Universidade Estadual de Campinas
Instituto de Matemática, Estatística e Computação Científica
Departamento de Matemática Aplicada

Construção de Rotas para Patrulhamento Urbano Preventivo

Washington Alves de Oliveira
Mestrado em Matemática Aplicada - Campinas - SP

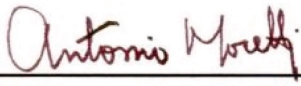
Orientador: Prof. Dr. Antônio Carlos Moretti

Co-Orientadora: Profa. Dra. Margarida Pinheiro Mello

Construção de Rotas para Patrulhamento Urbano Preventivo

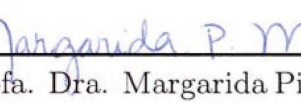
Este exemplar corresponde à redação final da dissertação devidamente corrigida e defendida por Washington Alves de Oliveira e aprovada pela comissão julgadora.

Campinas, 10 de Julho de 2008



Prof. Dr. Antônio Carlos Moretti

Orientador



Profa. Dra. Margarida Pinheiro Mello

Co-Orientadora

Banca Examinadora:

Prof. Dr. Antônio Carlos Moretti

Profa. Dra. Maria de Los Angeles Gonzalez Lima

Prof. Dr. Akebo Yamakami

Prof. Dr. Aurélio Ribeiro Leite de Oliveira

Dissertação apresentada no Instituto de Matemática, Estatística e Computação Científica, UNICAMP, como requisito parcial para obtenção do título de MESTRE em MATEMÁTICA APLICADA.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DO IMECC DA UNICAMP
Bibliotecária: Maria Júlia Milani Rodrigues

Oliveira, Washington Alves de

OL4c Construção de rotas para patrulhamento urbano preventivo / Washington Alves de Oliveira -- Campinas, [S.P. :s.n.], 2008.

Orientadores : Antônio Carlos Moretti ; Margarida Pinheiro Mello

Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Programação inteira. 2. Heurística. 3. Otimização combinatória. 4. Problema de roteamento de veículos. I. Moretti, Antônio Carlos. II. Mello, Margarida Pinheiro. III. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. IV. Título.

Título em inglês: Building preventive patrol routes.

Palavras-chave em inglês (Keywords): 1. Integer programming. 2. Heuristics. 3. Combinatorial optimization. 4. Vehicle routing problem.

Área de concentração: Pesquisa operacional

Titulação: Mestre em Matemática Aplicada

Banca examinadora:

Prof. Dr. Antônio Carlos Moretti (IMECC-UNICAMP)
Profa. Dra. Maria de Los Angeles Gonzalez Lima (Univ. Simon Bolivar)
Prof. Dr. Akebo Yamakami (FEEC-UNICAMP)
Prof. Dr. Aurélio Ribeiro Leite de Oliveira (IMECC-UNICAMP)

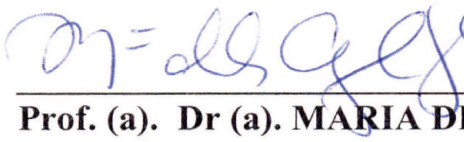
Data da defesa: 10/07/2008

Programa de pós-graduação: Mestrado em Matemática Aplicada

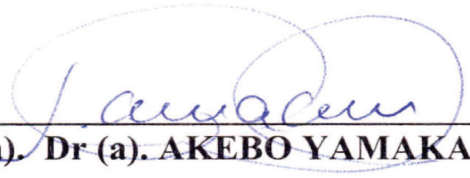
Dissertação de Mestrado defendida em 10 de julho de 2008 e aprovada

Pela Banca Examinadora composta pelos Profs. Drs.

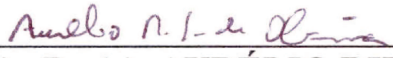
Prof.(a). Dr(a). ANTONIO CARLOS MORETTI



Prof. (a). Dr (a). MARIA DE LOS ANGELES GONZALEZ LIMA



Prof. (a). Dr (a). AKEBO YAMAKAMI



Prof. (a). Dr (a). AURÉLIO RIBEIRO LEITE DE OLIVEIRA

Pode ser chamado de divino quem é capaz de vencer por modificar suas táticas de acordo com a situação inimiga pois, dos cinco elementos, nenhum é sempre predominante; das quatros estações, nenhuma dura para sempre; dos dias, uns são longos e outros curtos e a lua cresce e míngua.

Sun Tzu

Agradecimentos

Agradeço:

A Deus por sentí-lo do meu lado dando força quando não mais a tinha.

Aos meus pais, nordestinos de grande coragem, pela educação que me deram e por acreditarem no meu esforço e capacidade.

Aos professores e grandes orientadores Antônio C. Moretti e Margarida P. Mello, que tiveram paciência e dispuseram de tempo para me passar valiosos conhecimentos.

Ao professor Marko Rojas Medar, que me ajudou com grandes conselhos do início ao final deste trabalho.

Aos meus quatro irmãos, maravilhosas pessoas que sempre acreditaram em mim. Também ao meu amigo Luciano, um dos poucos que acompanhou-me durante a trajetória deste desafio.

Aos meus amigos da universidade, com quem compartilhei alegrias e decepções com nossas inúmeras tarefas a serem estudadas e aprendidas ao longo do curso de mestrado.

Aos meus amigos e profissionais que trabalham na área de segurança pela ajuda e idéias importantes.

Resumo

Nesta dissertação estudamos um aspecto do problema de planejamento do pratulhamento urbano preventivo: a construção de rotas a serem percorridas pelos veículos da força policial no patrulhamento preventivo. De modo geral, a elaboração de rotas visa garantir uma boa visibilidade para o patrulhamento, de modo a proporcionar sensação de segurança para a população, permitir o atendimento rápido em caso de ocorrências, fazer vigilância de determinados estabelecimentos (hospitais, escolas, etc.). O planejamento deve levar em conta os recursos disponíveis, normalmente o número de veículos, visar agilidade e uma distribuição equânime de trabalho. O produto final é um módulo computacional capaz de automaticamente gerar rotas atendendo um dado conjunto de especificações, que possa ser utilizado pelos departamentos responsáveis pela segurança pública.

Para tanto, fizemos uma adaptação do modelo para o *Problema de Rotas de Cobertura multi-veículo (m-PRC)*. Este modelo consiste em um programa linear inteiro cujo tamanho e complexidade torna inviável a aplicação de métodos exatos para sua solução. Soluções subótimas são obtidas aplicando-se as heurísticas propostas por M. Hachicha et. al. (2000), e outras contribuídas por nós. Neste modelo alguns pontos geográficos devem ser obrigatoriamente visitados, enquanto outros devem ficar suficientemente próximos das rotas traçadas. Procuramos gerar rotas de tamanho menor possível, para que cada circuito seja percorrido um maior número de vezes durante o turno de serviço.

As heurísticas foram implementadas em MATLAB e sua validação, assim como a do modelo, foi feita através da resolução de problemas gerados aleatoriamente. Além disso, obtivemos dados relativos à cidade de Vinhedo, S.P., e formulamos rotas para patrulhamento preventivo pela Guarda Civil Municipal. Os resultados são promissores, e a análise das soluções obtidas será utilizada para aprimorar o modelo.

Palavras-chave: Programação inteira e combinatória, heurísticas, roteamento de veículos e cobertura de conjuntos.

Abstract

In this text we study one aspect of the urban community policing: routine patrol route planning. We seek routes that guarantee visibility, as this has a sizable impact on the community's perceived safety and allows for quick emergency responses, and that provide surveillance of public buildings (e.g., hospitals, schools). The planning is restricted to the availability of vehicles and strives to achieve balanced and short routes. We construct a computerized module, capable of automatic generation of routes for a given vehicle fleet and lists of sites that must be visited. Such a module could be of interest to Police, Public Safety Departments, Municipal Service Agencies.

The module implements an adaptation of the model for the *multi-vehicle covering tour problem*. It constitutes an integer program whose size and complexity makes the use of an exact method impractical. Suboptimal solutions are obtained with several heuristics, some by M. Hachicha et. al. (2000), and others of our own devising. In this model a set of locations must be visited, whereas another subset must be close enough to the planned routes. The heuristics aim to construct short routes so that one could make several rounds during a work shift.

The implementation was done in MATLAB and its validation, as well as the model's, was based on the solution of randomly generated problems. Furthermore, data from the city of Vinhedo, SP, was obtained and tentative routes planned for the patroling of a choice of locations by the Municipal Guard. Their appraisal by the personnel in charge of the route planning will, without a doubt, help us improve the model and heuristics.

Keywords: Integer programming, combinatorial programming, heuristics, vehicle routing, set covering.

Sumário

1 Introdução **1**

2 Apresentação do Problema **3**
2.1 Apresentação do Problema em Estudo 3
2.1.1 Relação do Problema em Estudo e o m-PRC 3
2.1.2 Problema Proposto 4
2.2 Problema de Rotas de Cobertura Multiveículo 8
2.2.1 m-PRC 8
2.2.2 Formulação do m-PRC 10
2.3 Modelo Matemático do Problema em Estudo 13
2.3.1 m-PRC para Construção de Rotas de Patrulhamento Preventivo . . . 13
2.3.2 Formulação do m-PRC para Construção de Rotas de Patrulhamento Preventivo 13

3 Heurísticas **15**
3.1 Introdução 15
3.2 Cobertura de Conjuntos 16
3.2.1 Problema de Cobertura de Conjunto 17
3.2.2 Heurística Primal1 17
3.2.3 Primal1 e o m-PRC 20
3.3 Caixeiro Viajante 21
3.3.1 Problema do Caixeiro Viajante 21
3.3.2 PCV e o m-PRC 23
3.3.3 Heurística GENI 23
3.3.4 Heurística US 26
3.3.5 Heurística GENIUS 28
3.4 Rota de Cobertura 1-veículo 30
3.4.1 Problema da Rota de Cobertura 31
3.4.2 Heurística H-1-PRC 32
3.4.3 Testes Comparando H-1-PRC e H-1-PRC Modificada 34
3.5 Rotas de Cobertura Multiveículo 35
3.5.1 Problema de Rotas de Cobertura Multiveículo 36

3.5.2 Heurística 2-opt* 36
3.5.3 Heurística da Economia 41
3.5.4 Heurística da Varredura 46
3.5.5 Heurística Primeiro 1-PRC/Segundo m-PRC 51

4 Novas Heurísticas **55**
4.1 Introdução 55
4.2 Teste de Cobertura 56
4.2.1 Passos do Teste de Cobertura 57
4.3 Heurística H-1-PRC Gananciosa 59
4.3.1 Passos H-1-PRC Gananciosa 60
4.3.2 Testes Comparando H-1-PRC, H-1-PRC Modificada e H-1-PRC Gananciosa 62
4.4 Heurística 2-opt Balanceada 62
4.4.1 Passos 2-opt Balanceada 64
4.5 Heurística da Economia Adaptada 65
4.5.1 Passos da Heurística da Economia Adaptada 66
4.6 Heurística Gananciosa para o m-PRC 66
4.6.1 Passos da Heurística Gananciosa para o m-PRC 68
4.7 Heurística Divide-Espaço 69
4.7.1 Passos da Heurística Divide-Espaço 71
4.8 Implementações das Heurísticas 72

5 Exemplares Gerados e Resultados Computacionais **75**
5.1 Definição da Constante c 75
5.1.1 Constante c nos Exemplares Gerados Aleatoriamente 75
5.1.2 Constante c no Problema em Estudo 75
5.2 Construção dos Exemplares 76
5.2.1 Exemplares Gerados Aleatoriamente 76
5.2.2 Exemplar do Problema em Estudo 76
5.3 Resultados Computacionais 79
5.3.1 Análise dos Resultados 81

6 Conclusões e Trabalhos Futuros **93**
6.1 Conclusões 93
6.2 Trabalhos Futuros 94

Lista de Tabelas

3.1 Soluções de exemplares do PCV 30
3.2 Soluções para um exemplar do PCV 30
3.3 Resultados obtidos com as heurísticas H-1-PRC 35

4.1 Resultados obtidos com as heurísticas H-1-PRC 62

5.1 Exemplares gerados aleatoriamente com distribuição uniforme 77
5.2 Exemplar gerado com dados reais 78
5.3 Heurísticas e valores escolhidos nos testes computacionais 80
5.4 Resultados das heurísticas estudadas 81
5.5 Resultados das novas heurísticas 82
5.6 Resultados das heurísticas mais eficientes 83
5.7 Diferenças N_k e o número de testes realizados por cada heurística 84
5.8 Resultados obtidos com os dados reais 85

Lista de Figuras

2.1 Exemplo de uma seleção de vértices em uma região do mapa 5
2.2 Seleção de vértices para o conjunto $T \subset V$ em uma região do mapa 6
2.3 Seleção de vértices para o conjunto $T \subset V$ e o conjunto W em uma região do mapa 7
2.4 Seleção de vértices para cada conjunto $T \subset V$, W e V em uma região do mapa 8
2.5 Exemplo de grafo não direcionado para o conjunto $V \cup W$ 9
2.6 Ocorrência de subrotas desconectadas e como as restrições as eliminam . . . 12

3.1 Processo de resolução de um exemplar do m-PRC em três fases principais . . 16
3.2 Primeira iteração da heurística PRIMAL 19
3.3 Segunda iteração da heurística PRIMAL 19
3.4 Terceira iteração da heurística PRIMAL 20
3.5 Ocorrência de subrotas desconectadas que satisfazem as restrições de designação 23
3.6 Inserção Tipo 1 e Tipo 2 de v entre v_i e v_j 24
3.7 Heurística GENI aplicada em um exemplo com 8 vértices 26
3.8 Remoção Tipo 1 e Tipo 2 do vértice v_i da rota 27
3.9 Configuração das soluções do PCV para os exemplares `berlin52` e `ch150` . . 31
3.10 Solução inicial viável de um m-PRC 37
3.11 Caso A e Caso B: bases replicadas no Passo 1 da heurística 2-opt* 38
3.12 Novas rotas para os casos A e B da Figura 3.11 39
3.13 Diminuição no número de rotas 40
3.14 Diminuição significativa no número de vértices de uma rota 40
3.15 Exemplo do Procedimento de Seleção por Economia 45
3.16 Exemplo do processo de varredura 49
3.17 Uma solução inicial viável na heurística da varredura 52
3.18 Uma solução inicial viável na heurística primeiro 1-PRC/segundo m-PRC . . 54

4.1 Processo de resolução de um exemplar do m-PRC em três fases principais . . 56
4.2 Cada vértice deve pertencer a uma única rota 58
4.3 Um vértice de W sendo coberto por duas rotas 59
4.4 $|S_{t,restante}| \geq 1$ para algum $v_t \in W$ 59
4.5 Duas possibilidades de escolha de um vértice para entrar na rota 61
4.6 Ocorrência de solução com rotas sobrepostas 63

4.7 Eliminação de rotas sobrepostas 64
4.8 Troca de vértices entre rotas opostas 65
4.9 Ocorrência de multicobertura de vértices em W 69
4.10 Construção de duas rotas ruins no método ganancioso 70
4.11 Início da análise do espaço de busca na heurística divide-espaço 72
4.12 Espaço de busca dividido em três setores em um primeiro teste 73
4.13 Espaço de busca dividido em três setores em um segundo teste 74
4.14 Ocorrência de posição ruim da base para a heurística divide-espaço 74

5.1 Configuração geográfica do exemplar obtido na cidade de Vinhedo 78
5.2 Configuração do exemplar obtido na cidade de Vinhedo reduzido 79
5.3 Relação: melhor de todas heurísticas por classe de exemplares 88
5.4 Relação: melhor de todas heurísticas por classe de exemplares 89
5.5 Rotas obtidas com a heurística da varredura para a cidade de Vinhedo . . . 90
5.6 Alternativa de rotas para a cidade de Vinhedo: apenas um veículo visita a base 91

Notação Utilizada

PRC	Problema de rota de cobertura.
m**-PRC**	Problema de rotas de cobertura multiveículo.
PCV	Problema do caixeiro viajante.
PRV	Problema de roteamento de veículos.
PC	Problema de cobertura de conjuntos.
GCM	Guarda civil municipal.
%O	Desvio da solução ótima.
N	Número de cidades ou dimensão do problema.
tempo	Tempo de processamento em segundos.
MS	Melhor solução encontrada.
R	Relação: valor obtido ao dividir a solução de uma dada heurística pelo melhor valor obtido entre todas as heurísticas por exemplar.
SO	Solução ótima.
SH	Solução obtida por uma dada heurística.
p	Parâmetro usado para definir a p-vizinhança.
***	Dado não informado: processamento encerrado por tempo ou pela falta de memória do computador.
H1PRC-GENIUS	Heurística H-1-PRC original usando a heurística GENIUS.
H1PRC-GENI	Heurística H-1-PRC original usando somente a heurística GENI.
H1PRC-MOD	Heurística H-1-PRC modificada (sugestão de mudança na heurística).
H1PRC-GAN	Heurística H-1-PRC gananciosa (nova heurística).
m	Número fixo de veículos no m-PRC.
N_k	Número de vértices na rota k.
Dif. N_k	Maior diferença entre os valores N_k das rotas construídas.
r	Valor que limita a diferença entre os valores N_k das k rotas de veículo.

Capítulo 1

Introdução

Um dos principais objetivos do patrulhamento urbano preventivo, feito com viaturas ou carros de patrulha, é garantir presença territorial, ou seja, a viatura com seus respectivos integrantes devem cobrir a maior área possível, para serem vistos e poder atender o maior número de pessoas com rapidez, caso seja necessário.

Entendemos por patrulhamento urbano preventivo feito com uma viatura, o seu deslocamento em velocidade baixa, em média 40 km/h, em um trajeto dentro de uma área urbanizada, tal que a viatura e seus integrantes estejam devidamente caracterizados (pintura, acessórios, fardas, etc.), representando um setor de segurança, e tem a finalidade principal de prevenir e coibir atos ilegais.

Existem algumas restrições associadas ao serviço de segurança, tanto de ordem humana e operacional, quanto de ordem material. Por exemplo, em um dia de serviço podemos ter a quantidade de carros ou de homens reduzida e o número de solicitações, por ocorrências, pode ser grande. Esses fatores tornam a tarefa de distribuição do efetivo, a fim de cobrir uma região, mais complicada. Mesmo assim, algumas tarefas associadas ao patrulhamento preventivo devem ser cumpridas. O carro de patrulha deve visitar pontos sensíveis durante seu trajeto, como escolas, postos de saúde, agências bancárias, locais de eventos, etc., de tal forma que durante este trajeto outros locais sejam cobertos para prevenir atos ilegais e atender a população próxima.

O objetivo desta dissertação é propor uma forma de construção de rotas dentro de uma região mapeada, a fim de encontrar uma boa solução para a difícil tarefa de distribuição adequada de viaturas. Tais rotas poderiam ser usadas como rotas de patrulhamento urbano preventivo por alguns departamentos responsáveis em promover segurança, como as Polícias, as Guardas Civis Municipais e empresas de segurança privada. Normalmente tais rotas são escolhidas manualmente, mas se a área e as restrições aumentam, essa escolha pode demorar dias e os resultados podem ser insatisfatórios com relação a alguns requisitos, como por exemplo, uma má cobertura da área e viaturas patrulhando em rotas semelhantes.

Quando observamos tal problema matematicamente, através de um modelo de otimização, percebemos a dificuldade de encontrar uma boa solução, porque o serviço é muito dinâmico e existem várias restrições, como por exemplo, a necessidade de alterar as rotas quando é

necessário o atendimento de ocorrências, as questões de hierarquia do serviço, etc. Temos como proposta de trabalho construir tais rotas de patrulhamento usando um modelo de programação linear inteira, que será chamado de Problema de Rotas de Cobertura multiveículo (m-PRC), originalmente conhecido como *multi-vehicle Covering Tour Problem* (m-CTP). Contudo algumas restrições do problema original serão ignoradas e outras acrescentadas de acordo com a nossa proposta.

Este modelo é um problema combinatório classificado como NP-difícil, cuja solução, através de algoritmos exatos de otimização demandaria grande esforço computacional. Para que o tempo computacional na obtenção de uma boa solução seja satisfatório, vamos considerar na busca desta solução uma combinação de técnicas heurísticas. O modelo do problema e as formas heurísticas de solução propostos por Hachicha, Hodgson, Laporte e Semet [16] foram estudados e usados, porém algumas alterações foram sugeridas.

Para validar a nossa proposta de construção de rotas, foram gerados aleatoriamente alguns exemplares para m-PRC e a partir de dados reais, aplicamos testes na construção de rotas de patrulhamento preventivo para a Guarda Civil Municipal da cidade de Vinhedo-SP.

No Capítulo 2 descrevemos o problema em estudo que originou esta pesquisa, suas características dentro do contexto de segurança e a aplicação feita na Guarda Municipal de Vinhedo. Ainda neste capítulo apresentamos o modelo m-PRC estudado na literatura, suas utilizações, suas características e o modelo matemático construído para o problema em estudo. As heurísticas que foram estudadas e usadas na resolução dos exemplares e as alterações sugeridas são apresentadas no Capítulo 3. No Capítulo 4 são descritas sugestões de novas heurísticas para o m-PRC. Os resultados computacionais são apresentados no Capítulo 5, e por fim, no Capítulo 6 apresentamos as conclusões e algumas propostas de trabalhos futuros.

Capítulo 2

Apresentação do Problema

Neste capítulo apresentamos o problema em estudo que originou o tema desta dissertação, suas características e as relações existentes entre a questão de distribuição de viaturas em uma dada região mapeada e o m-PRC. Também apresentamos com detalhes o modelo m-PRC, suas características e o modelo criado para o problema em estudo.

2.1 Apresentação do Problema em Estudo

Duas mudanças de nomenclatura, com relação às formulações de problemas de otimização relacionados com rotas de veículos, foram feitas. O que normalmente chamamos de depósito é chamado de **base**, que em nosso problema refere-se ao lugar físico onde os veículos ficam estacionados aguardando o início do serviço de patrulhamento dentro de uma região. Às vezes os veículos são chamados de **viaturas** ou **carros de patrulha**, que é o nome dado, frequentemente, ao veículo utilizado no serviço de segurança pública.

2.1.1 Relação do Problema em Estudo e o m-PRC

Um dos principais objetivos do patrulhamento urbano preventivo, feito com viaturas, é garantir presença territorial, ou seja, a viatura com seus respectivos integrantes, e em um determinado trajeto, devem cobrir a maior área possível. A visibilidade da viatura e de seus integrantes é muito importante neste serviço, é o que causa nas pessoas a chamada sensação de segurança. Assim, quanto maior for a área coberta e de forma adequada, maior será a sensação de segurança, e também possibilitará maior contato com a população. Essas são algumas características que previnem diversos tipos de atos ilegais.

Algumas tarefas associadas ao patrulhamento preventivo devem ser cumpridas por uma equipe de profissionais. As viaturas devem visitar locais sensíveis com relação a segurança e de importância para a comunidade, durante seu trajeto, como escolas, postos de saúde, agências bancárias, locais de eventos, etc., de tal maneira que durante este trajeto outros locais sejam cobertos, dentro de uma distância razoável de visibilidade, atuando preventivamente, como praças, bares, etc.

Essas características, nas quais existem locais (endereços) dispersos em uma região, tais que alguns deles devem ser visitados obrigatoriamente e outros devem ser cobertos por um veículo, foram as que fizeram escolhermos a formulação m-PRC para construir rotas de patrulhamento preventivo. O fato de uma viatura poder, pelas características do serviço e pela forma que as ruas estão configuradas em uma região, passar por vários locais, facilita a inclusão de outros locais que podem ser visitados. Aproximando-se ainda mais da formulação m-PRC.

Nos dias atuais, questões relacionadas com custo de combustível raramente causam restrições ao patrulhamento em cidades do estado de São Paulo. Desta forma as viaturas poderiam circular por todas as ruas da cidade, e a questão de cobertura da área estaria resolvida. Porém a quantidade de ruas para se rondar é muito grande comparada com o número de viaturas e profissionais normalmente disponíveis. Além disso, o tempo de um percurso incluindo todas as ruas de um setor (parte da região destinada a um veículo), seria alto, e uma viatura não conseguiria cumprir todo o trajeto em um turno de serviço. Outra questão que ocorre na escolha de rotas de patrulhamento inclui quais serviços se pretende realizar, pois existem equipes especializadas em um determinado tipo de serviço e outras que devem preocupar-se com serviços mais gerais. Também existem as questões de responsabilidade pela segurança em certos locais, pois as responsabilidades do município diferem das do estado. Assim, para se construir rotas a ser usadas pelos departamentos relacionados com o serviço de segurança preventiva, não podemos escolher todos os endereços de uma dada região como pontos possíveis de visita, isso irá depender do serviço que se pretende executar e de quem é a responsabilidade pela segurança nesta região.

O problema em estudo que queremos discutir envolve construir rotas de patrulhamento preventivo para Guarda Civil Municipal (GCM) da cidade de Vinhedo, SP, que é um dos setores responsáveis pela segurança de prédios e patrimônios municipais. Entre tais patrimônios estão todas as escolas municipais, hospitais municipais, postos de saúde municipais, etc. Outra questão importante é que as ruas também são bens municipais, então também é responsabilidade da GCM executar patrulhamento na maioria das ruas da cidade. Porém, como foi dito anteriormente, nem sempre é viável para uma viatura passar por um número grande de ruas. Portanto é interessante que o tamanho de cada rota seja minimizado, para que cada circuito seja completado o maior número de vezes possível por uma viatura, durante um turno de serviço e o serviço de cobertura pré-estabelecido para aquela equipe seja cumprido. Esta minimização proposta se aproxima mais ainda da formulação m-PRC.

2.1.2 Problema Proposto

Pretendemos construir sobre uma região rotas de patrulhamento para que uma segurança preventiva razoável, no sentido de cobertura da área, possa ser feita utilizando-se viaturas. Podemos afirmar em princípio que uma boa segurança não se baseia apenas em uma boa rota de patrulhamento, mas, do ponto de vista de cobertura da área, usar caminhos inteligentes pode ser muito relevante na qualidade do serviço de segurança. Para encontrar tais rotas, propomos seguir os seguintes passos:

- Primeiramente seleciona-se sobre o mapa da região, em que se pretende efetuar o patrulhamento, os locais (endereços) de importância para o setor responsável, bem como as esquinas e cruzamentos de ruas, utilizando as coordenadas geográficas. A idéia é considerar tais locais como vértices, que irão compor o conjunto de vértices onde a formulação m-PRC está definida.

- Em segundo lugar seleciona-se entre esses locais, aqueles que devem ser visitados obrigatoriamente pela viatura (conjunto $T \subset V$), aqueles que deve ser cobertos por uma rota (conjunto W) e o restante é considerado como pontos possíveis de visitas (conjunto $V \setminus T$).

Por exemplo, vemos na Figura 2.1 a ilustração de uma parte do mapa da cidade de Campinas-SP, onde é selecionada uma área onde pretendemos construir rotas de patrulhamento. Nesta área é feita a seleção de vértices conforme descrito acima, onde os locais selecionados incluem todas as esquinas, os cruzamentos de ruas, as escolas, hospitais, agências bancárias, etc. As coordenadas geográficas de cada ponto são guardadas e, em seguida, esses pontos são enumerados.

Figura 2.1: Exemplo de uma seleção de vértices em uma região do mapa

Após determinar os locais que devem ser visitados, por exemplo todos os prédios públicos, incluindo a base, a união dos vértices relacionados com esses locais formará o conjunto T, como pode ser visto na Figura 2.2, na qual os vértices estão representados por estrelas.

Após definir os locais que devem ser cobertos, como por exemplo agências bancárias, praças e locais com índice de criminalidade importante, a união dos vértices relacionados com essas escolhas formará o conjunto W, como pode ser visto na Figura 2.3, onde os vértices estão representados por circunferências.

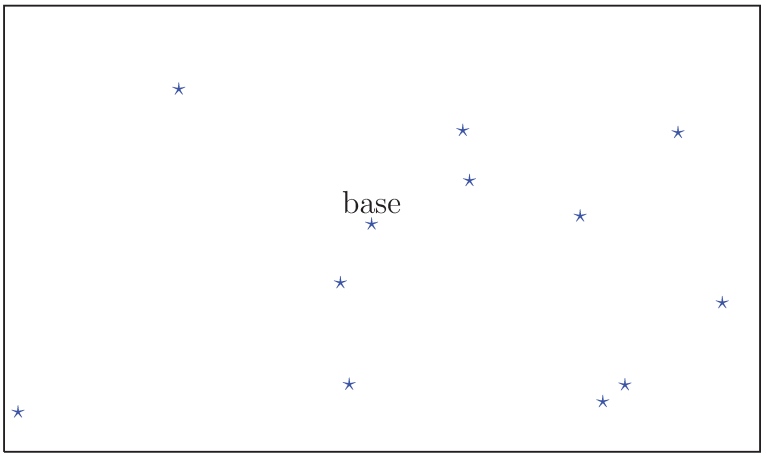


Figura 2.2: Seleção de vértices para o conjunto $T \subset V$ em uma região do mapa

Os locais restantes após essas escolhas são locais que podem ser visitados, ou seja, podem fazer parte de uma rota quando é necessário cobrir um local do conjunto W que fica longe de um ponto obrigatório de visita. Por exemplo, em um bairro podemos ter uma praça, perigosa em certos horários do dia, que fica longe de uma escola, sendo que a escola é um ponto obrigatório de visita. Assim escolhe-se um vértice fora de $T \cup W$ que fica a uma distância razoável da praça, e o local correspondente a esse vértice é incluído na rota, com isso a praça será coberta. Portanto teremos uma presença obrigatória na escola e uma ação preventiva de segurança na praça durante o percurso de uma viatura. Esses últimos vértices estão representados por pontos na Figura 2.4 juntamente com os demais vértices.

Na Figura 2.4 vemos ainda como os vértices selecionados ficam distribuídos na região e na Figura 2.5 vemos um pequeno exemplo de como o grafo para a formulação m-PRC é construído usando tais vértices. Agora podemos aplicar as heurísticas para que boas rotas sejam construídas.

Ao se refletir sobre até que ponto uma rota é eficaz para o patrulhamento preventivo podemos chegar a alguns tipos de questionamentos, como por exemplo, de que adianta uma rota bem construída se, quando uma viatura está visitando um ponto arbitrário v_i na rota, um crime está ocorrendo em outro ponto arbitrário v_j da rota, longe de v_i. Ou por que usar um caminho mais curto, se podemos maximizar o caminho para o patrulhamento das viaturas na região. Torna-se, portanto, obrigatório descrevermos quais são os principais objetivos propostos para construção de rotas. Temos os seguintes tópicos a considerar.

Ao assumir o serviço o responsável (administrador) pela segurança da área tem que garantir uma ação preventiva na região conforme a disponibilidade de agentes e viaturas no dia. Note que esse número pode variar de um dia para o outro, podemos ter viaturas em manutenção, uma quantidade baixa de motoristas em condições de trabalho, ou até mesmo, uma viatura pode quebrar ou se acidentar durante a patrulhamento. Neste último caso o ideal é reconstruir as rotas de acordo com o novo número de viaturas. No entanto, supondo

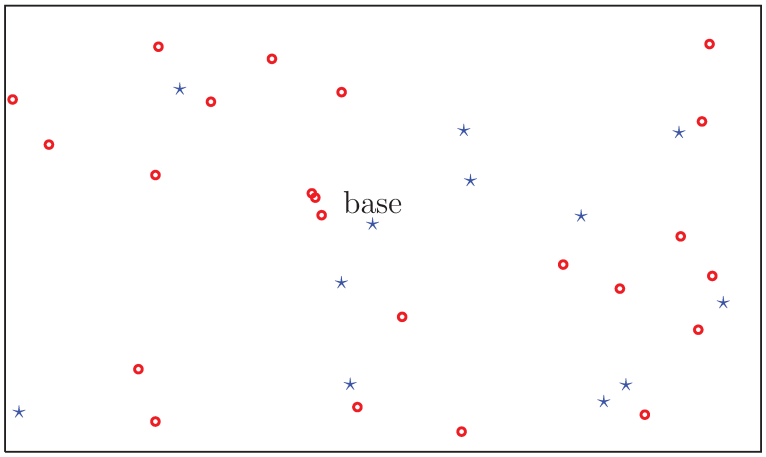


Figura 2.3: Seleção de vértices para o conjunto $T \subset V$ e o conjunto W em uma região do mapa

que sempre temos agentes suficientes para dirigir uma certa quantidade de viaturas, o número de rotas será igual ao número de viaturas disponíveis. Essa será uma condição inicial para a construção das rotas. Porém, pensando em uma aplicação, nada impede que mais de uma viatura utilize a mesma rota.

O tamanho das rotas e o número de locais visitados por cada viatura poderia ser uma restrição para o nosso problema, porém estamos presos ao um número fixo inicial de viaturas. Com isso, e pelo fato das rotas poderem ser construídas de diversas maneiras, se fixarmos um tamanho de rota, além da possibilidade do problema tornar-se inviável, pois as viaturas teriam uma quantidade limitada, por exemplo, em quilômetros para rodar, locais deixariam de ser cobertos ou visitados e teríamos um problema inicial a resolver, que é decidir um limite suficiente para uma viatura circular dentro de uma região. Todavia, distância percorrida e números de locais visitados, não constituem normalmente uma restrição para uma atividade de patrulhamento preventivo, a não ser que pretenda-se balancear esses valores para que alguns profissionais não fiquem sobrecarregados. Inicialmente não pretendemos discutir a questão de uma rota ser maior que outra, assim não teremos restrições de tamanho de rota. Para o nosso problema em estudo, não temos restrições de capacidade, pois os veículos transitam apenas com agentes e equipamentos, sem efetuar entregas. Por outro lado, as heurísticas foram implementadas de tal forma que os números de vértices, de uma rota para outra, fiquem próximos em quantidade, mas isso não evita que uma rota seja muito maior que a outra em termos de quilometragem.

Restrições de hierarquia do serviço, que podem forçar certos locais para uma determinada viatura, ou questões mais gerais de patrulhamento, como paradas para refeições ou atendimento de ocorrências, não estão inclusas em nosso propósito de trabalho.

