidade de um arco não seja respeitada; e a capacidade do veículo foi convertida na máxima distância que o veículo pode se deslocar, sendo um arco por período (dia). Seu modelo foi aplicado na manutenção preditiva e preventiva de linhas férreas. Para isso, utilizou um *software* comercial, conseguindo encontrar solução ótima para apenas uma das 23 instâncias adaptadas de Golden, Dearmon e Wong (1983) para o modelo proposto.

Huang e Lin (2014) combinam um algoritmo colônia de formigas (ACO) com um método de busca local, especificamente o método de melhoria 2-opt. Apresentam um modelo de roteamento periódico em arcos com pontos de recarga,

denominado *Periodic Arc Routing with Refill Points* (PARPRP), modelo que foi aplicado em um caso real de irrigação de árvores na cidade Kaohsiung em Taiwan. O método de resolução foi dividido em duas fases, sendo a primeira a aplicação do algoritmo ACO e, em seguida, o método de busca local, empregado para melhorar as soluções geradas na primeira fase. Os autores destacam que a taxa de melhoria com a aplicação do 2-opt pode chegar a 67,9%, o que foi considerado extraordinário.

Batista (2016, não publicado) apresenta uma modificação para modelo proposto pelo autor em 2014, esse novo modelo foi denominado Problema de Roteamento em Arcos Capacitado e Periódico com Movimento Contínuo (PCARPMC). Nessa nova formulação os veículos não necessitam terminar suas rotas no nó de origem de seu percurso, podendo terminar em qualquer nó, desde que esse seja início da rota de qualquer outro veículo. Maiores detalhes a respeito da formulação matemática do modelo serão tratados no capítulo 3. Juntamente com o novo modelo o autor propõe uma classificação taxonômica e uma bibliometria, com o objetivo de classificar os trabalhos que tratam a respeito do PCARP. Até o momento apenas uma simples classificação pode ser encontrada em Lacomme, Prins e Ramdame-Chérif (2005), a qual foi abordada anteriormente.

Um resumo dos aspectos abordados por Batista (2016, não publicado) para a proposição da classificação taxonômica é apresentado no QUADRO 1.

QUADRO 1 - RESUMO DA PROPOSTA TAXONÔMICA

Característica	Possibilidades
Período	Dependente; Independente.
Horizonte de tempo	Cíclico; Acíclico.
Serviço	Regular; Irregular.
Roteirização	Múltipla; Contínua.
Dimensão da frota	Um veículo; Múltiplos veículos; Irrestrito.
Frequências	Intervalo de tempo; Combinação de dias; Dinâmica.
Rigor da demanda	Flexível; Rígida.
Grafo	Direcionado; Não-direcionado.
Foco	Teórico; Aplicado.
Abordagem de resolução	Exato; Heurístico; Ambos.

FONTE: Batista (2016, não publicado).

O período é classificado em dependente e independente. Por exemplo, o modelo proposto por Lacomme, Prins e Ramdame-Chérif (2002a) para a coleta de

resíduos urbanos é classificado como dependente, pois o período depende da quantidade de lixo produzido. Já o monitoramento de rede rodoviária proposta por Moroy, Amaya e Langevin (2013) é classificado como independente, pois segundo Batista (2016, não publicado, p.6) “[...] uma frequência acaba sendo determinada para esse tipo de serviço”.

O horizonte de tempo pode ser cíclico quando a demanda se repete ao longo de todo o horizonte de tempo, por exemplo, os problemas de monitoramento e os serviços de inspeção de linhas de energia elétrica. Problemas acíclicos são aquele em que a demanda de uma tarefa é acumulada desde o último atendimento, exemplo dessa classificação é a supressão de poeira em estradas (RIQUELME-RODRÍGUEZ; GAMACHE; LANGEVIN, 2014).

Os serviços são classificados em regular quando uma mesma frequência envolve todo o horizonte de tempo ou serviços irregulares quando existe diferentes frequências para cada subperíodo do horizonte de tempo.

A roteirização é subdividida em múltipla e contínua. A roteirização contínua ocorre quando o veículo pode realizar sua rota de forma contínua sem ser obrigado a voltar a um depósito, o único modelo proposto na literatura até o dado momento que se enquadra nessa classificação são os propostos por Batista (2014, 2016, não publicado). Já a roteirização com múltiplas visitas ocorre quando o veículo inicia em um depósito, realiza o atendimento de mais de um arco (cliente), e retorna ao depósito.

A dimensão da frota foi dividida em um veículo, múltiplos veículos e irrestrito. Problemas irrestritos ocorre quando a função objetivo determina a dimensão da frota. Problemas com essas características são abordados por Chu, Labadi e Prins (2006) e Mei, tang e Yao (2011).

A frequência é uma das principais características do PCARP. Essa está classificada em: intervalo de tempo; combinação de dias, que consiste em determinar para cada arco uma combinação de dias de atendimento; frequência dinâmica definida por Batista (2016, não publicado):

A frequência dinâmica não é definida pelos espaços entre serviços e nem uma quantidade de vezes necessária para atender cada arco. As demandas acabam de uma forma ou de outra dependendo sempre do período de atendimento. Quanto mais tempo um arco demora para ser atendido maior é a sua demanda. (BATISTA, 2016, não publicado, p. 12).

O rigor da demanda pode ser rígido, quando não permite atrasos no atendimento de um arco ou flexível permitindo atrasos de serviço ao custo de uma punição, como por exemplo, nos modelos propostos por Batista (2014, 2016, não publicado).

Com relação ao foco dos trabalhos, esses são divididos em aplicados e teóricos. Na grande maioria, os trabalhos acabam citando aplicações “ [...] as aplicações dão sentido ao desenvolvimento das pesquisas e exemplificam bem os propósitos de modelos matemáticos.” (BATISTA, 2016, não publicado, p.15). Enquanto que os teóricos são baseados em métodos e técnicas de resolução.

E por fim as abordagens de resolução são classificadas em exatas, heurísticas ou ambas. Uma classificação dos trabalhos pode ser vista no QUADRO 2.

QUADRO 2 - ABORDAGENS DE RESOLUÇÃO

ABORDAGENS	TRABALHOS
EXATA	Batista e Scarpin (2015); Riquelme-Rodríguez, Gamache e Langevin (2014); Batista e Scarpin (2014); Chu, Labadi e Prins (2003).
HEURÍSTICA	Zhang <i>et al.</i> (2016); Riquelme-Rodríguez, Gamache e Langevin (2016); Huang e Lin (2014); Mei <i>et al.</i> (2011); Kansou e Yassine (2009); Marzolf, Trépanier e Langevin (2006); Chu, Labadi e Prins (2006); Lacomme, Prins e Ramdame – Chérif (2005); Ghiani <i>et al.</i> (2005); Lacomme, Prins e Ramdame–Chérif (2002b); Lacomme, Prins e Ramdame–Chérif (2002c).
AMBAS	Riquelme-Rodríguez, Gamache e Langevin (2014); Monroy, Amaya e Langevin (2013); Chu, Labadi e Prins (2004); Chu, Labadi e Prins (2004).

FONTE: Batista (2016, não publicado) adaptado pelo autor.

Segundo a bibliometria realizada por Batista (2016, não publicado) o ano que ocorreu o maior número de publicações foi o ano de 2014 com um total de 5 publicações seguido do ano de 2005 com 3 publicações. Ao todo são 25 autores desenvolvendo pesquisas referentes ao tema PCARP, sendo que André Langevin do Canadá e Chistian Prins da França são os autores com maior número de publicações. As revistas que mais contem publicações são *Computers and Operations Research* e *European Journal of Operations Research*.

2.4 MÉTODOS DE SOLUÇÃO

Enquanto os problemas lineares contínuos possuem o algoritmo simplex para sua solução de forma exata e com grande eficiência, os problemas lineares discretos (problemas de Programação Inteira – PI) carecem de um algoritmo com a mesma eficiência. Os PI's podem ser resolvidos por meio de métodos exatos, como *Branch-and-Bound* - B&B, *Restrição Surrogate*, métodos de Planos de Corte e método híbrido *Branch-and-cut* - B&C. No entanto, na maioria das ocasiões, as técnicas e algoritmos de solução são específicas para cada tipo de problema de programação inteira. Para Arenales (2007), os métodos exatos mais bem-sucedidos são baseados no método B&B e de Planos de Corte.

Em virtude dessa realidade, técnicas e algoritmos computacionais eficientes que visam encontrar soluções viáveis, com tempo e custo razoáveis, mas sem garantia de otimalidade para os problemas de programação inteira, têm sido desenvolvido. Esses algoritmos são denominados de heurísticos. O termo heurístico é derivado do grego *heuriskein*, que significa descobrir ou achar; em pesquisa operacional, o termo refere-se a uma técnica que busca alcançar boas soluções utilizando um esforço computacional razoável (GOLDBARG, 2005).

Como visto na seção anterior, grande parte dos métodos desenvolvidos para solucionar o PCARP são métodos heurísticos. Deste modo, as próximas seções são dedicadas a discutir as propriedades e características das heurísticas *relax-and-fix* e *Relaxation Induced Neighborhood Search* (RINS), uma vez que essas serão utilizadas como método de solução para os modelos PCARP abordado nesse trabalho.

2.4.1 Heurística *relax-and-fix*

A heurística *Relax-and-Fix* (R&F) foi descrita por Wolsey (1998) e consiste em um método de decomposição de um modelo de programação inteira mista em submodelos menores disjuntos, que podem ser resolvidos rapidamente, porém sem a garantia de resolução do problema original de forma ótima. A heurística tem sido amplamente aplicada para solucionar problemas de otimização combinatória, em especial, problemas de dimensionamento de lotes, como se pode confirmar em:

Belvaux e Wolsey (2000), Mercé e Fontan (2003), Ferreira *et al.* (2010), Wu *et al.* (2011) e outros.

A heurística propõe solucionar o problema em etapas, sendo que, cada etapa, resolve de forma exata um subproblema derivado do problema original (CHERRI, 2013). Para tanto, baseia-se na partição das variáveis inteiras ou binárias do modelo, criando R conjuntos disjuntos $Q^r, r = 1, \dots, R$. Observa-se que o conjunto de variáveis originais é dividido em três grupos: um contendo variáveis inteiras e binárias; um segundo contendo as variáveis relaxadas (livres); e o último com variáveis fixas (com seu valor fixado). O número R de conjuntos determina o número de iterações da heurística (FERREIRA, 2007).

A principal característica do *relax-and-fix* é a solução de submodelos que são menores que o modelo original e, possivelmente, mais fáceis de serem resolvidos (FERREIRA *et al.*, 2010). Apesar da heurística resolver diversos subproblemas, em vez de apenas um, como ocorre na resolução do MIP original, há redução no esforço computacional, que é proporcionado pela diminuição do número de variáveis inteiras (CUNHA, 2013). Observa-se ainda que no final de cada iteração um conjunto de variáveis são fixados, impedindo que seus valores possam ser alterados posteriormente. O critério utilizado para fixar essas variáveis tem forte ligação com o grau do submodelo gerado e a dificuldade para solucioná-lo (WOLSEY, 1998).

O comportamento clássico dessa heurística é descrito por Pochet e Wolsey (2006). Considere um problema inteiro misto (MIP), supondo que as variáveis binárias y são particionadas em R conjuntos disjuntos Q^1, Q^2, \dots, Q^R . Então, resolve-se sequencialmente os R MIP's, denotados por MIP^r com $1 \leq r \leq R$. Ao final do processo, espera-se determinar uma solução heurística factível para o MIP original.

Por exemplo, considere um problema de planejamento de produção, em que Q^1 é composto por todas as variáveis y associadas a períodos de tempo 1, Q^2 aquelas associadas ao período de tempo 2 e assim por diante. No primeiro MIP^1 , impõem-se a integralidade das variáveis y associadas a Q^1 e relaxa-se a restrição de integralidade em todas as outras partições Q .

MIP^1 :

$$\min z = cx + fy \tag{14}$$

Sujeito à:

$$Ax + By \geq b \quad (15)$$

$$x \in \mathbb{R}_+^n \quad (16)$$

$$y_j \in \{0,1\} \quad \forall j \in Q^1 \quad (17)$$

$$y_j \in [0,1] \quad \forall j \in Q - Q^1 \quad (18)$$

Sabe-se que, para problemas de otimização inteiro-misto, a dificuldade de resolução pode estar associada ao número de variáveis inteiras. Quanto maior esse número, mais difícil pode ser sua resolução (SILVA, 2013). Portanto ao resolver um subproblema MIP^r torna-se mais fácil, uma vez que o número de variáveis inteiras associadas ao submodelo é inferior as associadas MIP original.

Resolvendo o subproblema MIP^1 de forma exata, obtém-se (x^1, y^1) uma solução ótima para MIP^1 . Em seguida, fixam-se as variáveis de Q^1 em seus respectivos valores y^1 , mantém-se a restrição de integralidade para as variáveis y_j , $\forall j \in Q^2$ e relaxam-se as variáveis y_j , $\forall j \in Q - (Q^1 \cup Q^2)$, obtendo-se, assim, o MIP^2 .

Para um subproblema MIP^r , com $2 \leq r \leq R$, tem-se as variáveis y associadas a $Q^1 \cup \dots \cup Q^{r-1}$, fixadas de acordo com as soluções obtidas na resolução dos MIP^{r-1} anteriores. Adiciona-se a restrição de integralidade para as variáveis de Q^r e mantêm-se relaxadas as variáveis de $Q - (Q^1 \cup \dots \cup Q^r)$.

MIP^r :

$$\min z = cx + fy \quad (19)$$

Sujeito à:

$$Ax + By \geq b \quad (20)$$

$$x \in \mathbb{R}_+^n \quad (21)$$

$$y_j = y_j^{r-1} \in \{0,1\} \quad \forall j \in Q^1 \cup \dots \cup Q^{r-1}, 2 \leq r \leq R \quad (22)$$

$$y_j \in \{0,1\} \forall j \in Q^r, 2 \leq r \leq R \quad (23)$$

$$y_j \in [0,1] \forall j \in Q - (Q^1 \cup \dots \cup Q^r), 2 \leq r \leq R \quad (24)$$

Resolvendo MIP^r , obtém-se uma solução (x^r, y^r) , com $2 \leq r \leq R$. Caso um subproblema MIP^r seja infactível para algum $r \in \{1, \dots, R\}$, a heurística falhou ou então (x^R, y^R) é uma solução viável (factível) obtida pela heurística *relax-and-fix* para o problema MIP original. O QUADRO 3 descreve o algoritmo *relax-and-fix*.

QUADRO 3- ALGORITMO *RELAX-AND-FIX*

Inicialização – defina uma partição para as variáveis inteiras ou binárias criando R subconjuntos, $Q^r, r = 1, \dots, R$.

Para $r = 1, \dots, R$ **faça**

Passo 1 - Relaxe as variáveis inteiras/binárias para os subconjuntos $Q^i, i = r + 1, \dots, R$, mantenha a integralidade das variáveis do subconjunto Q^r .

Se $r = 1$ resolva o subproblema gerado.

Se não $r \geq 2$ fixe as variáveis do subconjunto $Q - (Q^r \cup Q^{r+1} \cup \dots \cup Q^{R-1})$.

Passo 2 - Resolva o subproblema resultante do passo 1.

Passo 3 - Se o problema gerado no passo 1 for infactível, pare. Caso contrário continue o procedimento voltando ao passo 1.

Fim Para

FONTE: O autor (2017)

A heurística *relax-and-fix*, segundo a literatura (MERCÉ & FONTAN, 2003; MORAES & SANTOS, 2012; CUNHA, 2013; FERREIRA, 2007), tem demonstrado bom desempenho para determinar soluções iniciais para modelos de dimensionamento de lotes. No entanto, apresenta uma fraqueza, uma vez que um subproblema é otimizado apenas para um pequeno subconjunto de variáveis, a fixação dessas pode afetar negativamente a otimização em subproblemas posteriores, gerando assim, soluções de baixa qualidade ou até mesmo infactíveis.

Observa-se que a heurística deve ser interrompida caso algum subproblema gerado no passo 1 da R&F seja infactível. Isso significa que não existe solução factível para o atual subproblema, considerando as variáveis fixadas previamente. Fato que poder ser comum, caso sejam fixadas todas as variáveis de períodos antecedentes. Afim de tentar evitar esse *status* da solução, pode-se inserir um passo a mais no algoritmo. Esse passo pode contemplar a fixação de partes do conjunto de variáveis inteiras (FEDERGRUEN *et al.*, 2007), o que nem sempre garante subproblemas factíveis.

O processo de escolha das partições das variáveis do problema afeta o desempenho da heurística, uma vez que seu objetivo é resolver submodelos inteiros mistos menores (MORAES; SANTOS, 2012). A heurística *relax-and-fix* com partição do conjunto de variáveis por período foi utilizada primeiramente por Dillenberger *et al* (1994) para resolver um modelo de dimensionamento de lotes multimáquinas, multiperíodos e multi-itens.

O número de iteração nesse caso é dado pelo número de períodos do modelo. Outras estratégias de decomposição do modelo em períodos, itens e níveis são descritas em Cunha (2013) e Ferreira (2007). É comum também a combinação de duas ou mais dessas estratégias para a formulação de novas estratégias híbridas, como pode ser visto no trabalho realizado por Ulbricht (2015), para a resolução de um modelo de planejamento da produção multiestágio com estoques intermediários limitados.

Mercé e Fontan (2003) propuseram a heurística do tipo *relax-and-fix*, como método de solução para o problema clássico de dimensionamento de lotes com restrição de capacidade. Para uma dada iteração k , os autores dividem o horizonte de planejamento em três seções: a primeira é composta de variáveis fixadas parcialmente ou totalmente; a segunda seção é composta pelas variáveis binárias ou inteiras; e por fim, a terceira é composta por variáveis relaxadas.

Federgruen *et al.* (2007) aplicam duas heurísticas de intervalos progressivos para a solução de um modelo de dimensionamento de lotes do tipo *big bucket*, capacitado com *set up* e consideram o *relax-and-fix* como um caso particular da heurística de intervalos progressivos. Para determinar quantas variáveis inteiras serão fixadas é definido um parâmetro τ .

A fim de evitar infactibilidade que pode ocorrer no passo 2 do algoritmo R&F, Escudero e Salmeron (2005) apresentam uma variante para o algoritmo *relax-and-fix*, o qual é denominado *Enhanced Fix-and-Relax* (EFR), que detecta problemas de infactibilidade. O algoritmo melhorado tem o potencial de voltar passos atrás se houver inviabilidade no modelo, que, por sua vez, proporciona uma abordagem mais próxima do problema original.

Observa-se que, na pior das hipóteses, o EFR acaba resolvendo o modelo original. Os autores propõem variações da heurística para as partições do conjunto de variáveis. A cada variável é atribuído um valor, e as variáveis são ordenadas de forma decrescente de valor, as estratégias variam então pelo valor atribuído a cada variável, e um parâmetro k que determina o número de iterações da heurística. O parâmetro k é definido por $k = \frac{n}{n'}$, onde n' indica o número de variáveis inteiras e n o número total de variáveis inteiras que se deseja por iteração. O *Enhanced Fix-and-Relax* é utilizado como método de solução para o problema de agendamento aplicado no setor de energia elétrica na Espanha.