Portanto os objetivos principais são: construir rotas que consigam visitar pontos obrigatórios e que cubram outros pontos, tal que cada rota tenha um tamanho mínimo. Porém

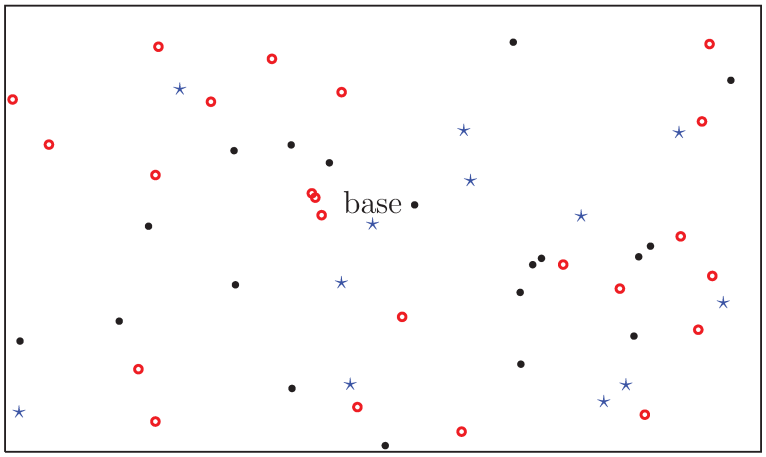


Figura 2.4: Seleção de vértices para cada conjunto $T \subset V$, W e V em uma região do mapa

esse tamanho mínimo não faz referência a diminuição de custos, e sim, que uma viatura circulando normalmente sobre esta rota, possa efetuar um ciclo completo o maior número de vezes possível durante um turno de serviço. Isto proporciona segurança nas proximidades e locais inclusos na rota e um aumento na possibilidade de ser acionada por um cidadão, já que um número maior de pessoas irá vê-la.

2.2 Problema de Rotas de Cobertura Multiveículo

Nesta seção apresentamos a formulação e os conceitos fundamentais do problema de otimização denominado por Problema de Rotas de Cobertura multiveículo (m-PRC), que será a base para o estudo proposto neste trabalho, e algumas de suas aplicações. A formulação usada neste trabalho se baseia na apresentada por Hachicha, Hodgson, Laporte e Semet [16].

2.2.1 m-PRC

O m-PRC é definido sobre um grafo $G = (V \cup W, E)$ não direcionado, onde $V \cup W$ compõem o conjunto de vértices, $V = \{v_0, \ldots, v_n\}$, $W = \{v_{n+1}, \ldots, v_l\}$ e $E = \{(v_i, v_j) : v_i, v_j \in V \cup W, i < j)\}$ o conjunto de arestas, ou seja, o subgrafo induzido por E é um grafo completo cujo conjunto de nós é V. O vértice v_0 é a base, onde estão disponíveis m veículos idênticos. O conjunto V é composto dos vértices que **podem** ser visitados e inclui um subconjunto T de vértices que **devem** ser visitados ($v_0 \in T$); W é um conjunto de vértices que **devem ser cobertos** coletivamente pelos m veículos. Uma matriz de distância ou tempo de viagem $C = (c_{ij})$ satisfazendo a desigualdade triangular é definida em E. Interpretamos c_{ji} como c_{ij} se $j > i$. O m-PRC consiste em atribuir um conjunto de m rotas de veículo de tamanho total mínimo, satisfazendo um certo conjunto de restrições:

1. Existem no máximo m rotas de veículo, e cada uma delas inicia e termina na base;

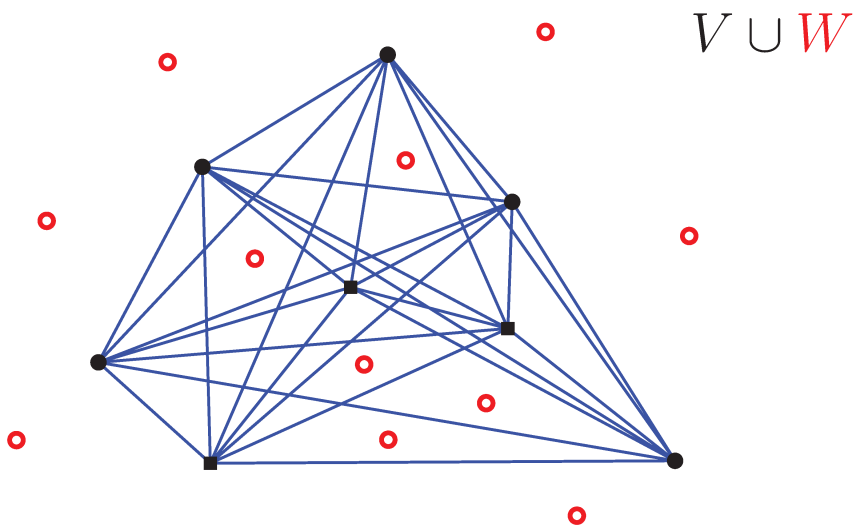


Figura 2.5: Exemplo de grafo não direcionado para o conjunto $V \cup W$

2. Cada vértice de V pertence a no máximo uma rota, enquanto cada vértice de T, com exceção da base, pertence a exatamente uma rota;

3. Cada vértice de W deve ser coberto por uma rota, no sentido de que deva situar-se dentro de uma distância c de um vértice de V pertencente a uma rota (assumimos que v_0 não cobre todos os vértices de W);

4. O número de vértices de uma rota (excluindo a base) não pode exceder um determinado valor P;

5. O tamanho de cada rota não pode exceder um determinado valor Q.

Vemos na Figura 2.5 um exemplo do grafo $G = (V \cup W, E)$ não direcionado. Note que este grafo é completo apenas no conjunto V, e apesar da matriz $C = (c_{ij})$ estar definida para todos os pares de nós em $V \cup W$, os vértices de W não são visitados.

Aplicações para o m-PRC surgem em algumas situações. Um exemplo é o problema de localização de caixas de correios (Labbé e Laporte [19]), onde deve-se alocar simultaneamente caixas de correios e uma coleção de rotas ótimas devem ser construídas dentre um conjunto de cidades candidatas, de tal maneira que usuários estejam localizados dentro de uma distância razoável de uma caixa de correio. Um outro exemplo é o desenho de rotas para equipes médicas móveis de atendimento em países em desenvolvimento, onde os serviços são realizados pelas equipes médicas em um número selecionado de vilarejos, então veículos viajam através deste número limitado de vilarejos e toda localidade não visitada deve estar dentro de uma curta distância de uma visitada, tal que as pessoas possam ir, por exemplo, a pé a este local para serem atendidas (Foord [12]; Hodgson, Laporte e Semet [18]; Oppong e Hodgson[24]; Swaddiwudhipong et al. [29]). Problemas similares são encontrados em

vários países ocidentais, em prevenção médica por equipes de saúde (Brown e Fintor[5]), na indústria leiteira (Simms [28]) e livrarias ou sistemas bancários movéis.

O m-PRC com $T = V$ reduz-se para um Problema de Roteamento de Veículos (PRV) com unidades de demanda (Fisher [11], Laporte [20]).

Em Hachicha, Hodgson, Laporte e Semet [16] três heurísticas são desenvolvidas e comparadas para a solução aproximada de alguns exemplares do m-PRC. No Capítulo 3 descreveremos estas heurísticas e outras desenvolvidas por nós.

2.2.2 Formulação do m-PRC

Para focalizar melhor o m-PRC, apresentamos sua formulação como um problema de programação linear inteira dada em [16].

Definição das variáveis.

- x_{ijk} representa o número de vezes que o veículo k usa a aresta (v_i, v_j) em seu trajeto. Se $i = 0$ esta variável toma valores 0,1 ou 2 (2 no caso de uma viagem de retorno após uma única visita), se $i > 0$, x_{ijk} será binário.
- $y_{hk} = \begin{cases} 1, & \text{se } v_h \in V \text{ é visitado pelo veículo } k \text{ na solução} \\ 0, & \text{caso contrário.} \end{cases}$

Definição das restrições.

1. Cada vértice v_l de W é coberto pelo menos uma vez:

$$\sum_{k=1}^{m} \sum_{v_h \in S_l} y_{hk} \geq 1, \quad (v_l \in W). \tag{2.1}$$

$$S_l = \{v_h \in V : c_{hl} \leq c\} \text{ : cobertura para cada } v_l \in W, \quad |S_l| \geq 2. \tag{2.2}$$

2. Cada vértice v_h de V, com exceção da base v_0, pertence a no máximo uma rota:

$$\sum_{k=1}^{m} y_{hk} \leq 1 \quad (v_h \in V \setminus \{v_0\}). \tag{2.3}$$

3. A solução conterá duas arestas usadas pelo veículo k e incidentes ao vértice v_h, ou nenhuma:

$$\sum_{i=0}^{h-1} x_{ihk} + \sum_{j=h+1}^{n} x_{hjk} = 2y_{hk} \quad (v_h \in V \setminus \{v_0\}, k = 1, \ldots, m). \tag{2.4}$$

4. As restrições anteriores não impedem soluções com subrotas desconectadas. Então são necessárias restrições que as eliminem. Se o vértice v_h não aparece na solução, então o lado direito das restrições 2.5 é igual a zero, ou seja, redundante. Caso contrário, pelo menos duas arestas irão conectar S e seu complemento $V \setminus S$, para todo subconjunto próprio de V, isto é válido desde que um vértice v_h visitado pertença a S e existam vértices de T fora de S. De fato, observe a Figura 2.6 e suponha que v_h pertence a uma subrota que não contém o nó base. Se considerarmos o conjunto S constituído pelos nós desta subrota, temos que o nó base pertence a $T \setminus S$. Como cada nó pertence a no máximo uma rota, garantido pela restrição 2.3, o somatório duplo no lado esquerdo de 2.5 terá valor nulo. Como, neste caso, o lado direito tem valor 2, vemos que este vetor (x, y) não satisfaz 2.5.

$$\sum_{k=1}^{m} \sum_{\substack{v_i \in S, v_j \in V \setminus S \\ \text{ou} \\ v_j \in S, v_i \in V \setminus S}} x_{ijk} \geq 2 \sum_{k=1}^{m} y_{hk} \quad (S \subset V,\ 2 \leq |S| \leq n-2,\ T \setminus S \neq \emptyset,\ v_h \in S). \tag{2.5}$$

5. No máximo m veículos entram e deixam o depósito:

$$\sum_{j=1}^{n} x_{0jk} \leq 2 \quad (k = 1, \ldots, m). \tag{2.6}$$

6. Uma rota não pode conter mais que P vértices de V:

$$\sum_{h=1}^{n} y_{hk} \leq P \quad (k = 1, \ldots, m). \tag{2.7}$$

7. Uma rota não pode ultrapassar um tamanho Q específico:

$$\sum_{i=0}^{n-1} \sum_{j=i+1}^{n} c_{ij} x_{ijk} \leq Q \quad (k = 1, \ldots, m). \tag{2.8}$$

8. Restrições de integralidade:

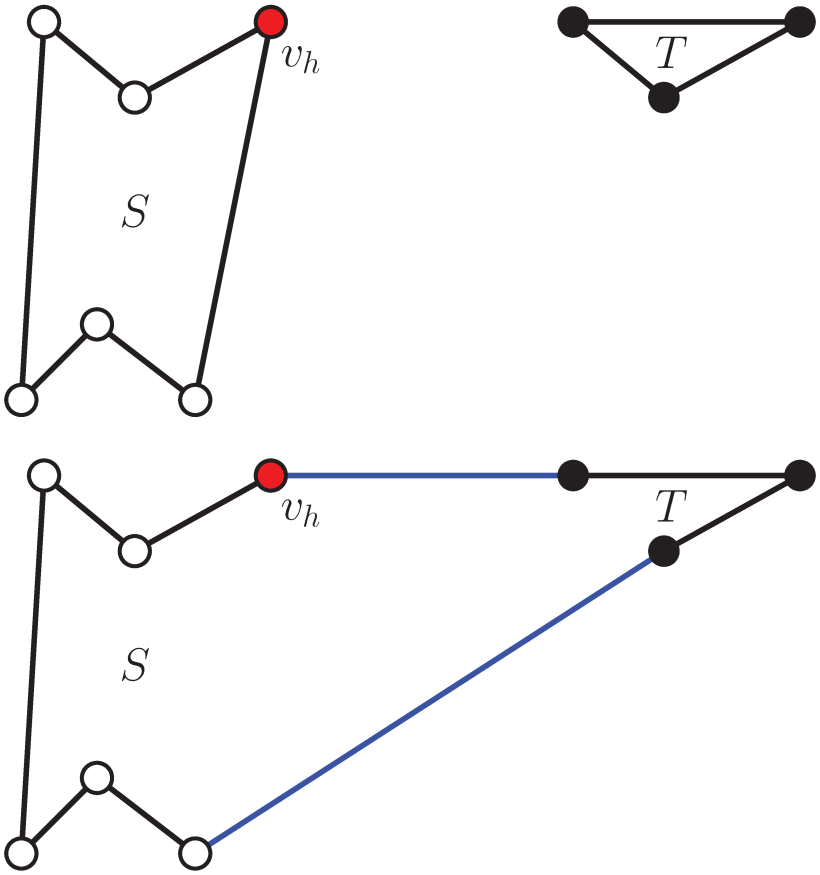


Figura 2.6: Ocorrência de subrotas desconectadas e como as restrições as eliminam

$$
\begin{aligned}
&\sum_{k=1}^{m} y_{hk} = 1 \quad (v_h \in T \setminus \{v_0\}),\\
&y_{0k} = 1 \quad k = 1, \ldots, m,\\
&y_{hk} \in \{0,1\} \quad (v_h \in V \setminus \{v_0\},\ \ k = 1, \ldots, m),\\
&x_{0jk} \in \{0,1,2\} \quad (v_j \in V \setminus \{v_0\},\ \ k = 1, \ldots, m),\\
&x_{ijk} \in \{0,1\} \quad (v_i, v_j \in V \setminus \{v_0\},\ \ i < j,\ \ k = 1, \ldots, m).
\end{aligned}
\tag{2.9}
$$

Definição da função objetivo.

- A soma total dos tamanhos das rotas é minimizada:

$$z = \min \sum_{k=1}^{m} \sum_{i=0}^{n-1} \sum_{j=i+1}^{n} c_{ij} x_{ijk}. \tag{2.10}$$

2.3 Modelo Matemático do Problema em Estudo

Neste seção apresentamos uma formulação para o m-PRC que é semelhante à formulação encontrada em Hachicha, Hodgson, Laporte e Semet [16], porém com algumas alterações que modelam o que chamamos de Construção de Rotas de Patrulhamento Urbano Preventivo.

2.3.1 m-PRC para Construção de Rotas de Patrulhamento Preventivo

Este problema refere-se ao grafo já definido no início da Seção 2.2.1. No entanto, o conjunto de restrições sofre algumas alterações. A seguir apresentamos o novo conjunto:

1. Existem **exatamente** m rotas de veículo, e cada uma delas inicia e termina na base;
2. Cada vértice de V pertence a no máximo uma rota, enquanto cada vértice de T, com exceção da base, pertence a exatamente uma rota;
3. Cada vértice de W deve ser coberto por uma rota, no sentido de que deva situar-se dentro de uma distância c de um vértice de V pertencente a uma rota (assumimos que v_0 não cobre todos os vértices de W);
4. A diferença entre o número de vértices de diferentes rotas não pode exceder um determinado valor $\boldsymbol{r}$.

Comparando com a definição que foi apresentada na Seção 2.2.1, vemos uma alteração no item 1. Agora devemos construir exatamente m rotas de veículo que é o número exato de veículos disponíveis. Excluímos as restrições de tamanho de rota e de número máximo de vértices por rota, porém acrescentamos a restrição de que as diferenças entre os números de vértices de diferentes rotas devem ser limitadas por uma constante r. Essa última é uma tentativa de não sobrecarregar uma equipe de profissionais atuando em uma determinada rota. Observe que isto não garante que os tamanhos das rotas sejam próximos.

Note que quando o conjunto T cobre todos os vértices do conjunto W, o m-PRC reduz-se a um PRV sobre o conjunto T.

2.3.2 Formulação do m-PRC para Construção de Rotas de Patrulhamento Preventivo

Existe grande semelhança entre esta nova formulação e a apresentada na Seção 2.2.2. Algumas alterações foram feitas para que o novo conjunto de restrições apresentado na Seção 2.3.1 fosse satisfeito.

Para satisfazer o item 1 deste novo conjunto temos que alterar a restrição 2.6, substituindo a desigualdade por uma igualdade, ou seja, a nova restrição 2.11. Esta alteração garante que exatamente m rotas sejam construídas.

$$\sum_{j=1}^{n} x_{0jk} = 2 \quad (k = 1, \ldots, m). \tag{2.11}$$

Para atender ao item 4 do novo conjunto de restrições, incluímos um novo dado de entrada, o número r. Substituímos na formulação as restrições 2.7 e 2.8 pela nova restrição 2.12. Esta alteração garante que a diferença entre o número de vértices de diferentes rotas seja limitado por uma constante.

$$|\sum_{h=1}^{n} y_{hp} - \sum_{h=1}^{n} y_{hq}| \leq r \quad (p = 1, \ldots, m; \quad q = 1, \ldots, m; \quad p \neq q; \quad r \in \mathbb{N}) \tag{2.12}$$

Temos pouca esperança de que a variável x_{0jk} assuma valor 2, pois uma viagem de retorno utilizando a mesma aresta (υ_0, υ_j) para algum $j = 1, \ldots, n$ só seria admitida se uma viatura visitasse apenas um local dentro da sua região de patrulhamento, o que normalmente não ocorre, devido ao tamanho da área a ser coberta e o número reduzido de viaturas.

Devido às alterações na formulação m-PRC, outras foram sugeridas nas heurísticas estudadas que estão descritas nos Capítulos 3 e 4. As principais mudanças foram feitas em relação às questões de capacidade e para tentar garantir um equilíbrio entre a quantidade de vértices nas rotas.

Capítulo 3

Heurísticas

3.1 Introdução

Apresentamos neste capítulo as três principais heurísticas usadas para resolver o m-PRC que foram estudadas e estão descritas em Hachicha, Hodgson, Laporte e Semet [16]. Para tal resolução a Heurística da Economia, a Heurística da Varredura e a Heurística Primeiro 1-PRC/Segundo m-PRC, como são conhecidas, utilizam uma combinação de outras heurísticas, sendo que cada uma dessas outras heurísticas resolve um problema de programação linear conhecido, durante o processamento das heurísticas principais.

As heurísticas que estão combinadas e o problema de programação linear que elas resolvem são: Heurística GENI [13] usada na construção de um ciclo hamiltoniano para o Problema do Caixeiro Viajante (PCV), a Heurística Primal1 [4] usada para encontrar uma boa solução do Problema de Cobertura de Conjuntos (PC), e outras duas de pós-otimização: a US [13] usada para melhorar uma solução do PCV e a 2-opt* [16] usada para melhorar uma solução do m-PRC. No entanto, a combinação da GENI com a US é conhecida como GENIUS, e a combinação desta última com a Primal1 é conhecida como H-1-PRC [16], a qual é usada para construir uma solução do problema 1-PRC.

Cada iteração ou teste de solução das heurísticas: da economia, da varredura e primeiro 1-PRC/segundo m-PRC resolve o problema m-PRC por um procedimento que se divide em três fases principais. O quadro ilustrado na Figura 3.1 representa essas três fases, qual heurística atua em cada fase e como estão combinadas. Nele, o retângulo que representa a Fase 1 mostra que um exemplar é processado de acordo com uma diferente heurística: da economia, da varredura ou primeiro 1-PRC/segundo m-PRC, ou seja, o lado esquerdo do primeiro retângulo informa que temos três alternativas de heurística a ser usada. Cada uma delas busca nesta fase, conforme uma estratégia particular, selecionar os vértices em rotas iniciais viáveis. Após essa seleção, o retângulo que representa a Fase 2 mostra que a heurística H-1-PRC, a qual é combinação da heurística Primal1 e GENIUS, constrói cada rota individualmente processando os conjuntos de vértices separados adequadamente na Fase 1. Por fim, o quadro que representa a Fase 3, mostra que as rotas construídas na Fase 2 são unidas formando uma solução para o m-PRC. Então essa solução é pós-otimizada pela

heurística 2-opt*, a qual tenta melhorá-la através de trocas de arestas. De acordo com a heurística da economia, da varredura ou primeiro 1-PRC/segundo m-PRC, e conforme a dimensão do exemplar avaliado, essas três fases são repetidas um determinado número de vezes. Então a melhor solução obtida entre essas repetições é selecionada como a solução final de exemplar de acordo com a respectiva heurística usada.

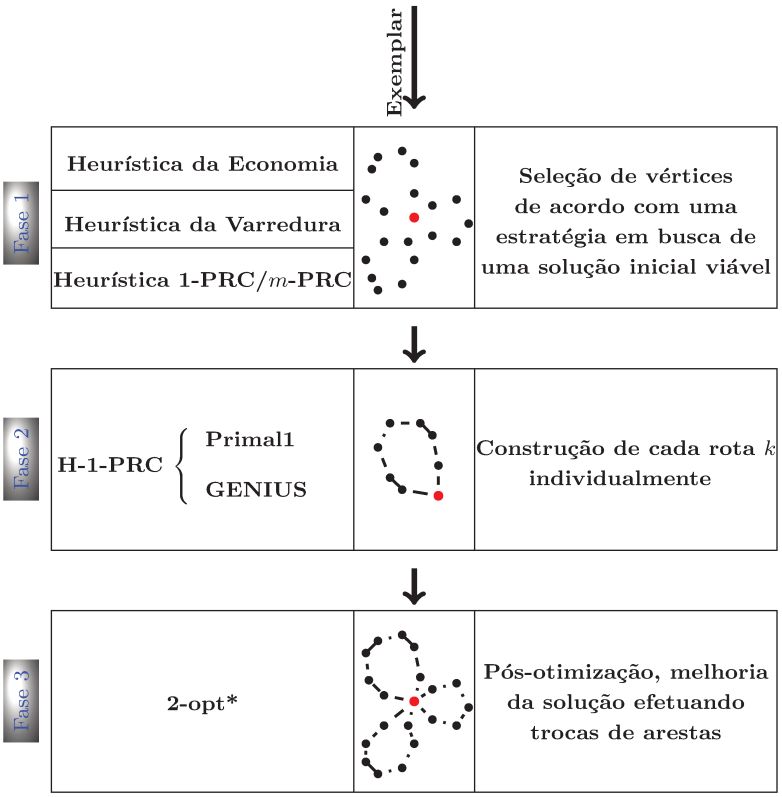


Figura 3.1: Processo de resolução de um exemplar do m-PRC em três fases principais

Nas quatro primeiras seções, que seguem neste capítulo, descrevemos sucessivamente cada problema de programação envolvido, a heurística que o resolve e como estão relacionados com as heurísticas principais para a solução do m-PRC, sendo que essas últimas estão descritas na Seção 3.5. As heurísticas que foram denominadas da economia, varredura e primeiro 1-PRC/segundo m-PRC, recebem esses nomes devido a forma que cada uma delas analisa e seleciona os vértices do exemplar na Fase 1.

3.2 Cobertura de Conjuntos

Apresentamos nesta seção a formulação e as características do PC, a heurística Primal1 associada a este problema e sua relação com o m-PRC.

3.2.1 Problema de Cobertura de Conjunto

Dado uma coleção de subconjuntos $\mathcal{F}$ de N, podemos definir um Problema de Cobertura de Conjuntos, como segue. Seja $M = \{1, \ldots, m\}$ um conjunto finito e seja $\{M_j\}_{j \in N}$, $N = \{1, \ldots, n\}$, uma coleção de subconjuntos de M. Dizemos que os subconjuntos $F \subseteq N$ cobre M se $\bigcup_{j \in F} M_j = M$. No problema em questão $\mathcal{F} = \{F : F \text{ cobre } M\}$. Esses problemas são frequentemente formulados como problemas de programação inteira binário.

Seja $A_{m \times n}$ a matriz de incidência da coleção $\{M_j\}_{j \in N}$, tal que, $a_{ij} = 1$ se $i \in M_j$, e $a_{ij} = 0$ caso contrário. Analogamente, uma seleção de conjuntos da coleção $\{M_j\}_{j \in N}$ pode ser representado por um vetor binário de dimensão n. Então a seleção $x \in B^n$ corresponde a uma cobertura se satisfaz $Ax \geq \mathbf{1}$, onde $\mathbf{1}$ é o vetor m-dimensional com componentes iguais a 1. Sendo $c = (c_j)$ um vetor n-dimensional o custo de M_j, desejamos uma cobertura de custo mínimo, ou seja, $\min\{cx : Ax \geq \mathbf{1}, x \in \{0,1\}^n\}$.

No nosso problema em estudo a matriz de incidência do problema de cobertura associado ao conjunto W é construída definindo para cada linha l ($\upsilon_l \in W$) e cada coluna h ($\upsilon_h \in V$) um coeficiente binário δ_{lh} igual a 1 se $c_{lh} \leq c$, e 0 caso contrário.

3.2.2 Heurística Primal1

Em Balas e Ho [4] encontramos alguns algoritmos para resolver o PC usando planos de cortes, subgradientes e heurísticas. As heurísticas são denominadas PRIMAL e DUAL, por explorar o par primal-dual do problema. Nesta subseção, identificamos e explicamos como funciona a heurística Primal1, a qual está entre as heurísticas chamadas de PRIMAL. Considere o problema de cobertura de conjuntos:

$$(PC) \qquad \min\{cx \mid Ax \geq \mathbf{1}, x \in \{0,1\}^n\}$$

onde $A_{m \times n} = (a_{ij})$ é uma matriz binária, e $\mathbf{1} = (1, \ldots, 1)$ tem m componentes. Tome ainda os conjuntos $M = \{1, \ldots, m\}$ de linhas e $N = \{1, \ldots, n\}$ de colunas. Denotamos,

$$M_j = \{i \in M \mid a_{ij} = 1\}, j \in N; \quad N_i = \{j \in N \mid a_{ij} = 1\}, i \in M;$$

chamaremos de *cobertura* um vetor $x \in \{0,1\}^n$ que satisfaz $Ax \geq \mathbf{1}$ e $S(x) = \{j \in N \mid x_j = 1\}$ o seu *suporte*. Procuramos uma cobertura tal que o seu suporte tenha a menor cardinalidade possível, e vamos chamá-la de *boa cobertura*.

As heurísticas que são usadas em Balas e Ho [4] para gerar boas coberturas são do tipo gananciosa, e constroem uma cobertura através de uma sequência de passos. Cada passo consiste da seleção de uma variável x_j que minimiza uma certa função f de coeficientes de x_j. Essas heurísticas recebem diferentes nomes de acordo com a função f usada para avaliar as variáveis durante o seu processamento. Mais adiante, vamos identificar nessa diferenciação a heurística Primal1. Se k_j denota o número de coeficientes positivos de x_j naquelas linhas da matriz A, do conjunto de restrições avaliadas e ainda não cobertas, a forma geral de avaliar a função f é $f(c_j, k_j)$.

As heurísticas consideram apenas um subconjunto de variáveis de cada vez, por ser mais barato computacionalmente, mas como toda linha de A deve ser coberta de qualquer maneira, uma cobertura para ser construída deve conter no mínimo uma das variáveis tendo coeficiente positivo em uma dada linha. Sendo assim, as heurísticas restringem a escolha em cada passo ao conjunto de variáveis tendo um coeficiente positivo em alguma linha específica $i^* \in M$. Denotando por R o conjunto de linhas ainda não cobertas e por S o suporte do vetor x atual, os passos da heurística usada podem ser descritos como segue.

Heurística PRIMAL

Passo 1. Faça $R = M$, $S = \emptyset$, $t = 1$, e vá para o Passo 2.

Passo 2. Se $R = \emptyset$, vá para o Passo 3. Caso contrário, tome $k_j = |M_j \cap R|$, escolha $i^* \in R$, tal que $|N_{i^*}| = \min_{i \in R} |N_i|$, e escolha $j(t)$ tal que

$f(c_{j(t)}, k_{j(t)}) = \min_{j \in N_{i^*}} f(c_j, k_j)$.

Faça $R \leftarrow R \setminus M_{j(t)}$, $\quad S \leftarrow S \cup \{j(t)\}$, $\quad t \leftarrow t + 1$, e vá para o Passo 2.

Passo 3. Considere os elementos $i \in S$ em ordem, e se $S \setminus \{i\}$ é o suporte de uma cobertura, faça $S \leftarrow S \setminus \{i\}$. Quando todos $i \in S$ tiverem sido considerados, S define uma boa cobertura.

Podem ser usadas as seguintes funções f:

1. $f(c_j, k_j) = c_j$;
2. $f(c_j, k_j) = \frac{c_j}{k_j}$;
3. $f(c_j, k_j) = \frac{c_j}{\log_2 k_j}$;
4. $f(c_j, k_j) = \frac{c_j}{k_j \log_2 k_j}$;
5. $f(c_j, k_j) = \frac{c_j}{k_j \ln k_j}$.

No caso 3 e 4, $\log_2 k_j$ é substituído por 1 quando $k_j = 1$; e no caso 5, $k_j \ln k_j$ é substituído por 1 quando $k_j = 1$ ou 2. Temos que k_j não assume valor nulo, pois como $|N_{i^*}| \geq 1$ e, para $j \in N_{i^*}$, $M_j \cap R$ contém pelo menos o índice i^*, por construção, segue que $k_j = |M_j \cap R| \geq 1$.

O caso 1 é simplesmente a escolha da variável de menor custo da variável em cada passo. No caso 2 passamos a dar maior peso de escolha àquela variável que tem maior número de linhas com coeficientes positivos ainda não cobertos, pois estamos minimizando a função f. Nos casos 3, 4 e 5 aumentamos gradativamente este peso de escolha.

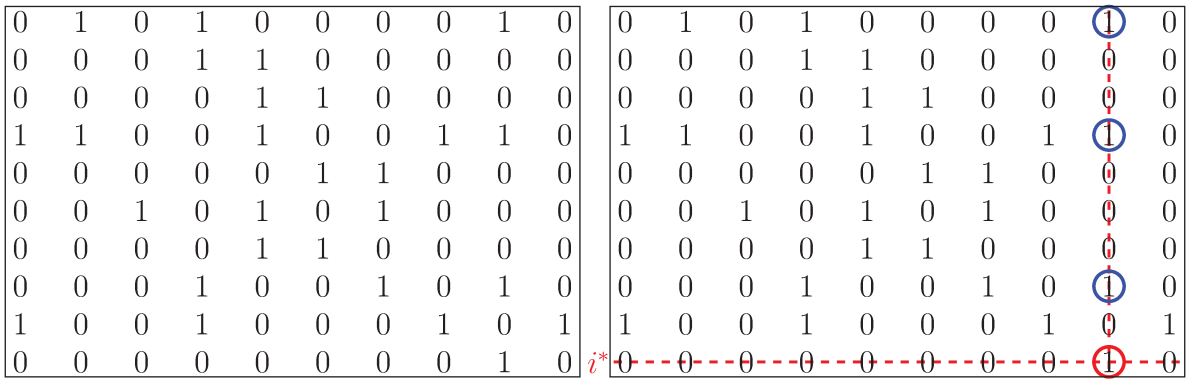


Figura 3.2: Primeira iteração da heurística PRIMAL

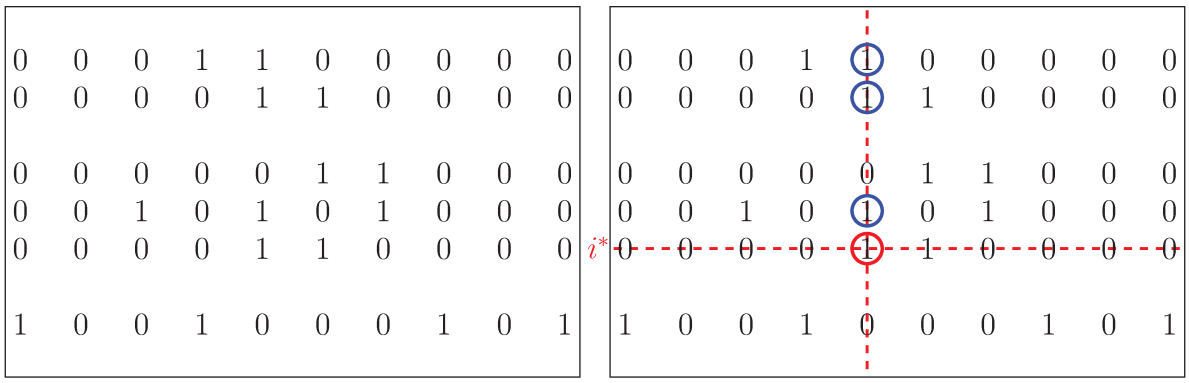


Figura 3.3: Segunda iteração da heurística PRIMAL

Observando o exemplo ilustrado nas Figuras 3.2, 3.3 e 3.4, temos uma idéia mais clara do procedimento descrito. Nestas figuras as matrizes contém as linhas da matriz original cujos índices pertencem a R.

A Figura 3.2 ilustra uma matriz quadrada de dimensão 10, que representa a matriz A de um PC. Para entender a heurística PRIMAL aplicada a este exemplo, basta observar o comportamento da matriz de incidência A. Quando usamos a função do item 3, a heurística escolhe primeiramente como i^* a décima linha da matriz, como ela tem apenas a variável x_9 com coeficiente positivo igual a 1 na décima linha, ao minimizar a função f a variável x_9 é escolhida para entrar no suporte, e o conjunto M_9, que contém as linhas 1, 4, 8 e 10, é excluído do conjunto de linhas, pois já estão cobertos pela variável x_9. A matriz resultante desta exclusão está exibida no lado esquerdo da Figura 3.3.

Ainda na Figura 3.3, a nova iteração da heurística escolhe como i^*, por exemplo, a sétima linha da matriz. Note que k_5 é maior que k_6 e, supondo que $c_5 = c_6$, neste passo é escolhida a variável x_5 para entrar no suporte. Consequentemente as linhas 2, 3, 6 e 7 que pertencem ao conjunto M_5 serão eliminadas do conjunto de linhas.

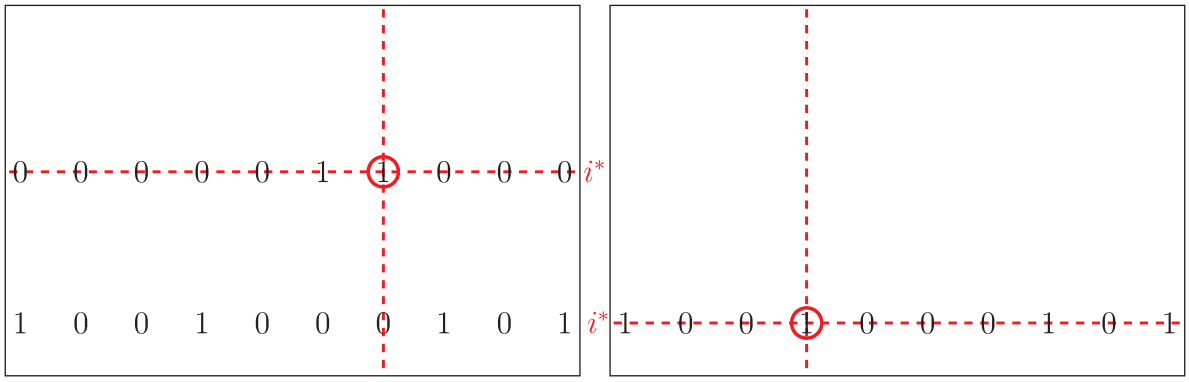


Figura 3.4: Terceira iteração da heurística PRIMAL

No lado esquerdo da Figura 3.4 vemos que restaram apenas duas linhas. Claramente deve ser escolhido como i^* a quinta linha. Supondo que o $c_7 \leq c_6$, já que nesta etapa a minimização é feita apenas comparando o custo c_j, a variável x_7 entra no suporte e apenas a quinta linha é eliminada, pois esta é coberta pela variável x_7.

E por fim, no lado direito da Figura 3.4, ao supor que o menor entre os custos c_1, c_4, c_8 e c_{10} seja c_4, a variável x_4 entra no suporte e a última linha do conjunto de restrições é coberta. Portanto neste exemplo arbitrário, o conjunto $\{x_9, x_5, x_7, x_4\}$, representado pelo vetor $x = (0, 0, 0, 1, 1, 0, 1, 0, 1)$, é a cobertura construída pela heurística PRIMAL.

Balas e Ho [4] afirmam que, nos piores casos, quando se usou apenas uma função f durante o processo de minimização, nunca o desvio do ótimo superou 10,8%. Contudo os resultados se aproximam do ótimo quando se muda a maneira de se obter a cobertura ou quando há uma combinação das funções.

A combinação é feita da seguinte forma: primeiro toma-se uma função f e usa-se o procedimento acima até se obter uma cobertura, depois considerando a ordem em que as variáveis foram incluídas na cobertura, retira-se todas aquelas que estão associadas a uma multicobertura (ou como em [4], quando uma restrição correspondente é satisfeita com folga). Em seguida, o processo é refeito com outra opção de f até se obter uma nova cobertura. Não se refaz mais o processo quando não há mais multicoberturas, ou quando um número inteiro σ é atingido.

O procedimento chamado de heurística Primal1 é obtido quando tomamos $\sigma = 3$ e usamos a função 3 para a primeira cobertura, uma diferente função (1 ou 2) para a segunda cobertura, e novamente uma terceira função (1 ou 2), diferente da segunda, para a terceira cobertura.

3.2.3 Primal1 e o m-PRC

Uma das tarefas do problema m-PRC é garantir a cobertura de um determinado conjunto W. Durante a construção das rotas de cobertura temos que incluir vértices de V, que cobrem

os vértices de W.

Temos na formulação do m-PRC, descrita no Capítulo 2, os conjuntos $S_l = \{v_h \in V : c_{hl} \leq c\}$ para todo $v_l \in W$. Em um determinado momento desta construção devemos escolher, dentre uma certa quantidade de vértices em V, um vértice v_h para incluir em uma rota. Então é criada uma matriz, tal que seu coeficiente δ_{hl} é igual a 1 se $v_l \in W$ pode ser coberto por $v_h \in V$, ou seja, $c_{hl} \leq c$, e 0 caso contrário. Também é feita uma lista incluindo cada custo c_h, referente ao custo de inclusão de cada vértice $v_h \in S_l$ na rota. Desta forma, usando esta matriz de coeficientes binários e os custos de inclusão de vértices (que estão neste caso associados às arestas), a heurística Primal1 é usada para decidir qual vértice será incluso em uma rota. Em Hachicha, Hodgson, Laporte e Semet [16] são usadas duas sequências distintas (3-1-2 e 3-2-1) de escolha das funções f (indicadas na seção anterior) durante a construção de uma rota, e selecionada a melhor rota entre elas no processo final.

3.3 Caixeiro Viajante

Apresentamos nesta seção a formulação e as características do PCV, as heurísticas GENI, US e GENIUS associadas a este problema e suas relações com o m-PRC. Também, no final desta seção, apresentamos uma tabela com testes feitos em exemplares do PCV para validar o uso das heurísticas citadas.

3.3.1 Problema do Caixeiro Viajante

O Problema do Caixeiro Viajante (PCV) é um dos problemas amplamente estudados em otimização combinatória. Ele pode ser definido como em Gendreau, Hertz e Laporte [13]. Seja $G = (V, A)$ um grafo, onde $V = \{v_1, \ldots, v_n\}$ é o conjunto de vértices e $A = \{(v_i, v_j) : v_i, v_j \in V\}$ é o conjunto de arcos. A cada arco (v_i, v_j) está associado um custo não negativo c_{ij}. O PCV consiste em encontrar o circuito de custo mínimo passando por todos os vértices exatamente uma vez. Tal circuito é conhecido como Hamiltoniano. O problema é simétrico se e somente se $c_{ij} = c_{ji}$ para todo i, j, e satisfaz a desigualdade triangular se e somente se $c_{ij} + c_{jk} \geq c_{ik}$ para todo i, j, k. Um caso particular denominado *Euclidiano* ocorre quando os custos correspondem a distância euclidiana.

A interpretação mais comum para o PCV é aquela em que um vendedor deve visitar n cidades percorrendo uma rota que inicia e termina numa mesma cidade, sendo que cada visita é feita exatamente uma vez. O custo ou tempo de viagem de uma cidade i para uma cidade j é c_{ij}. Desejamos encontrar a ordem na qual ele deveria fazer esta rota, para completá-la o mais rápido ou com menos custo possível. Este problema ocorre em uma multiplicidade de formas, por exemplo: um carro de entregas tem uma lista de clientes e ele deve visitá-los em um determinado dia, levando em conta o custo da viagem. É possível com frequência resolver PRVs adaptando algoritmos usados para resolver o PCV. No PRV um dos vértices do problema é designado como sendo o depósito e um número de rotas de veículos de menor custo deve ser desenhadas, partindo do depósito para o restante dos vértices, sujeito a algumas restrições.

É justamente um tipo de adaptação que será feita com a heurística GENIUS, construída para resolver o PCV, para buscar boas soluções para o m-PRC.

Uma formulação de programação linear inteira binária para o PCV, que mostra o conjunto de restrições é descrita em Wolsey [30], como segue.

Definição das variáveis.

$$x_{ij} = \begin{cases} 1, & \text{se o vendedor viaja diretamente da cidade } i \\ & \text{para a cidade } j, \\ 0, & \text{caso contrário.} \end{cases}$$

x_{ii} não está definido para $i = 1, \ldots, n$.

Definição das restrições.

- Ele deixa a cidade i exatamente uma vez:

$$\sum_{j:j\neq i} x_{ij} = 1 \text{ para } i = 1, \ldots, n.$$

- Ele chega na cidade j exatamente uma vez:

$$\sum_{i:i\neq j} x_{ij} = 1 \text{ para } j = 1, \ldots, n.$$

- Vetores que satisfazem as restrições anteriores, conhecidas como designação, podem produzir um conjunto de subrotas desconectadas, como ilustrado na Figura 3.5. Precisamos assim de mais restrições que garantam conectividade, impondo que o vendedor deva passar de um conjunto de cidades para outro. Para isso, existem algumas possibilidades na literatura, por exemplo:

  cut-set

$$\sum_{i\in S}\sum_{j\notin S} x_{ij} \geq 1 \text{ para todo } S \subset N, S \neq \emptyset.$$

  e **subtour elimination**

$$\sum_{i\in S}\sum_{j\in S} x_{ij} \leq |S| - 1 \text{ para todo } S \subset N,\ 2 \leq |S| \leq n-1.$$

Definição da função objetivo.

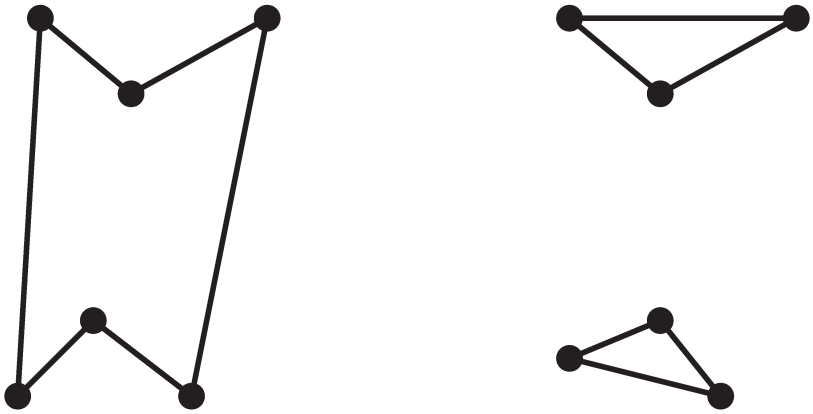


Figura 3.5: Ocorrência de subrotas desconectadas que satisfazem as restrições de designação

- O soma do tempo ou custo total da viagem é minimizado:

$$z = \min \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} c_{ij} x_{ij}.$$

3.3.2 PCV e o m-PRC

Com relação ao m-PRC, após decidido quais vértices entram em cada rota, temos exatamente que construir um ciclo Hamiltoniano de tamanho mínimo para cada rota, assim torna-se importante a formulação do PCV e conhecer uma boa heurística para resolvê-lo, devido a sua complexidade.

Heurísticas para o PCV podem ser classificadas em *procedimento de construção de rota*, a qual consiste em construir gradualmente uma solução ao adicionar um novo vértice em cada passo, e *procedimento de melhoria da rota*, a qual tem a finalidade de melhorar uma solução viável ao efetuar trocas de vértices testando várias configurações. Os melhores métodos combinam essas duas características em um procedimento de duas fases. Por conta disso, em Hachicha, Hodgson, Laporte e Semet [16] escolheu-se usar a heurística GENIUS, que é um método de duas fases extremamente eficiente e encontra uma boa solução para o PCV.

3.3.3 Heurística GENI

Criada por Gendreau, Hertz e Laporte [13], a heurística *Generalized Insertion Procedure* (GENI) demonstra ser igualmente eficiente para problemas simétricos e assimétricos. A característica principal desta heurística é que a inserção de um vértice v, não necessariamente ocorre entre dois vértices adjacentes da rota. No entanto, após sua inserção, aqueles vértices

escolhidos tornam-se adjacentes a v. Os vértices são escolhidos um a um de forma arbitrária para inserção a partir de uma rota inicial. Suponha que desejamos inserir v entre algum par v_i e v_j de vértices da rota. Para uma dada orientação da rota, seja v_k um vértice no caminho de v_j para v_i, e v_l um vértice no caminho de v_i para v_j. Se um vértice v_h está na rota, vamos chamar os seus dois vértices adjacentes por v_{h-1}, referindo-se ao seu predecessor e v_{h+1}, ao seu sucessor. A inserção de v entre v_i e v_j pode ser feita de duas maneiras. Vamos classificá-las de Tipo 1 e Tipo 2. A Figura 3.6 ilustra essas duas possibilidades. Na sua parte superior temos um exemplo de inserção do Tipo 1, e na parte inferior uma do Tipo 2. Nela observamos configurações que são dadas a partir de uma escolha de orientação da rota. As linhas contínuas representam caminhos entre dois extremos, enquanto as pontilhadas representam vértices consecutivos.

Inserção Tipo 1

Neste primeiro tipo temos que tomar $v_k \neq v_i$ e $v_k \neq v_j$. Inserir v na rota resulta em eliminar os arcos (v_i, v_{i+1}), (v_j, v_{j+1}), e (v_k, v_{k+1}) e em substituí-los pelos arcos (v_i, v), (v, v_j), (v_{i+1}, v_k) e (v_{j+1}, v_{k+1}). Isto significa que os dois caminhos $(v_{i+1}, \ldots, v_j)$ e $(v_{j+1}, \ldots, v_k)$ são invertidos. Note que tomando $j = i + 1$ e $k = j + 1$ leva a um procedimento de inserção entre vértices adjacentes da rota. O custo deste último também é testado e comparado com os demais.

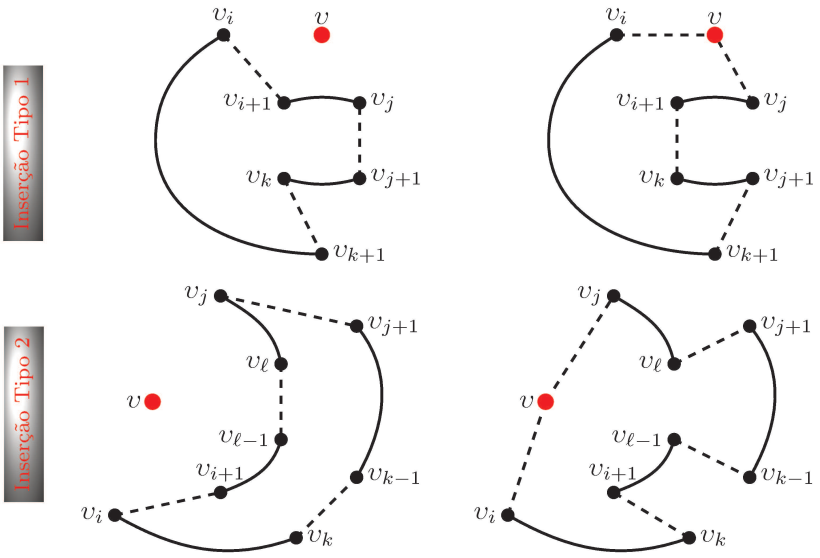


Figura 3.6: Inserção Tipo 1 e Tipo 2 de v entre v_i e v_j

Inserção Tipo 2

Neste segundo tipo temos que tomar $v_k \neq v_j$ e $v_k \neq v_{j+1}$; $v_l \neq v_i$ e $v_l \neq v_{i+1}$. Inserir v na rota resulta em eliminar os arcos $(v_i, v_{i+1}), (v_{l-1}, v_l), (v_j, v_{j+1})$ e (v_{k-1}, v_k) e em substituí-los pelos arcos $(v_i, v), (v, v_j), (v_l, v_{j+1}), (v_{k-1}, v_{l-1})$ e (v_{i+1}, v_k). Com isso os dois caminhos $(v_{i+1}, \ldots, v_{l-1})$ e $(v_l, \ldots, v_j)$ são invertidos.

A heurística GENI considera as duas possibilidades de orientações da rota para cada possibilidade de inserção de vértices. Temos que o número potencial de escolhas para v_i, v_j, v_k e v_l é grande, então são limitadas as formas de escolhas desses vértices, como segue.

Para algum vértice $v \in V$, define-se sua p-vizinhança $N_p(v)$ como sendo o conjunto dos p vértices na rota mais próximos de v, com respeito as distâncias c_{ij}'s; se v tem menos que p vizinhos na rota, eles todos pertencem a $N_p(v)$. Assim, para um dado parâmetro p, primeiro selecionamos $v_i, v_j \in N_p(v)$, $v_k \in N_p(v_{i+1})$, e $v_l \in N_p(v_{j+1})$. Na prática, p é um número relativamente pequeno. Agora os passos da heurística GENI podem ser descritos.

Heurística GENI

Passo 1. Selecione um subconjunto de três vértices arbitrariamente e crie uma rota inicial. Inicialize a p-vizinhança de todos os vértices.

Passo 2. Escolha arbitrariamente um vértice v fora da rota e calcule o menor custo de inserção de v considerando as duas possibilidades de orientação da rota e os dois tipos de inserção. Refaça a p-vizinhança de todos os vértices, considerando o fato de que v está agora na rota.

Passo 3. Se todos os vértices já fazem parte da rota, pare. Caso contrário vá para o Passo 2.

No Passo 2, a escolha arbitrária de um novo vértice para inserção na rota, pode ser feita de várias maneiras. Na nossa implementação, em cada iteração incluímos todos os vértices que faltam entrar na rota em um vetor e através de uma permutação aleatória dos componentes, escolhemos o próximo vértice a ser inserido.

O que faz essa heurística eficiente é o fato de que o número de inserções potenciais é restringido pela vizinhança do vértice a ser adicionado na rota. Também, a busca de melhoria está condicionada a configurações promissores, pois os arcos introduzidos ligam vértices que estão próximos um do outro. Vale notar que, mesmo em problemas euclidianos, inserir um vértice novo na rota pode às vezes resultar em um menor custo. Isto nunca acontece em procedimentos de inserção padrão, ou seja, entre dois vértices adjacentes da rota. Para ilustrar, considere o exemplo de [13] ilustrado na Figura 3.7. Neste exemplo, $p = 4$ e os vértices são inseridos na rota em ordem numérica crescente. Um fenômeno interessante ocorre quando o vértice 6 é inserido: o procedimento então produz um rota que

se cruza. Isto ocorre porque a melhor inserção disponível, entre os vértices 1 e 2, é inviável, já que o vértice 1 não pertence a $N_4(6)$. Ocorre também uma redução de custo, de 238.6 para 229.0, quando o vértice 8 é inserido. Neste exemplo a solução final não é a ótima. A melhor rota $(1, 7, 6, 2, 3, 5, 8, 4, 1)$ tem um custo de 223.0 e pode ser obtida com $p = 5$.

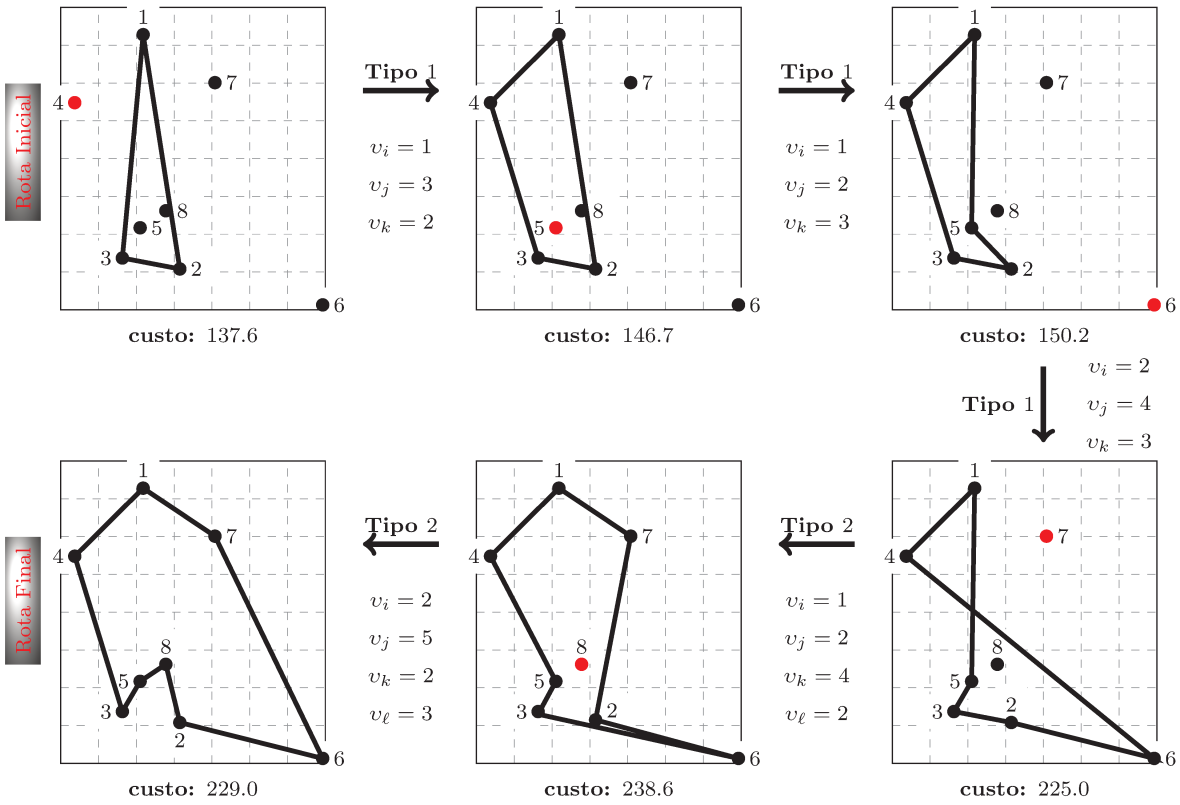


Figura 3.7: Heurística GENI aplicada em um exemplo com 8 vértices

3.3.4 Heurística US

Foi desenvolvida também por Gendreau, Hertz e Laporte [13] uma heurística de pós-otimização buscando melhorar a solução de um PCV, a qual consiste em remover um vértice de uma rota viável e inseri-lo novamente. Esses processos são chamados de *Unstringing* e *Stringing* (US) e podem ser aplicados em rotas produzidas por qualquer heurística. O procedimento de inserção do vértice é feito de maneira idêntica ao Passo 2 da heurística GENI. O procedimento de remoção é o inverso. Quando um vértice v_i é removido da rota, temos que considerar duas maneiras de reconectá-la. Vamos classificá-las de Tipo 1 e Tipo 2. A Figura 3.8 ilustra essas duas possibilidades. Na sua parte superior temos um exemplo de remoção do Tipo 1, e na parte inferior uma do Tipo 2. Nela, observamos configurações que são dadas a partir

de uma escolha de orientação da rota. As linhas contínuas representam caminhos entre dois extremos, enquanto as pontilhadas representam vértices consecutivos.

Remoção Tipo 1

Seja $v_j \in N_p(v_{i+1})$, e, para uma dada orientação da rota seja $v_k \in N_p(v_{i-1})$ um vértice no caminho $(v_{i+1}, \ldots, v_{j-1})$. Então os arcos $(v_{i-1}, v_i), (v_i, v_{i+1}), (v_k, v_{k+1})$ e (v_j, v_{j+1}) são eliminados, enquanto os arcos $(v_{i-1}, v_k), (v_{i+1}, v_j)$ e (v_{k+1}, v_{j+1}) são inseridos. Os dois caminhos $(v_{i+1}, \ldots, v_k)$ e $(v_{k+1}, \ldots, v_j)$ mudam de orientação.

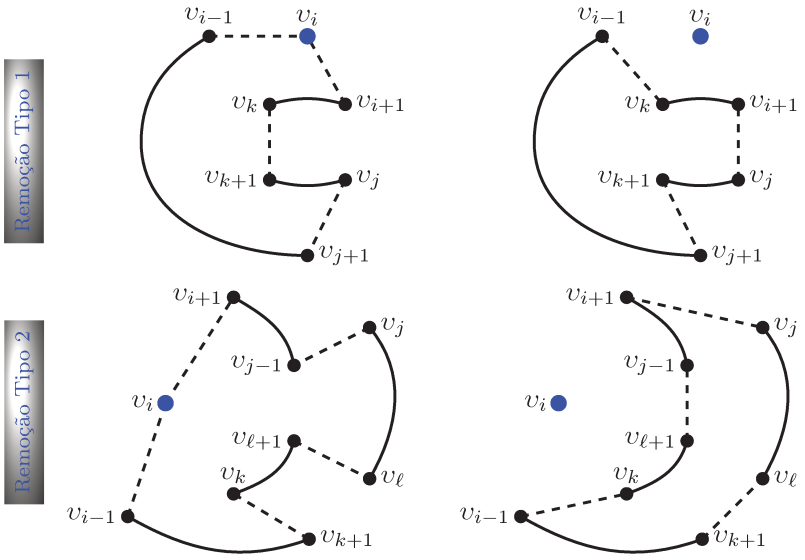


Figura 3.8: Remoção Tipo 1 e Tipo 2 do vértice v_i da rota

Remoção Tipo 2

Seja $v_j \in N_p(v_{i+1})$, para uma dada orientação da rota, seja $v_k \in N_p(v_{i-1})$ um vértice no caminho $(v_{j+1}, \ldots, v_{i-2})$. Também seja $v_l \in N_p(v_{k+1})$ no caminho $(v_j, \ldots, v_{k-1})$. Então os arcos $(v_{i-1}, v_i), (v_i, v_{i+1}), (v_{j-1}, v_j), (v_l, v_{l+1})$ e (v_k, v_{k+1}) são eliminados, enquanto os arcos $(v_{i-1}, v_k), (v_{l+1}, v_{j-1}), (v_{i+1}, v_j)$ e (v_l, v_{k+1}) são inseridos. Os dois caminhos $(v_{i+1}, \ldots, v_{j-1})$ e $(v_{l+1}, \ldots, v_k)$ mudam de orientação. Considerando um PCV com n cidades, os passos da heurística US podem ser descritos.

Heurística US

Passo 1. Considere uma rota inicial τ de custo z. Faça $\tau^* := \tau, z^* := z$ e $t := 1$.

Passo 2. Iniciando da rota τ, aplique o procedimento de remoção e inserção com o vértice v_t, considerando em cada caso os dois tipos possíveis de operação e as duas orientações da rota. Seja $\tilde{\tau}$ a rota obtida e $\tilde{z}$ seu custo . Faça $\tau := \tilde{\tau}$ e $z := \tilde{z}$.

- Se $z < z^*$, faça $\tau^* := \tau, z^* := z$ e $t := 1$; repita o Passo 2.
- Se $z \geq z^*$, faça $t := t + 1$.
- Se $t = n + 1$, Pare: a melhor rota disponível é τ^* e seu custo é z^*. Caso contrário, repita o Passo 2.

Realmente, um vértice para ser removido de uma rota não precisa necessariamente estar localizado entre dois vértices pertencentes à sua p-vizinhança. Assim, reinserir um vértice na posição que ele ocupava antes de sua remoção da rota, pode não ser permitido. Porém, a melhor rota conhecida sempre é armazenada.

3.3.5 Heurística GENIUS

Este procedimento se divide em duas fases, conhecidas como *Generalized Insertion Procedure* (**GENI**), que é a inserção de vértices na rota. E *Unstringing and Stringing* (**US**), um procedimento de pós-otimização, que tenta melhorar o custo de uma rota, removendo vértices e inserindo-os novamente. Ambos os procedimentos foram descritos com detalhe nas duas seções anteriores.

Note que as formas de inserções e remoções de vértices não permitem em nenhum momento criação de subrotas. Isto é um fato muito importante para garantir um resultado final com uma solução viável.

Heurística GENIUS

Passo 1. Construa um ciclo Hamiltoniano τ de custo z usando a heurística **GENI**.

Passo 2. Aplique a heurística **US** sobre τ.

Após a implementação desta heurística e feitos alguns testes com exemplares do PCV verificou-se sua grande eficiência na construção de rotas, assim como mostra a Tabela 3.1. Ela apresenta os resultados obtidos usando como entrada alguns exemplares retirados da biblioteca on-line TSPLIB [10]. A tabela se divide em duas partes, uma aplicando apenas a heurística GENI e a outra usando a pós-otimização US. Nela, $p = 4$ é o parâmetro da p-vizinhança. Nas suas colunas, N indica o número de cidades ou dimensão do exemplar, MS é a melhor solução encontrada, %O indica o desvio da solução ótima, enquanto SO é o valor da solução ótima do exemplar. O tempo é dado em segundos. Note que, apenas com o uso da GENI, o desvio da solução ótima não ultrapassa os 10,34 % e com a pós-otimização todos estão abaixo dos 8%. Mas os resultados dos exemplares de dimensões pequenas são muito bons. Veremos mais adiante, como usar esta heurística combinada com outras, na construção das rotas do m-PRC.

Foram feitos cinco testes para cada exemplar usando a heurística GENI na construção do ciclo Hamiltoniano, com uma escolha aleatória de vértices para inserção e a melhor solução foi inserida na tabela. A mesma quantidade de testes foram feitos para a GENIUS. Esta estratégia de escolha de vértices para inserção na rota repetidas vezes aumentam as chances de encontrarmos soluções melhores a cada tentativa. Um exemplo é o problema `bays29`, que no ótimo a função objetivo tem valor 2020. Na escolha de vértices em ordem numérica crescente o resultado final para função objetivo é 2072 e com outras escolhas arbitrárias ficou muito próximo do ótimo, com valor 2026. Porém, para este exemplo em questão, oito testes resultaram na solução ótima, isto sem a fase de pós-otimização.

Todos exemplares testados são simétricos, pois foi o critério usado durante a implementação, pensando em resolver exemplares para o m-PRC simétricos, porém esta heurística também funciona bem para problemas assimétricos, como é descrito em [13]. Os exemplares foram processados em código Matlab em um micro com processador Intel Celeron de 1.73GHz, 512 MB de memória RAM e sistema operacional Windows.

Observando as Tabelas 3.1 e 3.2 notamos uma diferença considerável que existe entre o tempo de processamento da implementação GENI e a GENIUS. Especialmente com relação aos exemplares `ch150`, `p654` e `pcb1173`, nos quais os processos foram interrompidos quando atingiram desvio do ótimo abaixo dos 3%, 5% e 8% respectivamente. Para exemplares de dimensão muito grande, este fato e a qualidade da aplicação GENI podem ser usados para decidir, dentro de um desvio requerido da solução ótima, se é viável aplicar a pós-otimização US.

Como ilustração, encontramos na Figura 3.9 duas configurações do PCV obtidas como solução dos exemplares `berlin52` e `ch150`, usando a heurística GENIUS.

O fato da heurística US aumentar muito o tempo, durante o processo de busca de uma boa solução, pode ser notado ao observarmos como os testes de melhoria se dão no seu método apresentado na Seção 3.3.4. São testados se a remoção e inserção de todos os vértices um a um alteram a configuração da rota trazendo melhoria ou não. Além de termos uma quantidade grande de configurações possíveis a serem testadas, após detectada uma melhora na rota, inicia-se novamente os testes em todos os vértices, inclusive os que já tínhamos testados. Este procedimento tende a melhorar uma solução inicial. No entanto, alguns testes que não

Heurística GENI, $p = 4$					
Exemplar	N	tempo	MS	%O	SO
`bays29`	29	0.47	2026.00	0.30	2020
`berlin52`	52	1.25	7799.88	3.42	7542
`ch150`	150	5.81	6845.27	4.86	6528
`p654`	654	105.36	36816.72	6.27	34643
`pcb1173`	1173	498.91	62772.60	10.34	56892
Heurística GENIUS, $p = 4$					
`bays29`	29	60.42	2020.00	0.00	2020
`berlin52`	52	235.05	7544.36	0.03	7542
`ch150`	150	12668.65	6692.97	2.53	6528
`p654`	654	50974.70	36370.15	4.98	34643
`pcb1173`	1173	151349.20	61439.30	7.99	56892

Tabela 3.1: Soluções de exemplares do PCV

Heurística GENIUS Aplicada no Exemplar `Berlin52`, $p = 4$					
Teste	Custo GENI	%O	Custo GENIUS	%O	tempo
1º	8176.66	8.41	7544.36	0.03	235.05
2º	8461.72	12.19	7974.98	5.74	382.31
3º	8654.36	14.75	8130.48	7.80	858.17
4º	8300.74	10.06	7544.36	0.03	718.37
5º	8538.63	13.21	7716.68	2.31	1471.16

Tabela 3.2: Soluções para um exemplar do PCV

retornaram melhora de solução anteriormente, são repetidos.

Concluímos que para exemplares de dimensão grande, cuja solução ótima é conhecida, podemos aplicar apenas a heurística GENI no processo de construção do ciclo Hamiltoniano, ou para melhorar o tempo da heurística US, ao invés de recomeçar os testes a partir do primeiro vértice toda vez que uma melhora é detectada, podemos continuar os testes a partir do último vértice testado. Ou ainda, interromper o processo em um determinado tempo ou desvio do ótimo, como foi feito nos exemplares `ch150`, `p654` e `pcb1173`.

3.4 Rota de Cobertura 1-veículo

Apresentamos nesta seção o problema conhecido como 1-PRC e a heurística H-1-PRC. Esta última é usada para resolver o primeiro problema, ou seja, resolve individualmente cada rota dentre as m rotas no processo de solução do m-PRC. Também apresentamos algumas alterações sugeridas na heurística H-1-PRC, e testes são apresentados para justificar tais alterações.

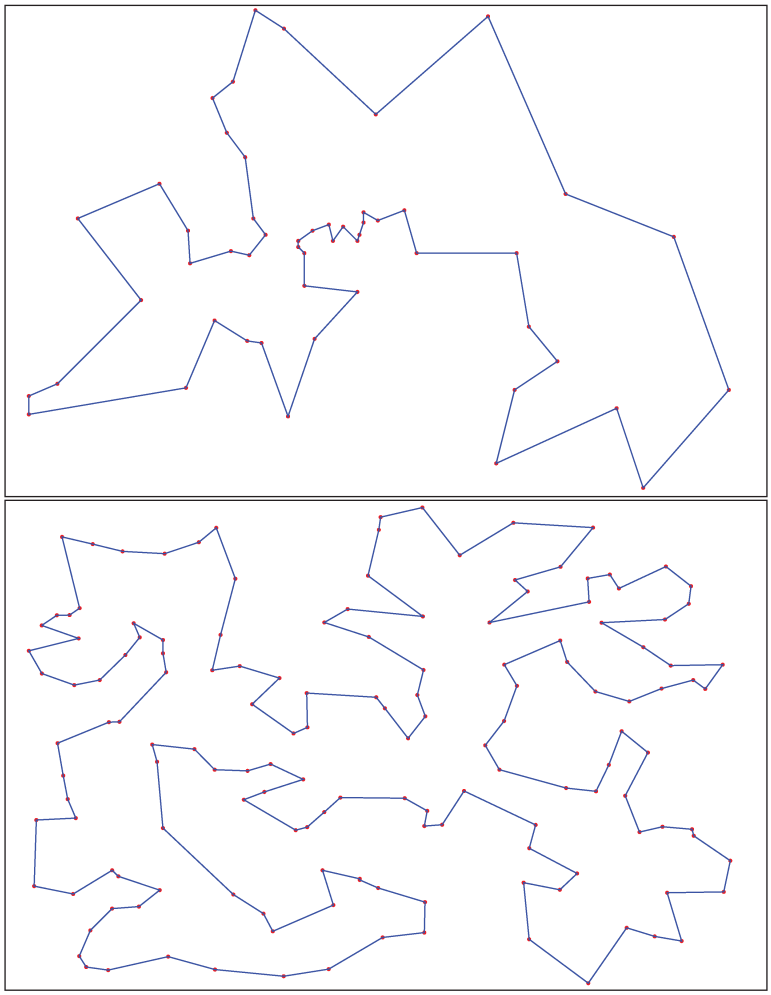


Figura 3.9: Configuração das soluções do PCV para os exemplares `berlin52` e `ch150`

3.4.1 Problema da Rota de Cobertura

O Problema de Rota de Cobertura (PRC), ou 1-PRC, nada mais é que um m-PRC com $m = 1$, então para obtermos a formulação deste problema basta tomar $m = 1$ na formulação apresentada na seção 2.2.2.