O critério utilizado para fixar as variáveis tem forte influência no grau de dificuldade de resolução do subproblema gerado, podendo até mesmo eliminar a solução ótima do MIP original, uma vez que as variáveis fixadas correspondem à solução ótima do subproblema da iteração corrente. Visando reduzir essa imposição, Cunha (2013) propõe uma variação para R&F, a qual foi chamada *relax-and-fix forward* orientada a períodos com *pool* (RF-FPSP). A estratégia seleciona um conjunto com as soluções mais promissoras encontradas a cada iteração, o qual é denominado *pool* de soluções.

Para a primeira iteração, a heurística RF-FPSP é semelhante a R&F, exceto por armazenar no *pool* as k melhores soluções encontradas para esse subproblema, que passam para a próxima etapa da heurística. A partir da segunda iteração, para cada uma das k soluções do passo anterior, é executado um R&F tradicional, do qual também são selecionadas as k melhores soluções para o processo da partição seguinte. Cunha (2013) destaca a importância de controlar o tamanho do *pool* para evitar um crescimento exponencial, o que inviabilizaria a utilização desse procedimento. Outras estratégias de decomposição e fixação de variáveis são descritas em Toledo *et al.* (2015).

A heurística *relax-and-fix* também é utilizada de forma híbrida com meta-heurísticas, como a Busca Tabu. A heurística R&F é utilizada para fornecer uma solução inicial para a Busca Tabu e/ou para a reconstrução de soluções (PEDROSO e KUBO, 2005; CHERRI, 2013). Pedroso e Kubo (2005) propuseram um algoritmo híbrido para solução do problema de dimensionamento de lotes com uma ou mais máquinas. O método híbrido consiste em uma Busca Tabu, e a heurística R&F é utilizada para inicializar e gerar novos pontos de partida da Busca Tabu.

O método executa diversas vezes a heurística *relax-and-fix*, tanto para a inicialização quanto para a reconstrução de soluções. Logo, é importante que ela seja capaz de encontrar soluções factíveis. Para isso, os autores propõem uma variante do R&F, denominada *relax-and-fix-one-product*. Nessa heurística, as partições são definidas por período/máquina e itens, com o objetivo de tornar menores os subproblemas, uma vez que, em alguns modelos de dimensionamento de lotes, com multi-itens, são de difícil solução.

SILVA (2013) apresenta um estudo sobre problema de dimensionamento de lotes com múltiplas plantas, múltiplos itens e múltiplos períodos. Para trabalhar com instâncias maiores, propõem um método híbrido (*Math-Heurística*) que combina o

método *relax-and-fix*, com a restrição de *local branching*. A proposta consiste em flexibilizar a fixação das variáveis de uma partição, adicionando a cada subproblema uma restrição do tipo local *branching*, buscando evitar infactibilidade e encontrar soluções de melhor qualidade.

2.4.2 Heurística *Relaxation Induced Neighborhood Search*

A heurística *Relaxation Induced Neighborhood Search* (RINS) é uma heurística de busca local apresentada por Danna, Rothberg e Pape (2005) com o objetivo de intensificar a busca por soluções viáveis de melhor qualidade, para problemas de programação inteiro mista (MIP). A RINS cria e explora uma vizinhança em torno do nó corrente da árvore de *branch-and-cut*. Para isso, leva em consideração a melhor solução inteira viável até aquele momento e a solução relaxada do nó. Para descrever as vizinhanças da RINS, a heurística foi baseada na ideia desenvolvida por Fischetti e Lodi (2003) para a heurística de Ramificação Local (*Local Branching*), a qual descreve a vizinhança por meio de sub-MIP's e a exploração é feita pela resolução desses MIP's. A RINS pode ser vista como um grande método de busca em vizinhanças que usa a relaxação contínuo para fixar variáveis definindo assim sua vizinhança (DANNA; ROTHBERG; PAPE, 2005).

Ao explorar uma árvore de *branch-and-cut*, a cada nó, duas soluções estão disponíveis, sendo: a solução incumbente, que é a melhor solução inteira até aquele momento, portanto, atende às restrições de integralidade; e a solução relaxada, que não é uma solução inteira, mas seu valor na função objetivo é sempre melhor do que a incumbente. Deste modo, as soluções incumbente e relaxada conseguem atingir apenas um dos seguintes objetivos, a integralidade ou a otimização do valor da função objetivo (DANNA; ROTHBERG; PAPE, 2005).

Ao examinar os valores das variáveis de ambas as soluções, observa-se que algumas variáveis assumem valores diferentes entre as soluções incumbente e relaxada, mas é importante notar que muitas assumem o mesmo valor. A RINS baseia-se na ideia de fixação das variáveis com valores em comum formando assim uma solução parcial, a qual pode ser estendida para uma solução completa, inteira e que pode atingir um melhor valor para função objetivo.

A heurística RINS consiste, após encontrar a primeira solução viável para cada nó da árvore global de *branch-and-cut*, nos seguintes passos:

1 - Crie um sub-MIP, em que variáveis com valores iguais entre a solução *incumbent* e a relaxada contínua corrente serão fixadas;

2 - Defina um corte com base no valor da função objetivo da solução *incumbente* para os nós da árvore de branch-and-cut, que será explorado na resolução do sub-MIP;

3 - Resolva o sub-MIP criado.

O sub-MIP gerado pode ser de difícil resolução, portanto, não é possível *a priori* determinar o tamanho da árvore de enumeração que será gerada para sua resolução. Nesse sentido, para evitar que o solver gaste muito tempo determinando a solução para o sub-MIP, Danna, Rothberg e Pape (2005) propõem a definição de um parâmetro nl que limita o número máximo de nós a serem explorados por sub-MIP.

A cada nó da árvore do *branch-and-cut* em que a heurística RINS for executada uma diversificação no espaço de busca estará sendo realizada. Nota-se que a aplicação da RINS em nós consecutivos geram vizinhança muito similares. Para melhorar a diversificação é proposto um parâmetro f ($1 \leq f \leq 100$) que determina que a heurística será aplicada a cada f nós explorados.

O QUADRO 4 a seguir descreve o algoritmo para heurística RINS, onde f_{inc} e f_{rel} são as soluções incumbente e relaxada respectivamente na exploração do f -ésimo nó da árvore global.

QUADRO 4 - ALGORITMO RINS

A cada f nós explorados da árvore global de branch-and-cut faça:

Passo 1 – fixe todas as variáveis de mesmo valor entre as soluções f_{inc} e f_{rel} criando um sub-MIP com essas variáveis fixadas.

Passo 2 – Resolva o sub-MIP gerado no passo 1, explorando no máximo nl nós da árvore de B&C.

Passo 3 – Atualize f_{inc} para melhor solução inteira encontrada na resolução do sub-Mip se houver.

Passo 4 – retornar para a exploração do MIP global.

FONTE: O autor 2017.

Outra forma de explorar o espaço entre duas soluções é fixar todas as variáveis em comum entre a solução incumbente e a relaxada contínua, e resolver o sub-MIP resultante de forma exata.

3 MODELO MATEMÁTICO

3.1 APLICAÇÃO

Diversas são as aplicações do PCARP. Entre elas, destaca-se a aplicação em casos específicos de serviços de manutenção. A manutenção é importante para garantir o funcionamento e a segurança de equipamentos. Além disso, melhora a qualidade e reduz os custos de possíveis falhas e quebras, por isso é importante à empresa elaborar um bom plano de manutenção preditiva e preventiva para seus equipamentos.

Batista (2014) desenvolve um modelo matemático pensado na manutenção preditiva e preventiva de malhas ferroviárias. As empresas responsáveis por essa manutenção dispõem de equipamentos que devem ser colocados em locais estratégicos para garantir o funcionamento adequado dos trilhos. Os trens (veículos) responsáveis por esse serviço percorrem cada trecho da malha ferroviária em determinados períodos de tempo. Esses percursos a serem vistoriados possuem diferentes frequências de manutenção, percursos com grande fluxo de trens demandam manutenções com maior frequência do que trechos com menor fluxo. Por exemplo, em uma mesma malha pode-se ter trechos que demandam manutenção a cada 10 dias e outros a cada 30 dias. Caso essa periodicidade não seja atendida, então implica-se uma penalização na função objetivo. Essa punição viabiliza a modelagem do problema. O trecho com a maior periodicidade define o horizonte de tempo.

Uma das peculiaridades dos modelos descritos por Batista (2014, 2016, não publicado) é a não necessidade de voltar a um ponto específico ao final do dia, pois o veículo se desloca de maneira lenta, podendo iniciar o dia em certo ponto da malha e terminar em outro. Outra característica dos modelos propostos é a possibilidade do veículo folgar em qualquer dia do horizonte de tempo, em qualquer um dos nós da malha ferroviária. A possibilidade de folgar apresenta papel importante no modelo proposto, pois, assim pode-se evitar penalizações (BATISTA, 2014). Nesses modelos, as restrições de capacidades dos veículos estão associadas à máxima distância que o carro pode se deslocar, especificamente 1 trecho por dia.

3.2 FORMULAÇÃO MATEMÁTICA

Os modelos propostos em Batista (2014, 2016, não publicado), partem de um grafo não direcionado $G = (X, E)$ em que $X = \{x_1, x_2, \dots, x_n\}$ possui $|X| = n$ nós, e $E = \{e_1, e_2, e_3, \dots, e_m\}$ possui $|E| = m$ arestas, nk carros definidos pelo conjunto $K = \{1, 2, \dots, nk\}$. Cada aresta e é formada por um par de nós $x_{ij} = (x_i, x_j)$, com um custo associado c_{ij} .

Batista (2014, 2016, não publicado) considerou a capacidade dos veículos, sendo a máxima distância que estes podem se deslocar, 1 trecho por dia. Os trechos a serem percorridos pelos trens de manutenção, definidos a priori no modelo, serão tratados doravante como arcos ou arestas da malha ferroviária. O horizonte de tempo H é formado por np períodos $H = \{1, 2, \dots, np\}$, onde cada período é representado por p . A demanda nesses modelos é considerado a periodicidade de cada aresta, que é expressa na quantidade máxima de períodos em que uma aresta deve ser atendido ao menos uma vez $MP(x_{ij})$. A seguir são descritas as formulações matemáticas para ambos os modelos.

3.2.1 Modelo Batista (2014)

A seguir descreve-se o modelo apresentado por Batista (2014).

Modelo:

$$\min Z = \sum_{[i,j] \in E} \sum_{k=1}^{nk} \sum_{p=1}^{np} c_{ij} * x_{ijkp} + \sum_{[i,j] \in E} \sum_{p=1}^{np} PU_{ij} * p_{ijp} \quad (25)$$

Sujeito à:

$$\sum_{[i,j] \in E} x_{ijkp} + f_{jkp} - \sum_{[i,j] \in E} x_{jik,p+1} - f_{jk,p+1} = 0 \quad \forall j \in X, \forall k \in K, \forall p \in H \quad (26)$$

$$\sum_{[i,j] \in E} x_{ijkp} + \sum_{[i,j] \in E} x_{jikp} + \sum_{i=1}^n f_{ikp} = 1 \quad \forall p \in H, \forall k \in K \quad (27)$$

$$\sum_{k=1}^{nk} \left(x_{ijkp} + x_{jikp} + x_{ijk,p+1} + x_{jik,p+1} + \dots \right) + p_{ijp} \geq 1 \quad \forall [i,j] \in E, \forall p \in H \quad (28)$$

$$x_{ijkp}, p_{ijp}, f_{ikp} \in \{0,1\} \quad \forall [i,j] \in E, \forall k \in K, \forall p \in H \quad (29)$$

Onde:

$$\begin{array}{l}
 x_{ijkp} \left\{ \begin{array}{l} 1, \text{ se o carro } k \text{ se desloca do ponto } i \text{ para o ponto } j \text{ no período } p \\ 0, \text{ caso contrário} \end{array} \right. \\
 p_{ijp} \left\{ \begin{array}{l} 1, \text{ se o arco } x_{ij} \text{ não respeita a periodicidade no período } p \\ 0, \text{ caso contrário} \end{array} \right. \\
 f_{ikp} \left\{ \begin{array}{l} 1, \text{ se o carro } k \text{ fica parado no ponto } i \text{ no dia } p \\ 0, \text{ caso contrário} \end{array} \right.
 \end{array}$$

Cada variável p_{ijp} está associada a uma punição PU_{ij} , caso uma aresta não tenha sua periodicidade atendida. Essa variável permite atrasos no atendimento das arestas, se necessário. A variável f_{ikp} permite que um veículo folgue em um determinado dia, possibilitando em alguns casos obter uma solução com menor quantidade de deslocamentos. Batista (2016) destaca a importância da folga:

“Nos problemas reais podem ocorrer imprevistos e outros problemas que atrasem o atendimento à alguns arcos. A possibilidade de um carro folgar na modelagem permite que quando ocorra algum problema dessa natureza a folga compense o tempo permitindo adiantamento do movimento futuro.” (BATISTA, 2016, p. 34, não publicado).

As variáveis p_{ijp} e f_{ikp} foram propostas para que o modelo possua viabilidade.

O objetivo do modelo (25) é minimizar custos de deslocamentos e atrasos. A restrição (26) garante o fluxo diário dos veículos permitindo folgar. A restrição (27) garante que todos os veículos terão uma designação para cada dia p , isto é, cada veículo no dia p pode se deslocar de i para j , ou de j para i , ou ainda permanecer parado em um ponto i . A restrição (28) se refere à periodicidade que deve ser atendida em cada aresta. Observa-se que se a variável p_{ijp} assumir valor 1, indica que a aresta e_{ij} teve sua periodicidade atrasada em um dia implicando em uma punição PU_{ij} na função objetivo. Observa-se que, nos casos em que um conjunto de arestas R é formado por arestas que demandam apenas uma passagem em todo o horizonte de tempo, a restrição (28) pode ser simplificada e a variável p_{ijp} descartada, dando origem a restrição (30):

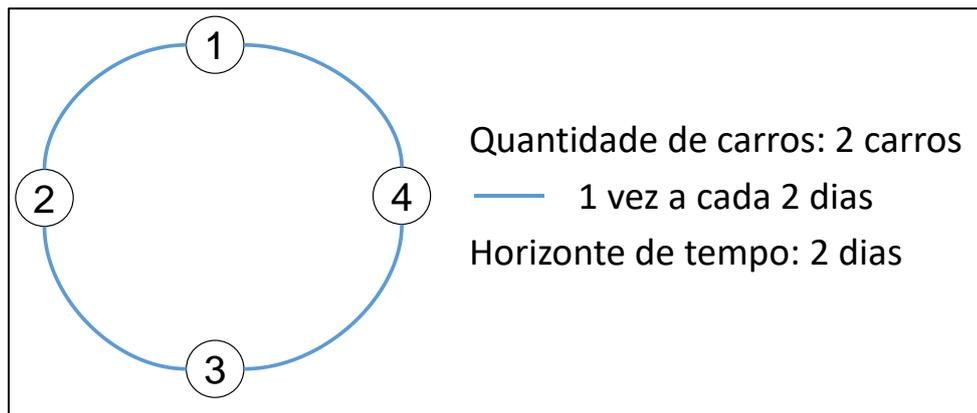
$$\sum_{k=1}^{nk} \left(x_{ijk1} + x_{jik1} + x_{ijk,2} + x_{jik,2} + \dots \right) \geq 1 \quad \forall [i, j] \in R \quad (30)$$

Por fim, a restrição (29) determina que todas as variáveis do modelo sejam binárias.

3.2.2 Modelo Batista (2016)

Para entendimento da nova formulação proposta por Batista (2016, não publicado) considera-se o exemplo ilustrado na FIGURA 4.

FIGURA 4 - EXEMPLO PCARPMC



FONTE: Batista (2016, não publicado).

O exemplo é formado por: um horizonte de tempo (H) de 2 dias, 2 carros de capacidades iguais e arestas de demandas iguais a 1 atendimento no horizonte H . A única forma desse problema ser viável seria a possibilidade de que, a quantidade de carros que sai no primeiro dia de algum vértice seja igual a quantidade de carros que retornam a esse vértice no último dia. Por exemplo, o carro 1 realizar a rota 2-3-4 e o carro 2 a rota 4-1-2, no fim do horizonte de tempo o carro 1 está no vértice em que o carro 2 iniciou sua rota e vice-versa. A partir dessas ideias Batista (2016, não publicado) propõem uma nova formulação para o modelo.

Parâmetros:

$G = (X, E)$: Grafo não direcionado;

X : conjunto de vértices (nós);

E : Conjunto de arestas;

$[i, j]$: aresta formado pelos pontos i e j ;

c_{ij} : custo associado a aresta $[i, j]$;

PU_{ij} : punição associada ao atraso da aresta $[i, j]$;

H : horizonte de tempo;

np : número de períodos de H ;

K : conjunto de veículos;

MP_{ij} : periodicidade do arco $[i, j]$;

S : subconjunto de arestas $[i, j]$ com $MP_{ij} < np$;

R : subconjunto de arestas $[i, j]$ com $MP_{ij} = np$.