Em Current e Schilling [9] encontramos uma aplicação genérica do PRC no desenho de redes de transportes, onde todos os pontos que não estão na rota de veículo principal encontram-se dentro do alcance dela. Em ReVelle e Laporte [26] o PRC é chamado de Problema do Circo Ambulante (*Traveling Circus Problem*), porque é o problema enfrentado por um circo que faz paradas em um número de locais durante uma estação, para sempre ser acessível por populações não visitadas. Semelhante mas diferente são os Problemas de Cobertura por Caminho mais Curto (*Shortest Covering Path Problem*) estudados por Current, Pirkul e Rolland [8]; a Rota de Cobertura em um Grafo (*Tour Cover of a Graph*)

encontrado em Arkin e Hassin [3], nos quais os pesos estão nos vértices no lugar das arestas; e uma versão contínua do problema chamado *Geometric Covering Salesman Problem* descrito em Arkin, Halldórsson e Hassin [2], no qual um conjunto especificado de vértices a priori para a rota de cobertura não é discreto, mas é uma região do plano.

3.4.2 Heurística H-1-PRC

Esta heurística está descrita em Gendreau, Laporte e Semet [14] e combina a heurística Primal1 para o problema de cobertura com a heurística GENIUS para o PCV. Ela é usada para resolver cada 1-PRC no processo de construção do m-PRC. Assim, durante a aplicação da heurística H-1-PRC, temos que garantir a cobertura do conjunto W, conforme a formulação m-PRC, com $m = 1$. Para encontrar uma boa cobertura usa-se a heurística Primal1. Então, dado um conjunto de vértices nos quais queremos construir uma rota, definimos c_h como o custo de inserção de um vértice v_h na rota usando as configurações da heurística GENI. Cria-se assim, um PC para ser usado internamente no processo de construção desta rota definido da seguinte forma:

$$\text{minimizar} \quad \sum_{v_h \in V} c_h y_h$$

sujeito a

$$\sum_{v_h \in S_l} y_h \geq 1, \quad \forall v_l \in W, \quad S_l = \{v_h \in V : c_{hl} \leq c\}$$

$$y_h = 1 \qquad \forall v_h \in T$$

$$y_h \in \{0, 1\}, \qquad \forall v_h \in V \setminus T.$$

A heurística Primal1 constrói uma solução para o problema acima por um método ganancioso, incluindo gradualmente uma variável y_h na solução ao minimizar uma certa função $f(c_h, b_h)$, onde b_h é o número de vértices $v_l \in W$, satisfazendo $c_{hl} \leq c$, mas ainda não cobertos. As funções f sugeridas em Balas e Ho [4] estão descritas na Seção 3.2.2. Diante disso, e do que foi descrito sobre a heurística GENIUS na Seção 3.3.5, podemos descrever os passos da heurística H-1-PRC.

Heurística H-1-PRC

Passo 1 (*inicialização*). Faça $H := T$, $\bar{z} := \infty$. A definição atual de f é (3).

Passo 2 (*construção da rota*). Construa um ciclo Hamiltoniano de custo z com todos os vértices de H, usando GENIUS.

Passo 3 (*critério de parada*). Se pelo menos um vértice de W não é coberto por um vértice de H, vá para o Passo 4. Se $z \leq \bar{z}$, faça $\bar{z} := z$ e $\bar{H} := H$. Se a definição atual de f é a última, pare com a melhor solução dada pela rota $\bar{H}$ de custo $\bar{z}$. Caso contrário, remova de H todos os vértices associados com multicobertura de vértices em W e vá para o Passo 2 com a próxima definição de f.

Passo 4 (*seleção de vértice*). Calcule para todo vértice $v_h \in V \setminus H$ um coeficiente c_h igual ao custo de sua inserção mais barata na rota atual H. Determine o melhor vértice v_{h^*} para incluir em H de acordo com a definição atual de f. Faça $H := H \cup \{v_{h^*}\}$ e vá para o Passo 2.

Na implementação da heurística H-1-PRC percebemos que algumas alterações poderiam ser feitas visando melhoria no método. Como contribuição explicamos abaixo tais alterações e descrevemos uma nova heurística para o 1-PRC.

Alterações na heurística H-1-PRC

Note que no Passo 3, em uma dada iteração da heurística, os vértices de H associados com multicobertura de vértices em W devem ser removidos e aplica-se o Passo 2, que é a construção de uma solução para o PCV. Isto é feito no mínimo três vezes de acordo com as definições de f.

Ora, primeiro lembramos que $T \subseteq H$, então devemos verificar apenas as multicoberturas de $H \setminus T$, pois não faz sentido retirar vértices de T, associados com multicobertura, pois são de visita obrigatória. Em segundo lugar, a heurística GENI demonstrou ser rápida e eficiente, porém o processo de pós-otimização US demonstrou ser muito mais lento comparando com a GENI. Então, ao invés de simplesmente remover os vértices da rota, fazê-lo utilizando os dois tipos de remoção descritos na heurística US. Assim, mantemos uma rota construída com menos vértices, e esperamos que ela tenha um custo reduzido, pois as remoções são feitas testando boas configurações. Então não é necessário retornar ao Passo 2, para refazer uma solução do PCV, e a heurística passaria diretamente para o Passo 4. Isso se estivermos preocupados com o tempo de processamento, sendo que outra opção poderia ser apenas usar a GENI em vez da GENIUS no Passo 2 em exemplares de dimensão grande.

Outro detalhe é uma mudança no procedimento do Passo 4 para inserção de vértices v_{h^*} de $V \setminus H$ na rota. A heurística escolhe gradualmente um vértice v_{h^*} para adicionar na rota,

retorna ao Passo 2 e resolve um novo PCV até que todos vértices de W sejam cobertos. Ao invés disto, ainda utilizando a heurística Primal1 e de acordo com a definição atual da f, podemos selecionar gradualmente v_{h^*} entre os vértices de $V \setminus H$ e incluí-los dentro de um mesmo passo, até a cobertura total de W. Agora incluímos os vértices escolhidos um a um usando os dois tipos de inserções definidos na heurística GENI. Note que assim, podemos dizer que o conjunto H sempre é uma rota, pois nunca vamos tratá-lo como um conjunto de vértices desconectados. Novamente não precisamos refazer o Passo 2, e retornamos ao Passo 3 com a nova definição de f. Desta forma, a construção de um ciclo para o PCV usando a heurística GENIUS é feita apenas uma vez, quando usamos o conjunto T no Passo 2. Obviamente, menos operações serão feitas e vamos aproveitar as características das inclusões e remoções de vértices na rota que foram apresentadas nas Seções 3.3.3 e 3.3.4.

Note também que se após a primeira passagem no Passo 2 todos os vértices de W já estiverem cobertos, não faz sentido continuar, pois não podemos retirar vértices do conjunto T, e os outros passos seriam redundantes. Portanto temos que incluir um critério de parada, que seria tomar como solução a rota construída pela heurística GENIUS no Passo 1, caso isso ocorra. Com essas mudanças os passos da heurística H-1-PRC podem ser reescrito. E vamos chamá-los de heurística H-1-PRC modificada ou H-1-PRC-MOD.

Heurística H-1-PRC-MOD

Passo 1 (*inicialização*). Faça $H := T$, $\bar{z} := \infty$. A definição atual de f é (3). Com o conjunto H, resolva um PCV, de custo z, usando a heurística GENIUS. Se todos os vértices de W já estão cobertos, pare com a melhor solução construída com os vértices de H. Caso contrário vá para o Passo 3.

Passo 2 (*critério de parada*). Se $z \leq \bar{z}$, faça $\bar{z} := z$ e $\bar{H} := H$. Se a definição atual de f é a última, pare com a melhor solução dada pela rota $\bar{H}$ de custo $\bar{z}$. Caso contrário, remova de $H \setminus T$ todos os vértices associados com multicobertura de vértices em W utilizando os dois tipos de remoção de vértices e as duas orientações da rota, como descrito na heurística US. Vá para o Passo 3 com a próxima definição de f.

Passo 3 (*seleção de vértice*). Dentre os vértices em $V \setminus H$ selecione gradualmente v_{h^*} utilizando a heurística Primal1 de acordo com a definição atual de f. Os custos para a escolha são dados pelos dois tipos de inserção de vértices e orientações da rota de acordo com a heurística GENI. Inclua cada v_{h^*} na rota H usando a GENI até que todos vértices de W sejam cobertos. Denote por z o custo da rota e vá para o Passo 2.

3.4.3 Testes Comparando H-1-PRC e H-1-PRC Modificada

Aqui escolhemos alguns exemplares, que foram construídos e estão descritos no Capítulo 5, para testar e comparar a eficiência das heurísticas H-1-PRC e H-1-PRC modificada. Os

exemplares foram construídos para o m-PRC, porém neste caso basta tomar $m = 1$. Três tipos do H-1-PRC foram testados: H1PRC-GENIUS (heurística H-1-PRC original); H1PRC-GENI (heurística H-1-PRC original usando somente a GENI); e H1PRC-MOD (heurística H-1-PRC modificada). Os exemplares foram testados em código Matlab em um micro com processador Intel Celeron de 1.73GHz, 512 MB de memória RAM e sistema operacional Windows.

$p = 4$	**H1PRC-GENIUS**			**H1PRC-GENI**			**H1PRC-MOD**		
Exemplar	R	SH	tempo	R	SH	tempo	R	SH	tempo
RPP-100-1	1	274.51	22.73	1	274.51	6.16	1	274.51	4.64
RPP-100-2	1	328.04	74.56	1	328.04	5.26	1	328.04	12.25
RPP-200-1	1.007	383.11	404.41	1.026	390.66	28.12	1	380.55	27.83
RPP-200-2	1	417.59	264.81	1.030	429.97	3.61	1.008	420.98	32.78
RPP-600A-1	1.015	617.84	23423	1.012	616.47	306.98	1	608.82	431.02
RPP-600A-2	***	***	***	1.037	660.00	118.91	1	636.61	525.22

Tabela 3.3: Resultados obtidos com as heurísticas H-1-PRC

Na Tabela 3.3, $p = 4$ é o parâmetro da p-vizinhança. Nas suas colunas, R indica a relação obtido ao dividir a solução de uma dada heurística pelo melhor valor obtido entre todas elas, referente a função objetivo. SH indica a solução obtida por uma dada heurística e o tempo é dado em segundos.

Observando a Tabela 3.3 concluímos que a heurística H-1-PRC modificada realmente, por efetuar menor número de operações, se destaca com relação ao tempo de processamento. Podemos notar que a estratégia usada, aproveitando as características das heurísticas GENI e US, com relação as inclusões e remoções de vértices na rota, separadamente, também trouxe boas respostas com relação ao valor da função objetivo, comparado com as demais. O tempo de processamento para H1PRC-GENI se destaca, porém perde com relação a função objetivo, o que era previsto. As dimensões dos exemplares que usamos aqui, estão descritas na Seção 5.2.1. Note que, conforme a dimensão do problema aumenta o tempo de solução se torna importante, como vemos nos exemplares `RPP-600A-1` e `RPP-600A-2`.

Com essas obervações podemos optar pelo uso da H-1-PRC modificada na solução dos exemplares m-PRC. No Capítulo 5 são indicados quais procedimentos foram usados e no Capítulo 4 é apresentada uma sugestão de uma nova heurística para o 1-PRC.

3.5 Rotas de Cobertura Multiveículo

Apresentamos nesta seção as heurísticas de construção das rotas e de pós-otimização para o m-PRC, suas características e como estão combinadas as heurísticas apresentadas anteriormente. Incluímos as três heurísticas que foram estudadas e estão descritas em [16], que são denominadas Heurística da Economia, Heurística da Varredura, Heurística Primeiro 1-PRC/ Segundo m-PRC e a de pós-otimização 2-opt*, uma adaptação da heurística original

2-opt construída para o PCV [7]. Uma mudança importante no Passo 1 da Heurística da Economia é descrito. Com a intenção de melhorar os resultados e para comparações, três novas heurísticas de construção de rotas e outra de pós-otimização são sugeridas no Capítulo 4. Todas as mudanças necessárias no que diz respeito ao nosso problema em estudo são comentadas.

3.5.1 Problema de Rotas de Cobertura Multiveículo

O problema considerado agora é uma generalização do 1-PRC. Gostaríamos de construir várias rotas sobre um subconjunto de V. E para isso vamos utilizar heurísticas descritas em [16]. A formulação e as características deste problema foram apresentadas no Capítulo 2.

3.5.2 Heurística 2-opt*

Esta heurística é descrita em [16] como uma adaptação da heurística original 2-opt de trocas de arestas destinada a solução de um PCV [7]. Vamos apresentar a heurística 2-opt* da forma que é encontrada em [16] para depois identificar quais adaptações devem ser feitas na intensão de resolver o nosso problema em estudo. A 2-opt* foi criada para resolver uma classe de problemas m-PRC que admite utilizar uma quantidade de veículos abaixo daqueles disponíveis, levando em conta restrições de capacidade. Respeitando esta classe de problemas apresentamos a seguir os passos desta heurística. Tendo uma solução viável para o m-PRC contendo μ rotas de veículo ($\mu \leq m$), esta heurística tenta melhorar a solução viável usando no máximo μ veículos.

Heurística 2-opt*

Passo 1 (*replicação do depósito*). Crie uma única rota, fazendo μ cópias da base e considere todas as combinações de pares de arestas nas rotas atuais. Veja as Figuras 3.10 e Figura 3.11.

Passo 2 (*combinando arestas*). Dada a combinação de arestas, $\{(v_r, v_s), (v_t, v_u)\}$, remova $(v_r, v_s), (v_t, v_u)$ da rota e considere sucessivamente as duas opções de novas arestas: (i) (v_r, v_t) e (v_s, v_u); (ii) (v_r, v_u) e (v_s, v_t). Isto cria um ou dois ciclos (veja a Figura 3.12). Cada ciclo só é viável se (i) ele contém no mínimo uma cópia da base e (ii) todo ciclo iniciando e terminando em uma base, satisfaz as restrições de capacidade do veículo.

- Se ambas opções derem soluções inviáveis ou não melhorarem a melhor solução conhecida, repita o Passo 2 com a próxima combinação de arestas, ou pare se todas já foram consideradas.
- Caso contrário, guarde a melhor opção viável e vá para o Passo 3.

Passo 3 (*reconstruindo as rotas*). Construa uma solução do m-PRC usando apenas uma base, onde cada ciclo ligando duas bases ou a mesma base define uma rota. Denote por $\tilde{\mu}$ o número de rotas de veículo nesta solução. Note que $\tilde{\mu}$ pode ser menor que μ se a solução no final do Passo 2 contém apenas uma aresta ligando duas cópias da base (veja a Figura 3.13). Faça $\mu := \tilde{\mu}$ e vá para o Passo 1.

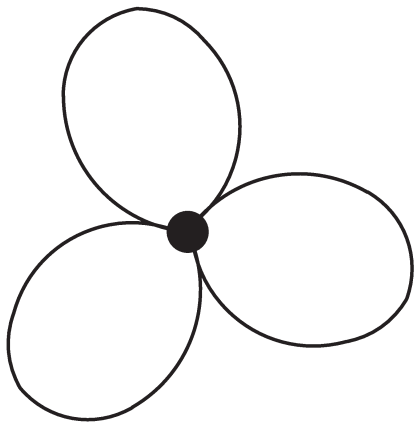


Figura 3.10: Solução inicial viável de um m-PRC

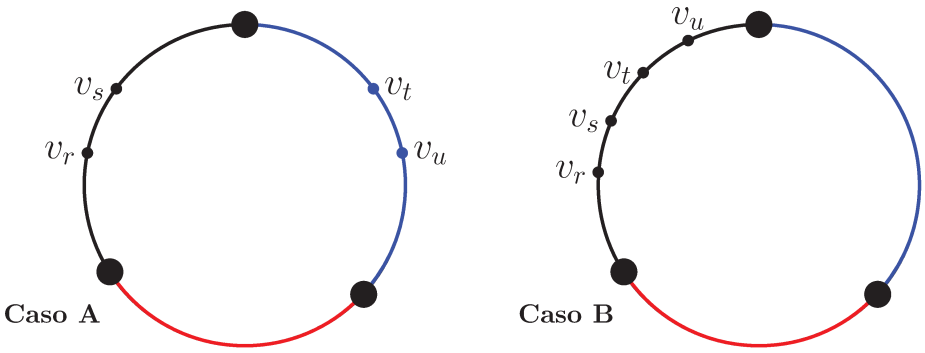


Figura 3.11: Caso A e Caso B: bases replicadas no Passo 1 da heurística 2-opt*

No Passo 1 a heurística faz μ cópias da base como no múltiplo PCV descrito em Lenstra e Rinnooy Kan [21] e considera todas as combinações de pares de arestas possíveis entre as rotas. A Figura 3.10 ilustra uma solução inicial com $\mu = 3$, e a Figura 3.11 mostra como obtemos uma única rota após a replicação da base. Os pontos grandes representam três bases distribuídas em uma única rota. Ela, também mostra dois tipos possíveis de combinações de pares de arestas. O Caso A que representa uma combinação entre rotas opostas e o Caso B que representa outra numa mesma rota.

No Passo 2 é feito uma recombinação entre as arestas analisadas, a Figura 3.12 ilustra o efeito de remover duas arestas e as quatro possibilidades de formação de novas arestas. Os casos são obtidos se as arestas removidas (v_r, v_s) e (v_t, v_u) estão na mesma rota entre-bases ou não, e de acordo com a opção de religação. Nos Casos $A1$ e $A2$, duas novas rotas de veículos são obtidas ao recombinar as antigas. O Caso $B1$ é inviável pois uma das rotas não contém réplica da base, que neste caso, é o mesmo que criar uma subrota desconectada em uma das rotas. Apenas uma das rotas é modificada no Caso $B2$, rearranjando-se a ordem em que os vértices são visitados.

O Passo 3 funciona como um inverso do Passo 1, ou seja, as cópias das bases são colapsadas a fim de construirmos uma nova solução. Se os novos ciclos criados no Passo 2 são viáveis e diminuem custos, eles são transformados em novas rotas. Ou seja, cada ciclo ligando duas bases ou a mesma base. Com isso, temos uma nova solução m-PRC desejada.

Agora vamos identificar o que precisamos alterar na heurística 2-opt* para atender ao nosso problema em estudo. Não podemos admitir que o número de rotas na solução inicial viável de um m-PRC diminua, ou seja, queremos que a solução final pós-otimizada continue com m rotas. Também, queremos que o número de vértices nas diferentes rotas continuem próximos depois de aplicarmos um processo de pós-otimização. Para isso, foram sugeridas algumas mudanças, como segue abaixo.

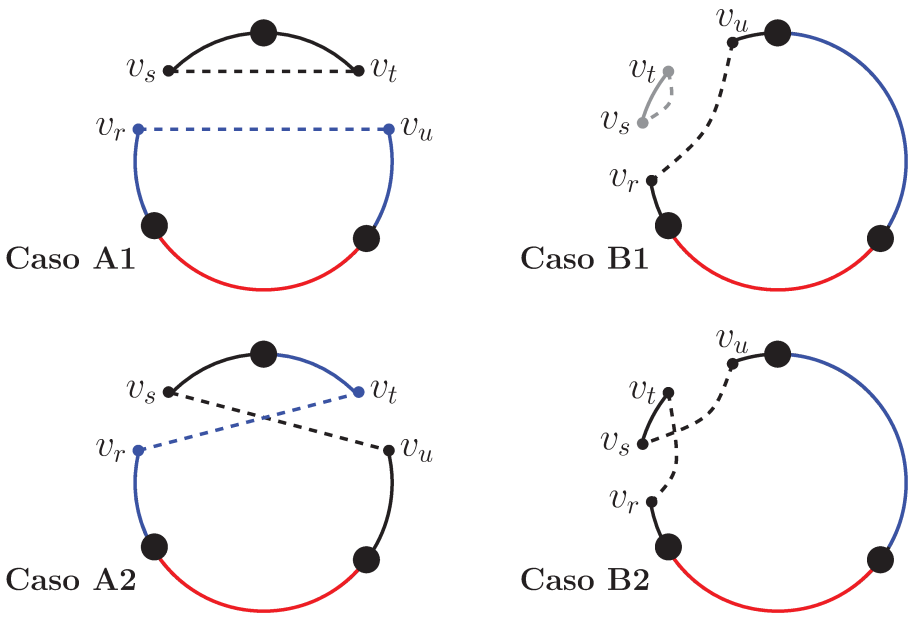


Figura 3.12: Novas rotas para os casos A e B da Figura 3.11

Alterações na heurística 2-opt*

Como não queremos que haja diminuição no número de veículos, diferentemente do que foi descrito no início desta seção, a aplicação desta heurística deve ser alterada como segue. Dada uma solução viável para o m-PRC contendo m rotas de veículo, esta heurística tenta melhorar a solução usando exatamente m veículos.

Durante alguns testes da implementação feita da 2-opt* notamos que por diversas vezes o número de vértices de algumas rotas diminuía consideravelmente e de outras aumentava muito, comparando com a solução anterior corrente. Provavelmente, pelo fato do nosso modelo em estudo não considerar restrições de capacidade, tamanho de rota e nem de número limite de vértices a serem visitados, algumas recombinações como ilustradas na Figura 3.14 acabam sendo permitidas. Nela, vemos que após o primeiro passo de replicação da base a solução inicial é composta de três rotas distintas com 5, 6 e 7 vértices. Seguindo os passos da heurística vemos na figura que uma recombinação de arestas foi permitida, porém, note que na solução final a diferença de vértices entre a menor e a maior rota são de 7 vértices. Esse exemplo representa um problema com três rotas pequenas e, neste caso, se o limite das diferenças entre o número de vértices nas diferentes rotas fosse de um vértice, esta solução final seria inviável.

Para que não ocorra uma diminuição considerável no número de vértices nas diferentes rotas foi feita uma modificação no Passo 2 da heurística 2-opt*. Observe novamente a Figura 3.12. No Caso A1 e A2, novas soluções só serão permitidas se o número de vértices N_k em cada nova rota k obtida, a partir de uma recombinação das rotas antigas, não seja

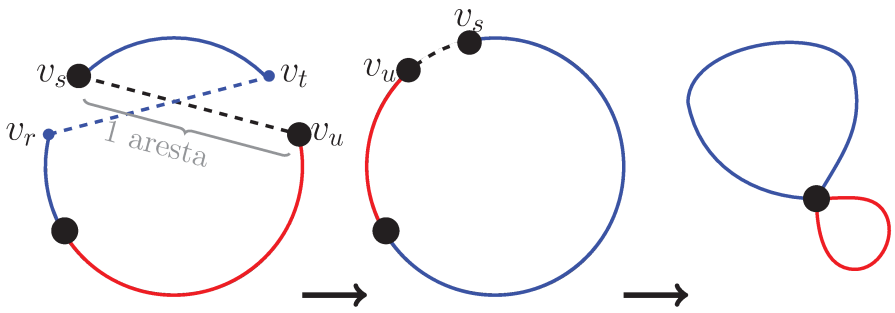


Figura 3.13: Diminuição no número de rotas

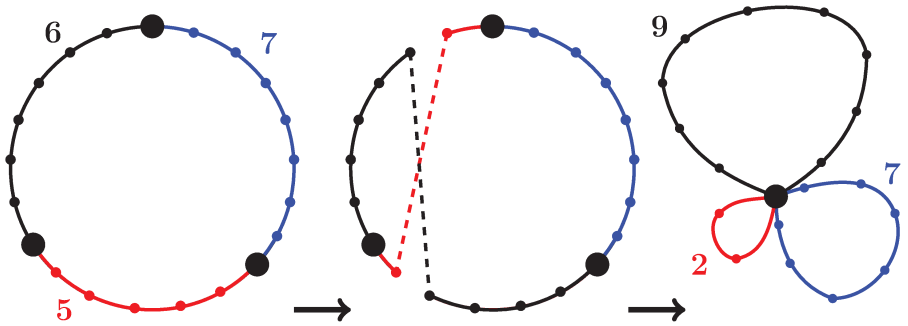


Figura 3.14: Diminuição significativa no número de vértices de uma rota

menor que o número de vértices da menor rota dada como solução inicial viável do m-PRC. Para ficar mais claro, dada uma solução viável para a aplicação da heurística 2-opt*. Com relação ao número de vértices, se a rota k com número de vértices igual a ρ é a menor, então este valor limita inferiormente o número de vértices de todas as possíveis rotas obtidas em uma nova solução. Portanto, neste momento incluímos um tipo de restrição, que não é normalmente usada, pois limita inferiormente o número de vértices que um veículo deve visitar.

Para manter o número fixo de m rotas tomamos $\mu = m$ no Passo 1 e no Passo 2 não permitimos recombinações que tenham apenas uma aresta entre duas cópias de bases. Note que limitar o tamanho dos ciclos por um valor ρ já garante que o número de rotas não muda, pois $\rho \geq 1$ em qualquer solução viável inicial, o que implica em no mínimo duas arestas. A Figura 3.13 ilustra essa possibilidade. Nela, aparece indicada uma aresta isolada em um ciclo com extremos v_s e v_u, após o Passo 2 da heurística. Consequentemente, caso essa recombinação fosse permitida, a nova solução após o Passo 3 teria $m - 1$ rotas. Levando em conta essas considerações, os passos da heurística 2-opt* devem ser reescritos da seguinte maneira:

Heurística 2-opt*-modificada

Passo 1 (*replicação da base*). Chame de ρ o número de vértices na menor rota da solução viável inicial. Crie uma única rota, fazendo m cópias da base e considere todas as combinações de pares de arestas nas rotas atuais. Veja as Figuras 3.10 e 3.11.

Passo 2 (*combinando arestas*). Dado o par de arestas, $\{(v_r, v_s), (v_t, v_u)\}$, remova (v_r, v_s), (v_t, v_u) da rota e considere sucessivamente as duas opções de novas arestas: (i) (v_r, v_t) e (v_s, v_u); (ii) (v_r, v_u) e (v_s, v_t). Isto cria um ou dois ciclos (veja a Figura 3.12). Cada ciclo só é viável se (i) ele contém no mínimo uma cópia da base e (ii) todo ciclo iniciando e terminando em uma base, com relação ao número de vértices, deve ter no mínimo ρ vértices.

- Se ambas opções derem soluções inviáveis ou não melhorar a melhor solução conhecida, repita o Passo 2 com o próximo par de arestas, ou pare se todos já foram considerados.
- Caso contrário, guarde a melhor opção viável e vá para o Passo 3.

Passo 3 (*reconstruindo as rotas*). Construa uma solução do m-PRC usando apenas uma base, onde cada ciclo ligando duas bases ou a mesma base define uma rota. Vá para o Passo 1.

3.5.3 Heurística da Economia

Esta heurística, criada por Hachicha, Hodgson, Laporte e Semet [16] determina em uma primeira etapa uma solução viável inicial para o m-PRC aplicando uma versão de construção de rotas semelhante a heurística criada por Clarke e Wright [6] para o Problema de Roteamento de Veículos (PRV). Em uma segunda etapa, através de um método ganancioso, ela remove da solução vértices de V, garantindo a cobertura do conjunto W. A seguir, ela considera em cada rota k construída anteriormente, o subconjunto W_k de W cobertos pelos vértices da rota k, e V_k, os vértices de V cobrindo W_k. Ela então aplica a heurística H-1-PRC para o subproblema induzido por $V_k \cup W_k$ para cada veículo k. Se no final desta etapa um vértice $v \in V$ pertence a mais de uma rota, ele é guardado somente em uma, naquela onde os custos são mais reduzidos. A segunda etapa é então reaplicada e finalmente a solução é pós-otimizada por meio da heurística 2-opt*.

Note que nessas etapas apresentadas não fica claro como o conjunto $T \subset V$ é avaliado e distribuído nas rotas. Este fato, e algumas alterações que achamos necessárias são explicadas oportunamente. Antes, vamos descrever os passos da heurística da economia da forma que são encontrados em [16], para que as alterações fiquem mais claras.

Passo 1 (*solução de um PRV em V*). Construa uma solução do PRV usando todos os vértices de V por meio do procedimento de construção de rotas descrito em Clarke e Wright [6]. Isto é, rotas de veículo são gradualmente construídas de acordo com a maior economia.

Passo 2 (*remoção de vértice*). Calcule para cada vértice de V incluído em uma rota de veículo a economia obtida por removê-lo da solução e por reconectar através de uma aresta os vértices que eram adjacentes a ele na rota. Classifique esses vértices em uma lista decrescente de economia. Considere cada vértice da lista e remova-o se isto não causa uma não cobertura em algum vértice de W.

Passo 3 (*H-1-PRC*). Considere todas k rotas de veículo. Determine W_k, o conjunto de vértices de W cobertos por algum vértice da rota k, e V_k, os vértices de V cobrindo W_k. Aplique a heurística H-1-PRC para o subproblema induzido por $V_k \cup W_k$. Se algum vértice v pertence a mais de uma rota, ele é guardado em apenas uma, naquela onde os custos são mais reduzidos. Chame de $\tilde{k}$ a nova rota obtida, desta maneira, da rota k. Se $\tilde{k}$ é viável e tem um custo menor do que da rota k, então substitua k por $\tilde{k}$. Caso contrário, a rota k não é alterada.

Passo 4 (*remoção de vértice*). Reaplique o Passo 2.

Passo 5 (*2-opt**). Pós-otimize por meio da 2-opt*.

A heurística da economia recebe este nome justamente pela forma que ela constrói uma solução inicial no Passo 1. Usando todos os vértices de V, ela utiliza o procedimento de construção de rotas descrito em [6] como um PRV. As rotas são gradualmente construídas analisando a economia obtida nas trocas de vértices. Devido as características deste procedimento de construção de rotas, que deve iniciar com uma solução viável de um PRV e efetua várias trocas de vértices que permitem mudanças no número de rotas, preferimos alterar esta etapa inicial usando outro procedimento, que se adapta melhor ao nosso problema em estudo. A idéia de economia se mantém, no entanto fazemos uma adaptação do procedimento de construção de rotas para o PRV descrita por Altinkemer e Gavish [1].

Para utilizar esta alteração no Passo 1 temos que decidir primeiro quantos vértices podem ser inclusos em uma determinada rota k inicial a ser construída no Passo 1. Como todos os vértices de V entram na solução inicial, o valor que procuramos, definido como N_k inicial, é obtido da seguinte maneira.

- Seja $N_k = \left\lfloor \frac{|V|-1}{m} \right\rfloor$, o número de vértices permitidos para a rota k no Passo 1 da heurística da economia, e $\beta = |V| - 1 - m.N_k$. Temos que $\beta < m$. Faça $k := 1$.

- Enquanto $\beta \neq 0$, faça:
  - $N_k := N_k + 1$.
  - $k := k + 1$ e $\beta := \beta - 1$.

Seguindo os passos acima, se $\beta = 0$, todas as rotas construídas no Passo 1 desta heurística terão igualmente N_k vértices, caso contrário, as rotas indexadas por $1, \ldots, k = \beta$ terão N_k vértices, e as demais terão um vértice a menos. Usamos na divisão $|V|-1$, pois são os vértices de V menos a base que serão escolhidos para entrar em cada rota. Esse procedimento faz com que a diferença entre o número de vértices de uma rota para outra seja de no máximo 1 na fase inicial. Todavia, esta diferença pode ser alterada durante os Passos 2 e 3 da heurística da economia. No entanto, escolhemos este critério inicial na tentativa de manter os valores N_k próximos.

Conhecido os valores N_k, precisamos entender o procedimento usado no Passo 1 para construir uma solução inicial. Primeiramente consideramos que cada vértice de V produz uma rota, como ilustrado no item (a) da Figura 3.15. Nela, representamos o conjunto V composto de 10 vértices, onde o vértice 1 é a base e os demais são escolhidos gradualmente para compor uma rota.

A distância economizada ao ligar um par de vértice (i, j), $i \neq j$ é dada como segue:

$$S_{ij} = d_{1,i} + d_{1,j} - d_{i,j}. \tag{3.1}$$

Na Figura 3.15 temos que $2 \leq i, j \leq 10$. A economia é calculada para cada par de vértices e o máximo dela é selecionado. Os dois vértices referente a essa maior economia são ligados para iniciar a primeira rota. Suponha que os vértices $i = 4$ e $j = 7$ iniciam a rota 1. Então verificamos agora o valor máximo da Equação 3.1 referentes a $S_{4,k}$ e $S_{k,7}$ para $k \in \{2, 3, 5, 6, 8, 9, 10\}$. Supondo que $N_1 = 3$ e $S_{8,7}$ é o novo máximo, o vértice 8 entra na rota 1, o que pode ser visto no item (b) da Figura 3.15, e assim, a rota 1 é encerrada. Retornamos a Equação 3.1 para selecionar o novo máximo, porém agora com $i, j \in \{2, 3, 5, 6, 9, 10\}$. Suponha que o novo valor seja $S_{3,10}$. Então os vértices 3 e 10 iniciam uma segunda rota. Se novamente $N_2 = 3$, falta entrar apenas mais um vértice na segunda rota. Avaliamos agora $S_{3,k}$ e $S_{k,10}$ para $k \in \{2, 5, 6, 9\}$. Conforme o item (c) da Figura 3.15 $S_{3,2}$ é o maior valor selecionado e o vértice 2 entra na rota 2. Na sequencia observamos no item (d) que os vértices 5 e 9 iniciam a terceira e última rota, sendo completada pelo vértice 6.

Note que incluímos vértices em cada rota k até atingir o seu valor N_k e ao mesmo tempo avaliamos a Equação 3.1 até todos os vértices de V serem utilizados. Seguindo o exemplo da Figura 3.15, podemos descrever um conjunto de etapas para ser usado no Passo 1 da heurística da economia. Vamos chamá-lo de Procedimento de Seleção por Economia, como segue.

Procedimento de Seleção por Economia

- Sejam τ_k o conjunto de vértices na rota k, $l = |V \setminus \{v_0\}|$ e $I = \{1, \ldots, l\}$ o conjunto de índices dos vértices $v_i \in V \setminus \{v_0\}$, tal que, $1 \leq i \leq l$.
- $k := 1$.
- Enquanto $k \leq m$.
  - $\tau_k := \{v_0\}$.
  - $(t, u) := \arg\max_{\substack{(i,j)\in I \\ i\neq j}} \{S_{ij}\}$, $\tau_k := \tau_k \cup \{v_t, v_u\}$ e $I := I \setminus \{t, u\}$. Elimine uma ligação de v_t e uma de v_u da base. Conecte v_t e v_u por uma aresta.
  - Enquanto $|\tau_k| \leq N_k$.
    - Sejam p e q os índices dos vértices que ainda têm alguma ligação com a base na rota τ_k. Para $s = \{p, q\}$ e $j \in I$.
    - $(t, u) := \arg\max_{(s,j)\in I} \{S_{sj}\}$, $\tau_k := \tau_k \cup \{v_u\}$ e $I := I \setminus \{u\}$. Elimine uma ligação de v_t e uma de v_u da base. Conecte v_t e v_u por uma aresta.
- $k := k + 1$.

No Procedimento de Seleção por Economia, temos que (t, u) representa o par de índices dos vértices selecionados ao se verificar o valor máximo de S_{ij}, tal que, $(i, j) \in I$ representam todos os pares de índices de vértices, onde $1 \leq i, j \leq l$ e $i \neq j$. Assim, t receberá algum índice i e u receberá algum índice j.

Agora vamos analisar o Passo 2 da heurística da economia. Nele, temos que usar um método ganancioso para eliminar vértices de V que não precisam pertencer a uma solução do m-PRC. Ou seja, vértices retirados de V, que não afetam a cobertura do conjunto W. Neste passo a heurística não refere-se ao conjunto $T \subset V$. Como este conjunto é de visita obrigatória, não faz sentido removê-los no Passo 2. Então neste passo vamos analisar a retirada de vértices em $V \setminus T$, e também, ao invés de criar uma lista de economia para escolher quem sai das rotas, achamos mais prudente fazer um pré-processamento que identifica entre os vértices de $V \setminus T$ quais podem sair da rota, e assim, retirá-los. Note que esse pré-processamento é um conjunto de passos que analisa quem pode sair sem afetar a cobertura de W. Vamos chamá-lo de Procedimento de Eliminação de Vértices sem Uso, que pode ser descrito como segue.

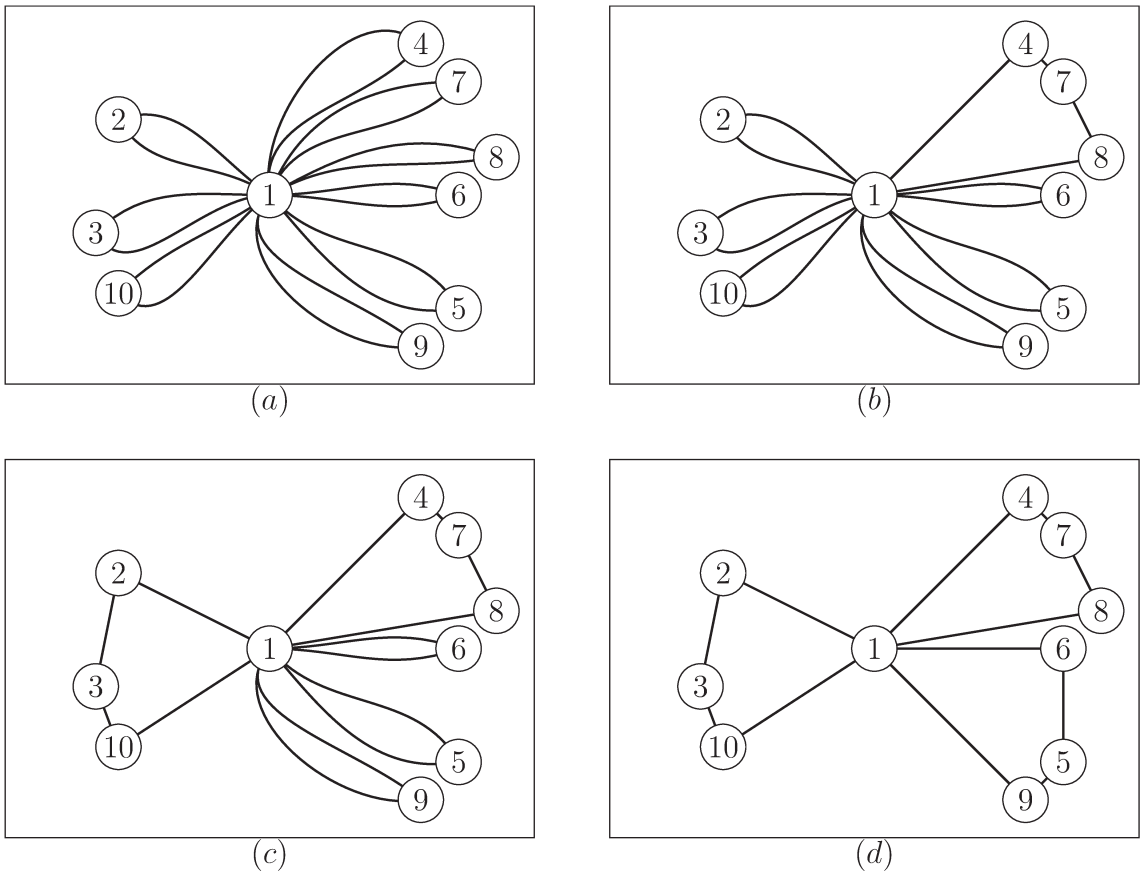


Figura 3.15: Exemplo do Procedimento de Seleção por Economia

Procedimento de Eliminação de Vértices sem Uso

Seja $l = |V \setminus T|$ e S um vetor ordenado tal que $S = \{v_1, v_2, \ldots, v_l\}$ para todo $v_h \in V \setminus T$, $h = 1, \ldots, l$. Defina R como o conjunto de vértices que podem ser retirados das rotas.

- $R := \emptyset$ e $h := 1$.
- Enquanto $h \leq l$, faça $S := S \setminus \{v_h\}$.
  - Se $S \cup T$ cobre todos os vértices de W, faça $R := R \cup \{v_h\}$ e $h := h + 1$.
  - Caso contrário, faça $S := S \cup \{v_h\}$ e $h := h + 1$.

Assim, os elementos do conjunto R podem ser retirados das rotas no Passo 2 da heurística da economia. Note que os vértices de $V \setminus R$ continuam em suas antigas rotas para serem usados no Passo 3.

O Passo 3 da heurística também não faz menção ao conjunto T ao escolher os subconjuntos V_k e W_k, no entanto acreditamos que fica subentendido que eventualmente ocorre um subconjunto $T_k \subset V_k$ para aplicar a heurística H-1-PRC ao subproblema induzido por $V_k \cup W_k$. De qualquer forma, ao menos a base estará em toda rota, assim é possível resolver o subproblema no Passo 3, tendo $T_k = \{v_0\}$.

Após essas explicações podemos reescrever os passos da heurística da economia, fazendo as alterações necessárias no Passo 1 e no Passo 2, como segue.

Heurística da economia

- **Passo 1** (*solução de um PRV em V*). Construa uma solução do PRV com todos os vértices de V usando o Procedimento de Seleção por Economia, descrito na Página 44.
- **Passo 2** (*remoção de vértice*). Verifique quais vértices de $V \setminus T$ incluído em uma rota de veículo podem ser removidos utilizando o Procedimento de Eliminação de Vértices sem Uso, descrito na Página 45. Remova tais vértices das rotas.
- **Passo 3** (*H-1-PRC*). Considere todas k rotas de veículo. Determine W_k, o conjunto de vértices de W cobertos por algum vértice da rota k, e V_k, os vértices de V cobrindo W_k. Aplique a heurística H-1-PRC para o subproblema induzido por $V_k \cup W_k$. Se algum vértice v pertence a mais de uma rota, ele é guardado em apenas uma, naquela onde os custos são mais reduzidos. Chame de $\tilde{k}$ a nova rota obtida, desta maneira, da rota k. Se $\tilde{k}$ é viável e tem um custo menor do que da rota k, então substitua k por $\tilde{k}$. Caso contrário, a rota k não é alterada.
- **Passo 4** (*remoção de vértice*). Reaplique o Passo 2.
- **Passo 5** (*2-opt**). Pós-otimize por meio da 2-opt*.

3.5.4 Heurística da Varredura

Criada por Hachicha, Hodgson, Laporte e Semet [16] para resolver o m-PRC, esta heurística aparece com o nome *modified sweep heuristic*, por ser uma adaptação do processo de construção de rotas para o PRV descrita em Gillet e Miller [15]. O nome varredura é escolhido por causa da característica em que os vértices são selecionados para pertencer a uma rota. Em [15], o processo de varredura descrito é dividido em duas etapas. Primeiro uma varredura no sentido anti-horário é feita, e as k rotas são selecionadas de acordo com as restrições do problema. Na segunda etapa, um processo inverso é feito para testar trocas de vértices entre rotas, considerando redução no custo através da minimização de uma função que depende do raio e ângulo de cada vértice numa rota.

Porém em [16] apenas a primeira etapa de varredura é feita, pois as tentativas de diminuição do custo através de trocas de vértices, são feitas de outras formas, considerando uma combinação de heurísticas.

Sendo assim, em uma primeira etapa a heurística da varredura constrói uma solução viável para o m-PRC usando a idéia do algoritmo de varredura descrito em [15]. O processo de varredura é aplicado nos vértices de $T \cup W$. Uma vez conhecida uma solução viável, ela é melhorada usando parte dos passos da heurística da economia. Várias soluções podem ser geradas.

A heurística da varredura foi criada para resolver uma classe de problemas m-PRC que atende a um certo conjunto de restrições, que em uma primeira etapa, devem ser avaliadas durante a construção de uma solução viável. Tais restrições referem-se a capacidade de veículos. Sabemos que no nosso problema em estudo, além de atender algumas restrições, temos que garantir um equilíbrio entre o número de vértices nas diferentes rotas. Por conta disso, tivemos que adaptar os passos da heurística da varredura de acordo com a nosso necessidade, principalmente na primeira etapa, onde temos que decidir quantos vértices irão compor cada rota viável inicial. Antes de explicarmos as adaptações feitas vamos descrever os passos da heurística da varredura da forma que é encontrada em [16], para que as alterações fiquem mais claras.

Passo 1 (*inicialização*). Determine o vértice v de $(T \cup W) \setminus \{v_0\}$ que tem a menor ordenada e considere uma linha tendo uma extremidade na base e passando por v. Relacione todos os vértices v_h de $T \cup W$ em ordem crescente do ângulo $\theta_h = \widehat{vv_0v_h}$.

Aplique os Passos 2 e 3 para cada ângulo θ_h determinado no Passo 1 ($h = 1, \ldots, \bar{h}$) e selecione a melhor de todas soluções.(O valor de $\bar{h}$ refere-se ao ângulo $\theta_{\bar{h}}$ do último vértice selecionado para a primeira rota a ser construída no Passo 2).

Passo 2 (*varredura*). Iniciando com a linha v_0v, rotacione um raio que tem um ponto fixo em v_0 no sentido anti-horário até um vértice v_h de $(T \cup W) \setminus \{v_0\}$ ser alcançado.

- Se $v_h \in T$, inclua-o na rota atual, se este não criar inviabilidade (isto é, não pode exceder o número de vértices P, nem o tamanho Q da rota atual, incluindo c_{0h}), e continue o processo de varredura. Se v_h não pode ser viável incluído na rota atual, complete a rota atual unindo seu último vértice a v_0, inicialize uma nova rota a partir de v_h, e continue o processo de varredura.
- Se $v_h \in W$, determine o vértice v_r de S_h que cobre o maior número de vértices de W e inclua-o na rota atual, se este é viável. Caso contrário complete a rota atual unindo seu último vértice a v_0, inicialize uma nova rota a partir de v_r, e continue o processo de varredura.

Passo 3 (*melhoria de solução*). Tente melhorar a solução do m-PRC atual aplicando os Passos 3-5 da heurística da economia.

Existem alguns detalhes que não estão claros nos passos apresentados acima. Vamos tentar entendê-los e ao mesmo tempo sugerir as adaptações que achamos necessárias. No Passo 1 da heurística da varredura, pede-se para determinar um vértice v de $(T \cup W) \setminus \{v_0\}$ de menor ordenada e uma linha imaginária tendo uma extremidade na base passando por v. Esta escolha não está bem clara em [16] e como é uma escolha totalmente arbitrária, vamos definir v como sendo o vértice mais próximo da base e chamá-lo de vértice inicializador do processo de varredura.

Ainda no Passo 1, pede-se para repetir a aplicação dos Passos 2 e 3 para cada ângulo θ_h, onde $h = 1, \ldots, \bar{h}$, e $\theta_{\bar{h}}$ o ângulo do último vértice selecionado para a primeira rota a ser construída no Passo 2. O processo de varredura no Passo 2 avalia quais vértices v_h de $(T \cup W) \setminus \{v_0\}$ podem entrar em cada uma das rotas. Assim, a primeira rota construída na primeira vez que o processo de varredura é feito no Passo 2 terá computado uma quantidade de ângulos $\theta_{\bar{h}}$ avaliados. Então os repetidos testes se darão ao refazer o processo de varredura e o Passo 3 para cada θ_h, com $h = 1, \ldots, \bar{h}$. Ou seja, serão feitos $\bar{h}$ testes e selecionado a melhor solução entre eles. Isso aumenta a chance de encontrarmos melhores soluções para

o m-PRC avaliado. Podemos prever melhores resultados comparando com a heurística da economia, pois nesta última é feito apenas um teste.

Vamos observar a Figura 3.16 para entender como é feito o processo de varredura. Nesta figura, não tomamos o vértice v como o mais próximo da base, para que, o processo de varredura ficasse melhor ilustrado. Nela, as circunferências representam os vértices do conjunto W e as estrelas do conjunto T. A partir do vértice inicializador v, podemos fazer uma analogia a abertura gradual de um leque no sentido anti-horário, que ao tocar um vértice em $T \cup W$, pára sua abertura e avalia se tal vértice pertence a T ou a W, incluindo vértices de T ou de $V \setminus T$, respectivamente, em uma rota. Também é ganancioso no processo de verificação da cobertura feita pelos vértices de $V \setminus T$. A abertura do leque continua após esta avaliação, porém deve-se decidir se muda ou continua na mesma rota. No nosso problema em estudo, isso será feito ao avaliar a quantidade de vértices permitidos para incluir nas diferentes rotas, ou seja, os valores N_k. Antes de definirmos esses valores, temos que conhecer o total de vértices que o processo de varredura seleciona. Note que, eventualmente este total é menor que $|T \cup W| - 1$, pois um vértice $v_r \in S_h$ cobrirá outros vértices $v_h \in W$ diferente daquele avaliado em W quando pará a abertura do leque. Esse fato é pertinente e ajuda a entender o item 2 do Passo 2. Nele, nada garante que ao inicializarmos uma nova rota a partir do vértice v_r, não estaremos deixando de analisar algum vértice de T que tem ângulo menor que θ_r.

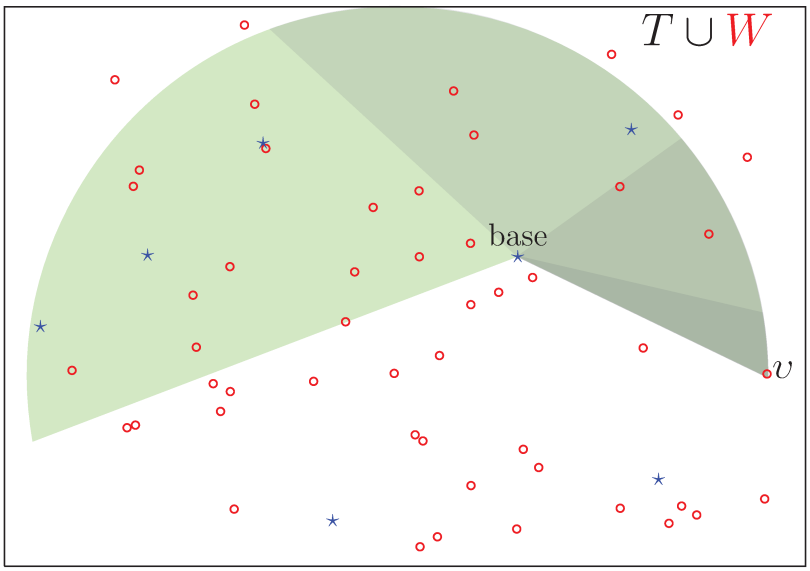


Figura 3.16: Exemplo do processo de varredura

Vamos descrever um conjunto de passos que executa o processo de varredura, porém com mais cuidado e de forma mais clara. Vamos chamá-lo de Procedimento de Abertura de um Leque, como segue.

Procedimento de Abertura de um Leque

- Escolha $v \in (T \cup W) \setminus \{v_0\}$ o mais próximo da base.
- Considere uma linha tendo uma extremidade na base e passando por v.
- Construa um vetor G ordenado com todos os vértices $v_h \in T \cup W$ em ordem crescente do ângulo $\theta_h = \widehat{vv_0v_h}$.
- Sejam $C_r = \{v_l \in W : c_{rl} \leq c\}$, $D := \emptyset$, $z := |G|$, então $G = \{v_1, v_2, \ldots, v_z\}$. Faça $h := 1$.
- Enquanto $h \leq z$
  - Se $v_h \in T$, $D = [D \quad v_h]$, $h := h + 1$.
  - Se $v_h \in W$, determine v_r de S_h que cobre o maior número de vértice de W, faça $D = [D \quad v_r]$, $T \cup W := (T \cup W) \setminus C_r$ e $h := h + 1$.
  - Se $v_h \notin W$, $h := h + 1$.
- D deve ser considerado como um vetor ordenado por construção.

Agora podemos avaliar os valores N_k. Isso será feito de acordo com a quantidade de vértices incluídos no vetor D a partir do Procedimento de Abertura de um Leque. Assim, vamos obter os N_k iniciais da seguinte maneira.

- Seja $N_k = \left\lfloor \frac{|D|}{m} \right\rfloor$, o número de vértices permitidos para a rota k no Passo 1 da heurística da varredura, e $\beta = |D| - m.N_k$. Temos que $\beta < m$. Faça $k := 1$.
- Enquanto $\beta \neq 0$, faça:
  - $N_k := N_k + 1$.
  - $k := k + 1$ e $\beta := \beta - 1$.

Fazemos os mesmos comentários feitos na Página 43, com relação aos diferentes valores N_k. Por fim, no Passo 3 da heurística da varredura são aplicados os mesmos procedimentos feitos nos Passos 3 a 5 da heurística da economia. Agora podemos descrever os passos da heurística da varredura com as alterações necessárias, como segue.

Heurística da varredura

Passo 1 (*inicialização*). Determine o vetor D através do Procedimento de Abertura de um Leque, descrito na Página 50. Considere D como um ciclo e obtenha os valores N_k.

Repita os Passos 2 e 3 iniciando com cada vértice v incluído na primeira rota construída no Passo 2 e selecione a melhor de todas as soluções.

Passo 2 (*seleção de vértices*). Iniciando de um vértice v arbitrário em D, construa uma rota k fazendo um passeio em D no sentido anti-horário e incluindo os vértices visitados neste passeio até atingir o valor N_k, complete tal rota unindo seu último vértice a v_0. Então considere o próximo vértice neste passeio como v, e repita este passo para a próxima rota até todos os vértices de D serem incluídos em alguma rota (veja a Figura 3.17).

Passo 3 (*melhoria de solução*). Tente melhorar a solução m-PRC atual aplicando os Passos 3-5 da heurística da economia.

A Figura 3.17 ilustra como a heurística da varredura obtém uma solução viável inicial. Nela, os vértices dos conjuntos W, T e $V \setminus T$ são representados por circunferências, estrelas e pontos respectivamente. Na parte superior da figura, temos um ciclo que representa o vetor ordenado D obtido do Procedimento de Abertura de um Leque. As cetas, representam como é feito o passeio sobre os vértices de D, selecionando os seus vértices para uma determinada rota em uma primeira iteração. Neste caso, é permitido incluir 4 vértices na primeira rota e temos $m = 3$. Na parte inferior da figura, já foi construída uma primeira solução do m-PRC. Obtemos 3 rotas com 4 vértices cada uma, excluindo a base. Esta solução fará parte dos quatro testes que serão feitos para este exemplo, pois como, 4 vértices puderam entrar na primeira rota, serão construídas quatro soluções viáveis para o m-PRC. Para cada uma delas, será aplicado o Passo 3, e a melhor selecionada.

3.5.5 Heurística Primeiro 1-PRC/Segundo m-PRC

Criada por Hachicha, Hodgson, Laporte e Semet [16], esta heurística recebe este nome porque se divide em duas etapas principais. Na primeira etapa, resolve o problema m-PRC induzido pelo conjunto $V \cup W$ tomando $m = 1$. Na segunda etapa, efetua divisões em rotas viáveis para o m-PRC construídas arbitrariamente.

Novamente esta heurística foi construída para uma classe de problemas m-PRC que atende restrições de capacidade durante a construção de rotas. Diferentemente, em nosso estudo temos que determinar quantos vértices podem entrar em cada uma das rotas antecipadamente, para depois construirmos tais rotas, analisando os valores N_k. Como a primeira etapa fornece uma rota de veículo, vamos determinar os valores N_k de acordo com o número de vértices incluídos nesta rota, como segue.

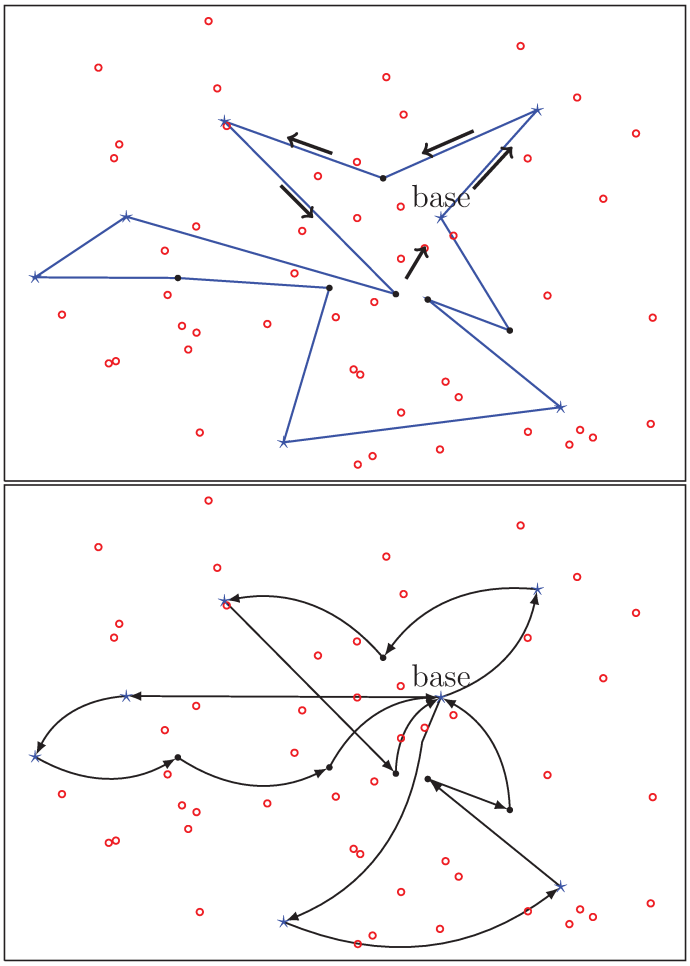


Figura 3.17: Uma solução inicial viável na heurística da varredura

- Seja τ a rota obtida na primeira etapa.
- Seja $N_k = \left\lfloor \frac{|\tau|-1}{m} \right\rfloor$, o número de vértices permitidos para a rota k no Passo 1 da heurística primeiro 1-PRC/segundo m-PRC, e $\beta = |\tau| - 1 - m.N_k$. Temos que $\beta < m$. Faça $k := 1$.
- Enquanto $\beta \neq 0$, faça:
  - $N_k := N_k + 1$.
  - $k := k + 1$ e $\beta := \beta - 1$.

Temos os mesmos comentários feitos na Página 43, com relação aos diferentes valores N_k. A seguir descrevemos os passos da heurística primeiro 1-PRC/segundo m-PRC com

as alterações necessárias, na tentativa de manter um equilíbrio no número de vértices nas diferentes rotas.

Heurística primeiro 1-PRC/segundo m-PRC

Passo 1 (*primeiro 1-PRC*). Construa uma solução para o problema m-PRC tomando $m = 1$ por meio da heurística H-1-PRC. Chame esta rota de τ e obtenha os valores N_k.

Repita os Passos 2 e 3 iniciando com cada vértice v incluído na primeira rota construída no Passo 2 e selecione a melhor de todas as soluções.

Passo 2 (*segundo m-PRC*). Iniciando de um vértice v arbitrário em τ, construa uma rota k fazendo um passeio em τ no sentido anti-horário e incluindo vértices visitados neste passeio até atingir seu valor N_k, complete tal rota unindo seu último vértice a v_0. Então considere o próximo vértice neste passeio como v, e repita este passo para a próxima rota até todos os vértices de τ serem incluídos em alguma rota (veja a Figura 3.18).

Passo 3 (*melhoria da solução*). Tente melhorar a solução m-PRC atual aplicando os Passos 3-5 da heurística da economia.

No parte superior da Figura 3.18 está ilustrado a solução de um m-PRC encontrada pela heurística H-1-PRC sobre o conjunto $V \cup W$ tomando $m = 1$. Nela, os vértices dos conjuntos W, T e $V \setminus T$ são representados por circunferências, estrelas e pontos respectivamente. As cetas, representam como é feito o passeio sobre os vértices de τ. Observe que, iniciamos o passeio sobre a solução τ a partir da base, e selecionamos os vértices que aparecem na sequencia até atingir um valor especificado N_k, em uma primeira iteração. Então inicia-se uma nova rota. Neste caso, é permitido incluir 4 vértices na primeira rota e temos $m = 3$. Na parte inferior da figura, já foi construída uma primeira solução do m-PRC. Obtemos 3 rotas, sendo a primeira com 4 vértices e as outras com 3 vértices, excluindo a base. Esta ilustração indica o processo feito para um teste de solução m-PRC, que fará parte dos quatro testes que serão feitos para este exemplo, pois como, 4 vértices puderam entrar na primeira rota, serão construídas quatro soluções viáveis para o m-PRC. Cada teste será encerrado no Passo 3, e o melhor selecionado.

Achamos que a idéia de iniciar a solução do problema m-PRC a partir da construção de uma rota sobre $V \cup W$ é muito interessante, pois esta primeira solução contem todos os vértices de T e as questões de cobertura do conjunto W já foram tratadas pela heurística H-1-PRC, ou seja, uma quantidade adequada de vértices em $V \setminus T$ estão nesta solução.

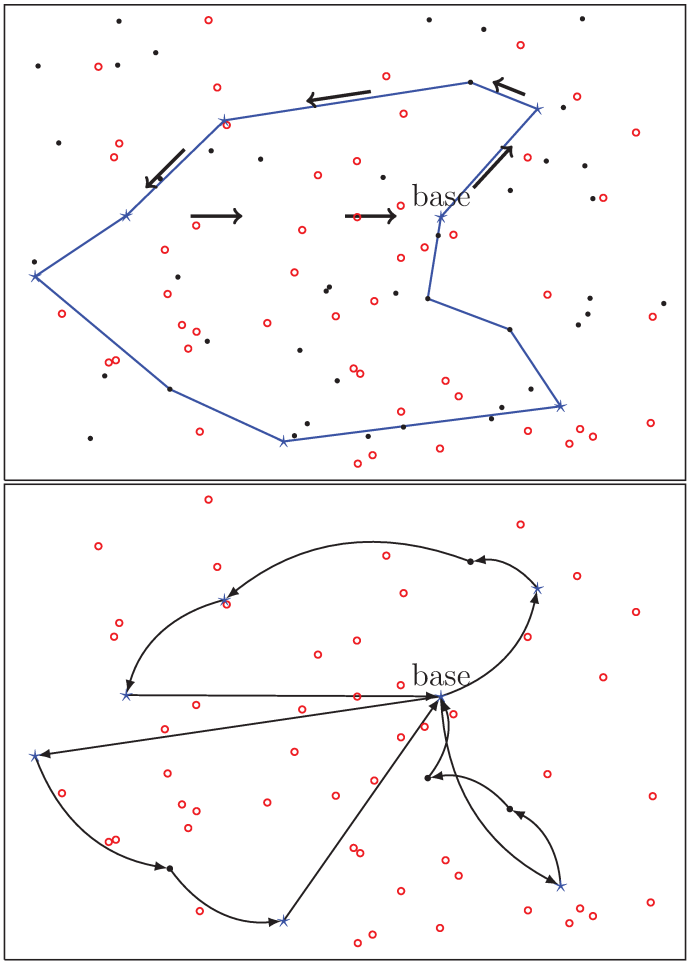


Figura 3.18: Uma solução inicial viável na heurística primeiro 1-PRC/segundo m-PRC

Capítulo 4

Novas Heurísticas

4.1 Introdução

Neste capítulo apresentamos de maneira geral as contribuições deste trabalho com relação à busca de boas soluções para o m-PRC. Ao longo do Capítulo 3 foram apresentadas todas as heurísticas estudadas nas referências. Porém, ao se analisar a qualidade dessas heurísticas e ao tentar adaptá-las ao nosso problema em estudo, foram surgindo idéias de mudanças, descritas no Capítulo 3, e de novas propostas de heurísticas. Neste capítulo apresentamos todos os procedimentos novos com relação à cobertura do conjunto W, ao processo de pós-otimização e quais estratégias usadas para manter o problema viável com relação à diferença entre o número de vértices nas diferentes rotas. As mudanças surgiram na tentativa de reduzir o número de operações durante a construção das rotas e de buscar melhores soluções comparadas com as heurísticas estudadas.

Como procedimento novo destacamos o chamado Teste de Cobertura. As heurísticas novas foram todas baseadas nas anteriores, porém tais mudanças não foram encontradas em nenhum artigo estudado até o momento. Entre elas está uma de pós-otimização para o m-PRC que chamamos de Heurística 2-opt Balanceada, uma para o problema 1-PRC denominada Heurística H-1-PRC Gananciosa e três heurísticas para o m-PRC, as quais foram chamadas de Heurística da Economia Adaptada, Heurística Gananciosa e Heurística Divide-Espaço. Essas heurísticas trabalham combinadas com outras, conforme ilustra o quadro apresentado na Figura 4.1. Este quadro é análogo ao quadro apresentado na Figura 3.1, na introdução do Capítulo 3. Então as explicações sobre as três fases principais das heurísticas de construção de rotas são as mesmas. No entanto, apesar das fases e as combinações serem semelhantes, as heurísticas mudaram. Observando o retângulo que representa a Fase 1, vemos que as heurísticas: da economia adaptada, gananciosa e divide-espaço são as alternativas de escolha para efetuarmos a resolução de um exemplar m-PRC. No retângulo que representa a Fase 2, substituímos a heurística H-1-PRC pela H-1-PRC modificada. Por fim, o quadro que representa a Fase 3, retiramos a heurística 2-opt* e acrescentamos duas alternativas de escolhas para a aplicação de pós-otimização. No caso da heurística divide-espaço usamos na pós-otimização o Teste de Cobertura, e nas heurísticas da economia adaptada e gananciosa

usamos a 2-opt balanceada.

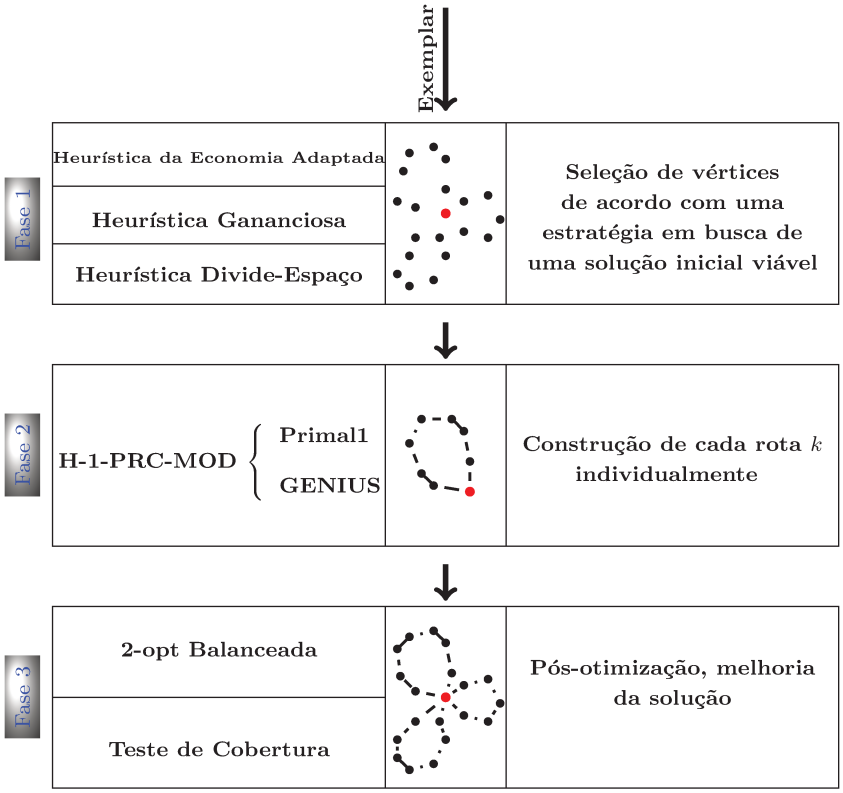


Figura 4.1: Processo de resolução de um exemplar do m-PRC em três fases principais

4.2 Teste de Cobertura

O Teste de Cobertura é um procedimento simples que surgiu durante a alteração no Passo 2 da heurística da economia, descrita na Seção 3.5.3. Neste passo são retirados vértices das rotas para testar melhorias no custo. Usamos o Teste de Cobertura para garantir a cobertura do conjunto W e para tratar as questões de multicoberturas. Para isso podemos seguir os seguintes passos:

4.2.1 Passos do Teste de Cobertura

- Crie uma lista com todos os vértices nas rotas em $V \setminus T$.
- Para cada v_k nesta lista remova-o e verifique se algum vértice de W não está coberto.
- Se algum vértice de W não está coberto, v_k não pode sair de sua rota. Caso contrário, v_k pode ser incluído em uma lista de remoção ordenado de forma decrescente por economia.

Testar se algum vértice de W não está coberto tornou-se importante durante as implementações, pois é possível conferir se uma solução atual realmente está cobrindo todos os vértices de W. Para isto, basta atualizar em cada passo o conjunto que vamos chamar de $S_{l,restante}$, onde $S_{l,restante} = \{v_h \in V_{k,corrente} : c_{hl} \leq c\}$, para todo $v_l \in W$, sendo que $V_{k,corrente} \subset V$ é o conjunto de vértices pertencente a rota k corrente.