As variáveis x_{ijkp} , p_{ijp} e f_{ikp} continuam a ser utilizadas na nova formulação e uma nova variável inteira (y_i) é acrescentada ao modelo em que:

$$y_i \quad \left\{ \begin{array}{l} \text{Contabiliza a quantidade de carros que sai do ponto } i \text{ no dia } 1 \text{ e chega} \\ \text{no ponto } i \text{ no dia } np. \end{array} \right.$$

Modelo:

$$\min Z = \sum_{[i,j] \in E} \sum_{k=1}^{nk} \sum_{p=1}^{np} c_{ij} * x_{ijkp} + \sum_{[i,j] \in E} \sum_{p=1}^{np} PU_{ij} * p_{ijp} \quad (31)$$

Sujeito à:

$$\sum_{[i,j] \in E} x_{ijkp} + f_{jkp} = \sum_{[i,j] \in E} x_{jik,p+1} + f_{jk,p+1} \quad \forall j \in X, \forall k \in K, p = 1, \dots, np - 1 \quad (32)$$

$$\sum_{[i,j] \in E} x_{ijk1} + f_{ik1} = y_i \quad \forall i \quad (33)$$

$$\sum_{[i,j] \in E} x_{ijk,np} + f_{jk,np} = y_j \quad \forall j \quad (34)$$

$$\sum_{[i,j] \in E} x_{ijkp} + \sum_{[i,j] \in E} x_{jikp} + \sum_{i=1}^n f_{ikp} = 1 \quad \forall p \in H, \forall k \in K \quad (35)$$

$$\sum_{k=1}^{nk} \left(x_{ijkp} + x_{jikp} + x_{ijk,p+1} + x_{jik,p+1} + \dots \right) + p_{ijp} \geq 1 \quad (36)$$

$$\forall [i, j] \in S, \forall p \in H$$

$$\sum_{k=1}^{nk} \sum_{p=1}^{np} (x_{ijkp} + x_{jikp}) \geq 1 \quad \forall [i, j] \in R \quad (37)$$

$$x_{ijkp}, p_{ijp}, f_{ikp} \in \{0,1\} \quad \forall [i, j] \in E, \forall k \in K, \forall p \in H$$

$$y_i \in N^+ \quad (38)$$

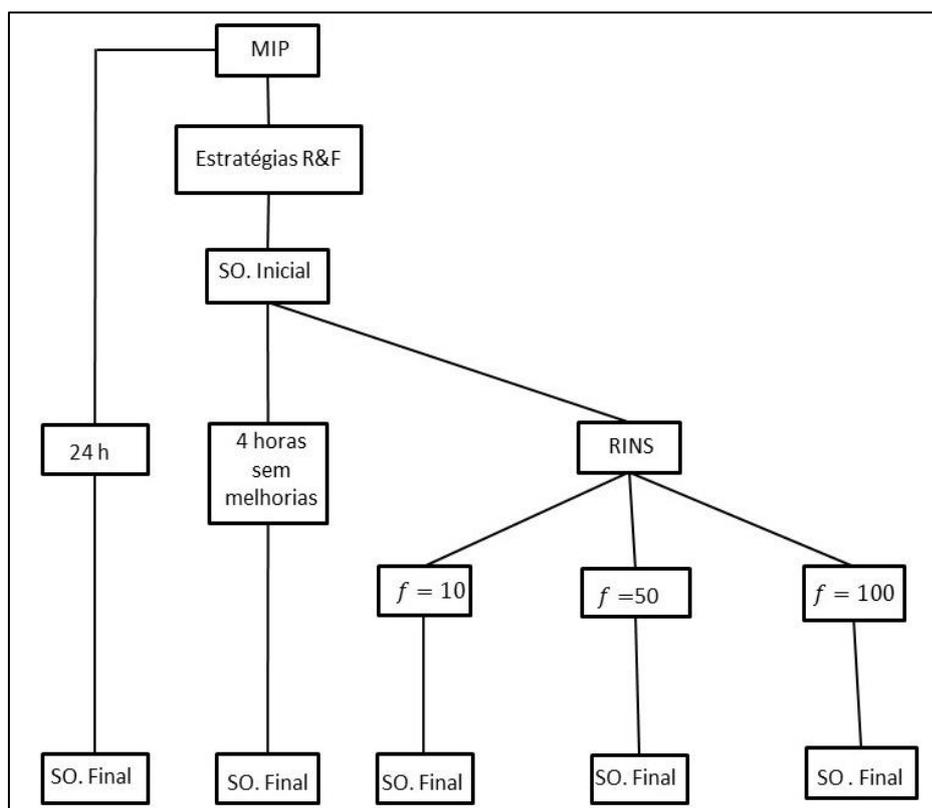
A função objetivo (31) permanece a mesma, a restrição (32) que garante o fluxo agora não necessita contabilizar o dia $np + 1$. As restrições (33) e (34) são restrições novas, que asseguram que a quantidade de veículos que sai de um vértice no dia 1, será igual a quantidade de veículos que retornam nesse vértice no último dia do horizonte de tempo. Restrição (35) garante que cada veículo terá uma designação para cada dia.

A restrição (28) do modelo de 2014 se tornou a restrição (36) e (37), a restrição (37) resume a restrição (30). A restrição que garante que os arcos com periodicidades igual a 1 passagem no horizonte de tempo sejam atendidos é dado na restrição (37). E por fim a restrição (38) garante que as variáveis x_{ijkp} , p_{ijp} e f_{ikp} sejam binárias e a variável y_i inteira positiva.

4 METODOLOGIA

Nesse capítulo será descrito o método de solução utilizado para os modelos propostos por Batista (2014, 2016, não publicado), assim como, as estratégias e parâmetros adotados para as heurísticas propostas nesse trabalho. A metodologia utilizada para o PCARPMC 2016 é exposta na FIGURA 5.

FIGURA 5 – METODOLOGIA



FONTE: O autor (2017).

A partir do modelo MIP original, foi aplicado dez estratégias diferentes para a heurística *relax-and-fix* (descritas nesse capítulo), visando gerar assim, soluções viáveis para o modelo. As soluções geradas foram comparadas para determinar a solução de melhor qualidade. Essa foi adotada como solução inicial para o problema. Em posse da solução inicial a metodologia foi subdividida em três etapas, onde todas partem da solução inicial encontrada na etapa anterior:

- 1 - O problema foi executado por 1.440 minutos (24 horas);
- 2 - A partir da solução inicial o problema foi executado utilizando a configuração default do solver, tendo como critério de parada 240 minutos (4 horas) caso não tenha melhorias na solução incumbente;

3 - A partir da solução inicial foi aplicado a heurística RINS com três parâmetros diferentes e critério de parada igual ao da etapa dois.

As soluções encontradas nestas etapas foram comparadas, avaliando a melhor estratégia de solução. Após determinar qual foi a melhor estratégia de solução, essa foi aplicada ao modelo proposto em 2014.

4.1 HEURISTICA *RELAX-AND-FIX*

A heurística *relax-and-fix*, como descrita no Capítulo 2, consiste em um procedimento de relaxação e fixação de um determinado conjunto de variáveis. Destaca-se que um dos fatores fundamentais para a eficiência do método está associado à decomposição. Foram propostas duas maneiras distintas de efetuar essa operação, a decomposição por períodos e a decomposição por adjacência dos nós (vértices).

As estratégias são descritas levando em consideração o modelo PCARPMC. Optou-se para o processo de relaxação e fixação, variáveis inteiras e binárias (X_{ijkp} , p_{ijp} , f_{ikp} e y_i). O processo de relaxação das variáveis inteiras consiste em torná-las reais não negativas, enquanto que as binárias, contínuas no intervalo $[0,1]$.

4.1.1 Decomposição por período

Os modelos do Capítulo 3 foram adaptados para serem resolvidos por meio do uso da heurística *relax-and-fix*. Para tanto, as variáveis binárias x_{ijkp} , p_{ijp} e f_{ikp} foram selecionadas para serem decompostas em R subconjuntos disjuntos definidos por Q^r , com $r = 1, \dots, R$. Denota-se por P^r o subproblema que está associado a partição r e $(x_{ijkp}^r, p_{ijp}^r, f_{ikp}^r)$ uma solução para o subproblema P^r , com $2 \leq r \leq R$. Observa-se que R representa tanto o número de períodos do horizonte de tempo, quanto o número de iterações da heurística.

As restrições de (39) a (43) a seguir substituem a restrição (29) e (38) dos modelos originais.

$$x_{ijkp} = x_{ijkp}^r \in \{0,1\} \quad \forall (i,j) \in E, \forall k \in K, \forall p \in \bigcup_{r=1, \dots, r-1} Q^r \quad (39)$$

$$p_{ijp} = p_{ijp}^r \in \{0,1\} \quad \forall (i,j) \in E, \forall k \in K, \forall p \in \bigcup_{r=1,\dots,r-1} Q^r \quad (40)$$

$$f_{ikp} = f_{ikp}^r \in \{0,1\} \quad \forall (i,j) \in E, \forall k \in K, \forall p \in \bigcup_{r=1,\dots,r-1} Q^r \quad (41)$$

$$x_{ijkp}, p_{ijp}, f_{ikp} \in \{0,1\} \quad \forall (i,j) \in E, \forall k \in K, \forall p \in Q^r \quad (42)$$

$$x_{ijkp}, p_{ijp}, f_{ikp} \in [0,1] \quad \forall (i,j) \in E, \forall k \in K, \forall p \in \bigcup_{r=r+1,\dots,R} Q^r \quad (43)$$

Nesta formulação, as restrições (39), (40) e (41) indica que os subproblemas P^r , com $r = 1, \dots, r-1$ são subproblemas que já foram resolvidos e cujas soluções $(x_{ijkp}^r, p_{ijp}^r, f_{ikp}^r)$, $r = 1, \dots, r-1$, permanecerão fixas até a resolução do subproblema P^R . A restrição (42) indica que a r -ésima partição possui suas variáveis definidas como binárias. E por fim, a restrição (43) indica que as variáveis pertencentes as partições Q^r , com $r = r+1, \dots, R$, são variáveis contínuas relaxadas (no intervalo $[0,1]$) na resolução do r -ésimo subproblema.

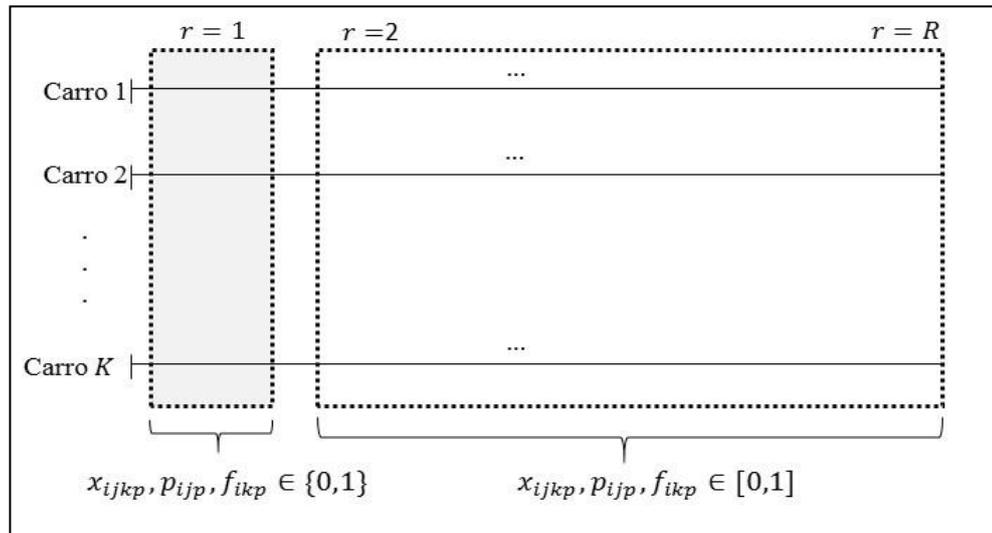
Para verificar o desempenho da heurística *relax-and-fix* como metodologia de solução para o PCARP, foram realizadas 4 estratégias distintas de decomposição das variáveis, considerando horizonte de tempo H dos modelos como estratégia de decomposição.

4.1.1.1 Estratégia *relax-and-fix forward* – R&F-F

Nessa estratégia o problema original foi dividido em R subconjuntos Q^r , com $r = 1, \dots, R$, a serem resolvidos de modo que cada subproblema P^r corresponda a um período/dia r do horizonte de tempo. A ordem da resolução é a cronológica, isto é, iniciando a resolução no subproblema correspondente ao primeiro período do horizonte de tempo e finalizando no subproblema correspondente ao último período subproblema P^R .

A FIGURA 6 ilustra o funcionamento dessa estratégia. Para o primeiro subproblema, as variáveis binárias referentes a $r = 1$, primeiro período do horizonte de tempo, foram mantidas binárias e as mesmas variáveis para os demais períodos foram relaxadas (valores no intervalo $[0,1]$).

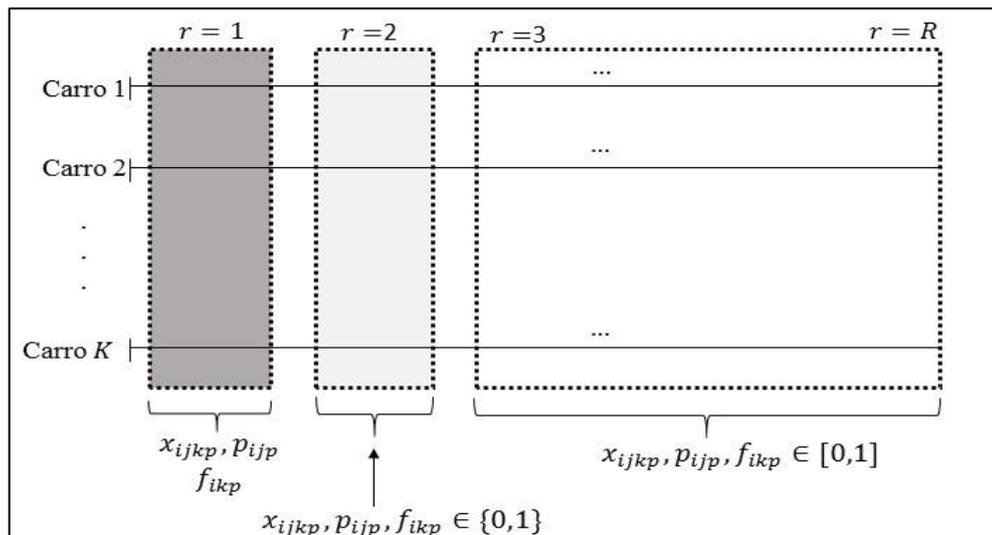
FIGURA 6 – ESTRATÉGIA R&F-F: SUBPROBLEMA INICIAL



FONTE: O autor (2017).

Após a resolução do primeiro subproblema, as variáveis binárias do primeiro período são fixadas, tornam-se parâmetros e as variáveis do segundo período ($r = 2$) são mantidas binárias e as demais para os períodos ($r = 3, \dots, R$) são relaxadas, conforme mostra a FIGURA 7.

FIGURA 7 - ESTRATÉGIA R&F-F: SEGUNDO SUBPROBLEMA

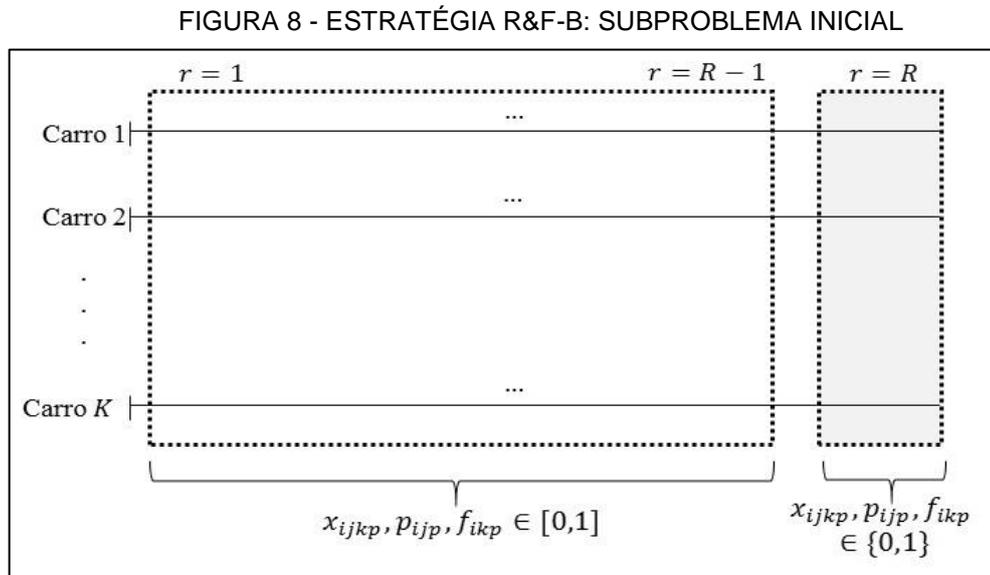


FONTE: O autor (2017)

O processo se repete até que o último subproblema P^R seja resolvido.

4.1.1.2 Estratégia *relax-and-fix backward* – R&F-B

Esta estratégia é similar à estratégia anterior, porém, a sequência de resolução dos subproblemas é iniciada em ordem cronológica inversa e a resolução inicia-se pelo subproblema P^R . A FIGURA 8 ilustra a configuração inicial dessa estratégia. O primeiro subproblema a ser resolvido é o P^R e as variáveis referentes a partição $r = R$ são mantidas binárias, enquanto que, as variáveis das partições $r = 1, \dots, R - 1$ são relaxadas.

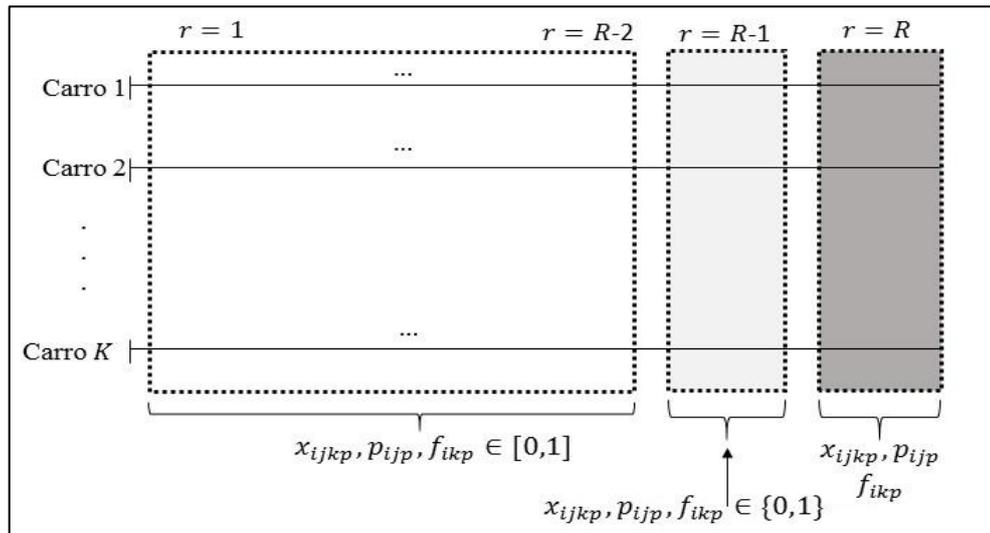


FONTE: O autor (2017).

Na FIGURA 9, os valores das variáveis pertencentes a partição Q^R são mantidas fixadas de acordo com a solução encontrada na resolução do subproblema P^R . Toma-se então as variáveis pertencentes ao subconjunto Q^{r-1} como binárias e as variáveis contidas nos subconjuntos Q^r , com $r = 1, \dots, R - 2$ como relaxadas.

Esse processo se repete até a resolução do subproblema P^1 referente ao primeiro dia do horizonte de tempo.

FIGURA 9 - ESTRATÉGIA R&F-B: SEGUNDO SUBPROBLEMA



FONTE: O autor (2017).

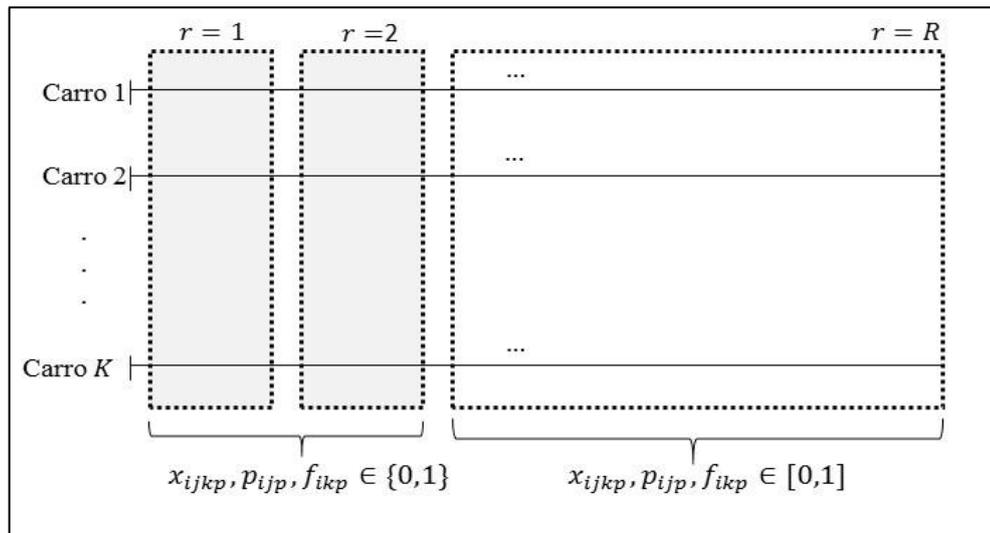
4.1.1.3 Estratégia relax-and-fix overlapping – R&F-O

Esta estratégia baseia-se na partição do problema original em R períodos com sobreposição de partições. Dada uma partição r , as variáveis pertencentes as partições Q^r e Q^{r-1} , com $2 \leq r \leq R$ são mantidas binárias e as demais variáveis subsequentes a esses períodos são relaxadas. Na iteração seguinte, as variáveis da partição Q^{r-1} são fixadas, as variáveis dos subconjunto Q^r e Q^{r+1} mantidas binárias e as demais variáveis relaxadas. O processo ocorre até que o último subproblema seja resolvido.

A FIGURA 10 ilustra a primeira iteração dessa estratégia, no qual o primeiro subproblema a ser resolvido possui as variáveis pertencentes aos subconjuntos Q^1 e Q^2 binárias e para os demais subconjuntos as variáveis são relaxadas.

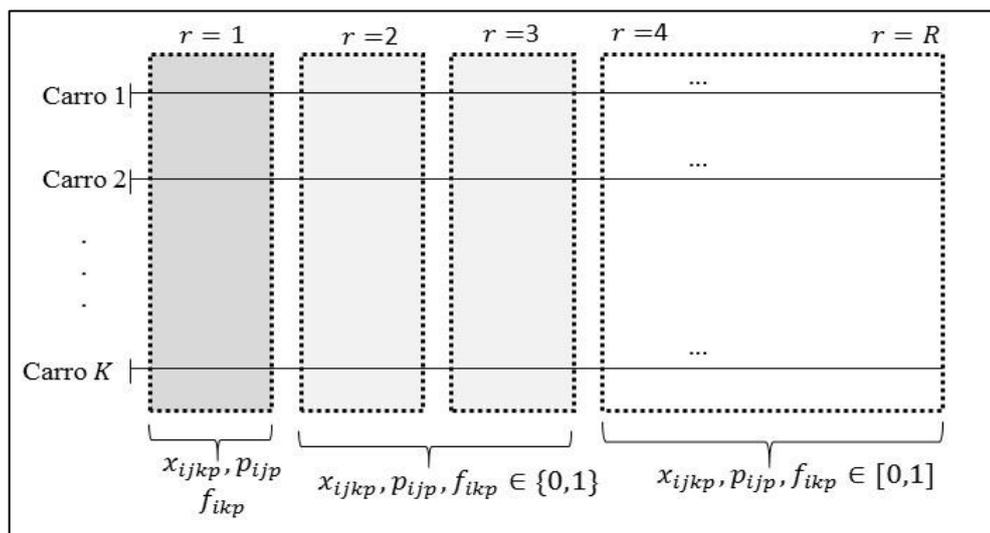
Na segunda iteração da heurística, as variáveis binárias do primeiro subconjunto Q^1 são fixadas (tornando-as parâmetros), as variáveis pertencentes a segundo e terceira partição ($r = 2$ e $r = 3$) são mantidas binárias e as demais variáveis são relaxadas, conforme ilustra FIGURA 11.

FIGURA 10 - ESTRATÉGIA R&F-O: SUBPROBLEMA INICIAL



FONTE: O autor (2017).

FIGURA 11 - ESTRATÉGIA R&F-O: SEGUNDO SUBPROBLEMA



FONTE: O autor (2017).

A heurística avança em ordem cronológica, utilizando o procedimento de sobreposição até a resolução do último subproblema.

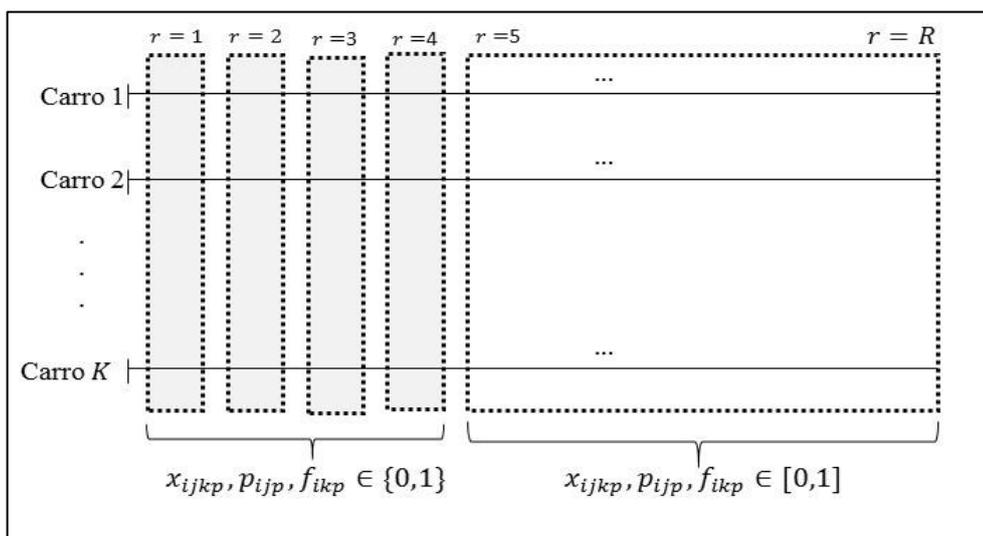
4.1.1.4 Estratégia relax-and-fix overlapping 2 – R&F-O2

A estratégia R&F-O2 é baseada na sobreposição de partições, similar à estratégia anterior. Nesta proposta, cada subproblema é resolvido mantendo as variáveis correspondentes aos períodos $r - 3$, $r - 2$, $r - 1$ e r binárias ($3 \leq r \leq R$) e as demais variáveis dos períodos subsequentes relaxadas. No próximo subproblema, as variáveis correspondentes aos períodos $r - 3$ e $r - 2$ são fixadas,

as variáveis dos períodos $r - 1$, r , $r + 1$ e $r + 2$ são binárias e as demais variáveis subsequentes relaxadas, assim sucessivamente até o último subproblema ser resolvido.

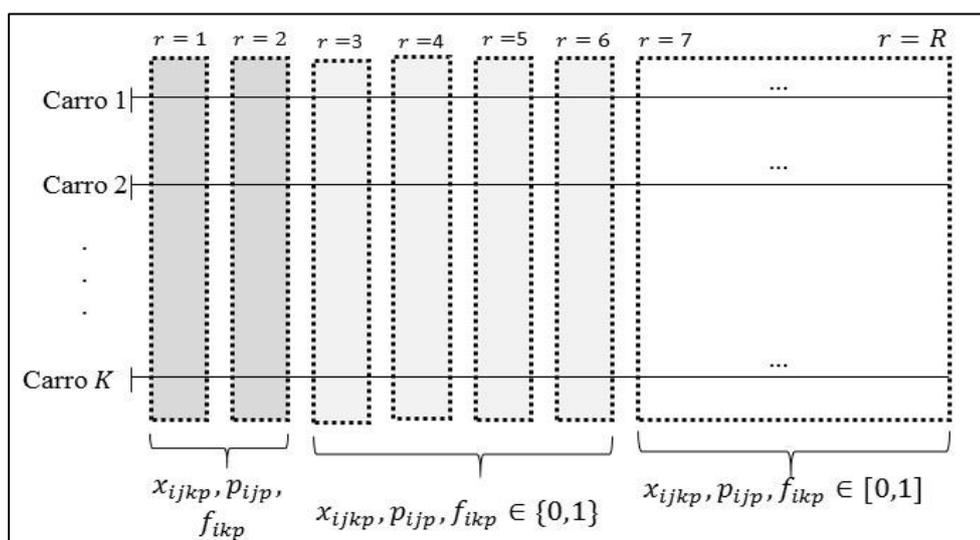
O funcionamento dessa estratégia é ilustrado na FIGURA 12, onde as variáveis subconjuntos Q^1 , Q^2 , Q^3 e Q^4 são binárias e para os demais subconjuntos são relaxadas.

FIGURA 12- ESTRATÉGIA R&F-O2: SUBPROBLEMA INICIAL



FONTE: O autor (2017).

FIGURA 13 - ESTRATÉGIA R&F-O2: SEGUNDO SUBPROBLEMA



FONTE: O autor (2017)

Após a resolução desse primeiro subproblema, as variáveis dos subconjuntos Q^1 e Q^2 tornam-se parâmetros (fixadas), as variáveis pertencentes a Q^3 , Q^4 , Q^5 e Q^6

binárias e para os demais subconjuntos as variáveis são relaxadas, como ilustrado na FIGURA 13.

A heurística prossegue fixando as variáveis de cada subproblema até que o último seja resolvido, caso todos os subproblemas anteriores sejam factíveis, caso contrário a heurística não foi capaz de determinar uma solução viável para o modelo original.

4.1.2 Decomposição por nós adjacentes: estratégia *relax-and-fix* nós adjacentes – R&F-NA

Essa estratégia de decomposição por nós foi baseada em adjacência ou vizinhança de vértices (nós). Para tanto, vamos definir alguns conceitos referentes a vizinhança da teoria de grafos.

Se (i, j) é uma aresta, então os vértices i e j são os extremos desta aresta. Dizemos também que i e j são adjacentes ou vizinhos. Denota-se por $N(i)$ o conjunto de vértices pertencentes a vizinhança de i e $N[i] = N(i) \cup \{i\}$ como vizinhança fechada de i .

O modelo descrito no capítulo 3 baseia-se em um grafo não direcionado $G = (X, E)$. Para inicializar a heurística R&F-NA, escolhe-se um nó $i^* \in X$ para ser o nó raiz (nó de origem), as variáveis binárias x_{ijkp} , p_{ijp} , f_{ikp} e y_i são decompostas em dois subconjuntos disjuntos na primeira iteração. Sendo $N[i^*]$ o conjunto formado pela vizinhança fechada de i^* , as variáveis x_{ijkp} , p_{ijp} , f_{ikp} e y_i são mantidas binárias para todos os nós pertencentes a $N[i^*]$ e as variáveis para os demais nós pertencentes a $X - \{i^*\}$ são relaxadas.

A partir da segunda iteração, as variáveis x_{ijkp} , p_{ijp} , f_{ikp} e y_i são decompostas em três subconjuntos disjuntos.

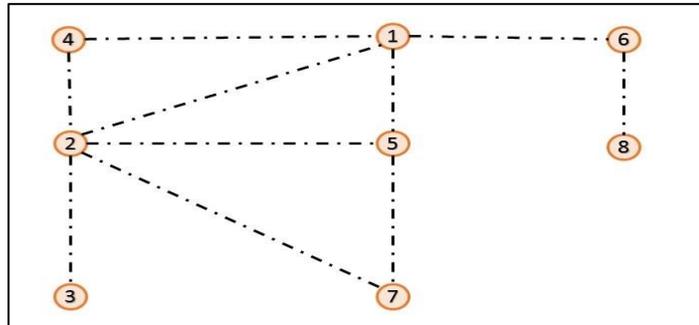
Primeiro subconjunto – formado por nós pertencentes à vizinhança fechada $N[i^*]$. Todas as variáveis que possuem relação com esses nós terão seus valores fixados de acordo com a solução do subproblema da iteração anterior;

Segundo subconjunto – Formado pela vizinhança fechada $N[N(i^*)]$. As variáveis serão mantidas binárias para todos os nós pertencentes a essa vizinhança;

Terceiro subconjunto – Formado por nós pertencente ao subconjunto $X - N[N(i^*)]$. As variáveis serão relaxadas para todos os nós pertencentes a este conjunto.

Para ilustrar o funcionamento dessa estratégia, considera-se o grafo não direcionado $G = (X, E)$ com $|X| = 8$ e $|E| = 10$, representado na FIGURA 14.

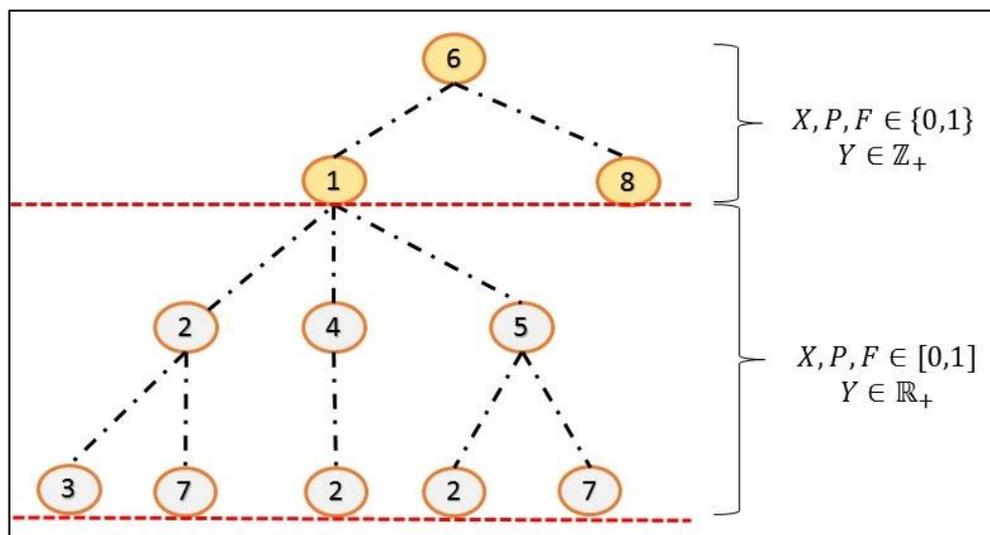
FIGURA 14 - GRAFO



FONTE: O autor (2017).

As FIGURA 15, 16 e 17 ilustram o esquema de funcionamento dessa estratégia. A FIGURA 15 ilustra a configuração inicial, onde foi adotado o nó 6 como nó raiz. Para esse primeiro subproblema, as variáveis x_{ijkp} , p_{ijp} , f_{ikp} e y_i são mantidas binárias para todos os nós pertencentes à vizinhança fechada $N[6]$ e relaxadas para todos os nós pertencentes a $X - \{6\}$.

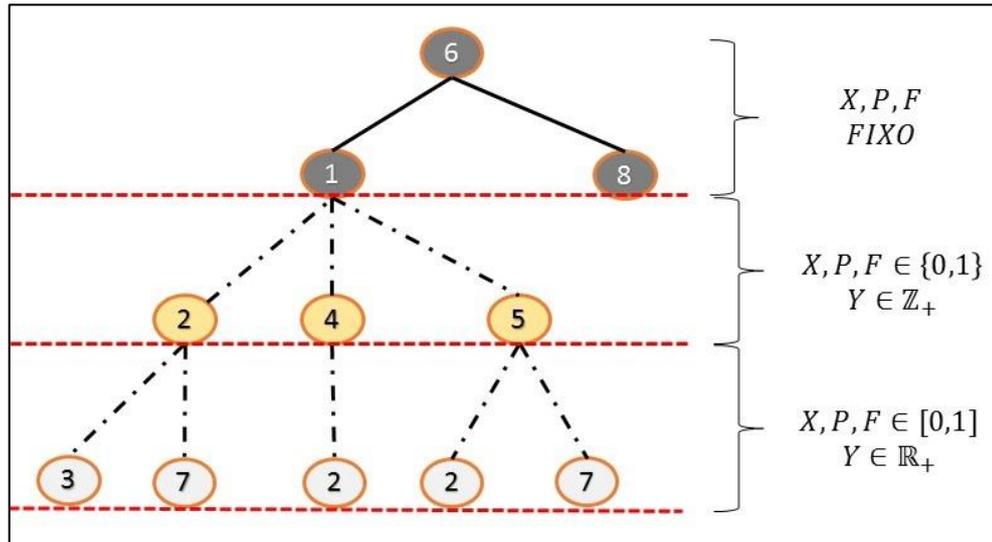
FIGURA 15 – CONFIGURAÇÃO INICIAL R&F-NA



FONTE: O autor (2017).

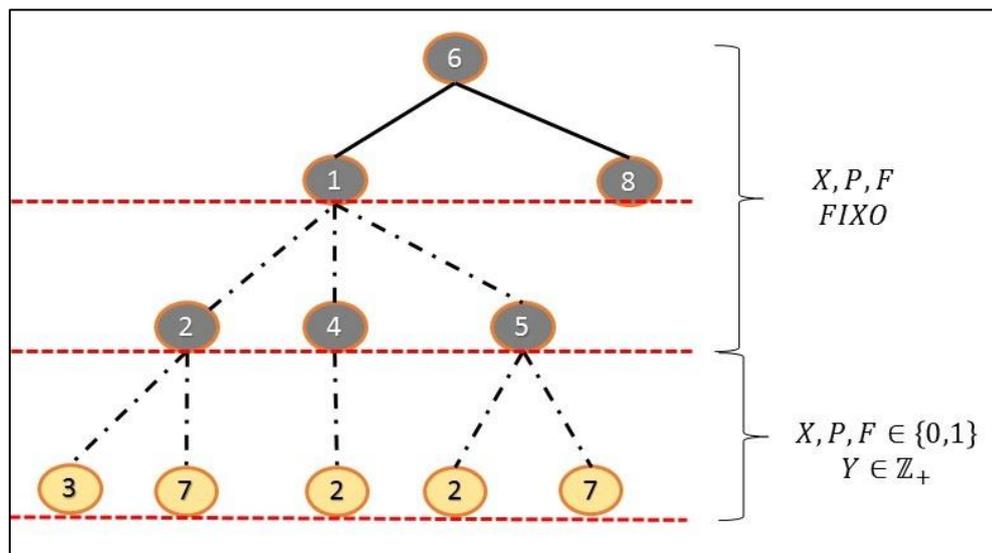
Na segunda iteração tem-se o primeiro subproblema resolvido. As variáveis referentes aos nós pertencentes a $N[6] = \{1, 6, 8\}$ são fixadas. Os nós pertencentes aos subconjuntos $N[1] = \{1, 2, 4, 5\}$ são mantidas binárias e relaxadas para os nós contidos em $X - N[N(6)]$, FIGURA 16.

FIGURA 16 – SEGUNDA ITERAÇÃO R&F-NA



FONTE: O autor (2017).

FIGURA 17 – TERCEIRA ITERAÇÃO R&F-NA



FONTE: O autor (2017).

Na terceira e última iteração para esse exemplo FIGURA 17, as variáveis referentes aos nós pertencentes a $N[6] \cup N[1] = (1, 8, 2, 4, 5 \text{ e } 6)$ são fixadas. Para os nós pertencentes a $N[2] \cup N[4] \cup N[5]$ as variáveis são mantidas binárias. No final do procedimento, caso nenhum subproblema gerado seja infactível encontra-se uma solução viável para o problema original. Em caso de um subproblema gerado ser infactível a heurística falhou, não sendo capaz de determinar uma solução viável.

Para a estratégia de decomposição em nós adjacentes não é possível determinar o número de iterações a priori para a heurística, uma vez que esse número está vinculado com o nó que é adotado como nó raiz. No exemplo, iniciando

pelo nó raiz 6 tem-se três iterações, caso inicialize tomando o nó 3 como raiz tem-se 4 iterações.