Portanto ao se retirar v_k da lista dos vértices de $V \setminus T$ nas rotas, se algum $S_{l,restante} = \emptyset$ então v_k não pode sair da sua rota atual. Vale lembrar também que, na formulação do m-PRC descrita no Capítulo 2, os conjuntos S_l são definidos com a exigência de que $|S_l| \geq 2$, ou seja, pelo menos dois vértices de V cobrem cada vértice de W. Essa exigência surge em Hachicha, Hodgson, Laporte e Semet [16] ao se definir a constante c, que é a distância mínima para que um vértice de W esteja coberto por um vértice de uma rota. No entanto, se durante o processamento de um exemplar verificarmos que $S_{l,restante} = 1$, então o único vértice em $S_{l,restante}$ deve ser colocado em T e v_l não precisa mais ser analisado. O Teste de Cobertura tem três considerações principais, como segue:

1. Como já foi dito, podemos testar as soluções quanto à cobertura do conjunto W.
2. O Passo 3 da heurística da economia trata de uma questão importante para melhoria da solução, pois não permite que um vértice v pertença a mais de uma rota (veja a Figura 4.2). Mas pode ocorrer que um vértice $v_l \in W$ esteja sendo coberto por dois vértices de V, estando esses em rotas distintas. Note que ao aplicar a heurística H-1-PRC nos subproblemas induzidos por $V_k \cup W_k$, para cada rota k, essa possibilidade pode ocorrer (veja a Figura 4.3).

   Não vemos clareza em [16], se quando o Passo 4 da heurística da economia é aplicado, pretende-se eliminar essa multicobertura de rotas separadas. O fato é que ao aplicar o Teste de Cobertura, da maneira que foi explicado acima, utilizamos os vértices de todas as rotas juntas, ao invés de separadas, e consideramos apenas a lista dos vértices v_k que podem sair de alguma rota. Assim, podemos antes de iniciar as remoções, identificar aqueles vértices que estão em rotas distintas e remover aquele que mais reduz custos.

3. Por último, podemos usar o conjunto $S_{l,restante}$ para verificar a qualidade da cobertura, pois se no final do procedimento todos $S_{l,restante}$ tiverem cardinalidade igual a 1, podemos sugerir que a parte do m-PRC que trata da cobertura é animadora. Assim podemos fazer uma análise da cardinalidade do conjunto $S_{l,restante}$.

A Figura 4.2 ilustra a possibilidade de um vértice pertencer a mais de uma rota. Neste exemplo v_r pertence às Rotas 1 e 2, e v_s pertence às Rotas 2 e 3. O lado direito da figura mostra a atuação do Passo 3 da heurística da economia: v_r está agora apenas na Rota 1, e v_s na Rota 2.

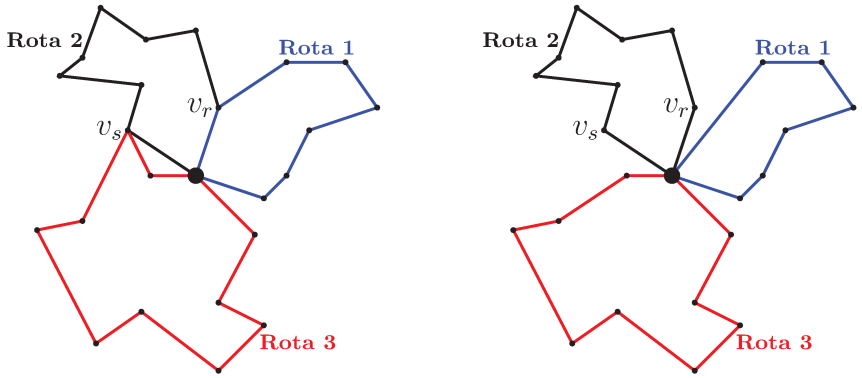


Figura 4.2: Cada vértice deve pertencer a uma única rota

A Figura 4.3 ilustra quando um vértice de W é coberto por dois vértices em rotas distintas, neste exemplo v_w está sendo coberto por v_r que pertence a Rota 2 e ao mesmo tempo por v_s que pertence a Rota 1. Por outro lado, v_z está sendo coberto por v_t da Rota 1 e por v_u da Rota 3. Do lado direito desta figura vemos um exemplo de como essa multicobertura pode ser resolvida. Note que esse tipo de multicobertura pode ser detectada usando o Teste de Cobertura.

A Figura 4.4 ilustra o que foi dito na terceira consideração do teste de cobertura, onde os vértices v_z, v_t e v_u, representados por pequenos círculos, pertencem a W e são cobertos por uma rota. Note que se pretendemos retirar da rota v_r, para que $|S_{z,restante}| = 1$ e $|S_{t,restante}| = 1$, o problema se tornaria inviável, pois v_u ficaria sem cobertura. Temos portanto que obrigatoriamente $|S_{t,restante}| \geq 1$. Mas quanto menor for a cardinalidade desses conjuntos mais animadora é nossa solução. Também não é difícil notar que isso faz referência aquilo que a heurística Primal1 chama de *boa solução* para o problema de cobertura, pois estamos da mesma forma querendo encontrar um suporte mínimo (veja a Seção 3.2.2, que trata da heurística Primal1) para o problema.

Observando as considerações anteriores, com relação a cobertura do conjunto W, sugerimos maneiras diferentes de obter rotas por meio da heurística H-1-PRC e novos testes na pós-otimização.

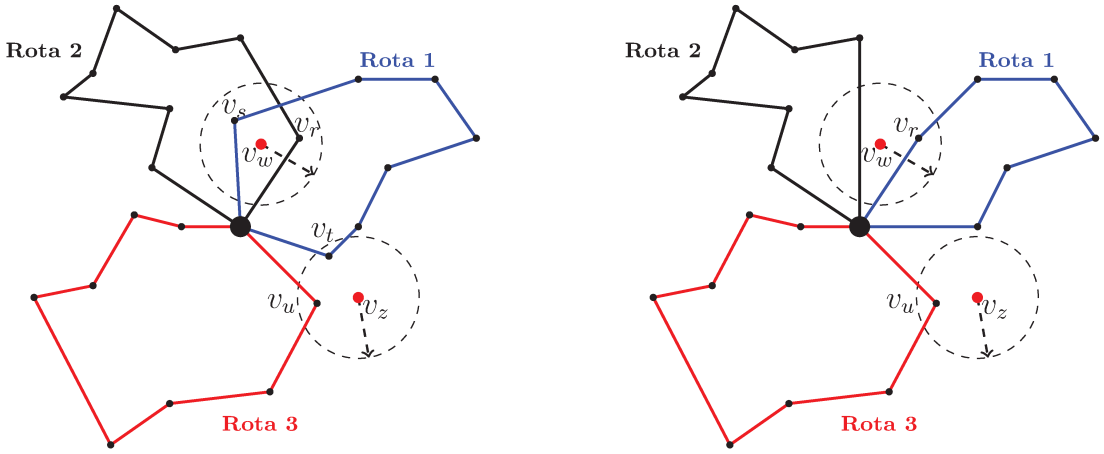


Figura 4.3: Um vértice de W sendo coberto por duas rotas

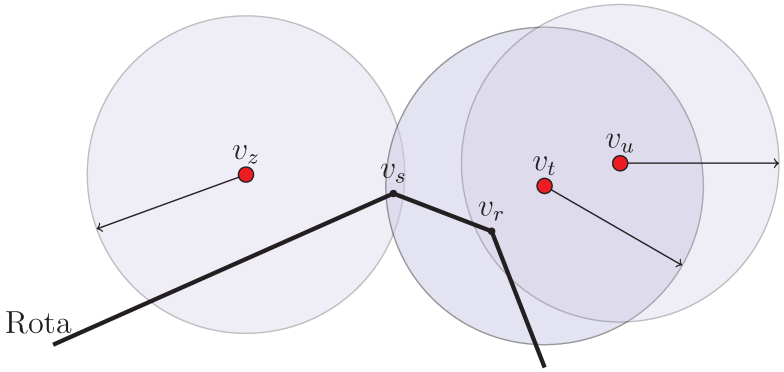


Figura 4.4: $|S_{t,restante}| \geq 1$ para algum $v_t \in W$

4.3 Heurística H-1-PRC Gananciosa

Como foi comentado anteriormente, com a idéia de fazer testes de cobertura durante o processo de solução do m-PRC, resolvemos propor mudanças no processo de análise da cobertura do conjunto W. Nesta seção apresentamos uma nova heurística, baseado na heurística H-1-PRC. Ele será chamado de H-1-PRC Gananciosa e podemos destacar duas coisas: o problema de cobertura é tratado novamente por um método ganancioso, porém não faremos uso da heurística Primal1 e a heurística GENIUS, que é um método eficiente, é usado sem modificações.

A principal mudança feita refere-se à forma como escolhemos os vértices de $V \setminus T$ para serem incluídos em uma rota. Esta escolha é feita em um único passo, ou seja, não incluímos vértices gradualmente como na heurística H-1-PRC, que por sua vez, usa os critérios da heurística Primal1.

A heurística H-1-PRC gananciosa, em uma primeira etapa, verifica se uma rota construída apenas com os vértices de T é uma solução viável para o 1-PRC. Caso contrário, em uma segunda etapa, decide quem são os novos vértices de $V \setminus T$ que poderão ser incluídos na rota. Este heurística recebe o nome de H-1-PRC gananciosa, justamente por causa da forma que decide quais vértices irão entrar na rota. Finalmente, em uma terceira etapa, ela constrói duas rotas de maneiras distintas e escolhe a melhor delas como solução final. A seguir descrevemos suas etapas.

4.3.1 Passos H-1-PRC Gananciosa

Passo 1 (*inicialização*). Encontre o conjunto de vértices $W_{coberto}$ que já estão cobertos pelos vértices de T. Faça $W := W \setminus W_{coberto}$. Construa um ciclo hamiltoniano τ sobre todos os vértices de T, usando a heurística GENIUS. Se $W = \emptyset$, pare, a rota τ de custo z é a solução. Caso contrário faça $R = \emptyset$ e vá para o Passo 2.

Passo 2 (*escolha de vértices*). Construa um vetor contendo todos os vértices $v_h \in V \setminus T$ em ordem decrescente da cardinalidade dos conjuntos $C_h = \{v_l \in W : c_{hl} \leq c\}$ para cada $v_h \in V \setminus T$. Para aqueles vértices com os conjuntos C_h com cardinalidades iguais, rearranje-os, ainda dentro do mesmo vetor, em ordem crescente de custo da sua inserção na rota τ utilizando as duas direções e os dois tipos de inserção da heurística GENI.

- Selecione o primeiro vértice v_h do vetor construído anteriormente, faça $T := T \cup \{v_h\}$, $R := R \cup \{v_h\}$ e $W := W \setminus C_h$. Se $W = \emptyset$ vá para o Passo 3. Caso contrário vá para o Passo 2.

Passo 3 (*construção da rota*). Construa dois ciclos Hamiltonianos da seguinte maneira:

- (*primeira rota*) τ_1, ao incluir na rota τ, todos os vértices do conjunto R, na ordem que ele foi construído no Passo 2, considerando as duas orientações da rota e os dois tipos de inserção de vértices da heurística GENI.
- (*segunda rota*) τ_2, com todos os vértices de T, usando a GENIUS.

A solução final será a melhor entre as rotas τ_1 e τ_2.

O Passo 2 da heurística H-1-PRC gananciosa procura diminuir a cardinalidade dos conjuntos $S_l = \{v_h \in V : c_{hl} \leq c\}$, para todo $v_l \in W$. Isto garante a cobertura de W e tende a reduzir o custo da rota.

Obtemos uma maior simplicidade na construção de uma rota com relação a heurística H-1-PRC original, pois a heurística H-1-PRC gananciosa, antes de iniciar a construção da

rota, escolhe em um único passo os vértices que vão pertencer a ela. Na heurística H-1-PRC original as rotas são construídas várias vezes durante os testes, enquanto que, no Passo 3 da heurística H-1-PRC gananciosa são construídas apenas duas.

Note que durante a construção da primeira rota no Passo 3, após o primeiro vértice v_h de R ser incluído na rota τ, a configuração muda e os custos de inserção dos vértices restantes já não são os mesmos que foram calculados anteriormente no Passo 2. No entanto, pretendemos evitar escolhas inadequadas de vértices como ilustra a Figura 4.5. Nela, os pequenos círculos são vértices de W cobertos pelos vértices de $V \setminus T$ representados por quadrados. Note que, para este exemplo, o rearranjo feito na lista de vértices v_h no Passo 2, garante que v_u será escolhido para entrar na rota ao invés de v_s, pois podemos perceber na ilustração que a inserção de v_u é mais barata.

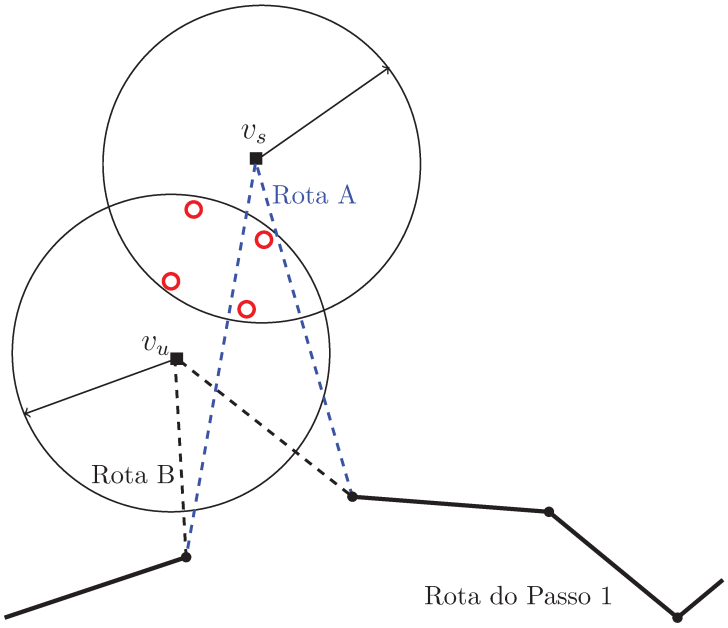


Figura 4.5: Duas possibilidades de escolha de um vértice para entrar na rota

Efetuamos a construção de dois ciclos hamiltonianos para comparação, pois durante a inserção de cada vértice em uma rota a configuração muda totalmente, e pretendemos verificar pelo menos duas formas de construção. Em uma já temos inicialmente uma rota construída com o conjunto T dos vértices que devem ser visitados, e na outra, a construção da rota é feita após todos os vértices que farão parte dela tiverem sido escolhidos. Esperamos obter melhores respostas com relação a heurística H-1-PRC original, principalmente com relação ao tempo de processamento.

4.3.2 Testes Comparando H-1-PRC, H-1-PRC Modificada e H-1-PRC Gananciosa

Aqui escolhemos os mesmos exemplares testados na Seção 3.4.4, que foram construídos e estão descritos no Capítulo 5, e comparamos a eficiência das heurísticas H-1-PRC original (H1PRC-GENIUS), H-1-PRC Modificada (H1PRC-MOD) e H-1-PRC Gananciosa (H1PRC-GAN).

$p = 4$	**H1PRC-GENIUS**			**H1PRC-MOD**			**H1PRC-GAN**		
Exemplar	R	SH	tempo	R	SH	tempo	R	SH	tempo
RPP-100-1	1	274.51	22.73	1	274.51	4.64	1	274.51	7.95
RPP-100-2	1	328.04	74.56	1	328.04	12.25	1	328.04	15.16
RPP-200-1	1.007	383.11	404.41	1	380.55	27.83	1.029	391.84	53.41
RPP-200-2	1	417.59	264.81	1.008	420.98	32.78	1	417.59	110.97
RPP-600A-1	1.014	617.84	23423	1	608.82	431.02	1.085	660.50	398.87
RPP-600A-2	***	***	***	1	636.61	525.22	1.014	645.84	2112

Tabela 4.1: Resultados obtidos com as heurísticas H-1-PRC

A Tabela 4.1 mostra os resultados obtidos com as heurísticas H-1-PRC. Nela, $p = 4$ é o parâmetro da p-vizinhança. Nas suas colunas, R indica a relação obtido ao dividir a solução de uma dada heurística pelo melhor valor obtido entre todas elas, referente a função objetivo. SH indica a solução obtida por uma dada heurística e o tempo é dado em segundos.

Observando a Tabela 4.1 concluímos que a heurística H-1-PRC gananciosa também se destaca com relação ao tempo de processamento, comparando com a H-1-PRC original. Repare na terceira coluna de cada heurística que, com relação a H-1-PRC modificada os tempos de processamentos são bem próximos, só há uma diferença considerável no exemplar RPP-600A-2. Observando os valores da função objetivo, que estão na segunda coluna de cada heurística, podemos notar que a estratégia usada pela heurística H-1-PRC gananciosa traz resultados que não diferem muito das outras duas. Assim obtemos um procedimento mais simples que também retorna resultados próximos dos daqueles já obtidos. No entanto, se observarmos a coluna referente as relações, a heurística H-1PRC modificada se destaca com relação as demais.

4.4 Heurística 2-opt Balanceada

Nesta seção vamos apresentar mais um procedimento, que pode ser usado como um passo a mais na heurística 2-opt*-modificada, apresentada na Seção 3.5.2. Esta última é uma adaptação da heurística 2-opt* descrita por Hachicha, Hodgson, Laporte e Semet [16] e usada para pós-otimizar uma solução viável do m-PRC. Alinhada com nosso problema em estudo, esta adaptação foi feita para manter o número de vértices nas diferentes rotas próximos.

O número de recombinações possíveis de rotas, ao se efetuar trocas de arestas, na 2-opt*-modificada diminui com relação a 2-opt*. Também, durante alguns testes, as soluções

finais para o m-PRC ocorriam com rotas sobrepostas, como está ilustrado na Figura 4.6. Nela, observamos que a Rota 1 e a Rota 2 estão sobrepostas com duas arestas se cruzando. Pensado em uma forma de eliminar rotas sobrepostas, que normalmente aumentam o custo final do problema, vamos incluir um elemento a mais na pós-otimização para buscar um conjunto de rotas com custo menor e mais adequado ao nosso problema em estudo.

Conforme ilustrado na Figura 4.7, o objetivo é testar trocas de vértices entre as rotas e verificar se nessas trocas o custo total reduz. Ao invés da Rota 1 ter no seu caminho o vértice v_s ela passa a ter o vértice v_r, essa troca é feita com a Rota 2 que passa a ter em seu caminho o vértice v_s.

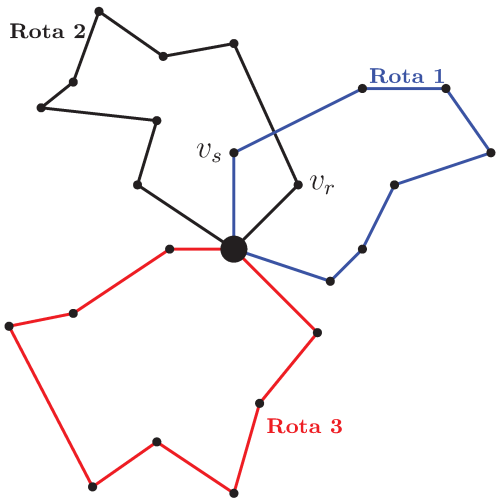


Figura 4.6: Ocorrência de solução com rotas sobrepostas

Note que essas trocas de vértices são adequadas para o nosso problema em estudo. Como este procedimento troca vértices um pelo outro, não afetamos as cardinalidades das rotas envolvidas.

Aqui, a nossa preocupação principal é manter um equilíbrio entre o número de vértices nas rotas envolvidas. Assim, decidimos realizar possíveis trocas de vértices em uma solução m-PRC após a aplicação da heurística 2-opt*-modificada. E denominamos este novo procedimento de heurística 2-opt Balanceada. Então, dada uma solução m-PRC, podemos aplicar os passos descritos a seguir.

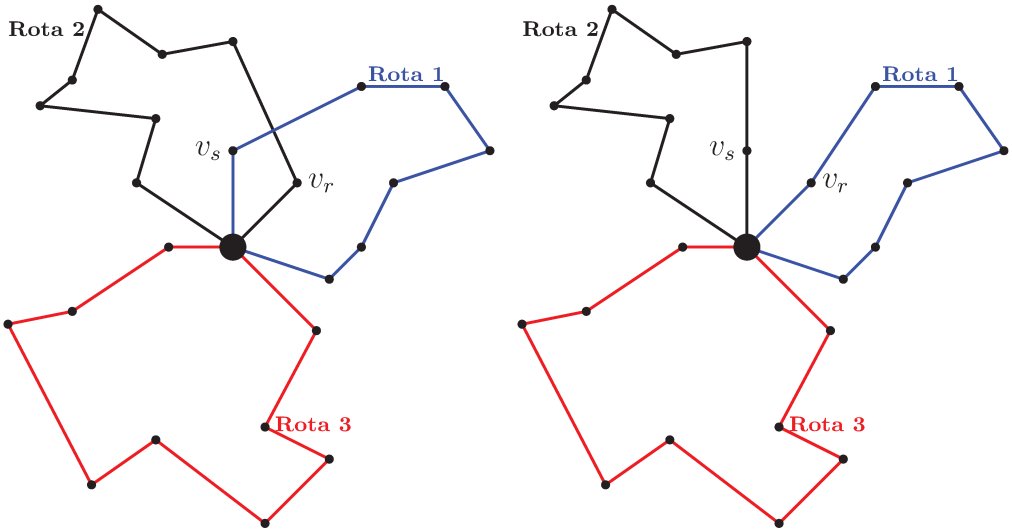


Figura 4.7: Eliminação de rotas sobrepostas

4.4.1 Passos 2-opt Balanceada

Passo 1 Aplique a heurística 2-opt*-modificada.

Passo 2 (*troca de vértices*). Considere todas as combinações de pares de vértices $\{v_r, v_s\}$ das rotas.

- **Passo 2.1**. Dado um par de vértices, considere a possível troca desses vértices de posição. Removendo cada vértice da sua posição original e incluindo na posição do outro (veja a Figura 4.7).
- **Passo 2.2**. Se não houver melhora da melhor solução conhecida, repita o Passo 2.1 com o próximo par de vértices ou pare se todos já foram considerados. Caso contrário, efetue a troca de vértices, guarde o melhor custo obtido e retorne ao Passo 2.

Na Figura 4.8 observamos melhor como podem ser feitos os testes de trocas de vértices quando os pares de vértices estão em rotas distintas. Analogamente ao que foi apresentado sobre a heurística 2-opt*, podemos replicar a base de acordo com o número de rotas e testar a troca entre todos os pares de vértices em rotas opostas. Nesta figura trocamos o vértice v_s pelo vértice v_r. Note que na heurística 2-opt balanceada também podem ocorrer recombinações de vértices em uma mesma rota e o Passo 2 pode ser aplicado em uma solução m-PRC independente da 2-opt*-modificada.

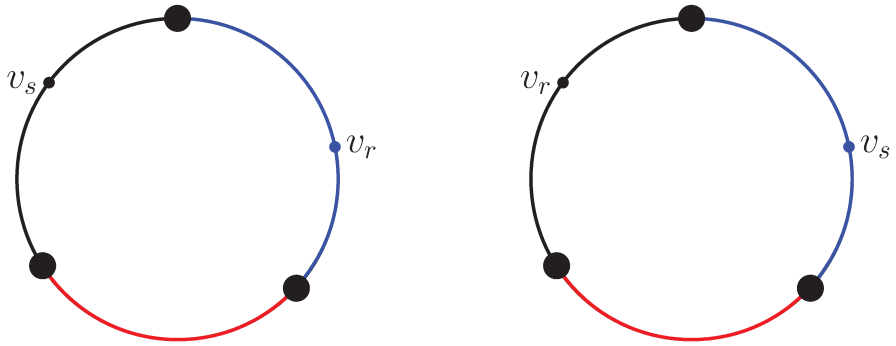


Figura 4.8: Troca de vértices entre rotas opostas

4.5 Heurística da Economia Adaptada

O Passo 1 da heurística da economia, descrita na Seção 3.5.3, usa o procedimento encontrado em Clarke e Wright [6] para encontrar uma solução factível inicial para o m-PRC. Este procedimento, que maximiza a economia ao efetuar inclusões e trocas de vértices entre rotas, constrói gradativamente as rotas de veículo de maneira aparentemente fácil. Porém, durante as implementações verificamos que este procedimento, além de difícil implementação, não consegue controlar de forma eficiente os valores N_k e, principalmente conservar o valor fixo m de veículos. A alternativa de mudar o procedimento, levando em conta o que é descrito por Altinkemer e Gavish [1], facilitou a implementação, porém verificamos muitas sobreposições entre as rotas em alguns testes feitos.

Construir rotas sobrepostas em uma primeira etapa pode aumentar muito o custo total. Embora sejam tratadas pelos Passos 3 a 5 da heurística da economia, em alguns casos, elas não são totalmente eliminadas. O ideal é que tais soluções iniciais fiquem com poucas sobreposições para que a solução final seja melhor. Isso ocorre com mais facilidade, pelas próprias características de escolhas de rotas iniciais, nas heurísticas da varredura e primeiro 1-PRC/segundo m-PRC.

Propomos nesta seção uma forma nova de construir a solução viável inicial para a heurística da economia, que chamaremos de Heurística da Economia Adaptada, pois a diferença entre elas ocorre apenas no Passo 1, e também usa um método ganancioso maximizando a economia ao se incluir um vértice na rota. Incluímos o uso da heurística GENI nesta construção, pois achamos que a qualidade das suas diferentes formas de inserção de vértices escolhidos dentro de uma vizinhança devem ser aproveitadas.

Vamos descrever abaixo os passos desta nova heurística. No lugar do Passo 1 da heurística da economia para obtenção de rotas iniciais viáveis é incluído um procedimento novo dividido em três passos. Definimos a q-vizinhança $M_q(v_h)$ como sendo os q vértices fora da rota mais próximos de v_h pertencente a rota. O critério de proximidade usado é a distância euclidiana. Este procedimento será repetido por duas vezes (iniciando com um vértice mais próximo e com um vértice mais longe da base) na nova heurística. E usamos os outros passos da

heurística da economia sem alteração. Os valores N_k iniciais para esta nova heurística é obtido usando o mesmo procedimento que na heurística da economia, que foi apresentado na Seção 3.5.3.

4.5.1 Passos da Heurística da Economia Adaptada

Passo 1 Escolha o valor N_k para cada rota k sobre o conjunto V (veja a seção 3.5.3). Faça $S := V \setminus \{v_0\}$.

Passo 2 Se $S = \emptyset$ pare e vá para o Passo 4. Caso contrário, inicialize uma rota k unindo a base ao vértice v_h de S mais próximo (mais longe no segundo teste). Faça $S := S \setminus \{v_h\}$.

Passo 3 Calcule os conjuntos $M_q(v) \subset S$ de todos os vértices v na rota k atual. Chame de U a união desses conjuntos e calcule para cada vértice v_h de U o custo de sua inserção na rota k atual usando as duas direções e os dois tipos de inserção usados na heurística GENI. Inclua na rota k atual o vértice v_h mais econômico se isto não supera o valor N_k, faça $S := S \setminus \{v_h\}$ e repita o Passo 3. Caso contrário vá para o Passo 2 (construção de uma nova rota).

Passo 4 Aplique os Passos de 2 a 5 da heurística da economia.

Note que os três primeiros passos da heurística da economia adaptada são usados para escolher as rotas iniciais, construindo para cada rota k um ciclo Hamiltoniano usando a heurística GENI, e ao mesmo tempo respeitam o número fixo de veículos e os valores N_k. Os testes demonstram que esse procedimento diminui o custo na solução final comparado com a heurística da economia original, pois além de diminuir rotas iniciais sobrepostas, ele faz dois testes de construção de rotas, o que aumenta a chance de obter soluções melhores.

4.6 Heurística Gananciosa para o m-PRC

Devido às sugestões de mudanças nas heurísticas, que foram sendo pensadas ao longo das implementações das heurísticas descritas no Capítulo 3, resolvemos criar uma nova heurística, que, devido à sua característica de escolha de vértices iniciais para obtenção de rotas, vamos chamar de Heurística Gananciosa para o m-PRC. Note que estamos seguindo a mesma idéia que encontramos em [16], onde os nomes das heurísticas para o m-PRC, sempre fazem referência ao procedimento usado para obtenção das rotas iniciais nos primeiros passos de cada heurística.

Por curiosidade, queríamos verificar o que ocorria quando a escolha de um vértice para inclusão em uma rota fosse sempre o mais próximo do anterior. Isto é feito nos Passos 1 e 2 desta nova heurística, analisando o conjunto $T \cup W$. No Passo 3, é construída cada rota k

separadamente utilizando a heurística H-1-PRC modificada, a qual testamos e achamos mais conveniente. Aplica-se o Teste de Cobertura no Passo 4, pois se existir algum vértice em alguma rota, sem necessidade, ele é retirado. E por fim pós-otimizamos usando a heurística 2-opt balanceada, que é mais adequada para o nosso propósito de trabalho.

O processo usado no Passo 1 da heurística gananciosa é muito simples. Construímos um ciclo gradativamente a partir da base, analisando sempre o vértice mais próximo no conjunto $T \cup W$. Ao selecionar um vértice em $T \cup W$, avaliamos se tal vértice pertence a T ou a W, incluindo vértices de T ou de $V \setminus T$, respectivamente, em uma rota. É ganancioso no processo de verificação da cobertura feita pelos vértices de $V \setminus T$. Sempre que incluímos um vértice no ciclo, os vértices de W que ele cobre não serão mais analisados. Após a construção do ciclo várias soluções são geradas e testadas de acordo com o tamanho do ciclo e dos valores N_k. Assim, temos que conhecer o total de vértices que o processo seleciona durante a construção do ciclo. Note que, eventualmente este total é menor que $|T \cup W| - 1$, pois um vértice $v_r \in S_h$ cobre outros vértices $v_h \in W$ diferente daquele avaliado em W, e eventualmente vértices de T também cobrem vértices de W, o que faz diminuir a quantidade de vértices analisados em W.

Vamos descrever um conjunto de passos que executa o processo de escolha dos vértices mais próximos para entrar no ciclo. Ele é muito semelhante ao que chamamos de Procedimento de Abertura de um Leque, descrito na Seção 3.5.4. Porém, a estratégia de seleção de vértices para rotas iniciais é muito distinta. Vamos chamá-lo de Procedimento de Seleção de Vértices GAN, como segue.

Procedimento de Seleção de Vértices GAN

- Iniciando de v_0, construa um ciclo G tomando todos os vértices $v_h \in T \cup W$ o mais próximo do anterior.
- Sejam $C_r = \{v_l \in W : c_{rl} \leq c\}$, $D := \{v_0\}$. Faça $G := G \setminus \{v_0\}$, $z := |G|$, então $G = \{v_1, v_2, \ldots, v_z\}$. Faça $h := 1$.
- Enquanto $h \leq z$
  - Se $v_h \in T$, $D = [D \quad v_h]$, $T \cup W := (T \cup W) \setminus C_h$ e $h := h + 1$.
  - Se $v_h \in W$, determine v_r de S_h que cobre o maior número de vértice de W, faça $D = [D \quad v_r]$, $T \cup W := (T \cup W) \setminus C_r$ e $h := h + 1$.
  - Se $v_h \notin W$, $h := h + 1$.
- D deve ser considerado como um ciclo iniciando da base.

Note que nos passos acima tomamos $G := G \setminus \{v_0\}$ pois a base pertence a toda rota e não precisamos computá-la com relação aos valores N_k. Agora podemos avaliar os valores N_k. Isso será feito de acordo com a quantidade de vértices incluídos em D a partir do Procedimento de Seleção de Vértices GAN. Tendo o conjunto D, os valores N_k são obtidos de forma idêntica ao apresentado na Seção 3.5.4, na heurística da varredura. Basta fazer $D := D \setminus \{v_0\}$. Agora podemos descrever os passos da heurística gananciosa para o m-PRC, como segue.

4.6.1 Passos da Heurística Gananciosa para o m-PRC

Passo 1 (*inicialização*). Determine o vetor D através do Procedimento de Seleção de Vértices GAN e obtenha os valores N_k.

Repita os Passos 2-5 iniciando com cada vértice v incluído na primeira rota construída no Passo 2 e selecione a melhor de todas as soluções.

Passo 2 (*seleção de vértices*). Iniciando de um vértice v arbitrário em D, construa uma rota k fazendo um passeio em D e incluindo os vértices visitados neste passeio até atingir o valor N_k, complete tal rota unindo seu último vértice a v_0. Então considere o próximo vértice neste passeio como v, e repita este passo para a próxima rota até todos os vértices de D serem incluídos em alguma rota.

Passo 3 (*H-1-PRC modificada*). Considere todas k rotas de veículo. Determine W_k, o conjunto de vértices de W cobertos por um vértice de rota k, e V_k, os vértices de V cobrindo W_k. Aplique a heurística H-1-PRC modificada para o subproblema induzido por $V_k \cup W_k$. Se algum vértice v pertence a mais de uma rota, ele é guardado em apenas uma rota, naquela onde os custos são mais reduzidos. Chame de $\tilde{k}$ a nova rota obtida, desta maneira, da rota k. Se $\tilde{k}$ é viável e tem um custo menor do que da rota k, então substitua k por $\tilde{k}$. Caso contrário, a rota k não é alterada.

Passo 4 (*Teste de Cobertura*). Aplique o Teste de Cobertura, utilizando todos os vértices das rotas. Verifique se existem vértices que podem ser retirados. Obedecendo a ordem de economia, retire-os das rotas sem causar não cobertura em algum vértice de W.

Passo 5 (*melhoria da solução*). Pós-otimize a solução m-PRC por meio da heurística 2-opt balanceada.

Note que várias soluções podem ser geradas de acordo com a dimensão do problema. Da forma que as rotas são construídas no Passo 3, dificilmente ocorre multicoberturas de vértices em W por vértices em rotas distintas. Mas mesmo assim, muitas das configurações podem ser como ilustrado na Figura 4.9. Nela, o vértice v_z é coberto pelo vértice v_t da Rota

1 e pelo vértice v_u da Rota 3. Neste caso nenhum dos vértices podem ser eliminados de suas rotas, pois caso fosse feito, ou v_w, ou v_r e v_s ficariam sem cobertura. Porém, quando ocorrer multicobertura como na Figura 4.3, o Passo 4 permitirá a remoção de um vértice de uma das rotas.