Como pode ser observado nessa estratégia optou-se pela busca em largura, onde dado um grafo G e um vértice de origem distinta i^* , a busca consiste em explorar sistematicamente as arestas de G até que todos os vértices de G acessíveis a partir de i^* sejam “descobertos” (CORMEM, 2002). Outros tipos de buscas podem ser desenvolvidas para a estratégia R&F-NA.

4.2 HEURÍSTICA *ENHANCED RELAX-AND-FIX*

Apresenta-se uma adaptação baseada no algoritmo *Enhanced Relax-and-fix* (E-R&F) propostas por Escudero e Salmeron (2005). Esse algoritmo ao detectar infactibilidade na resolução de um determinado subproblema, tem o potencial de retroceder um ou mais passos.

Ao retroceder um passo o algoritmo agrupa dois ou mais subproblemas em um único. Observa-se que ao realizar essa iteração a abordagem do subproblema tende a se aproximar do problema original. O algoritmo da heurística E-R&F é descrita no QUADRO 5.

QUADRO 5 - ALGORITMO *ENHANCED RELAX-AND-FIX*

(Continua)

Inicialização – defina uma partição para as variáveis inteiras ou binárias criando R subconjuntos, $Q^r, r = 1, \dots, R$.

Para $r = 1, \dots, R$ **faça**

Passo 1 – Relaxe as variáveis inteiras/binárias para os subconjuntos $Q^i, i = r + 1, \dots, R$;

Mantenha a integralidade das variáveis do subconjunto Q^r ;

Fixe as variáveis do subconjunto $Q - (Q^r \cup Q^{r+1} \cup \dots \cup Q^{R-1})$ se $r \geq 2$.

Se $r = 1$

– Resolva o subproblema gerado e retorne ao passo 1.

Senão

– Resolva o subproblema resultante do passo 1.

Se o problema for infactível então:

Para $aux = (r-1), \dots, 1$ **faça**

Mantenha relaxado os subconjuntos $Q^i, i = r + 1, \dots, R$;

Mantenha a integralidade das variáveis do subconjunto $Q^i, i = aux, \dots, r$.

Fixe as variáveis dos subconjuntos $Q^i, i = 1, \dots, (aux - 1)$.

Resolva o subproblema gerado.

(Conclusão)

Se o subproblema for factível **Exit Para.****Fim Para****Fim Para**

FONTE: O autor (2017)

Observa-se que na pior das hipóteses o E-R&F resolve o problema original, algo que pode ser comum para instâncias grandes e de difícil solução (ESCUDERO; SALMERON, 2005).

4.3 HEURÍSTICA RINS

A heurística RINS é uma estratégia de intensificação na busca por soluções viáveis de melhor qualidade. Para intensificar essa busca a heurística depende de dois parâmetros: f que determina a cada quantos nós da árvore de *branch-and-cut* a RINS deve ser executada e nl que indica quantos nós devem ser explorados no sub-MIP gerado.

Esses dois parâmetros necessitam de definição para a estratégia de busca, uma vez que, a execução da RINS em dois nós consecutivos geram vizinhanças muito similares e, ao adotar nl com um valor muito pequeno, pode não ser possível gerar melhorias na solução incumbente. Então, com o objetivo de melhorar o desempenho da heurística Danna, Rothberg e Pape (2005) propõem a execução da RINS a cada $1 \leq f \leq 100$ e com limite de $nl = 1.000$ nós. Nesse trabalho foram testados três diferentes valores para f sendo 10, 50 e 100 e nl fixo para todos os parâmetros anteriores em 1.000.

5 TESTES COMPUTACIONAIS

As estratégias propostas para a heurística *relax-and-fix* para a resolução dos modelos utilizados neste trabalho foram implementadas no Microsoft Visual Studio 2015, utilizando pacote de otimização Gurobi 7.0. Os testes foram realizados utilizando um computador com processador Intel Core i7-5500U com velocidade de 2.4GHz, memória de RAM de 8GB sob o sistema operacional Windows 10.

Os experimentos computacionais foram divididos em quatro conjuntos de testes. No primeiro conjunto, foram realizados testes com a heurística *relax-and-fix* clássica utilizando as estratégias descritas no capítulo 4 para o modelo proposto em Batista (2016). No segundo conjunto de testes foram descritos os resultados para heurística *Enhanced Relax-and-Fix* adaptada para as estratégias propostas para o modelo Batista (2016). No terceiro conjunto de testes foi aplicado a heurística RINS tendo como solução inicial a melhor solução encontrada nos testes anteriores. E por fim com base nos testes anteriores a melhor estratégia foi aplicada ao modelo proposto em Batista (2014).

5.1 CRITÉRIO DE PARADA

Os critérios de parada determinam o momento em que o processo deve ser encerrado. Geralmente são definidos por um certo número de iterações, por um tempo pré-determinado de execução ou pela diferença percentual (GAP) relativa entre a solução incumbente e o limitante inferior (ou superior, dependendo da função objetivo) da solução atual.

Nesse trabalho, adotou-se como critério de parada o tempo de execução. Para a primeira etapa de testes, referentes a execução da heurística *relax-and-fix*, foi adotado um tempo de 1 minuto por iteração para as estratégias *relax-and-fix forward* (R&F-F), *relax-and-fix backward* (R&F-B), *relax-and-fix overlapping* (R&F-O) e *relax-and-fix overlapping 2* (R&F-O2) e suas respectivas adaptações para a heurística *enchaced* (E-R&F). Para a estratégia baseada na decomposição por nós adjacentes, *relax-and-fix nós adjacentes* (R&F-NA) e sua adaptação para *Enchaced* (E-R&F-NA), optou-se por 10 minutos por iteração.

Nos testes, referentes à aplicação da heurística RINS, foi adotado um tempo total de 240 minutos (4 horas) caso não ocorresse melhorias na solução incumbente

como critério de parada. Esse tempo foi determinado pela média dos maiores tempos entre duas melhorias consecutivas na solução incumbente para cada uma das 23 instâncias, essas executadas na configuração *default* do *solver*, com tempo limite de 1.440 minutos (24 horas).

Para cada instância foi calculado o desvio entre a solução obtida pela heurística R&F (Z_H) e a melhor solução fornecida pelo *software* Gurobi 7.0 (Z_G) com um tempo limite de 1.440 minutos (24 horas). Este desvio é calculado da seguinte forma:

$$Desvio = \frac{Z_H - Z_G}{Z_G} \quad (44)$$

5.2 INSTÂNCIAS

Para avaliar as estratégias propostas nesse trabalho foram utilizadas as 23 instâncias proposta por Batista (2014), sendo estas adaptadas de Golden, Dearmon e Baker (1983). As características para cada uma das instâncias estão dispostas na TABELA 1.

TABELA 1- CARACTERÍSTICAS DOS 23 PROBLEMAS

Problema	Nós	Arestas	Períod.	Carros	Neces.	Dias
gdb1	12	22	2	1	32	38
gdb2	12	26	3	1	44	53
gdb3	12	22	2	2	31	19
gdb4	11	19	3	2	34	21
gdb5	13	26	4	2	53	32
gdb6	12	22	2	3	32	14
gdb7	12	22	3	3	37	45
gdb8	27	46	4	3	67	27
gdb9	27	51	3	4	78	24
gdb10	12	25	4	4	70	21
gdb11	22	45	2	1	50	60
gdb12	13	23	3	1	33	40
gdb13	10	28	2	2	33	20
gdb14	7	21	3	2	32	20
gdb15	7	21	4	2	48	29
gdb16	8	28	2	3	40	24
gdb17	8	28	3	3	52	21
gdb18	9	36	4	3	81	33
gdb19	8	11	3	4	29	12
gdb20	11	22	4	4	53	16
gdb21	11	33	2	5	46	12
gdb22	11	44	4	5	88	22
gdb23	11	55	2	6	82	17

FONTE: Batista (2014).

Observa-se que as instâncias possuem periodicidades que variam de 2 a 4, com o número de veículos entre 1 e 6. A coluna que indica a necessidade de cada problema refere-se à soma das necessidades de todos os arcos. O horizonte de tempo é definido pelo número de necessidades multiplicado por 1,2 arredondando para cima (BATISTA, 2016, não publicado). As instâncias propostas tornam os problemas complexos de serem resolvidos, fato esperado devido à natureza do problema *NP-hard*.

5.3 RELAX-AND-FIX CLÁSSICO APLICADO AO MODELO BATISTA (2016)

As tabelas apresentadas nessa seção são compostas por: primeira coluna referente à instância e as colunas subsequentes referentes à heurística, sendo que essas são subdivididas em colunas de solução (Z) e desvio em relação à solução do modelo executado no solver por 24h. O símbolo “-” indica que a estratégia gerou subproblemas ineficazes, não sendo capaz de determinar uma solução viável para o problema original, enquanto que, o símbolo “*” indica que o problema esgotou o tempo estabelecido em uma determinada iteração.

A coluna referente à estratégia RF-NA apresenta a melhor solução encontrada na execução da heurística. A estratégia foi inicializada n vezes (n = número de nós da instância), considerando a cada execução um nó distinto como nó raiz. Os tempos apresentados nas tabelas são a soma dos tempos gasto em cada execução. A última coluna refere-se aos valores da função objetivo de cada instância, esses valores foram determinados pelo *software* GUROBI 7.0 utilizando a configuração *default* do solver, com um tempo limite de 1.440 minutos (24 horas). O solver foi capaz de determinar a solução ótima para duas instâncias, gdb1 e gdb14.

A TABELA 2 apresenta os resultados computacionais relativos a aplicação das heurísticas, *relax-and-fix forward* (R&F-F), *relax-and-fix backward* (R&F-B), *relax-and-fix overlapping* (R&F-O), *relax-and-fix overlapping 2* (R&F-O2) e *relax-and-fix nós adjacentes* (R&F-NA).

TABELA 2 – SOLUÇÕES ENCONTRADAS COM A APLICAÇÃO DAS ESTRATÉGIAS

Problema	R&F-F		R&F-B		R&F-O		R&F-O2		R&F-NA		GUROBI
	Z	D	Z	D	Z	D	Z	D	Z	D	Z
gdb1	-		-		-		-		27440	59,84	451 ¹
gdb2	-		-		-		-		596	0,01	592
gdb3	-		-		-		-		348	0,02	340
gdb4	-		-		-		7562	2,00	7521	1,99	2519
gdb5	-		-		-		-		7747	-0,56	17733
gdb6	-		-		456	0	-		474	0,04	456
gdb7	-		-		-		11508	21,65	8502	15,74	508
gdb8	*		*		*		*		-		19363
gdb9	*		*		*		*		-		24465
gdb10	-		13754	17,46	-		-		-		745
gdb11	*		*		*		*		458	-0,002	459
gdb12	-		-		-		-		12498	22,72	527
gdb13	-		642	0,03	620	0	620	0	620	0	620
gdb14	-		147	0,01	-		145¹	0	145¹	0	145 ¹
gdb15	-		-		-		146	0,04	140	0	140
gdb16	-		192	0,05	182	0	182	0	182	0	182
gdb17	-		1178	6,01	168	0	-		168	0	168
gdb18	-		-		-		*		396	0,03	385
gdb19	2135	16,08	2145	16,16	125	0	127	0,02	127	0,02	125
gdb20	-		-		-		6280	21,51	267	-0,04	279
gdb21	-		-		-		214	0,01	212	0	212
gdb22	-		*		*		*		431	0,01	426
gdb23	-		340	0,01	-		21375	62,62	337	0,003	336

FONTE: O autor (2017).

NOTA: ¹Solução ótima.

A estratégia R&F-F apresentou o pior desempenho em relação às demais, conseguindo encontrar apenas uma solução factível e de baixa qualidade. Esgotou o tempo estabelecido para 3 problemas e, para os demais, gerou subproblemas infactíveis. A estratégia R&F-B esgotou o tempo limite de execução em 4 instâncias, porém, apresentou desempenho superior à primeira estratégia conseguindo resolver 7 das 23 instâncias, sendo 0,01 o menor desvio e 17,46 o pior desvio em relação a melhor solução. A média do desvio da heurística foi de 5,68.

A estratégia R&F-O atingiu performance melhor que as duas estratégias anteriores, alcançando 5 soluções viáveis com desvio igual a 0, sendo que 4 instâncias exauriram o tempo estabelecido por iteração. A estratégia R&F-O2 obteve 10 soluções viáveis, apresentando desvio igual a 0 para os problemas gdb13, gdb14 e gdb16, enquanto os problemas gdb15, gdb19 e gdb21 apresentaram desvio igual ou inferiores a 0,04, indicando que as soluções encontradas estão muito próximas da solução determinada pelo solver. A heurística apresenta média do desvio de 10,79. Observa-se que 5 problemas esgotaram o tempo estabelecido, não sendo capaz de determinar uma solução viável.

A estratégia R&F-NA apresenta o melhor desempenho quando comparada com as demais, não sendo capaz de determinar solução viável para apenas 3 instâncias (gdb8, gdb9 e gdb10). As instâncias gdb8 e gdb9 são de difícil resolução, pois apresentam uma quantidade elevada de vértices, arestas e necessidades. É possível evidenciar ainda que a solução é de baixa qualidade mesmo no uso do GUROBI com tempo limite de 24 horas.

A estratégia determinou duas soluções viáveis (gdb5 e gdb20) de melhor qualidade que a resolução pelo método exato. Obteve-se ao todo 20 soluções viáveis, sendo que dessas os problemas gdb13, gdb14, gdb15, gdb16, gdb17 e gdb21 apresentam desvio igual a 0, já os problemas gdb2, gdb3, gdb6, gdb11, gdb18, gdb19, gdb22 e gdb23 apresentaram desvio igual ou inferior a 0,04, ficando muito próximas da solução encontrada pelo método exato. O desvio médio da heurística é de 4,99, valor abaixo das demais estratégias. Observa-se ainda que em nenhum dos problemas esgotou-se o tempo limite por iteração.

A TABELA 3 apresenta os tempos em minutos gastos na execução de cada estratégia.

TABELA 3 – TEMPO DE EXECUÇÃO DAS ESTRATÉGIAS *RELAX-AND-FIX*

(Continua)

Problema	R&F-F	R&F-B	R&F-O	R&F-O2	R&F-NA	GUROBI
	t (min)					
gdb1	-	-	-	-	51,65	953,28
gdb2	-	-	-	-	150,26	1440
gdb3	-	-	-	-	117,55	1440
gdb4	-	-	-	8,98	93,47	1440
gdb5	-	-	-	-	221,15	1440
gdb6	-	-	4,33	-	63,30	1440

(Conclusão)						
gdb7	-	-	-	5,90	152,08	1440
gdb8	*	*	*	*	-	1440
gdb9	*	*	*	*	-	1440
gdb10	-	6,03	-	-	-	1440
gdb11	*	*	*	*	304,20	1440
gdb12	-	-	-	-	54,92	1440
gdb13	-	0,75	6	2,28	59,43	1440
gdb14	-	0,60	-	3,05	4,42	0,58
gdb15	-	-	-	11,93	125,05	1440
gdb16	-	2,47	11,1	7,25	125,88	1440
gdb17	-	1,67	15,67	-	154,43	1440
gdb18	-	-	-	*	193,22	1440
gdb19	2,05	0,38	0,38	2,53	48,73	1440
gdb20	-	-	-	7,20	178,25	1440
gdb21	-	-	-	5,15	171,23	1440
gdb22	-	*	*	*	201,55	1440
gdb23	-	9,20	-	19,11	270,50	1440
t médio (mim)	2,05	3,01	7,50	7,34	137,06	1418,86

FONTE: O autor (2017).

Observa-se que a resolução pelo *software* GUROBI evidencia, na maioria dos casos, tempo de execução superior às estratégias propostas. Esses resultados já eram previstos, uma vez que as estratégias possuem tempo predefinido (1 minuto por iteração para as decomposições em períodos e 10 minutos para decomposição em adjacência). O método exato conseguiu determinar solução ótima para as instâncias gdb1 e gdb14 com tempos de 953,28 e 0,58 minutos, respectivamente, enquanto a estratégia R&F-NA conseguiu determinar uma solução viável para gdb1 com tempo inferior, mas solução muito distante do ótimo. Já as estratégias R&F-O2 e R&F-NA conseguiram chegar à solução ótima para a instância gdb14, entretanto com tempo superior ao método exato.

A TABELA 4 descreve para cada estratégia quais instâncias obtiveram desvio igual a 0 e inferior ou igual 0,04 em relação a solução determinada pelo método exato e seus respectivos tempos de execução.

TABELA 4 – DESVIO E TEMPOS PARA AS MELHORES ESTRATÉGIAS

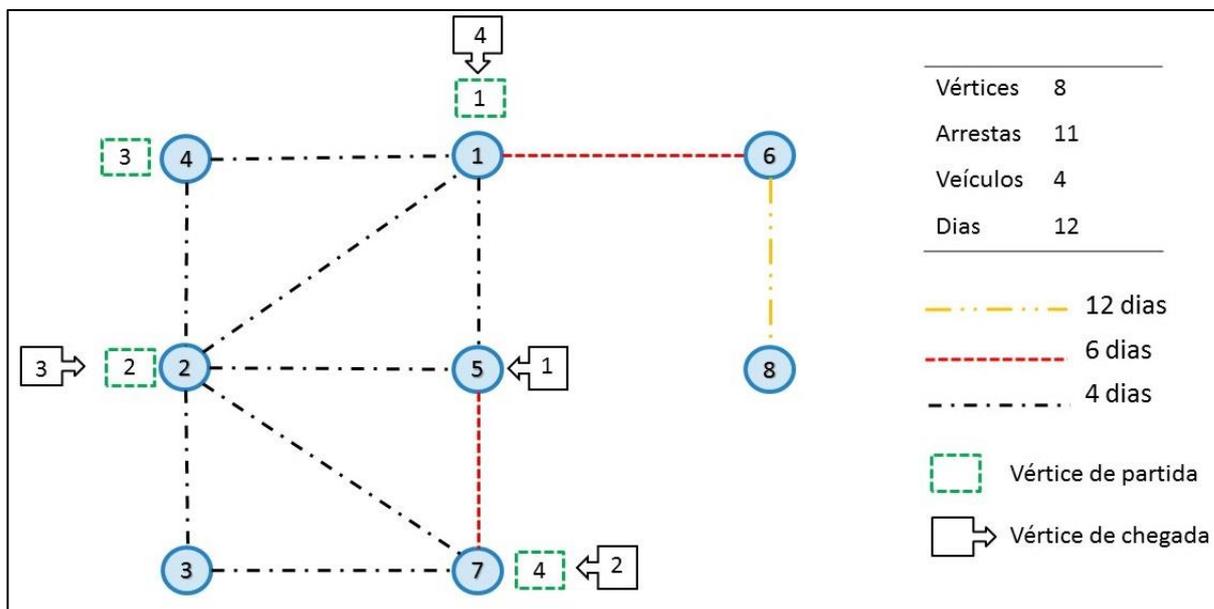
Estratégia	$D = 0$	t (min)	$D \leq 0,04$	t (min)
R&F-B			gdb13	0,75
			gdb14	0,60
			gdb23	9,20
R&F-O	gdb6	4,33		
	gdb13	6		
	gdb16	11,10		
	gdb17	15,67		
	gdb19	0,38		
R&F-O2	gdb13	2,28	gdb15	11,93
	gdb14	3,05	gdb19	2,53
	gdb16	7,25	gdb21	7,20
R&F-NA	gdb5	221,15	gdb2	150,65
	gdb11	304,20	gdb3	117,55
	gdb13	59,43	gdb6	63,30
	gdb14	4,42	gdb18	193,22
	gdb15	125,05	gdb19	48,73
	gdb16	125,88	gdb22	201,55
	gdb17	154,43	gdb23	270,50
	gdb20	178,25		
	gdb21	171,23		

FONTE: O autor (2017).