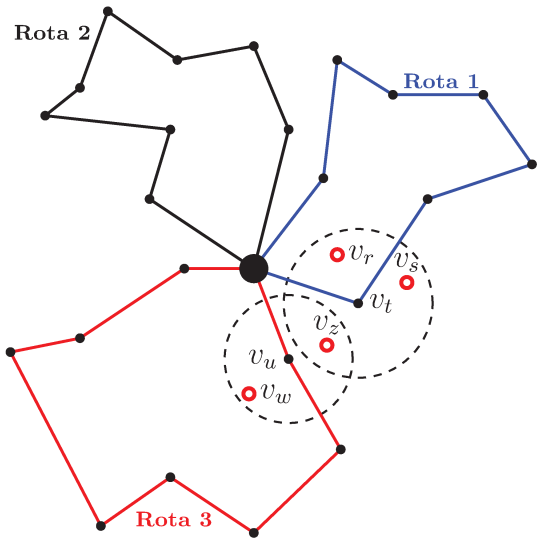


Figura 4.9: Ocorrência de multicobertura de vértices em W

O procedimento usado nos Passos 1 e 2 da heurística gananciosa pode retornar soluções viáveis não muito boas como, por exemplo, ilustrado na Figura 4.10. Nela, duas rotas de veículo são construídas incluindo sempre o vértice mais próximo, e temos como resposta uma configuração que provavelmente tem um custo alto comparado com outras configurações. Porém, como várias outras possibilidades de configurações serão testadas por este método, esperamos boas soluções finais.

4.7 Heurística Divide-Espaço

Nesta seção apresentamos uma nova heurística que foi criada após analisarmos as anteriores e verificarmos quais procedimentos e operações que poderiam ser evitadas e, ao mesmo tempo, buscando melhores resultados. Os primeiros passos de cada heurística de construção de rotas viáveis para o m-PRC fornecem uma solução inicial para ser melhorada pela heurística H-1-PRC. A solução inicial é obtida analisando-se os vértices dos conjuntos V ou $T \cup W$, que acabam sendo boas quando o subconjunto de vértices escolhidos para construir cada rota estão próximos ou aglomerados, pois assim as rotas finais ficam sem sobreposições e o custo é reduzido.

Podemos afirmar que a aplicação H-1-PRC é o passo mais importante das heurísticas, a qual combina a solução de um PCV e de um PC, garantindo as visitas obrigatórias e a

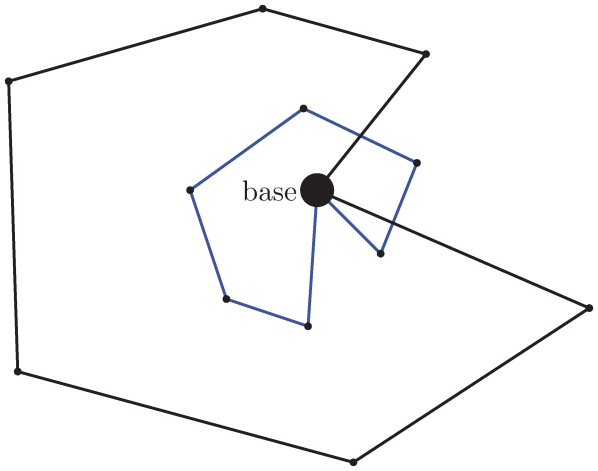


Figura 4.10: Construção de duas rotas ruins no método ganancioso

cobertura do conjunto W em cada aglomeração. Depois são analisadas algumas melhorias de pós-otimização.

Sendo assim, as heurísticas se dividem em três etapas principais, as escolhas das aglomerações, a aplicação H-1-PRC e uma pós-otimização. Durante essas três etapas vários testes são feitos. O que queremos evitar com essa nova heurística, são os primeiros testes para se escolher as aglomerações sobre o conjunto $V \cup W$, mas de forma inteligente para que a pós-otimização 2-opt* ou a 2-opt balanceada não seja usada. E apenas a H-1-PRC modificada seja aplicada em uma segunda etapa.

Vamos acompanhar o que vem escrito a seguir e as Figuras 4.11 e 4.12 para entender o procedimento usado nesta nova heurística que vamos chamar de Heurística Divide-Espaço.

Primeiro selecionamos a base no espaço onde está o nosso conjunto $V \cup W$, traçamos uma reta sobre a base e a partir desta reta vamos dividir todo esse espaço em setores. Encontramos na Figura 4.11 uma ilustração deste fato. Nela, a base é selecionada entre os vértices que estão representados por circunferências, estrelas e pontos, conforme os conjuntos W, T e $V \setminus T$ respectivamente. Uma linha horizontal é traçada sobre a base e tomamos ela como referência para analisarmos o espaço todo no sentido anti-horário, conforme indica o vetor.

Vamos dividir o espaço em m setores, onde cada setor é composto pelos vértices que pertencem a um setor angular no espaço. O tamanho dos setores angulares é obtido dividindo 2π por m. Cada parte do espaço com este tamanho vamos chamar de setor k, que faz referência a rota k.

Em cada setor k vamos ter um subconjunto $V_k \cup W_k$ de $V \cup W$ que é o conjunto onde aplicaremos a heurística H-1-PRC modificada para construção de cada rota k.

Desta forma não teremos possibilidades de construir rotas sobrepostas, então não precisamos em princípio aplicar a heurística 2-opt* ou a 2-opt balanceada, pois a recombinação de arestas ou a troca de vértices em rotas distintas provavelmente não serão permitidas.

Não vamos nos preocupar com as possíveis recombinações feitas numa mesma rota, pois isso também é tratado pela heurística H-1-PRC modificada ao usar a heurística GENIUS.

Vamos definir θ_n como sendo o ângulo a partir da linha traçada sobre a base no n-ésimo teste de solução m-PRC. A idéia é tomar a reta que passa pela base como referência e em cada teste de construção de rotas somamos uma pertubação no ângulo θ_n atual, ou seja, se a pertubação escolhida é α, no n-ésimo teste temos como ângulo inicial a partir da reta de referência $\theta_n = \theta_{n-1} + \alpha$, para $n = 1, \ldots, t$, onde $\theta_1 = 0$ e t é o número de teste que pretendemos fazer. Em cada teste obtemos um novo conjunto de setores para a construção de cada rota k. Podemos adiantar que optamos por fixar $t = 10$ para todos exemplares testados. Note que o tamanho do setor angular $\frac{2\pi}{m}$ é fixo e o que irá variar é o ângulo que se inicia a divisão do espaço. A Figura 4.12 ilustra como dividimos o espaço em três setores. Nela, os vértices estão representados por pontos, círculos e estrelas conforme o seu conjunto. Pretendemos construir três rotas de veículo, e a figura representa como os setores ficaram divididos em setores angulares de tamanho $\frac{2\pi}{3}$, os quais formam três subconjuntos de $V \cup W$. A Figura 4.12 representa o primeiro teste que será efetuado pela heurística divide-espaço e a Figura 4.13 representa um novo teste onde já foi feita uma pertubação no angulo θ_n.

Da maneira que segue, em cada teste a aplicação H-1-PRC modificada só poderá retornar um vértice com multicobertura se for por rotas distintas. Assim, como última etapa, aplicamos o Teste de Cobertura como uma pós-otimização.

Da forma que dividimos o espaço pode ocorrer falha de cobertura, pois como o valor c é fixo para todo conjunto de vértices, em algum setor podemos ter um vértice de W sem cobertura, pois o vértice de V que o cobre pode estar em um outro setor. Poderíamos corrigir esta falha definindo c_k para cada setor k, porém levando em conta a formulação m-PRC descrita, teríamos problema de inviabilidade. Note também que ao se completar um teste, pode ocorrer que a diferença entre os valores N_k seja inviável. Nesta heurística decidimos acrescentar mais um teste toda vez que um dos casos acima ocorrer, fazendo $\alpha := \frac{\alpha}{2}$, na tentativa de não desperdiçar um teste. Porém se neste novo teste a solução continuar infactível ela é eliminada.

Agora podemos descrever os passos da heurística divide-espaço como segue.

4.7.1 Passos da Heurística Divide-Espaço

Passo 1 (*divide-espaço*) Divida o Grafo induzido por $V \cup W$ em m setores conforme foi descrito anteriormente e $n := 0$.

Enquanto $n \leq t$, *faça* $\alpha = \frac{2\pi}{mt}$, $n := n+1$ *e repita os Passos 2 e 3 variando o ângulo* θ_n *a partir da reta referência, que passa pela base, tal que no* n*-ésimo teste* $\theta_n = \theta_{n-1} + \alpha$, *onde* $\theta_1 = 0$, t *é o número de teste pretendido e* m *é o número de veículos disponível. Se ocorrer falha de cobertura em algum setor* k, *faça* $\theta_n := \theta_n + \frac{\alpha}{2}$, *apenas uma vez para cada teste atual. Selecione a melhor de todas as soluções.*

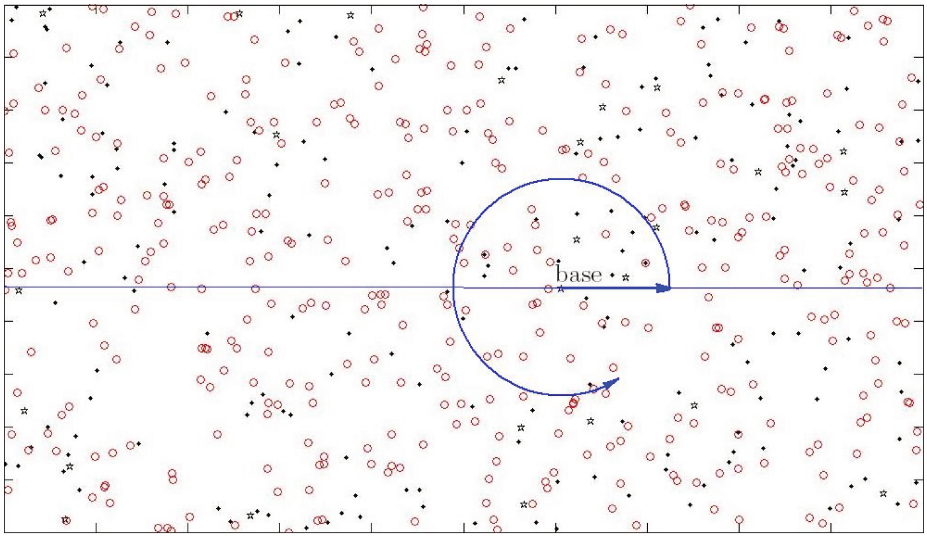


Figura 4.11: Início da análise do espaço de busca na heurística divide-espaço

Passo 2 (*H-1-PRC modificada*) Para cada setor k, $k \leq m$, construído a partir de θ_n, aplique a heurística H-1-PRC modificada para o subproblema induzido por $V_k \cup W_k$ pertencente ao setor k.

Passo 3 (*Teste de Cobertura*) Aplique o Teste de Cobertura sobre as rotas e retire vértices da rota por ordem de economia se for possível. Se ocorrer problema de inviabilidade com relação aos valores N_k repita o Passo 2 com $\theta_n := \theta_n + \frac{\alpha}{2}$ apenas uma vez para cada teste atual.

Note que esta heurística tem melhores aplicações quando a base está mais centralizada em nosso espaço de busca, pois os k subconjuntos $V_k \cup W_k$ estarão mais bem distribuídos. Podemos ver um exemplo na Figura 4.14, onde esta heurística não produzirá bons resultados porque os setores têm quantidade de vértices muito distintos. Porém para os exemplares gerados aleatoriamente foi decidido que a base ficaria centralizada de acordo com a idéia encontrada em [16]. Sendo assim, esperamos bons resultados com essa nova heurística.

4.8 Implementações das Heurísticas

Temos que salientar que para nenhuma das heurísticas apresentadas foram encontrados códigos já implementados, sendo assim todas as implementações foram produzidos, do início, em código Matlab a partir de idéias novas e através das heurísticas descritas nas referências.

Também não foram encontrados exemplares do m-PRC disponíveis em bibliotecas digitais. Por conta disso, ficou decidido implementar todas as heurísticas de construção de rotas

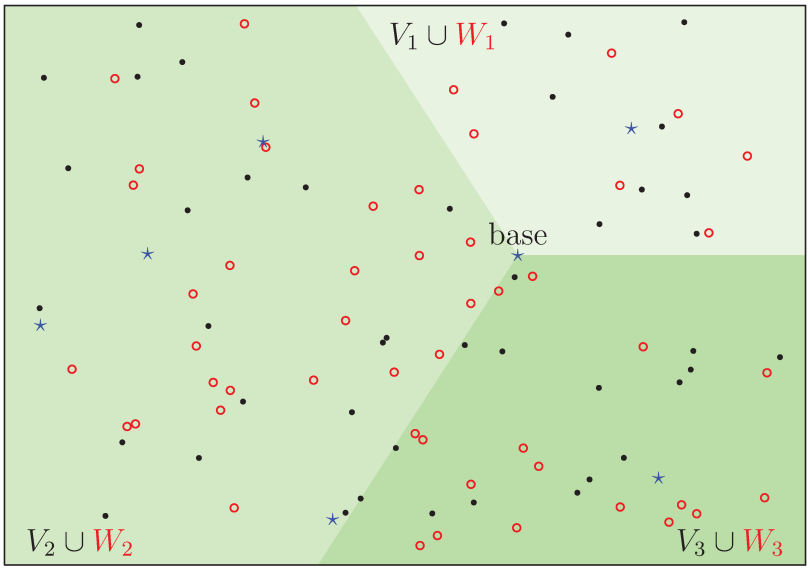


Figura 4.12: Espaço de busca dividido em três setores em um primeiro teste

estudadas em [16] com o foco de processar exemplares para m-PRC levando em conta nosso problema de construção de rotas de patrulhamento preventivo, ou seja, nos primeiros passos de cada heurística estudada tivemos a preocupação de tentar controlar os valores N_k das rotas.

Vale citar que a maior dificuldade de implementação foi encontrada na heurística GENIUS que requer grande número de testes e diversas configurações de inserções e remoções de vértices. Ainda em termos de dificuldade, segue a implementação do Passo 1 da heurística da economia original, depois a heurística H-1-PRC devido a combinação do problema de cobertura com um PCV e em seguida a heurística 2-opt balanceada devido ao grande número de recombinações possíveis de serem feitas.

Acreditamos que as implementações podem ser melhoradas, pois não nos preocupamos com a alocação de memória nem o melhor armazenamento de matrizes simétricas e esparsas. Como esse problema é um problema de grande complexidade achamos que uma nova implementação em um código compilado pode trazer melhores resultados com relação ao tempo de processamento.

Ficam como contribuições a implementação de todas as heurísticas estudadas e as sugestões de novas heurísticas.

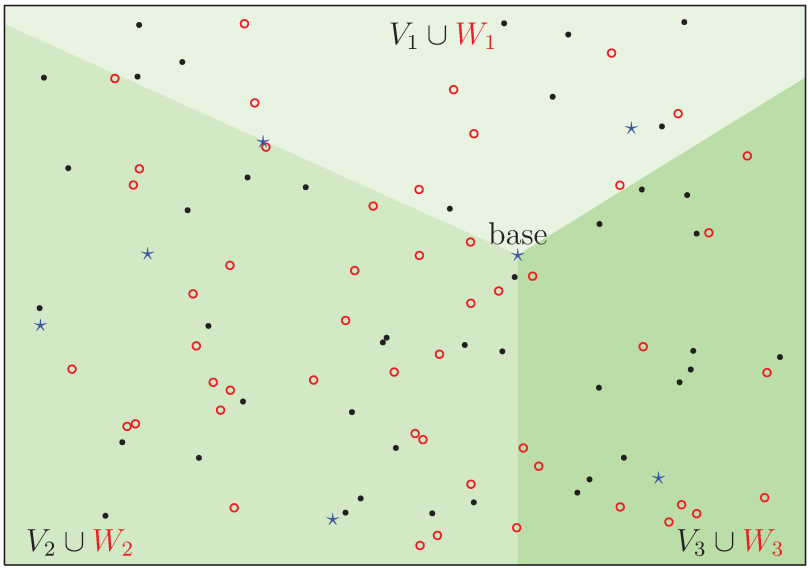


Figura 4.13: Espaço de busca dividido em três setores em um segundo teste

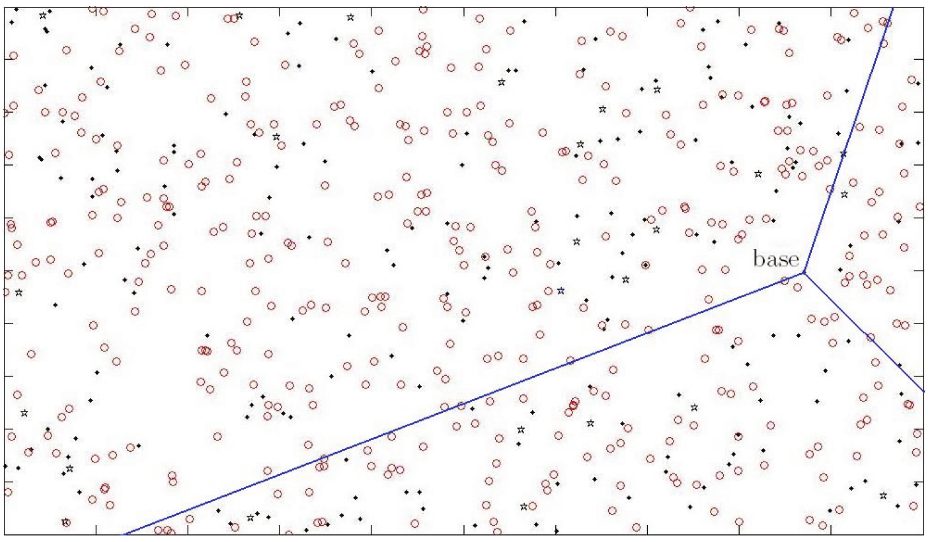


Figura 4.14: Ocorrência de posição ruim da base para a heurística divide-espaço

Capítulo 5

Exemplares Gerados e Resultados Computacionais

Neste capítulo apresentamos como foram construídos os exemplares gerados aleatoriamente e os resultados obtidos a partir das heurísticas estudadas. Os resultados são apresentados em tabelas, figuras e gráficos. É relatado como os dados do problema em estudo foram coletados, suas características principais, bem como os resultados obtidos.

5.1 Definição da Constante c

A definição da constante c na formulação do m-PRC é muito importante, pois ela garante que todos os vértices de W possam ser cobertos. Isso é essencial para garantir um problema viável.

5.1.1 Constante c nos Exemplares Gerados Aleatoriamente

Nas referências [16] e [14] o valor de c está definido como:

$$c = \max\{\max_{v_h \in V \setminus T} \ \min_{v_l \in W}\{c_{lh}\}, \quad \max_{v_l \in W}\{c_{l,h(l)}\}\},$$

onde $h(l)$ é o índice do vértice em $V \setminus T$ que é o segundo mais próximo a v_l. Com esta definição, cada vértice de $V \setminus T$ cobre pelo menos um vértice de W e cada vértice de W é coberto por pelo menos dois vértices de $V \setminus T$. Isso porque nessas referências ao definirem o conjunto $S_l = \{v_h \in V : c_{hl} \leq c\}$ para cada $v_l \in W$, é exigido que $|S_l| \geq 2$. Note que a definição de c garante esta cardinalidade do conjunto S_l.

5.1.2 Constante c no Problema em Estudo

No nosso problema em estudo o valor de c pode ser alterado de um teste para outro, e assim, as configurações das rotas obtidas como solução mudam. Uma escolha inadequada de c pode tornar o problema inviável, pois poderíamos ter endereços importantes sem cobertura. O

administrador responsável pela região escolhida para construir as rotas de patrulhamento preventivo decidiu que 150m seria uma distância adequada, para que um local fosse coberto no momento da passagem de uma viatura. Esse valor foi usado na resolução do exemplar criado com os dados obtidos naquela cidade. Como na nossa idéia todas as esquinas e cruzamentos são locais que tornaram-se vértices do conjunto V, sempre obtivemos $|S_l| \geq 1$.

5.2 Construção dos Exemplares

Nesta seção vamos apresentar os exemplares gerados aleatoriamente e o exemplar do problema em estudo com dados reais, suas características e diferenças.

5.2.1 Exemplares Gerados Aleatoriamente

Em [16] foram construídos e testados alguns exemplares gerados aleatoriamente. Pela falta de exemplares prontos em bibliotecas decidimos gerar nossos exemplares seguindo a mesma idéia deste artigo. Construímos exemplares para o m-PRC gerados aleatoriamente com distribuição uniforme. Foram criados 302 exemplares. Para cada exemplar foram gerados $|V| + |W|$ pontos numa região quadrada $[0, 100] \times [0, 100]$.

A localização da base ficou restrita ao quadrado $[35, 65] \times [35, 65]$. Os conjuntos T e V foram definidos ao tomar os primeiros $|T|$ e $|V|$ pontos, respectivamente. Todos os outros pontos foram atribuídos a W. Os exemplares testados neste caso são combinações dos seguintes valores: $|V|$= 50, 100, 200; $|W|$= 50, 100, 200, 400 e $|T|$=$\left\lceil \frac{|V|}{8} \right\rceil$, $\left\lceil \frac{|V|}{4} \right\rceil$, $\left\lceil \frac{|V|}{2} \right\rceil$. Dividimos 300 exemplares em classes, como podemos ver na Tabela 5.1. Nela observamos que os exemplares se dividem em 5 classes principais com dimensões iguais a 100, 150, 200, 300 e 400. Em cada uma dessas classes principais fizemos outras três subdivisões de acordo com o tamanho do conjunto T escolhido. Foram gerados 20 exemplares em cada subdivisão. No final da tabela aparecem seis exemplares que foram nomeados com as iniciais RPP, os quatro primeiros são de dimensões 100 e 200, os quais foram selecionados das classes de dimensões equivalentes. Os dois últimos são de dimensão igual a 600. Usamos, tais exemplares, denominados `RPP-100-1`, `RPP-100-2`, `RPP-200-1`, `RPP-200-2`, `RPP-600A-1` e `RPP-600A-2` em testes descritos nas Seções 3.4.3, 4.3.2 como forma de comparação das heurísticas que resolvem o 1-PRC.

5.2.2 Exemplar do Problema em Estudo

Foram disponibilizados pela Guarda Civil Municipal da cidade de Vinhedo-SP dados referentes a segurança da cidade. O administrador de segurança local providenciou uma lista com 102 endereços referentes a locais que seu departamento tem o dever de promover segurança. Entre esses endereços estão todas as escolas, creches, postos de saúde, hospitais públicos e outros prédios como prefeitura e secretarias. Formamos com esta lista o nosso conjunto T de visita obrigatória. Outra lista foi providenciada contendo 133 endereços importantes

$\lvert T\rvert = \left\lceil \frac{\lvert V\rvert}{8} \right\rceil, \left\lceil \frac{\lvert V\rvert}{4} \right\rceil, \left\lceil \frac{\lvert V\rvert}{2} \right\rceil$					
Exemplar	Quantidade	Dimensão	$\lvert V\rvert$	$\lvert W\rvert$	$\lvert T\rvert$
Classe 100-1	20	100	50	50	7
Classe 100-2	20	100	50	50	13
Classe 100-3	20	100	50	50	25
Classe 150-1	20	150	50	100	7
Classe 150-2	20	150	50	100	13
Classe 150-3	20	150	50	100	25
Classe 200-1	20	200	100	100	13
Classe 200-2	20	200	100	100	25
Classe 200-3	20	200	100	100	50
Classe 300-1	20	300	100	200	13
Classe 300-2	20	300	100	200	25
Classe 300-3	20	300	100	200	50
Classe 400-1	20	400	200	200	25
Classe 400-2	20	400	200	200	50
Classe 400-3	20	400	200	200	100
`RPP-100-1`	1	100	50	50	7
`RPP-100-2`	1	100	50	50	13
`RPP-200-1`	1	200	100	100	13
`RPP-200-2`	1	200	100	100	25
`RPP-600A-1`	1	600	200	400	25
`RPP-600A-2`	1	600	200	400	50

Tabela 5.1: Exemplares gerados aleatoriamente com distribuição uniforme

para se promover uma ação preventiva de segurança na cidade. Entre esses endereços estão várias praças, algumas ruas (neste caso tomamos o número central da rua como um ponto de vértice), alguns cruzamentos com semáforos e prédios públicos estaduais. Formamos com esta segunda lista o conjunto W a ser coberto. Essas duas listas foram encaminhadas para a empresa SSR-Tecnologia do setor de geoprocessamento daquela cidade, tal empresa é detentora de um mapa digital da cidade. A SSR-Tecnologia providenciou as coordenadas geográficas desses endereços em questão. Ela também forneceu as coordenadas de todos os cruzamento de ruas da cidade, as quais somaram 2496 locais. Com esses últimos locais formamos o conjunto $V \setminus T$ de visita opcional. Desta forma foi possível criar o grafo e a matriz de custos para a formulação m-PRC. As dimensões dos conjuntos selecionados para o exemplar construído com tais dados estão ilustrados na Tabela 5.2.

A Figura 5.1 ilustra como ficou a distribuição dos vértices do exemplar com os dados reais. Nela conseguimos notar que o problema em estudo difere dos problemas gerados

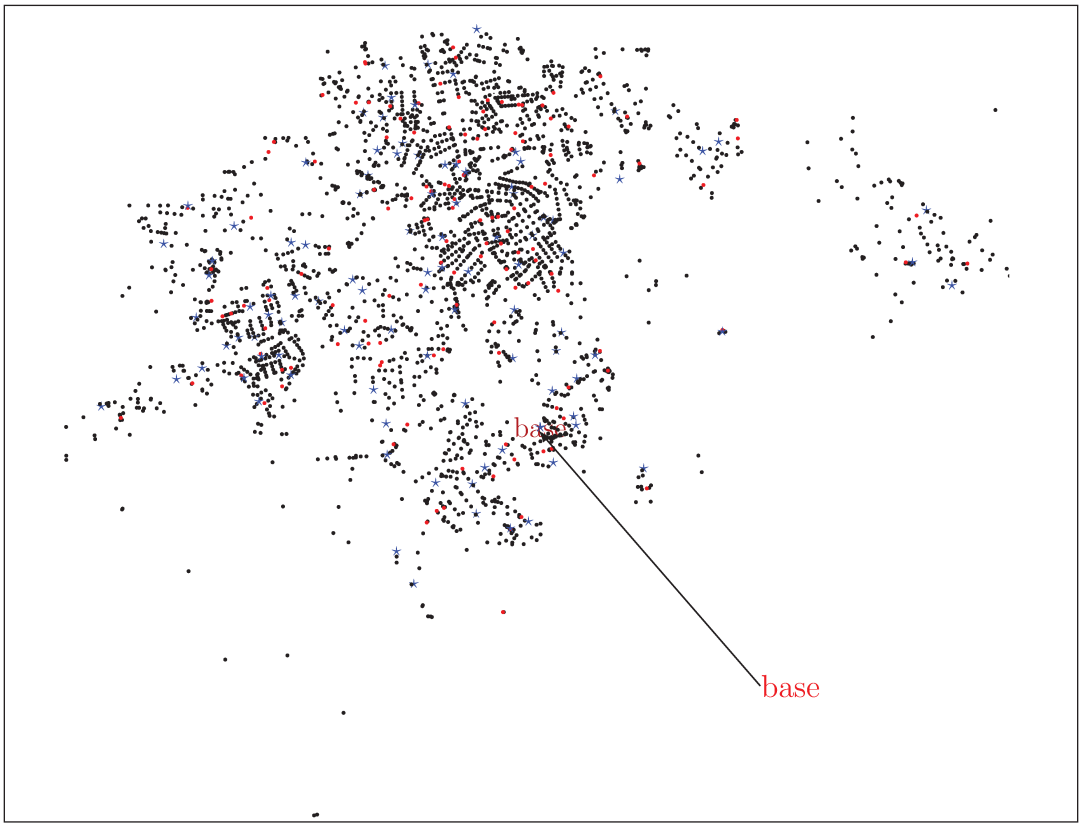


Figura 5.1: Configuração geográfica do exemplar obtido na cidade de Vinhedo

Exemplar	Quantidade	Dimensão	$\|V\|$	$\|W\|$	$\|T\|$
Dados Reais	1	2731	2496	133	102

Tabela 5.2: Exemplar gerado com dados reais

aleatoriamente, pois a distribuição dos pontos não é uniforme e a quantidade de vértices de visita opcional é muito maior que a quantidade de vértices em W. A não uniformidade dos dados já era previsto. Note que a base não fica bem centralizada na cidade, prevemos que esse fato pode prejudicar os resultados da heurística divide-espaço. Os pontos pretos representam o conjunto $V \setminus T$ de visita opcional, os pontos vermelhos representam o conjunto W a ser coberto e as estrelas azuis representam o conjunto T de visita obrigatória.

Devido a configuração do nosso problema em estudo e a grande quantidade de vértices em $V \setminus T$, efetuamos um pré-processamento e reduzimos a quantidade de vértices deste conjunto. Apenas os vértices que estão dentro de um raio de 150m de cada vértice de W foram selecionados, pois são os únicos que têm a possibilidade de fazer parte de uma solução final. A Figura 5.2 mostra a nova configuração reduzida do exemplar obtido com os dados reais, os quais somam agora 933 vértices no conjunto $V \setminus T$.

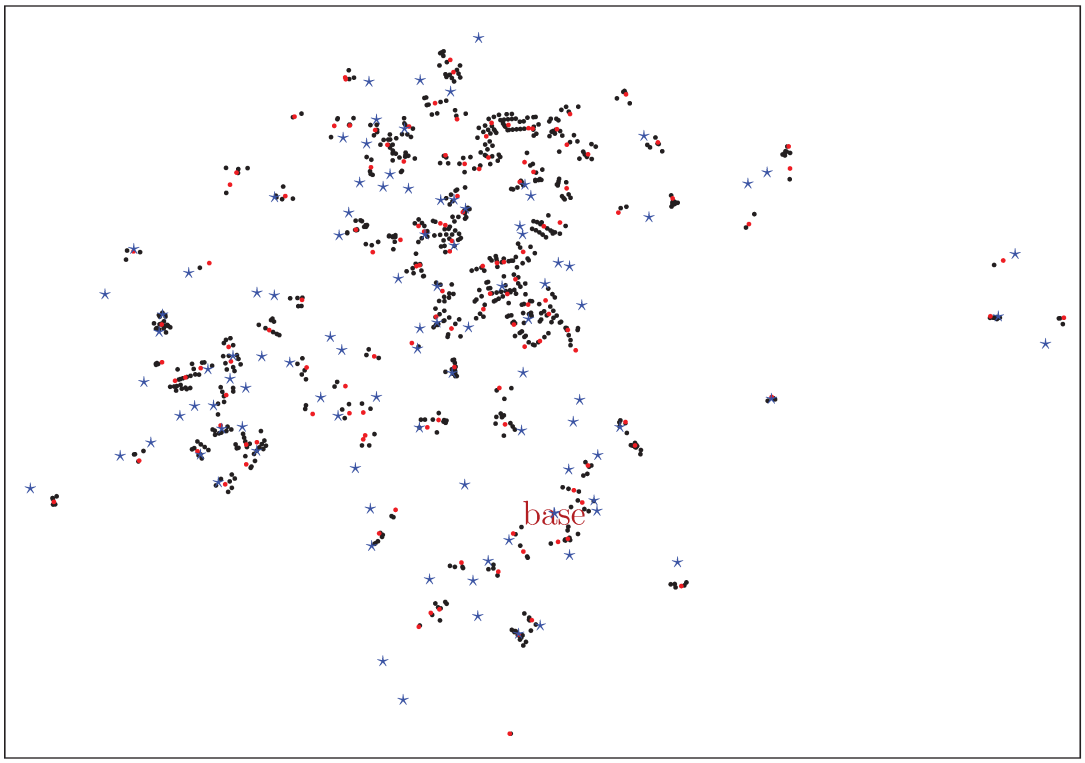


Figura 5.2: Configuração do exemplar obtido na cidade de Vinhedo reduzido

5.3 Resultados Computacionais

As implementações foram feitas em um código MATLAB. Para o caso dos exemplares gerados aleatoriamente foram usados dois tipos de máquinas, devido a necessidade de mais memória para os exemplares de maior dimensão. Os exemplares com dimensão de 400 vértices foram processados em um computador com processador Intel Pentium de 3.40GHz, 2 GB de memória RAM e sistema operacional Linux. Todos os outros, foram processados em um computador com processador Intel Celeron de 1.73GHz, 512 MB de memória RAM e sistema operacional Windows. O exemplar com dados reais foi resolvido em um computador com processador Intel Core 2.40GHz, 2 GB de memória RAM e sistema operacional Windows. Os resultados computacionais são apresentados nesta seção por meio de tabelas e gráficos. Também achamos importante ilustrarmos algumas figuras com as configurações das rotas obtidas como resposta.

Durante este trabalho foram apresentados vários procedimentos e heurísticas que podem

ser combinados na construção de rotas de cobertura, porém o número de combinações é grande e não achamos de interesse todos esses testes. Sendo assim escolhemos os procedimentos que se mostraram mais eficientes após alguns testes computacionais. Lembrando que as heurísticas de construção de rotas para o m-PRC se dividem em três fases principais, nas quais, a primeira é de seleção de vértices para rotas iniciais ou aglomerações, a segunda é de construção de cada rota individualmente e a terceira de pós-otimização. A Tabela 5.3 mostra a combinação de todas as heurísticas escolhidas e alguns valores fixados para os testes dos exemplares gerados aleatoriamente e para o exemplar com dados reais. Nela, a letra p representa o parâmetro da p-vizinhança usada na heurística GENIUS, fixamos seu valor igual a 4 em todos os testes. Na segunda fase usamos a heurística H-1-PRC modificada na construção de cada rota do problema 1-PRC em todos os testes, pois essa demonstrou ser mais promissora. O Teste de Cobertura foi usado como pós-otimização na heurística divide-espaço, e em todas as outras usamos a heurística 2-opt balanceada. Fixamos o número de veículos igual a 3 em todos os testes dos exemplares gerados aleatoriamente e variamos em 5, 6 e 7 o número de veículos para o exemplar com dados reais, sendo que essa variação, representa o número de viaturas que a Guarda Civil de Vinhedo normalmente utiliza para o serviço de patrulhamento preventivo, segundo dados fornecidos por eles.