É possível avaliar a qualidade das heurísticas no que concerne ao tempo. Observa-se que as estratégias R&F-B, R&F-O e R&F-O2 apresentam os menores tempos. Em contrapartida, estas possuem o maior número de problemas inactíveis e soluções com desvio mais elevado. O menor tempo de execução é de 0,38 min (23 segundos) referente à resolução do problema gdb19 pela estratégia R&F-O e o maior tempo é de 304,2 min obtido pela estratégia R&F-NA na solução do problema gdb11, o qual possui solução melhor que o problema resolvido pelo método exato. Destaca-se ainda que o maior tempo atingido pela heurística é 4,7 vezes menor que o tempo utilizado pelo solver.

A inactibilidade ocorreu com grande frequência nas estratégias em que a decomposição por período foi realizada, conforme observado na TABELA 2. Para compreender o elevado número de problemas inactíveis será analisada a instância gdb19 resolvida pela estratégia R&F-F. A FIGURA 18 ilustra a configuração da solução do subproblema 10 na resolução da instância gdb19.

FIGURA 18 – CONFIGURAÇÃO DA SOLUÇÃO SUBPROBLEMA 10 DA INSTÂNCIA GDB19



FONTE: O autor (2017).

Nota-se que a instância possui arestas com três periodicidades diferentes, que devem ser atendidas a cada 4, 6 e 12 dias com demandas 1, 2 e 3 passagens no horizonte de 12 dias. Observa-se que o veículo 1 inicia sua rota no vértice 1, veículo 2 no vértice 2, veículo 3 no vértice 4 e veículo 4 no vértice 7. O problema original foi decomposto em 12 subproblemas a serem resolvidos pela estratégia R&F-F, ocorrendo infactibilidade na resolução do 11º subproblema. A seguir são descritos os movimentos realizados por cada veículo até o 10º dia do horizonte de tempo.

Carro 1: 1 – 2 – 5 – 5 – 1 – 6 – 6 – 1 – 4 – 2 – 5.

Carro 2: 2 – 2 – 7 – 3 – 2 – 1 – 1 – 5 – 5 – 5 – 7.

Carro 3: 4 – 2 – 2 – 2 – 2 – 2 – 5 – 7 – 7 – 7 – 2.

Carro 4: 7 – 5 – 1 – 1 – 4 – 2 – 7 – 3 – 2 – 1 – 1.

No final do 10º dia, o veículo 1 está no vértice 5, o veículo 2 no vértice 7, o veículo 3 no vértice 2 e o 4º no vértice 1. As arestas 1 – 5 e 3 – 7 necessitam passagens no 11º dia, já as arestas 1 – 4, 1 – 6, 2 – 3, 2 – 4 e 6 – 8 no 12º dia, totalizando 7 necessidades. Considerando a existência de dois dias no horizonte de tempo e um total de 4 veículos, o problema disponibiliza 8 movimentos sobre o grafo que, teoricamente, poderia atender as 7 arestas que ainda exigem passagens com folga de um dia para um veículo.

Observa-se que a aresta 6 – 8 requer atendimento no 12º dia, que poderia ser atendido pelo veículo 4 que se encontra no vértice 1 no 10º dia. Todavia, devido às restrições do modelo, as quais exigem que no final do horizonte de tempo a quantidade de veículos em cada vértice seja igual ao número de veículos que saem do vértice no 1º dia, não é possível satisfazer tais restrições, tornando o problema infactível. Portanto, nesse caso, essas restrições não podem ser atendidas, pois com dois movimentos o veículo 4 não conseguiria voltar ao vértice que marca o início da rota dos veículos.

Além disso, um fator que pode contribuir para a infactibilidade é número de folgas por veículos. Na resolução do 11º subproblema, o veículo 1 havia folgado 2 vezes, o veículo 2 folgado 4 vezes, o veículo 3 folgado 6 vezes e o veículo 4 folgado 2 vezes totalizando 14 folgas até o 10º dia, o que comprometeu a viabilidade de solução do subproblema. A fim de evitar esse comportamento, foi adaptado o algoritmo *Enhanced Relax-and-Fix* que permite à heurística voltar algumas iterações caso ocorra infactibilidade, podendo estruturar de melhor forma as rotas e evitar essa configuração de solução na iteração corrente.

5.4 ENHANCED RELAX-AND-FIX APLICADO AO MODELO BATISTA (2016)

A heurística *Enhanced Relax-and-Fix* foi adaptada para todas as estratégias propostas nesse trabalho, com o objetivo de evitar a infactibilidade que ocorreu com grande frequência em algumas estratégias. Os resultados alcançados estão expostos na TABELA 5.

TABELA 5 – RESULTADOS DA HEURÍSTICA *ENHANCED RELAX-AND-FIX*

(Continua)

Problema	E-R&F-F		E-R&F-B		E-R&F-O		E-R&F-O2		E-R&F-NA		GUROBI
	Z	D	Z	D	Z	D	Z	D	Z	D	Z
gdb1	1451	2,22	*		42440	93,10	50443	110,85	3451	6,65	451 ¹
gdb2	25596	42,24	40572	67,53	56546	94,52	36610	60,84	596	0,01	592
gdb3	367	0,08	20316	58,75	12367	35,37	394	0,16	348	0,02	340
gdb4	17570	5,97	15500	5,15	9555	2,79	7562	2,00	6523	1,59	2519
gdb5	46756	1,64	20720	0,17	123818	5,98	34702	0,96	5795	-0,67	17733
gdb6	469	0,03	11478	24,17	456	0,00	*		474	0,04	456
gdb7	3559	6,01	4589	8,03	5578	9,98	11508		3554	6,00	508
gdb8	*		*		*		*		22355	0,15	19363
gdb9	*		*		*		*		48432	0,98	24465

(Conclusão)											
gdb10	52534		13754	17,46	40791	53,75	58777	77,90	4738	5,36	745
gdb11	*		*		*		*		458	0,00	459
gdb12	25537	47,46	*		33540	62,64	7546	13,32	1507	1,86	527
gdb13	620	0,00	642	0,04	620	0,00	620	0,00	620	0,00	620
gdb14	145	0,00	147	0,01	145	0,00	145	0,00	145	0,00	145 ¹
gdb15	144	0,03	147	0,05	140	0,00	146	0,04	140	0,00	140
gdb16	184	0,01	192	0,05	182	0,00	182	0,00	182	0,00	182
gdb17	169	0,01	1178	6,01	168	0,00	179	0,07	168	0,00	168
gdb18	154450	400,17	6391	15,60	58427	150,76	121399	314,32	396	0,03	385
gdb19	125	0,00	2145	16,16	125	0,00	127	0,02	125	0,00	125
gdb20	8271	28,65	15265	53,71	6286	21,53	6280	21,51	267	-0,04	279
gdb21	217	0,02	218	0,03	212	0,00	214	0,01	212	0,00	212
gdb22	45489	105,78	2465	4,79	71414	166,64	43483	101,07	431	0,01	426
gdb23	34399	101,38	340	0,01	357	0,06	21375	62,62	337	0,00	336

FONTE: O autor (2017).

NOTA: ¹Solução ótima.

Observa-se que aumentou o número de problemas que não conseguiram determinar uma solução, devido ao tempo de iteração, isto é, em um determinado subproblema a heurística não foi capaz de encontrar uma solução viável. Esse fato ocorre devido à estratégia retroceder um número muito grande de iterações fazendo com que o subproblema se aproxime do problema original, ou seja, de difícil resolução.

A estratégia R&F-F na versão clássica apresentou apenas uma solução viável, já na versão E-R&F-F conseguiu determinar 20 soluções viáveis, dessas: 4 são de mesmo valor que o método exato executado por 24 horas e 6 são de valores muito próximo à melhor solução encontrada pelo método exato. A estratégia E-R&F-B teve desempenho inferior à anterior, pois não determinou nenhuma solução viável com desvio igual a zero. Já a heurística E-R&F-O obteve o melhor desempenho entre as estratégias que realizam a decomposição por períodos, encontrando soluções para os problemas gdb6, gdb13, gdb14, gdb15, gdb16, gdb17, gdb19 e gdb21 com desvio igual a 0.

A estratégia E-R&F-NA para os problemas gdb13, gdb14, gdb15, gdb16, gdb17, gdb19, gdb21 e gdb23 apresentaram soluções com desvio igual 0 da melhor solução. Para os problemas gdb5, gdb11 e gdb20 a heurística obteve soluções com desempenho superior ao método exato executado por 24 horas. Soluções com

desvio igual ou abaixo de 0,04 são encontradas para os problemas gdb2, gdb3, gdb6, gdb18 e gdb22.

Com relação ao tempo de execução, houve aumento significativo, o que já era esperado, pois o número de problemas resolvidos cresceu. A TABELA 6 apresenta o tempo gasto em minutos para cada uma das estratégias.

TABELA 6 - TEMPO DE EXECUÇÃO DAS ESTRATÉGIAS *ENHANCED RELAX-AND-FIX*

Problema	E-R&F-F	E-R&F-B	E-R&F-O	E-R&F-O2	E-R&F-NA	GUROBI
	t (min)	t (min)	t (min)	t (min)	t (min)	t (min)
gdb1	28,20		29,83	15,80	114,37	953,28
gdb2	56,82	15,65	57,62	29,72	247,72	1440
gdb3	9,97	1,95	12,40	7,05	113,27	1440
gdb4	14,12	6,23	14,22	8,98	202,95	1440
gdb5	28,65	5,80	35,60	17,72	320,68	1440
gdb6	6,92	1,12	4,33		67,63	1440
gdb7	7,58	3,85	9,30	5,93	225,18	1440
gdb8					806,43	1440
gdb9					957,00	1440
gdb10	18,17	6,03	22,48	13,50	391,07	1440
gdb11					697	1440
gdb12	28,53		32,33	15,52	93,82	1440
gdb13	3,67	0,75	6,00	2,28	65,62	1440
gdb14	2,78	0,60	4,30	3,05	7,77	0,58
gdb15	21,05	5,13	21,03	11,93	121,77	1440
gdb16	11,95	2,47	11,10	7,25	147,98	1440
gdb17	14,28	1,67	15,47	9,73	154,43	1440
gdb18	38,57	15,73	41,02	26,55	180	1440
gdb19	2,05	0,38	1,88	2,53	46,20	1440
gdb20	11,45	2,57	9,88	7,20	255,30	1440
gdb21	3,60	1,90	4,88	5,15	200,13	1440
gdb22	25,07	3,93	25,53	17,78	293,40	1440
gdb23	1,03	9,20	20,17	19,12	220	1440
t médio (mim)	16,72	4,72	18,72	11,94	260,5	1418,86

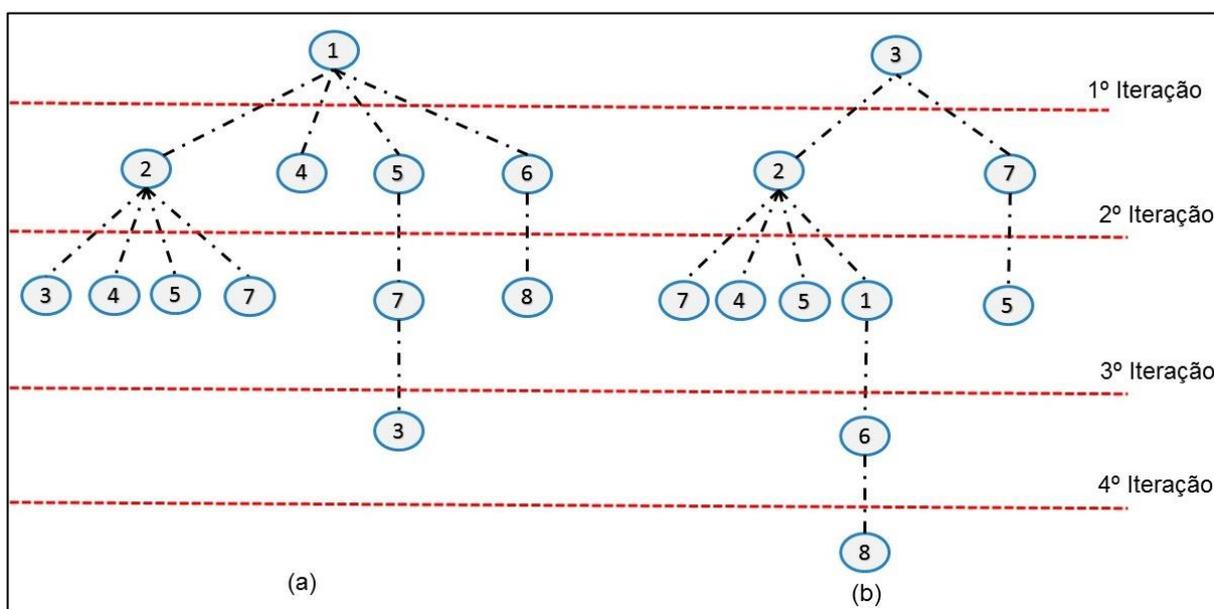
FONTE: O autor (2017).

A estratégia E-R&F-B possui o menor tempo médio, mas é a estratégia que apresenta os maiores desvios em relação às melhores soluções (TABELA 5). A estratégia E-R&F-NA apresenta os maiores tempos de execução (aproximadamente 4 horas de execução por instância), fato devido ao número de problemas resolvidos por essa estratégia, pois a estratégia é executada n vezes (n número de vértices) para cada instância. Isso justifica a diferença em tempo de execução entre alguns

problemas ao serem resolvidos por essa estratégia. Por exemplo, a instância gdb14 teve um tempo de 7,77 minutos ($n = 7$), enquanto que a gdb9, 957 minutos ($n = 27$).

Observa-se ainda que o número de vértices não é o único fator a influenciar no tempo de execução. Outro fator é número de iterações, que pode ser diferente para cada vértice n . Diferentemente da decomposição em períodos, onde o número de iterações é conhecido a priori, dado pelo número de dias do horizonte de tempo, na decomposição por adjacência em nós esse número pode variar dependendo do nó que será adotado como nó raiz. A FIGURA 19 a seguir ilustra esse fato para a instância gdb19. As características do grafo dessa instância podem ser vistas na FIGURA 18.

FIGURA 19 - NÚMERO DE ITERAÇÕES POR NÓ RAIZ



FONTE: O autor (2017).

Na FIGURA 19 (a), ao iniciar a resolução do problema adotando o vértice 1, tem-se um total de 3 iterações. Ao iniciar a resolução da instância gdb19 tomando o vértice 3 como nó raiz a heurística executou 4 iterações, como ilustrado na FIGURA 19 (b). Esse número tem forte influência no tempo, uma vez que, foi adotado um tempo de 10 minutos por iterações. Considerando esse caso, o mesmo problema pode ser resolvido no máximo em 10 ou 20 minutos dependendo do nó que vai ser adotado para dar início à estratégia. Na TABELA 7, estão listados o número médio de iterações por problema e o tempo médio por iteração.

TABELA 7 – MÉDIA DE ITERAÇÃO POR INSTÂNCIA

Problema	Nós	Arestas	Média de Iterações	Tempo Médio por Iteração (min)
gdb1	12	22	3,83	2,49
gdb2	12	26	3,75	5,50
gdb3	12	22	3,92	2,41
gdb4	11	19	3,91	4,72
gdb5	13	26	3,84	6,42
gdb6	12	22	4,25	1,33
gdb7	12	22	3,67	5,11
gdb8	27	46	7,11	4,20
gdb9	27	51	5,07	6,99
gdb10	12	25	3,67	8,88
gdb11	22	45	4,4	7,20
gdb12	13	23	4,15	1,74
gdb13	10	28	3,2	2,05
gdb14	7	21	2	0,55
gdb15	7	21	2	8,70
gdb16	8	28	2	9,25
gdb17	8	28	2	9,65
gdb18	9	36	2	10
gdb19	8	11	4,25	1,36
gdb20	11	22	3,64	6,38
gdb21	11	33	3	6,06
gdb22	11	44	2,82	9,46
gdb23	11	55	2	10

FONTE: O autor (2017).

Nota-se que o número de iterações realizado pela heurística E-R&F-NA está relacionado com o número de arestas que a instância apresenta. Instâncias que apresentam grafos completos possuem o menor número de iterações (gdb14, gdb15, gdb16, gdb17, gdb18 e gdb23). Já instâncias que apresentam menor quantidade de arestas possuem maior número de iterações, como no caso da instância gdb8, que apresentou o maior número de iterações entre todos os problemas.

Além disso, o número de vértices e iterações não são os únicos fatores a influenciar no tempo de resolução dos problemas, pois, por exemplo, as instâncias gdb13 e gdb17 possuem o mesmo número de arestas e iterações, mas apresentam 2,05 e 9,65 minutos por iterações, respectivamente. Logo, ao observar as características das instâncias gdb13 e gdb17 (TABELA 1), verifica-se que a

periodicidade, o número de carros e necessidades diferem, sendo que essas características aumentam a complexidade de resolução dos problemas.

5.5 TESTES HEURÍSTICA RINS

Para algumas instâncias, as estratégias propostas para a heurística *relax-and-fix* não foram capazes de determinar soluções viáveis de boa qualidade, apresentando desvio muito alto em relação a melhor solução encontrada pelo método exato executado por 24 horas. Esse fato ocorreu para as instâncias gdb1, gdb4, gdb7, gdb8, gdb9, gdb10 e gdb12. Objetivando melhorar essas soluções, foi proposto o uso da heurística RINS, cujos parâmetros adotados na aplicação da heurística, assim como o tempo de execução e desvio da melhor solução encontrada pelo método exato são apresentados na TABELA 8. Para todos os casos o parâmetro *nl* foi fixado em 1.000, isto significa que, a cada execução da RINS na árvore de *branch-and-cut*, a heurística explora 1.000 nós do subproblema.

TABELA 8 - RESULTADOS HEURÍSTICA RINS

(Continua)

Problema	R&F Z	$f = 10$		$f = 50$		$f = 100$		Sem RINS		GUROBI Z
		D	t (min)	D	t (min)	D	t (min)	D	t (min)	
gdb1	1451	2,22	240	2,22	240	0,00	382,25	0,00	332	451
gdb2	596	0,01	240	0,01	240	0,01	240	0,01	240	592
gdb3	348	0,00	392,63	0,01	460,37	0,02	240	0,02	240	340
gdb4	6523	0,79	245,62	0,80	253,2	0,80	299,57	0,80	265,55	2519
gdb5	5795	-0,67	240	-0,79	879,62	-0,68	253,73	-0,67	240	17733
gdb6	474	0,00	241	0,00	241	0,00	242	0,00	241	456
gdb7	3554	0,02	439,1	6,00	240	6,00	240	0,08	340,13	508
gdb8	22355	0,15	457	0,15	245	0,15	241	0,15	287	19363
gdb9	48432	0,90	353	0,86	303,75	0,86	530,68	0,86	505,9	24465
gdb10	4738	5,35	277,5	-0,01	524,62	5,35	265,1	5,35	242,5	745
gdb11	458	0,00	240	0,00	240	0,00	240	0,00	240	459
gdb12	1507	1,85	241	1,85	241	1,85	244	1,85	270	527
gdb13	620	0,00	240	0,00	240	0,00	240	0,00	240	620
gdb14	145									145
gdb15	140	0,00	240	0,00	240	0,00	240	0,00	240	140
gdb16	182	0,00	240	0,00	240	0,00	240	0,00	240	182
gdb17	168	0,00	241	0,00	241	0,00	241	0,00	241	168
gdb18	396	0,03	240	0,03	240	0,03	240	0,03	240	385
gdb19	125	0,00	240	0,00	240	0,00	240	0,00	240	125
gdb20	267	-0,04	240	-0,04	240	-0,04	240	-0,04	240	279

										(Conclusão)
gdb21	212	0,00	240	0,00	240	0,00	240	0,00	240	212
gdb22	431	0,01	319	0,01	240	0,01	240	0,01	241	426
gdb23	337	0,00	240	0,00	240	0,00	308,47	0,00	240	336
Média D		0,48	256,95	0,46	285,16	0,61	267,23	0,34	256,73	

FONTE: O autor (2017).

As estratégias desenvolvidas para a heurística R&F encontraram soluções de baixa qualidade para os problemas gdb1, gdb4, gdb7, gdb8, gdb9, gdb10 e gdb12. Ao aplicar a heurística RINS, os problemas gdb1, gdb7 e gdb10 apresentaram melhorias de 68,92%, 84,34% e 85,48%, respectivamente. Já os problemas gdb4, gdb8, gdb9 e gdb12 não apresentaram melhorias significativas. Para os demais problemas, a heurística RINS conseguiu poucas ou nenhuma melhoria. Esse fato ocorre devido à heurística R&F ter encontrado soluções de boa qualidade para os problemas.

Ao considerar a média de desvio como forma de avaliar os diferentes parâmetros aplicados na heurística, conclui-se que a utilização da configuração *default* para o solver é a melhor estratégia, permitindo, deste modo, que o solver determine os melhores momentos de executar a RINS. Com relação ao tempo de execução, os parâmetros aplicados não apresentaram grande diferenças, como pode ser observado pelo tempo médio de execução para cada estratégia.

5.6 TESTE APLICADO AO MODELO BATISTA (2014)

Após aplicar diferentes heurísticas R&F ao problema de Roteamento em Arcos Capacitado e Periódico com Movimento Contínuos (PCARPMC), foi possível verificar que a heurística *Enhanced Relax-and-Fix* juntamente com a estratégia de decomposição de variáveis em nós adjacentes apresentou desempenho superior as demais estratégias propostas. Portanto, nesse último conjunto de testes optou-se pela aplicação apenas da heurística E-R&F-NA no modelo proposto em Batista (2014). Nesse modelo os veículos têm a obrigatoriedade de retornar ao final do horizonte de tempo ao ponto em que iniciaram suas rotas. Optou-se pela aplicação no modelo 2014 uma vez que a mesma estratégia já foi aplicada ao modelo 2016 e o modelo 2014 possui resultados já publicados na literatura, o que facilitou a comparação.

A TABELA 9 apresenta os resultados. Esses resultados são comparados com os valores obtidos por Batista (2014), os quais foram executados no software CPLEX 12.4 (configuração *default*) com um tempo limite para obtenção de uma solução de 24 horas.

TABELA 9 - RESULTADOS E-R&F-NA APLICADO AO MODELO 2014

PROBLEMA	E-R&F-NA			CPLEX	
	Z	D	t (min)	Z	t (min)
gdb1	1456	-0,673	102,41	4450	650,00 ¹
gdb2	3613	-0,689	150,883	11622	1173,72 ¹
gdb3	344	0,012	137,9	340	615,10 ¹
gdb4	2562	-0,279	134,083	3555	826,58 ¹
gdb5	6754	2,859	320,117	1750	1440,00
gdb6	456	0,000	110,217	456	1440,00
gdb7	526	0,015	180,167	518	929,45 ¹
gdb8	23373	0,886	1186,4	12395	1440,00
gdb9	45450	2,378	1146,31	13454	1440,00
gdb10	3757	4,119	327,233	734	1440,00
gdb11	2451	-0,962	445,533	64437	525,17 ¹
gdb12	1503	1,804	100,467	536	1440,00
gdb13	620	0,000	63,783	620	1440,00
gdb14	145	0,000	11,767	145	0,07
gdb15	140	0,000	136,483	140	1052,25 ¹
gdb16	182	0,000	136,367	182	588,35 ¹
gdb17	168	0,000	165,867	168	1294,88 ¹
gdb18	389	-0,063	174,433	415	1440,00
gdb19	127	0,016	42,5	125	792,77 ¹
gdb20	1279	-0,439	288,467	2280	1131,72 ¹
gdb21	212	0,000	176,35	212	636,35 ¹
gdb22	434	0,002	156,167	433	1440,00
gdb23	336	0,000	269,867	336	1440,00

FONTE: O autor (2017).

NOTA: ¹ Problemas que esgotaram a memória do computador antes de chegar a uma solução ótima ou atingir o tempo limite.

Os valores em negrito correspondem a melhor solução encontrada para cada instância. Observa-se que *software* CPLEX foi capaz de determinar solução ótima apenas para o problema gdb14. Para 12 instâncias o *software* esgotou a memória do computador antes de chegar a uma solução ótima ou atingir o tempo limite de 24 horas.

Para os problemas gdb1, gdb2, gdb4, gdb11, gdb18 e gdb20 a heurística E-R&F-NA obteve soluções com desempenho superior ao método exato. Para os problemas gdb6, gdb13, gdb14, gdb15, gdb16, gdb17, gdb21 e gdb23 a heurística determinou soluções com desvio igual à zero da melhor solução conhecida.

Soluções com desvio igual ou inferior a 0,04 são encontradas para os problemas gdb3, gdb7, gdb19 e gdb22. Os piores desempenhos da estratégia correspondem à resolução das instâncias gdb5, gdb9 e gdb12 apresentando desvio superior a 1,8. Destaca-se ainda que a estratégia em nenhum problema esgotou a memória do computador, fato esperado uma vez que o objetivo da heurística R&F é a resolução de subproblemas menores.

Com relação ao tempo de execução, houve um ganho significativo, a E-R&F-NA teve um tempo médio de 260,99 minutos (aproximadamente 4 horas) por instância, já a resolução pelo método exato obteve um tempo médio de 1070,28 minutos (aproximadamente 17 horas) por instância.

A aplicação da E-R&F-NA no modelo 2014 apresentou bom desempenho, pois obteve 6 soluções melhores do que as já conhecidas, 11 soluções iguais ou muito próximas ao método exato e apenas 3 soluções com valores piores que os conhecidos. Com relação ao tempo de execução a estratégia foi em média 4 vezes mais rápida que o método exato.

6 CONCLUSÕES E TRABALHOS FUTUROS

Para tomada de decisões de níveis operacionais, o CARP tem sido amplamente aplicado, mas quando se faz necessário abordar problemas que envolvam tomadas de decisões de níveis tático e estratégicos, a aplicação do CARP torna-se inviável. O Problema de Roteamento em Arcos Capacitado e Periódico, uma generalização do CARP, consiste em determinar um número de tratamento igual à frequência de cada arco (decisão tática) e formar rotas em cada dia de modo a atender os arcos designados para esses dias (decisão operacional) (LACOMME; PRINS; RAMDANE-CHÉRIF, 2005). Portanto, resolver um PCARP implica tomadas de decisões em níveis tático e operacionais simultaneamente (BATISTA; SCARPIN, 2014).

O campo de aplicação do PCARP é bastante amplo, uma vez que grande parte dos serviços realizados ao longo de vias de uma rede possuem características como frequência e veículos de capacidades limitadas. O PCARP pode ser utilizado para abordar o problema de coleta de resíduos urbanos, mas, também, outras aplicações têm sido desenvolvidas nos últimos anos. As aplicações encontradas foram: monitoramento de redes rodoviárias, supressão de poeira em estradas, manutenção preditiva e preventiva de linhas férreas, irrigação de árvores e despejo de sal em estradas com acúmulo de neve.

Os problemas de roteirização em arcos representam um campo de pesquisa de grande interesse na área de otimização, mas pesquisas que tratam do PCARP ainda são embrionárias. Em um levantamento realizado por Batista (2016, não publicado) nos portais de pesquisa *Web-of-science* e *Scopus*, para uma proposta de classificação taxonômica do problema, foi encontrado um total de 14 artigos, sendo que, desses, dois eram do próprio autor. Portanto novas pesquisas com o objetivo de propor novos modelos ou métodos de soluções tornam-se necessárias para desenvolvimento e/ou entendimento do PCARP.

O PCARP possui uma natureza *NP-hard* e quanto maior for a rede (grafo) e o número de restrições, sejam estas vinculadas à capacidade, tempo ou periodicidades, mais difícil é determinar uma solução. Tal dificuldade é compensada pelo ganho que o modelo pode trazer ao problema.

Neste trabalho foram considerados dois modelos desenvolvidos por Batista (2014, 2016, não publicado). Estes possuem as seguintes peculiaridades:

flexibilidade no atendimento dos arcos, podendo atrasar o atendimento de um arco a um dado custo; a possibilidade do veículo folgar, possibilitando um melhor atendimento das demandas; e, no modelo 2016, possui a possibilidade de um veículo iniciar sua rota em um vértice qualquer e terminar em outro vértice desde que esse seja início da rota de algum veículo.

Como método de solução, propõe-se o uso da heurística *relax-and-fix*, que consiste na decomposição de um modelo de programação inteira mista em submodelos menores disjuntos, que podem ser resolvidos rapidamente, porém sem a garantia de resolução do modelo original de forma ótima. A heurística foi proposta por Wolsey (1998) e tem sido largamente aplicada para a resolução de modelos de dimensionamento de lotes.

Um fator que influencia no desempenho da heurística é a escolha das partições das variáveis do problema. Para os modelos aqui tratados, foram propostas duas formas diferentes de partições: a decomposição das variáveis em relação ao período e a decomposição em adjacência em nós. Para a decomposição em períodos, foram desenvolvidas 4 estratégias clássicas distintas: R&F- *forward*, R&F- *backward*, R&F- *overlapping* e R&F- *overlapping 2*. Para a decomposição em adjacência, foi proposta uma estratégia R&F- nós adjacentes.

Para avaliar o desempenho das 5 estratégias propostas foram utilizadas 23 instâncias propostas por Batista (2014). Os resultados computacionais mostram que as heurísticas que utilizaram a decomposição por período possuem desempenho inferior à decomposição por adjacência. A R&F-NA obtiveram tempo de processamento maior, mas em contrapartida determinaram soluções viáveis de melhor qualidade, conseguindo determinar soluções para 20 das 23 instâncias. O maior tempo gasto pela R&F-NA é 4,7 vezes menor que a execução pelo método exato.

Devido ao alto número de problemas infactíveis encontrados na aplicação da R&F clássica, foi proposta uma adaptação da heurística *Enhanced Relax-and-Fix* (ESCUDEIRO; SALMERON, 2005). Ao determinar uma solução infactível na resolução de um determinado subproblema, o algoritmo tem o potencial de retornar um ou mais passos, o que possibilita a reconfiguração da solução. Os resultados desse segundo conjunto de testes, mostram a potencialidade da estratégia, visto que o número de problemas infactíveis foi nulo. Mas, por outro lado, em algumas instâncias, a heurística retrocedeu um número muito grande de iterações

aproximando o subproblema do problema original, não sendo capaz de determinar uma solução viável, exaurindo o tempo predeterminado por iteração.

O tempo computacional da E-R&F-NA foi o maior entre as heurísticas, o que foi compensado pela qualidade das soluções, apresentando média dos desvios de 0,95 em relação às soluções encontradas pelo método exato executado por 24 horas. Como método de melhoria foi utilizada a heurística RINS com diferentes parâmetros. Foi possível verificar que a utilização do solver na configuração *default* apresentou desempenho superior aos demais parâmetros testados.

Ao aplicar a E-R&F-NA no modelo proposto em 2014 a estratégia apresentou bom desempenho, obtendo 6 soluções melhores que as já conhecidas e em nenhuma das instâncias esgotou a memória do computador, fato que foi comum na resolução pelo método exato (*software* CPLEX 12.4), o qual esgotou a memória em 12 dos 23 problemas. Com relação ao tempo de execução a estratégia foi em média 4 vezes mais rápida que o método exato.

A heurística *relax-and-fix* mostrou-se uma boa estratégia para determinar soluções viáveis para os modelos do PCARP estudados nesse trabalho. Na maioria dos casos essas soluções são de boa qualidade. Houve ganho em tempo computacional em relação à resolução dos modelos por um *software* comercial e nenhum caso esgotou a memória do computador, o que ocorreu com frequência na resolução pelo *software* CPLEX. Em casos em que as soluções determinadas pela R&F encontram-se distantes de soluções aceitáveis, estas podem ser aprimoradas com a aplicação de uma heurística de melhoria.

Como sugestão de trabalhos futuros destaca-se:

- Desenvolver uma heurística de melhoria para as soluções de baixa qualidade;
- Aplicar as estratégias propostas nesse trabalho em outros modelos da literatura;
- Propor novas restrições ao modelo com o objetivo de torna-lo mais eficiente computacionalmente;
- Adicionar uma restrição ao modelo que restrinja o número de folgas por veículo, evitando a infactibilidade ao aplicar a *relax-and-fix* que utiliza decomposição por períodos;
- Propor limitantes inferior para o modelo;

- Utilizar outras estratégias de exploração da heurística *relax-and-fix nós adjacentes*, por exemplo, explorar a árvore em profundidade. Na exploração em profundidade, o próximo nó a ser considerado inteiro e/ou binário será um dos filhos do nó corrente. Quando o nó não possuir mais filhos, retorna-se ao longo do caminho em direção ao nó raiz até encontrar o primeiro nó que tenha filho a ser explorado.

REFERÊNCIAS

ANDRADE, M. C. **A criação no processo decisório: o grafo como opção metodológica**. Rio de Janeiro: Livros Técnicos e Científicos, 1980.

ARAÚJO, R. R. **Um modelo de resolução para o problema de roteirização em arcos com restrição de capacidade**. 177 f. Dissertação (Mestrado em Engenharia de Produção). Universidade Federal do Rio Grande do Sul, Porto Alegre, 2003. Disponível em: <<https://www.lume.ufrgs.br/bitstream/handle/10183/4557/000412823.pdf?sequence=1>>. Acesso em: 15 fev. 2016.

ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. **Pesquisa Operacional**. Rio de Janeiro: Elsevier, 2007.

BALLOU, R. H. **Gerenciamento da cadeia de suprimentos/logística empresarial**. 5. ed. Porto Alegre: Bookman, 2006.

BATISTA, G. V. **Proposta de um Modelo Matemático para o Problema de Roteamento em Arcos Capacitado e Periódico**. 95 f. Dissertação (Mestrado em Métodos Numéricos em Engenharia). Universidade Federal do Paraná, Curitiba, 2014. Disponível em: <<http://acervodigital.ufpr.br/bitstream/handle/1884/35767/R%20-%20D%20-%20GUILHERME%20VINICYUS%20BATISTA.pdf?sequence=1>>. Acesso em: 29 nov. 2015.

BATISTA, G. V.; SCARPIN, C. T. Problema de Roteamento em Arcos Capacitado e Periódico Aplicado a um Contexto Real. **Revista Produção Online**, Florianópolis, Vol. 15, n. 3, p. 1080-1098, jul./set. 2015. Disponível em: <<https://www.producaoonline.org.br/rpo/article/view/1962>>. Acesso em: 10 dez. 2015.

BATISTA, G. V. **Iterative Restricted Search Space Aplicado ao Problema de Roteamento em Arcos Capacitados e Periódico com Movimento Contínuos**. 91 f. Qualificação (Doutorado em Métodos Numéricos em Engenharia). Universidade Federal do Paraná, Curitiba, 2016. Não publicado.

BELENGUER, J.; BENAVENT, E. A cutting plane algorithm for the capacitated arc routing problem. **Computers & Operations Research**, Valencia, v. 30, p. 705–728, 2003. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054802000461>>. Acesso em: 20 mar. 2016.

BELVAUX, G.; WOLSEY, L. A. bc-prod: A specialized Branch-and-Cut system for lot-sizing problems. **Management Science**, v. 46, n.5, p. 724-738, 2000. Disponível em: <<http://pubsonline.informs.org/doi/pdf/10.1287/mnsc.46.5.724.12048>>. Acesso em: 25 jan. 2016.

BEULLENS, P.; MUYLDERMANS, L.; CATTRYSSE, D.; VAN OUDHEUSDEN, D. A guided local search heuristic for the capacitated arc routing problem. **European Journal of Operational Research**, v. 147, n. 3, p. 629–643,

2003. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S037722170200334X>>. Acesso em: 25 jan. 2016

BOAVENTURA NETTO; OSWALDO, P. **Teoria e Modelos de Grafos**. São Paulo: Editora Edgard Blucher LTDA, 1979.

BOAVENTURA NETTO, P. O. **Grafos: teoria, modelos, algoritmos**. 4a ed. ed. São Paulo: Edgard Blucher, 2001.

CHERRI, L. H. **Um método híbrido para o problema de dimensionamento de lotes**. 69 f. Dissertação (Mestrado em Ciências). Universidade de São Paulo, São Paulo, 2013.

CHU, F.; LABADI, N.; PRINS, C. The periodic capacitated arc routing problem linear programming model, metaheuristic and lower bounds. **Journal of Systems Science and Systems Engineering**, v. 13, n. 4, p. 423–435, 2004. Disponível em: <<http://link.springer.com/article/10.1007/s11518-006-0174-y>>. Acesso em: 10 dez. 2015.

CHU, F.; LABADI, N.; PRINS, C. Heuristics for the periodic capacitated arc routing problem. **Journal of Intelligent Manufacturing**, v. 16, n. 2, p. 243–251, 2005. Disponível em: <<http://link.springer.com/article/10.1007/s10845-004-5892-8>>. Acesso em: 10 dez. 2015.

CHU, F.; LABADI, N.; PRINS, C. A Scatter Search for the periodic capacitated arc routing problem. **European Journal of Operational Research**, v. 169, p. 586–605, 2006. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221704005569>>. Acesso em: 10 dez. 2015.

CLARKE, G.; WRIGHT, J. W. Scheduling of Vehicles from a Central Depot to Number of Delivery Points. **OPNS**, Res. 12, p. 568-581, 1964. Disponível em: <<http://read.pudn.com/downloads160/doc/fileformat/721736/Scheduling%20of%20vehicles%20from%20a%20central%20depot%20to%20a%20number%20of%20delivery%20points.pdf>>. Acesso em: 11 dez. 2015.

CORBERÁN, A.; PRINS, C. Recent Results on Arc Routing Problems: Na Annotated Bibliography. **Networks**, p. 50–69, 2010. Disponível em: <<http://onlinelibrary.wiley.com/doi/10.1002/net.20347/abstract>>. Acesso em: 11 dez. 2015.

CORMEN T. H. **Algoritmos: Teoria e Prática**. Rio de Janeiro: Elsevier, 2002.

CUNHA, C. B. **Uma contribuição para o problema de roteirização de veículos com restrição operacionais**. 222 f. Tese (Doutorado). EPUSP, Departamento de Engenharia de Transporte, 1997.

CUNHA, C. B. Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais. **Transportes**, v. 8, n.2, p. 51-74, 2000. Disponível em: <http://www.gestori.com.br/website2/diversos/artigos/aspectos_praticos_de_aplicacao_da_roterizacao.pdf>. Acesso em: 15 jan. 2016.

CUNHA, A. L. **Métodos heurísticos para um problema de planejamento da produção em uma indústria química**. 109 f. Dissertação (Mestrado em Ciências). Universidade de São Paulo, São Paulo, 2013. Disponível em: <www.teses.usp.br/teses/disponiveis/.../cunhaAL_dissertacao_revisada.pdf>. Acesso em: 2 fev. 2016.

DANNA, E.; ROTHBERG, E.; LE PAPE, C. Exploring relaxation induced neighborhoods to improve mip solutions. **Math. Program**, Ser A, v. 102, p. 71–90, 2005. Disponível em: <<http://link.springer.com/article/10.1007/s10107-004-0518-7>>. Acesso em: 1 de set. 2016.

DILLENBERGER, C.; ESCUDERO, L. F.; WU ZHANG, A. W. On practical resource allocation for production planning and scheduling with pwrriod overlapping setups. **European Journal of Operational Research**, v. 75, p. 275-286, 1994. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0377221794900744>>. Acesso em: 4 mar. 2016.

DOULABI, S. H. H.; SEIFE, A. Lower and upper bounds for location-arc routing problems with vehicle capacity constraints. **European Journal of Operational Research**, vl. 224, n. 1, p.189–208, 2013. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221712004705>>. Acesso em: 6 mar. 2016.

DROR, M.; STERN, H.; TRUDEAU, P. Postman Tour on a Graph with Precedence Relation on Arcs. **Networks**, v. 17, p. 283-294, 1987. Disponível em: <<http://onlinelibrary.wiley.com/doi/10.1002/net.3230170304/abstract>>. Acesso em: 5 dez. 2015.

DROR, M. **Arc routing: theory , solutions , and applications**. Boston, Mass. [u.a.]: Kluwer Academic Publishers, 2000.

EGLESE, R. W. Routing Winter Gritting Vehicles. **Dicrete Applied Mathematics**, v. 48, n. 3, p. 231- 244, feb. 1994. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0166218X92000035>>. Acesso em: 5 dez. 2015.

EISELT, H. A.; GENDREAU, M.; LAPORTE, G. Arc Routing Problems, Part I: The Chinese Postman Problem. **Operations Research**, v. 43, n. 2, p. 231–242, 1995i. Disponível em: <<http://or.journal.informs.org/cgi/doi/10.1287/opre.43.2.231>>. Acesso em; 5 dez. 2015.

EISELT, H. A.; GENDREAU, M.; LAPORTE, G. Arc Routing Problems, Part II: The Rural Postman Problem. **Operations Research**, v. 43, n. 3, p. 399–414, 1995ii. Disponível em: <<http://or.journal.informs.org/content/43/3/399.full.pdf>>. Acesso em: 5 dez. 2015.

ESCUADERO, L. F.; SALMERON, J. On a fix-and-relax framework for a class of project scheduling problems. **Operations Research**, n. 140, p. 163-188, 2005.

Disponível em: <<http://link.springer.com/article/10.1007/s10479-005-3970-8>>. Acesso em: 15 fev. 2016.

FEDERGRUEN, A.; MEISSNER, J.; TZUR, M. Progressive interval heuristics for multi-item capacitated lot sizing problems. **Operations Research**, v. 55, n. 3, 2007. Disponível em: <<http://www.meiss.com/en/publications/interval-heuristics.html>>. Acesso em: 16 fev. 2016.

FERREIRA, D. **Abordagens para o problema integrado de dimensionamento e sequenciamento de lotes da produção**. 253 f. Tese (Doutorado em Engenharia da Produção). Universidade Federal de São Carlos, São Carlos, 2007. Disponível em: <https://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwiEuon-t7XRAhWBI5AKHS9EAXgQFggaMAA&url=https%3A%2F%2Frepositorio.ufscar.br%2Fbitstream%2Fhandle%2Fufscar%2F3305%2FTeseFinal_DeisemaraFerreira.pdf%3Fsequence%3D1&usg=AFQjCNFtxchE_tK7HUx9HAoZQ6XeHxi_hA&bvm=bv.143423383,d.Y2l>. Acesso em: 16 fev. 2016.

FERREIRA, D.; MORABITO, R.; RANGEL, S. Relax and fix heuristics to solve one-stage one-machine lot-scheduling models for small-scale soft drink plants. **Computers & Operations Research**, v. 37, n. 4, p. 684-691, 2010. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054809001695>>. Acesso em: 15 fev. 2016.

FISCHETTI, M.; LODI, A. Local Branching. **Math. Program. Ser. B**, v. 98, p. 23–47, 2003. Disponível em: <<http://link.springer.com/article/10.1007/s10107-003-0395-5>>. Acesso em: 3 de mar. 2016.

FRANCO, E. F. J.; OLIVEIRA, H. C. B. Adaptação da Meta-Heurística GRASP na Resolução do Problema de Roteamento de Veículos com Janela de Tempo. **PODes**, v. 4, n. 3, p. 271-287, 2012. Disponível em: <<https://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwjNk-nGuLXRAhWMIpAKHTEgC8UQFggcMAA&url=http%3A%2F%2Fwww.podesenvolvimento.org.br%2Finicio%2Findex.php%3Fjournal%3Dpodesenvolvimento%26page%3Darticle%26op%3Ddownload%26path%255B%255D%3D107%26path%255B%255D%3D189&usg=AFQjCNHHeR74otd04Wf1KLum2WvOfd4r1Q>>. Acesso em: 10 dez. 2015.

GHIANI, G.; IMPROTA, G. An algorithm for the hierarchical Chinese postman problem. **Operations Research Letters**, v. 26, n. 1, p. 27–32, 2000. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0167637799000462>>. Acesso em: 9 dez. 2015.

GOLDEN, B. L.; WONG, R. T. Capacitated Arc Routing Problem. **Networks**, v. 11, p. 305–315, 1981.

GOLDEN, B. L.; DEARMON, J. S.; BAKER, E. K. Computational experiments with algorithms for a class of routing problems. **Computers & Operations Research**, v. 10, n. 1, p. 47–59, 1983. Disponível em:

<<http://www.sciencedirect.com/science/article/pii/0305054883900266>>. Acesso em: 13 dez. 2015.

GOLDBARG, M. C. **Otimização Combinatória e Programação Linear: modelos e algoritmos**. Rio de Janeiro: Elsevier, 2005.

GROSS, J. L.; YELLEN, J. Handbook of Graph Theory. **Slam Review**, v. 48, n. 1, 2006.

GRÖTSCHEL, M.; YUAN, Y. Euler, Mei-Ko Kwan, Königsberg, and a Chinese Postman. **Documenta Mathematica**, v. I, n. Extra Volume: Optimization Stories, p. 43–50, 2012. Disponível em: <https://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwjx-umzubXRAhVDGJAKHfwtD3EQFggcMAA&url=https%3A%2F%2Fwww.math.uni-bielefeld.de%2Fdocumenta%2Fvol-ismp%2F16_groetschel-martin-yuan-ya-xiang.pdf&usg=AFQjCNHtRgMwr-WUQzqsx0FthGPhzn914w>. Acesso em: 5 dez. 2015.

HELGAUN, K. Na Effective Implementation of the Lin- Kernighan Traveling Salesman Heuristic. **European Journal of Operational Research**, v. 126, n. 1, p. 106-130, oct. 2000. Disponível em: <http://akira.ruc.dk/~keld/research/LKH/LKH-1.3/DOC/LKH_REPORT.pdf>. Acesso em: 15 mai. 2016.

HERTZ, A.; LAPORTE, G.; MITTAZ, M. A Tabu Search heuristic for the Capacitated Arc Routing Problem. **Operations Research**, v. 48, n. 1, p. 129–135, 2000. Disponível em: <<http://pubsonline.informs.org/doi/abs/10.1287/opre.48.1.129.12455>>. Acesso em: 12 dez. 2015.

HILLER, F. S.; LIBERMAN, G. J. **Introdução à pesquisa operacional**. Porto Alegre: AMGH, 2013.

HUANG, S. H.; LIN, T. H. Using Ant Colony Optimization to solve Periodic Arc Routing Problem with Refill Points. **Journal of Industrial and Production Engineering**, v. 31, n. 7, p. 441-451, 2014. Disponível em: <<http://www.tandfonline.com/doi/abs/10.1080/21681015.2014.980444?journalCode=tjci21>>. Acesso em: 16 mar. 2016.

KONOWALENKO, F. **Problema do carteiro chinês não-orientado e misto para a otimização de rotas na cidade de Irati / PR**. 108 f. Dissertação (Mestrado em Ciências). Universidade Federal do Paraná, Curitiba, 2012. Disponível em: <http://acervodigital.ufpr.br/bitstream/handle/1884/26945/Dissertacao_Flavia_Konowalenko_Final.pdf?sequence=1&isAllowed=y>. Acesso em: 13 dez. 2015.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. A genetic algorithm for the capacitated arc routing problem and its extensions. Applications of evolutionary computing, E.J.W. Boers (ed.), **Lecture Notes in Computer Sciences** 2037, Springer, p. 473-483, 2001. Disponível em:

<http://link.springer.com/chapter/10.1007%2F3-540-45365-2_49>. Acesso em: 15 dez. 2015.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. Evolutionary Algorithms for Multiperiod Arc Routing Problems. In: IPMU 2002 (Ed.); 9th Int. Conf. On Information Processing and Management of Uncertainty in Knowledge-Based systems. **Anais...** p.1–8, 2002a. Annecy, France: ESIA-University of Savoie. Disponível em: <<https://pdfs.semanticscholar.org/42f6/b650867dd7001bbf05df288bc9725f6e6d02.pdf>>. Acesso em; 15 dez. 2015.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. General arc routing problems solved by a cutting plane algorithm and a genetic algorithm. In: IFAC (Ed.); 15th Triennial World Congress. **Anais...**, 2002b. Barcelona, Spain. Disponível em: <<https://pdfs.semanticscholar.org/f6f1/3ffc831894bb69381185ad0f3682a88a5eac.pdf>>. Acesso em: 5 mar. 2016.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. Planning Problems in Arc Routing. In: F. U.-E. de Valencia (Ed.); Actes de PMS 2002 (EURO Working Group on Project Management and Scheduling). **Anais...** p.232–235, 2002c. Valencia, Espanha. Disponível em: <<http://www.isima.fr/~lacomme/doc/pms02.pdf>>. Acesso em: 16 dez. 2015.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. Evolutionary algorithms for periodic arc routing problems. **European Journal of Operational Research**, v. 165, p. 535–553, 2005. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221704002590>>. Acesso em: 17 dez. 2015.

LACOMME, P.; PRINS, C.; SEVAUX, M. A genetic algorithm for a bi-objective capacitated arc routing problem. **Computers & Operations Research**, v. 33, n. 12, p. 3473–3493, 2006. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054805000730>>. Acesso em: 16 dez. 2015.

MARZOLF, F.; TRÉPANIER, M.; LANGEVIN, A. Road network monitoring: algorithms and a case study. **Computers & Operations Research**, v. 33, n. 12, p. 3494–3507, 2006. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054805000717>>. Acesso em: 17 dez. 2015.

MEI, Y.; TANG, K.; YAO, X. A memetic algorithm for periodic capacitated arc routing problem. **IEEE Transactions on Systems, Man, And Cybernetics/ PART B: Cybernetics**, v. 41, n. 6, p. 1654–1667, 2011. Disponível em: <<https://www.ncbi.nlm.nih.gov/pubmed/21768050>>. Acesso em: 10 dez. 2015.

MERCÉ, C.; FONTAN, G. MIP-based heuristics for capacitated lotsizing problems. **Int. J. Production Economics**, v. 85, p. 97-111, 2003. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0925527303000902>>. Acesso em: 13 fev. 2016.

MINIEKA, E. The Chinese Postman Problem for Mixed Networks. **Managemente**

Sciences, v. 25, n. 7, 1979. Disponível em: <<http://pubsonline.informs.org/doi/abs/10.1287/mnsc.25.7.643?journalCode=mnsc>>. Acesso em: 7 dez. 2015.

MONROY, I. M.; AMAYA, C. A.; LANGEVIN, A. The periodic capacitated arc routing problem with irregular services. **Discrete Applied Mathematics**, v. 161, p. 691–701, 2013. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0166218X11002083>>. Acesso em: 18 mar. 2016.

MORAES, L. C. C.; SANTOS, M. O. **Heurísticas relax-and-fix para o problema de dimensionamento de lotes com janelas de tempo de produção**. Simpósio Brasileiro de Pesquisa Operacional, 2012. Disponível em: <<http://www.din.uem.br/sbpo/sbpo2012/pdf/arq0479.pdf>>. Acesso em: 14 fev. 2016.

NEGREIROS, M.; COELHO JUNIOR, W. R.; PALHANO, A. W. D. C.; et al. O problema do carteiro chinês, algoritmos exatos e um ambiente MVI para análise de suas instâncias: SISTEMA XNÊS. **Pesquisa Operacional**, v. 29, n. 2, p. 323–363, 2009. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0101-74382009000200005>. Acesso em: 10 dez. 2015.

ORLOFF, C. A. A Fundamental Problem in Vehicle Routing. **Networks**, v. 4, p. 35-64, 1974. Disponível em: <<http://onlinelibrary.wiley.com/doi/10.1002/net.3230040105/abstract>>. Acesso em: 13 dez. 2015.

PEARN, W. L. Approximate Solutions for the Capacitated Arc Routing Problem. **Computers and Operations Research**, v. 16, n. 6, p. 589-600, 1989. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0305054889900440>>. Acesso em: 16 jan. 2016.

PEARN, W. L. Augment-Insert Algorithms for capacitated Arc routing Problem. **Computers and Operations Research**, v. 18, n.2, p. 189-198, 1991. Disponível em: <<http://www.sciencedirect.com/science/article/pii/030505489190089A>>. Acesso em: 14 mar. 2016.

PEDROSO, J. P.; KUBO, M. Hybrid Tabu Search for lot sizing problems. Capítulo em **Lecture Notes in Computer Science**, v. 3636, Spring, Berlin, 2005. Disponível em: <<http://www.dcc.fc.up.pt/~jpp/publications/PDF/hybrid-ts.pdf>>. Acesso em: 23 fev. 2016.

PEREIRA, D. L. **Heurísticas e Algoritmo Exato para o Problema de Roteamento de Veículos de Coleta e Entrega Simultâneas**. 83 f. Dissertação (Mestrado em Ciência da Computação), Universidade Federal de Minas Gerais, Belo Horizonte, 2010. Disponível em: <<https://www.dcc.ufmg.br/pos/cursos/defesas/1209M.PDF>>. Acesso em: 3 de nov. 2016.

POCHET, Y. WOLSEY, L. A. Production Planning by Mixed Integer Programming. **Science Business Media**, Inc, New York, Springer, 2006. Disponível em: <<http://www.springer.com/us/book/9780387299594>>. Acesso em: 13 fev. 2016.

RIQUELME-RODRÍGUEZ, J. P.; LANGEVIN, A.; GAMACHE, M. Adaptive Large Neighborhood Search for the Periodic Capacitated Arc Routing Problem with Inventory Constraints. **Network**, v. 64, p. 125-139, 2014. Disponível em: <<http://onlinelibrary.wiley.com/doi/10.1002/net.21562/abstract>>. Acesso em: 19 mar. 2016

RIQUELME-RODRÍGUEZ, J. P.; LANGEVIN, A.; GAMACHE, M. Location arc routing problem with inventory constraints. **Computers & Operations Research**, v. 76, p. 84-94, 2016. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054816301447>>. Acesso em: 26 nov. 2016.

SHERAFAT, H. **Algoritmos Heurísticos de Cobertura de Arcos**. 174 f. Tese (Doutorado em Engenharia da Produção). Universidade Federal de Santa Catarina, Florianópolis, 2004. Disponível em: <<https://repositorio.ufsc.br/bitstream/handle/123456789/86744/203794.pdf?sequence=1>>. Acesso em: 13 set. 2016.

SILVA, D. Método híbrido para o problema de dimensionamento de lotes com múltiplas plantas. Dissertação (Mestrado). Universidade de São Paulo, São Paulo, 2013. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/55/55134/tde-08052013-140926/publico/LuizCherrrevisada.pdf>>. Acesso em: 23 fev. 2016.

TAGMOUTI, M.; GENDREAU, M.; POTVIN, J.-Y. Arc routing problems with time-dependent service costs. **European Journal of Operational Research**, v. 181, n. 1, p. 30-39, 2007. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0377221706004644>>. Acesso em: 13 jan. 2016.

TOLEDO, C. F. M.; ARANTES, M. S.; HOSSOMI, M. Y. B.; FRANÇA, P. M.; AKARTUNALI, K. A relax-and-fix with fix-and-optimize heuristic applied to multi-level lot-sizing problems. **Journal of Heuristics**, v. 21, n. 5, p. 687-717, 2015. Disponível em: <<http://link.springer.com/article/10.1007/s10732-015-9295-0>>. Acesso em: 13 fev. 2016.

ULBRICHT, G. **Um Modelo de Planejamento da Produção Multiestágio com Estoque Intermediário limitados**. 208 f. Tese (Doutorado em Métodos Numérico em Engenharia). Universidade Federal do Paraná, Curitiba, 2015. Disponível em: <<http://acervodigital.ufpr.br/handle/1884/41818>>. Acesso em: 14 fev. 2016.

VIEIRA, H. P. **Metaheurística para a Solução de Problemas de Roteamento de Veículos com Janela de Tempo**. 120 f. Dissertação (Mestrado em Matemática Aplicada), Universidade Estadual de Campinas, São Paulo, 2013. Disponível em: <http://www.ime.unicamp.br/~chico/tese_heloisa.pdf>. Acesso em: 15 jan. 2016.

WAN, H. F.; WEN, Y. P. Time-Constrained Chinese Postman Problems. **Computer and Mathematics with Applications**, v. 44, n. 3-4, p. 375-387, 2002. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0898122102001566>>. Acesso em: 10 dez. 2015.

WOLSEY, L. A. Wiley-Interscience Series in Discrete Mathematics and Optimization,

Integer Programming. **John Wiley & Sons**, 1998.

WU, L.; CUNHA, C. B. O problema de roteirização periódica de veículos. **Transportes**, v. 16, n. 1, p. 5-16, 2008. Disponível em: <<https://www.revistatransportes.org.br/anpet/article/view/8>>. Acesso em: 15 dez. 2015.

WU, T.; SHI, L.; SONG, J. Na MIP-based interval heuristic for the capacitated multi-level lot sizing problem with setup times. **Annals of Operations Research**, v. 196, n. 1, p. 65-650, 2012. Disponível em: <<http://link.springer.com/article/10.1007/s10479-011-1026-9>>. Acesso em: 15 fev. 2016.