Exemplares gerados aleatoriamente

1ª Fase	2ª Fase	3ª Fase	Número de veículos	p
Heurística da Economia Heurística da Varredura Heurística 1-PRC/m-PRC Heurística da Economia Adaptada Heurística Gananciosa	H1PRC-MOD	2-opt Balanceada	3	4
Heurística Divide-Espaço		Teste de Cobertura		

Exemplar com dados reais

Heurística da Varredura Heurística 1-PRC/m-PRC	H1PRC-MOD	2-opt Balanceada	5, 6 e 7	4
Heurística Divide-Espaço		Teste de Cobertura		

Tabela 5.3: Heurísticas e valores escolhidos nos testes computacionais

Custos e tempos médios por classe

	Economia			Varredura			1-PRC m-PRC		
Classe	relação	custo	tempo	relação	custo	tempo	relação	custo	tempo
Classe 100-1	1.20464	457.963	5.6358	1.03993	395.345	5.3561	1	380.166	6.6219
Classe 100-2	1.23804	552.174	5.6576	1.0358	461.976	7.2391	1	446.007	13.7078
Classe 100-3	1.16473	610.525	10.6328	1.00913	528.963	59.2344	1	524.176	108.788
Classe 150-1	1.20858	500.838	8.7172	1.02326	424.04	12.4843	1	414.402	11.547
Classe 150-2	1.20172	557.815	9.6672	1.01634	471.766	13.3595	1	464.181	20.1435
Classe 150-3	1.20641	633.424	11.7374	1.01531	533.088	60.1294	1	525.05	105.262
Classe 200-1	1.32977	667.159	43.9515	1.02744	515.48	108.431	1	501.712	97.2156
Classe 200-2	1.27904	694.744	71.636	1.00982	548.511	257.64	1	543.175	328.409
Classe 200-3	1.14262	775.106	154.877	1.00024	678.525	2886.66	1	678.361	3041.96
Classe 300-1	1.29496	671.247	80.3468	1.0153	526.281	319.708	1.0034	520.116	259.494
Classe 300-2	1.27569	693.452	113.353	1.02584	557.636	492.587	1	543.591	395.56
Classe 300-3	1.20626	800.685	219.889	1.00435	666.661	3739.28	1	663.775	4377.25
Classe 400-1	1.34145	828.184	970.625	1.01513	626.721	4760.12	1.00466	620.255	3689.24
Classe 400-2	1.33865	932.863	955.191	1.0081	702.517	6203.82	1	696.871	6630.7
Classe 400-3	1.20978	1030.45	491.246	1	851.769	12362.4	1.00937	859.751	13796.7

Tabela 5.4: Resultados das heurísticas estudadas

5.3.1 Análise dos Resultados

Chamamos de relação o valor obtido ao se dividir o resultado da função objetivo de cada heurística pelo melhor resultado obtido entre todas as heurísticas por exemplar testado. Assim, para cada exemplar a heurística mais eficiente tem a relação igual a 1. Isso facilita na comparação entre as eficiências das heurísticas com relação ao custo obtido, pois neste nosso estudo não temos valores ótimos para validar os resultados.

As Tabelas 5.4 e 5.5 mostram os custos e tempos médios por classe das seis heurísticas para construção de rotas, sendo que a 5.4 refere-se às três estudadas em [16] e a 5.5 às três heurísticas novas. Nelas a primeira coluna de cada heurística refere-se às relações, a segunda à função objetivo (custo) encontrada e a terceira ao tempo de processamento em segundos. A Tabela 5.6 mostra os resultados dos custos e tempos médios das heurísticas mais eficientes entre as seis analisadas. Observando essas três tabelas, e levando em conta o critério da relação, vemos que as melhores heurísticas são as da varredura, primeiro 1-PRC/segundo m-PRC e divide-espaço, pois só nessas aparecem valores iguais a 1 na coluna de relações. Como as tabelas representam valores médios, ou seja, o valor médio entre os 20 exemplares de cada classe, a relação neste caso é a divisão entre os valores médios de cada heurística pelo melhor valor médio encontrado entre elas, isso para cada classe.

Primeiro vamos comparar os resultados da Tabela 5.4, a qual apresenta os resultados das heurísticas estudadas em [16]. Nela, podemos observar que a heurística primeiro 1-PRC/segunda m-PRC se destacou com relação aos custos, pois foi melhor que as outras duas em 12 classes de exemplares. Note que neste caso a heurística da varredura foi mais

Custos e tempos médios por classe

	Economia Adaptada			Gananciosa			Divide-Espaço		
Classe	relação	custo	tempo	relação	custo	tempo	relação	custo	tempo
Classe 100-1	1.14629	435.78	15.4468	1.04391	396.86	7.4719	1.00194	380.904	8.3093
Classe 100-2	1.214	541.454	15.6594	1.07043	477.42	8.2156	1.05172	469.074	15.4967
Classe 100-3	1.13907	597.073	28.4687	1.05952	555.377	59.636	1.01273	530.846	147.833
Classe 150-1	1.15421	478.308	20.2687	1.04823	434.389	17.0282	1.00707	417.334	13.1171
Classe 150-2	1.17043	543.291	21.7248	1.05947	491.785	15.1563	1.01745	472.282	22.5811
Classe 150-3	1.1597	608.901	28.9096	1.0734	563.591	63.4656	1.02919	540.374	138.959
Classe 200-1	1.22622	615.21	101.133	1.08077	542.233	135.855	1.00064	502.032	65.2844
Classe 200-2	1.19865	651.077	198.353	1.07453	583.659	306.536	1.01954	553.786	307.34
Classe 200-3	1.133	768.586	400.899	1.06333	721.318	2505.12	1.00505	681.786	2011.24
Classe 300-1	1.17476	608.939	146.799	1.11228	576.555	539.375	1	518.352	140.244
Classe 300-2	1.24667	677.679	290.18	1.08493	589.76	519.959	1.03123	560.569	259.51
Classe 300-3	1.17018	776.737	630.481	1.09578	727.353	4377.69	1.02788	682.285	1953.89
Classe 400-1	1.23256	760.956	2238.29	1.10041	679.369	7785.5	1	617.378	1384.7
Classe 400-2	1.25352	873.539	2149.29	1.11266	775.377	9376.62	1.00972	703.644	2345.59
Classe 400-3	1.16853	995.316	1108.93	1.10688	942.803	13176.8	1.02135	869.952	6654.26

Tabela 5.5: Resultados das novas heurísticas

eficiente apenas na Classe 400-3. Adiantamos que, esta última heurística também se destacou com relação ao custo obtido como solução do exemplar com dados reais, isto sugere que esta heurística é mais eficiente para exemplares com dimensões maiores, porém outros testes devem ser feitos para tal comprovação. Ainda na Tabela 5.4, vemos que há um equilíbrio entre os tempos de processamento das heurísticas da varredura e primeiro 1-PRC/segunda m-PRC, porém a heurística da economia é sempre mais rápida, com exceção da Classe 100-1. Esse fato deve ocorrer por conta que na heurística da economia é feito apenas um teste de solução do m-PRC. Atribuímos os melhores custos obtidos pela heurística primeiro 1-PRC/segunda m-PRC à sua característica de obter uma solução inicial em uma primeira fase.

Vamos comparar agora os resultados da Tabela 5.5. Nela, encontramos os resultados médios das heurísticas novas. A heurística divide-espaço se destaca como a mais eficiente, ganha em todas as classes com relação aos custos médios obtidos. Note que na sequencia os melhores custos são da heurística gananciosa e depois da heurística da economia adaptada. Para esta última atribuímos os piores resultados pelo fato dela testar apenas duas soluções m-PRC por exemplar, porém, por conta disso, o seu tempo de processamento deve ser menor. Podemos confirmar isso comparando com os tempos da heurística gananciosa, que gera vários testes de solução por exemplar. Ainda na Tabela 5.5 notamos que a heurística divide-espaço encontra os melhores custos processando em um tempo médio abaixo das outras duas heurísticas em cinco classes, e nas outras classes, os tempos são próximos daqueles obtidos pela heurística da economia adaptada. Um fato importante, pois fixamos em 10 o número

Custos e tempos médios por classe

	Varredura			1-PRC m-PRC			Divide-Espaço		
Classe	relação	custo	tempo	relação	custo	tempo	relação	custo	tempo
Classe 100-1	1.03993	395.345	5.3561	1	380.166	6.6219	1.00194	380.904	8.3093
Classe 100-2	1.0358	461.976	7.2391	1	446.007	13.7078	1.05172	469.074	15.4967
Classe 100-3	1.00913	528.963	59.2344	1	524.176	108.788	1.01273	530.846	147.833
Classe 150-1	1.02326	424.04	12.4843	1	414.402	11.547	1.00707	417.334	13.1171
Classe 150-2	1.01634	471.766	13.3595	1	464.181	20.1435	1.01745	472.282	22.5811
Classe 150-3	1.01531	533.088	60.1294	1	525.05	105.262	1.02919	540.374	138.959
Classe 200-1	1.02744	515.48	108.431	1	501.712	97.2156	1.00064	502.032	65.2844
Classe 200-2	1.00982	548.511	257.64	1	543.175	328.409	1.01954	553.786	307.34
Classe 200-3	1.00024	678.525	2886.66	1	678.361	3041.96	1.00505	681.786	2011.24
Classe 300-1	1.0153	526.281	319.708	1.0034	520.116	259.494	1	518.352	140.244
Classe 300-2	1.02584	557.636	492.587	1	543.591	395.56	1.03123	560.569	259.51
Classe 300-3	1.00435	666.661	3739.28	1	663.775	4377.25	1.02788	682.285	1953.89
Classe 400-1	1.01513	626.721	4760.12	1.00466	620.255	3689.24	1	617.378	1384.7
Classe 400-2	1.0081	702.517	6203.82	1	696.871	6630.7	1.00972	703.644	2345.59
Classe 400-3	1	851.769	12362.4	1.00937	859.751	13796.7	1.02135	869.952	6654.26

Tabela 5.6: Resultados das heurísticas mais eficientes

de testes realizados pela heurística divide-espaço, assim, entendemos o porque do tempo reduzido, mas avaliamos que esta heurística é promissora pois encontra soluções melhores com mais facilidade.

Agora vamos analisar os resultados apresentados na Tabela 5.6. Nela, separamos as heurísticas que obtiveram relação igual a 1 para alguma classe de exemplares. Note que apesar da heurística primeiro 1-PRC/segundo m-PRC ter relação igual a 1 em 12 classes, ou seja, na maioria delas, os valores médios dos custos entre as três não diferem muito. E destacamos o tempo de processamento para a heurística divide-espaço a partir das classes com dimensões maiores que 150. Note que conforme a dimensão da classe aumenta, o tempo de processamento se torna importante, e isso é mais relevante nas heurísticas primeiro 1-PRC/segundo m-PRC e da varredura, pela suas próprias características de gerar soluções para testes. Lembrando que isso também ocorre para heurística gananciosa e pode ser notado na Tabela 5.5.

Concluímos a partir dos resultados apresentados nas três primeiras tabelas que, a heurística primeiro 1-PRC/segundo m-PRC é a mais eficiente e que a heurística divide-espaço consegue encontrar soluções melhores com menos testes efetuados, por isso se destaca com relação ao tempo de processamento.

Vamos observar os dados apresentados na Tabela 5.7. Como um dos focos do trabalho foi adaptar as heurísticas estudadas para que fosse possível manter um equilíbrio entre o número de vértices nas diferentes rotas, e por julgar muito importante a estratégia usada, para selecionar vértices em rotas iniciais, por cada heurística de construção de rotas para

Número de testes e diferenças N_k médios por classe

		Varredura		Gananciosa		1-PRC m-PRC		Divide-espaço	
Classe	r	Dif. N_k	testes	Dif. N_k	testes	Dif. N_k	testes	Dif. N_k	testes
Classe 100-1	2	1.2	4.1	1.1	4.1	0.7	3.4	1.4	10.8
Classe 100-2	2	1	5.2	0.8	5.2	0.9	5.1	1.5	11.2
Classe 100-3	2	0.4	8.4	0.4	8.4	0.4	8.4	2.3	10.2
Classe 150-1	2	1.4	5.3	1.4	5.3	0.9	4.1	1.7	10.6
Classe 150-2	2	1.4	5.7	1.4	5.7	0.9	5.3	2	12
Classe 150-3	2	0.3	8.3	0.3	8.3	0.3	8.3	2.4	10.4
Classe 200-1	3	1.2	8.5	1.8	8.5	1.3	7.2	2.7	10.4
Classe 200-2	3	1.5	9.9	1.1	9.9	0.6	9.6	2.9	11.5
Classe 200-3	3	1	17	0.9	17	1	17	3.4	10.2
Classe 300-1	3	1.7	9.6	2.1	9.6	1.3	7.5	2	10.6
Classe 300-2	3	1.9	10.7	1.5	10.7	1.2	9.9	2.7	10
Classe 300-3	3	1	17	1	17	1	17	2.9	11.8
Classe 400-1	4	3	15.1	2.6	15.1	1.7	13.5	3.1	12.5
Classe 400-2	4	1.3	19.4	1.6	19.4	0.9	18.9	3.6	12.2
Classe 400-3	4	0.4	33.4	0.4	33.4	0.4	33.4	5.1	12.8

Tabela 5.7: Diferenças N_k e o número de testes realizados por cada heurística

o m-PRC durante a primeira fase, que colabora para obtermos soluções finais com ou sem sobreposição de rotas. Apresentamos na Tabela 5.7 os valores médios do número de testes efetuados pelas heurísticas em cada classe de exemplares e as diferenças médias dos valores N_k obtidos nas soluções finais dos exemplares em cada classe. Pensando em apenas um exemplar e em uma heurística, o número de teste representa quantas soluções iniciais foram criadas e testadas pela heurística. Lembrando que as heurísticas, de acordo com uma estratégia, criam um determinado número de soluções, que são testadas e a melhor selecionada. A tabela apresenta os testes médios feitos entre os 20 exemplares, em cada classe por heurística. Com exceção das heurísticas da economia e da economia adaptada, que efetuam um e dois testes respectivamente por exemplar, a Tabela 5.7 apresenta os valores obtidos das outras quatro heurísticas. Observe que, para as heurísticas da varredura, gananciosa e primeiro 1-PRC/segundo m-PRC, conforme aumenta a dimensão dos exemplares, os testes também aumentam, mas esse aumento se mantém quase que iguais para as três heurísticas. Isso significa que, apesar de terem estratégias diferentes de obtenção de soluções iniciais, elas acabam avaliando o mesmo número de soluções nos exemplares até decidirem pela de menor custo. Algo diferente ocorre com a heurística divide-espaço. Nesta, fixamos que o número de testes de soluções para cada exemplar igual a 10. Lembrando que esse último valor representa quantas vezes dividimos o espaço de busca em m setores, ao perturbar um determinado ângulo. Porém os valores referentes ao número de testes médios da heurística divide-espaço apresentados na Tabela 5.7, varia entre 10 e 12 testes. Isto ocorre por que nos passos de seu algoritmo ocorre algumas repetições de teste, quando são detectadas inviabilidades por

Exemplar com dados reais

Heurística	custo (Km)	relação	tempo	r	Dif. N_k	veículos	testes	viável
Varredura	94.801	1	77831.08	6	2	5	37	sim
1-PRC m-PRC	99.486	1.04942	68459.89	6	3	5	38	sim
Divide-Espaço	92.196	-	25727.797	6	94	5	20	não
Varredura	104.800	1	63351.69	8	3	6	31	sim
1-PRC m-PRC	104.806	1.00006	51102.73	8	1	6	31	sim
Divide-Espaço	95.485	-	19611.42	8	81	6	20	não
Varredura	110.349	1	51985.00	8	4	7	27	sim
1-PRC m-PRC	111.234	1.00802	41490.68	8	2	7	26	sim
Divide-Espaço	99.384	-	18752.37	8	69	7	20	não

Tabela 5.8: Resultados obtidos com os dados reais

cobertura ou por diferença entre os valores N_k. Concluímos assim, que mesmo em exemplares com distribuição uniforme de vértices, a heurística divide-espaço detecta problemas de inviabilidade em algumas soluções testadas. Como o número de testes médios estão entre 10 e 12, podemos supor que o número de soluções inviáveis não é expressivo por exemplar testado, porém esperávamos que tal problema só iria ocorrer em exemplares sem distribuição uniforme, como o caso do exemplar gerado com os dados reais.

O que chamamos de Dif. N_k na Tabela 5.7, representa para cada exemplar, e cada heurística, a maior diferença entre o número de vértices nas diferentes rotas na solução final. Porém o valor apresentado na tabela, em cada linha, é a média encontrada entre os 20 exemplares, em cada classe, por cada heurística. O valor r, apresentado na segunda coluna da tabela, que varia de 2 a 4 conforme aumenta a dimensão dos exemplares, representa o dado de entrada que limita a maior diferença entre os valores N_k na solução final. Como temos a média dos valores Dif. N_k apresentados na tabela por classe, não podemos avaliar com certeza quantos exemplares entre os 20 de cada classe são viáveis, porém, devido as médias apresentadas, podemos sugerir que as heurísticas da varredura, gananciosa e primeiro 1-PRC/segundo m-PRC sempre encontram soluções finais viáveis para os exemplares. O mesmo não ocorre com a heurística divide-espaço, pois os valores Dif. N_k médios apresentados na tabela para esta heurística sugere duas coisas: o número de exemplares com soluções finais inviáveis encontras pela heurística divide-espaço é alto, principalmente nas Classes 100-3, 150-3, 200-3, 300-3 e 400-3; ou a maior diferença entre o número de vértices nas diferentes rotas dos poucos exemplares inviáveis é alta, tornando o valor médio Dif. N_k alto. Note também que os problemas de inviabilidade sempre ocorrem nas classes onde os exemplares têm $|T| = \left\lfloor \frac{V}{2} \right\rfloor$, ou seja, o tamanho do conjunto T, ou a sua distribuição acabam sendo determinantes para esta heurística encontrar ou não soluções finais viáveis.

Vamos observar os resultados apresentados nas Figuras 5.3 e 5.4. Elas ilustram os gráficos obtidos dos testes realizados com as classes de exemplares gerados aleatoriamente. Em cada

figura são apresentados três gráficos, que trás uma visualização melhor do que chamamos de relação entre os custos obtidos como solução de cada heurística. O eixo vertical representa a relação e no horizontal estão distribuídas as classes de exemplares. Cada heurística é representada por uma esfera colorida. Assim, a heurística que tem as melhores soluções terá a esfera da sua respectiva cor sobre a linha tracejada representando a relação igual a 1. O primeiro gráfico, indicado como Pior Caso, na parte superior da Figura 5.3, representa o pior valor encontrado da função objetivo entre os 20 exemplares de cada classe por heurística. O gráfico, que aparece no meio da figura, indicado por Melhor Caso, representa o melhor valor encontrado da função objetivo entre os 20 exemplares de cada classe por heurística. E o último gráfico, indicado por Média, representa os custos médios que foram apresentados nas Tabelas 5.4 e 5.5. Com essas explicações, fica fácil verificar na figura qual a heurística que se destacou em cada tipo de gráfico. Para o pior caso aparece sobre a linha tracejado dez esferas de cor verde, que representa o heurística primeiro 1-PRC/segundo m-PRC. Ou seja, ela obteve melhores resultados em dez classes de exemplares. Neste caso ainda, a heurística da varredura, representada pela esfera de cor branca venceu nas Classes 150-2 e 400-3. A divide-espaço venceu na Classes 300-1 e 400-2. E a heurística gananciosa obteve melhor resultado na Classe 100-1. No melhor caso a heurística divide-espaço se destacou sendo melhor em 9 classes, e primeiro 1-PRC/segundo m-PRC em outras 5 classes. Neste caso a heurística da varredura venceu na Classe 400-2. O último gráfico que representa a média só vem a colaborar com o que já foi descrito sobre a tabelas, pois conseguimos visualizar melhor ainda que a heurística primeiro 1-PRC/segundo m-PRC foi melhor em 12 das 15 classes testadas.

Como a diferença entre os resultados da função objetivo, de uma heurística para outra, não são tão expressivos, com exceção das duas piores, heurística da economia e economia adaptada. As esferas na Figura 5.3 aparecem sobrepostas, pois os valores das relações se diferem nas casas decimais. Sendo assim, para facilitar ainda mais a visualização dos resultados, somamos uma constante nas relações e construímos os gráficos apresentados na Figura 5.4. Nela podemos deduzir algumas coisas. Note que nos três gráficos sempre as piores heurísticas são as da economia e economia adaptada, sendo que a última, na grande maioria é melhor que a primeira. Portanto a estratégia usada nesta nova heurística demonstra melhores resultados. Sobrepondo as esferas no primeiro e no segundo gráfico, aparece a letra E, que informa um empate entre os resultados da função objetivo com as respectivas heurísticas que as esferas representam. Por exemplo, houve um empate nos custos obtidos pelas heurísticas da varredura, gananciosa e primeiro 1-PRC/segundo m-PRC na Classe 150-2 no gráfico que representa o melhor caso. Observando agora o gráfico que representa a média, notamos que, após a heurística da economia e economia adaptada, a heurística gananciosa é a pior entre elas, e fora a heurística mais eficiente primeiro 1-PRC/segundo m-PRC existe um equilíbrio nos resultados das heurísticas da varredura e divide-espaço.

Concluímos que os testes com exemplares gerados aleatoriamente revelam que as heurísticas primeiro 1-PRC/segundo m-PRC, da varredura e divide-espaço são as melhores. Damos destaque à primeiro 1-PRC/segundo m-PRC, que obteve custos mais reduzidos na maioria dos casos e à divide-espaço, que apesar dos problemas de inviabilidade obtém resultados

satisfatórios efetuando menor número de testes e consequentemente em menor tempo de processamento. Achamos que uma mistura das estratégias usadas pelas heurísticas da varredura e divide-espaço pode trazer bons resultados. A heurística gananciosa demonstrou que a estratégia de escolha de vértices iniciais, tomando sempre o vértice mais próximo do último escolhido é pouco eficiente, tanto no valor da função objetivo quanto no tempo de processamento.

Agora vamos comparar os resultados obtidos com os testes realizados no exemplar com dados reais. Decidimos testar este exemplar com apenas com as três heurísticas que se destacaram nos testes realizados com exemplares gerados aleatoriamente. Observamos na Tabela 5.8 os resultados obtidos com as heurísticas da varredura, primeiro 1-PRC/segundo m-PRC e divide-espaço. Nela, Os custos aparecem na primeira coluna e são dados em quilômetros. A segunda coluna fornece os valores das relações. A terceira coluna fornece os tempos de execução em segundos. Com essas três colunas podemos concluir que a heurística da varredura foi mais eficiente com relação aos custos neste problema, porém as respostas obtidas com a heurística primeiro 1-PRC/segundo m-PRC não diferem muito, sendo que no teste com 6 viaturas o custo é praticamente o mesmo. Efetuamos três tipos de testes variando o número de viaturas em 5, 6 e 7. E limitamos a maior diferença N_k em 6 e 8 vértices. Note que a heurística divide espaço realmente não obtém bons resultados em exemplares onde os a distribuição dos vértices não é uniforme, e principalmente pela posição não centralizada da base. Pois, apesar dos custos serem menores, as soluções finais são inviáveis. Note que, os valores Dif. N_k são muito altos. Isso sugere que esta heurística não tem uma estratégia interessante para processar exemplares com dados reais. Outro fato que confirma isso é o número de testes efetuados por esta heurística, pois apesar de ser fixado inicialmente em 10 testes, ela testou 20 soluções, oque indica que todas as soluções testadas eram inviáveis. Concluímos que, as heurísticas da varredura e primeiro 1-PRC/segundo m-PRC retornaram resultados satisfatórios e mantêm um bom equilíbrio entre o número de vértices na diferentes rotas. Isso é importante, pois em uma aplicação real, não permitimos que alguma equipe de agentes utilizando uma viatura fique sobrecarregada.

A Figura 5.5 ilustra como a heurística da Varredura define os circuitos a serem percorridos por 5 carros de patrulhas na cidade de Vinhedo. Fugindo do modelo m-PRC para rotas de patrulhamento proposto neste trabalho e após uma análise de aplicação das rotas no cotidiano da GCM, verificamos que não é preciso que todas as viaturas durante o turno de serviço passem pela base. Basta que apenas uma viatura tenha a base em seu trajeto, garantindo sua segurança. As equipes que estão atuando na segurança permanecem em comunicação constante via rádio, então conforme a necessidade é possível escolher de forma dinâmica qual viatura deva passar pela base durante um período do dia. A Figura 5.6 mostra um exemplo, tal que, apenas uma das viaturas tem a base em seu percurso de patrulhamento, e quando necessário troca-se essa viatura por outra.

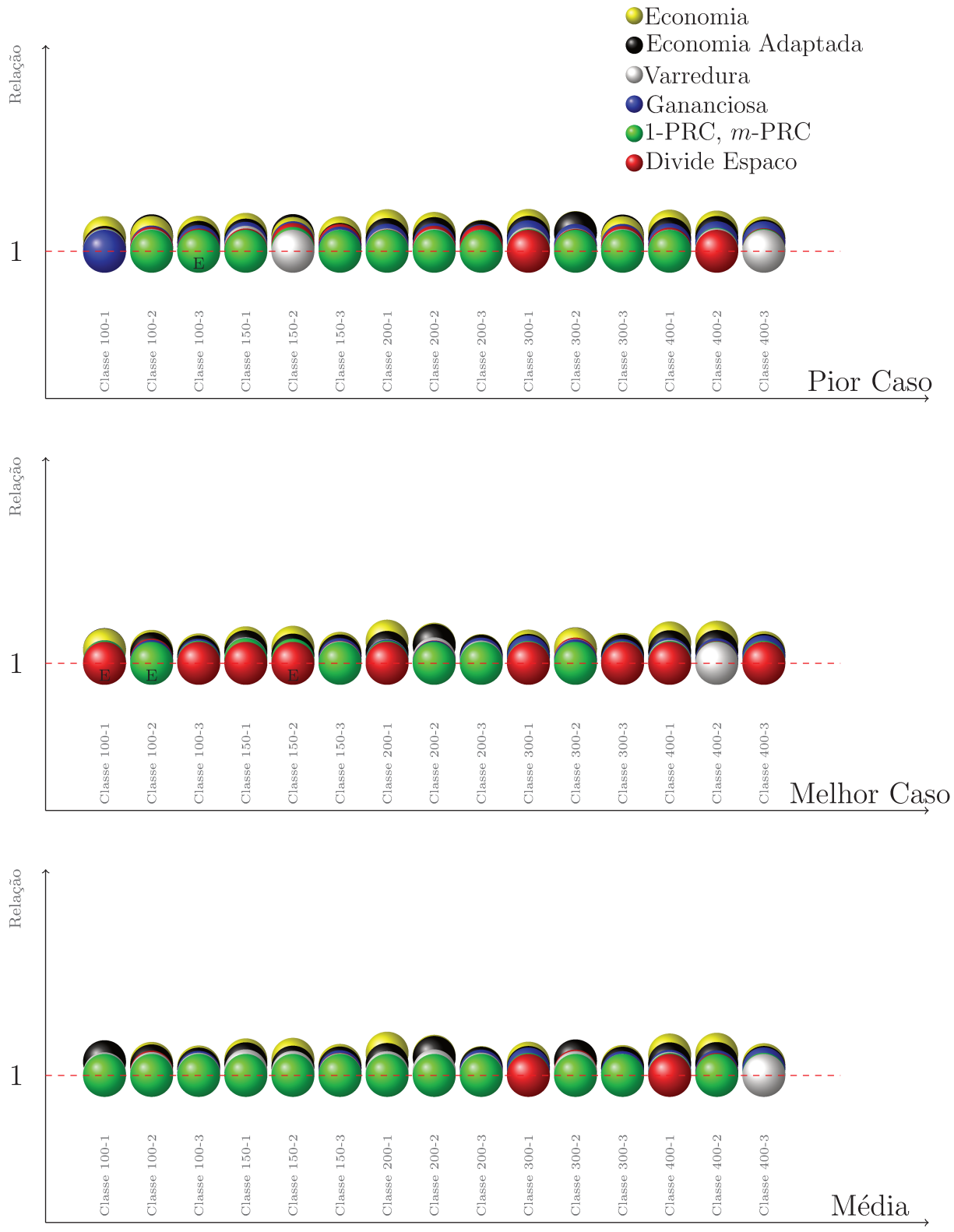


Figura 5.3: Relação: melhor de todas heurísticas por classe de exemplares

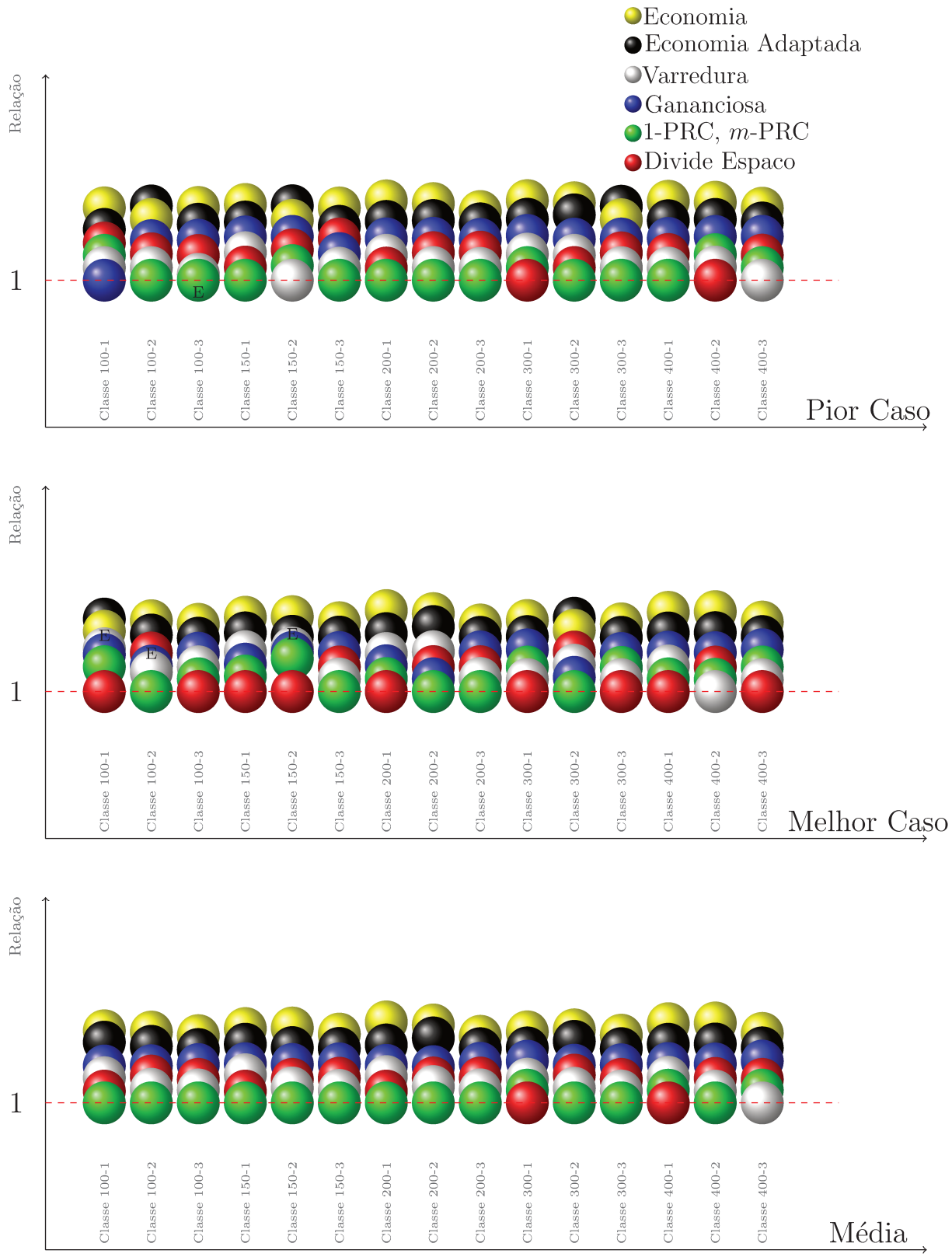


Figura 5.4: Relação: melhor de todas heurísticas por classe de exemplares

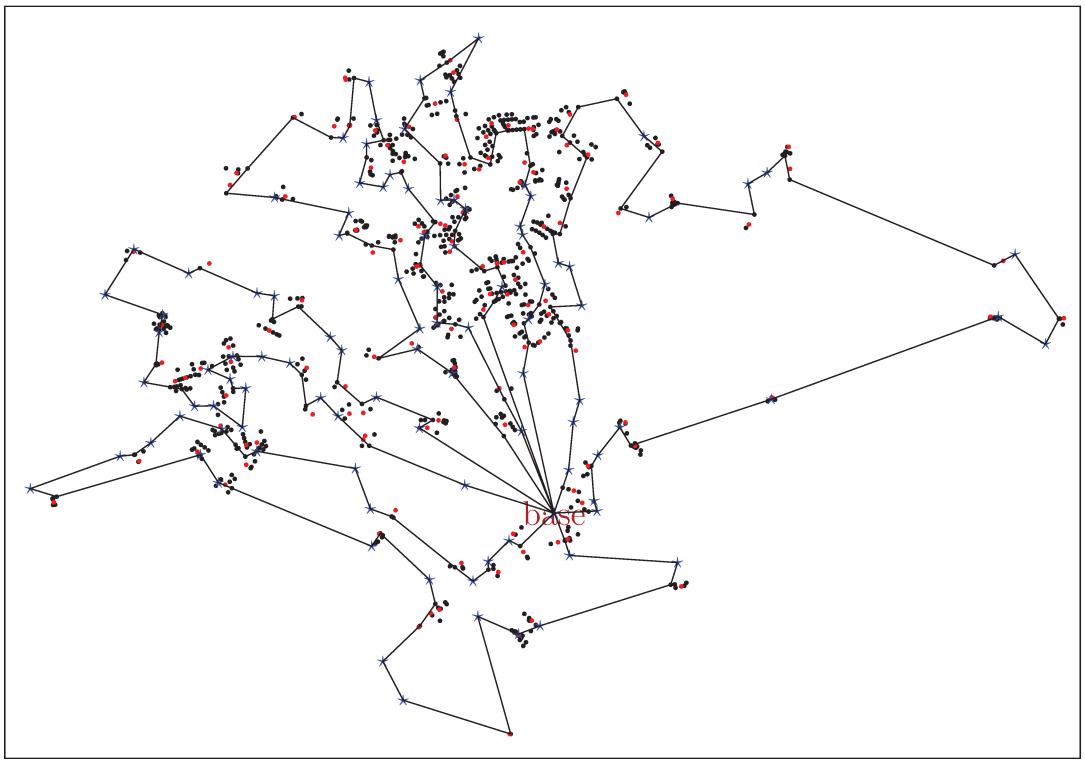


Figura 5.5: Rotas obtidas com a heurística da varredura para a cidade de Vinhedo

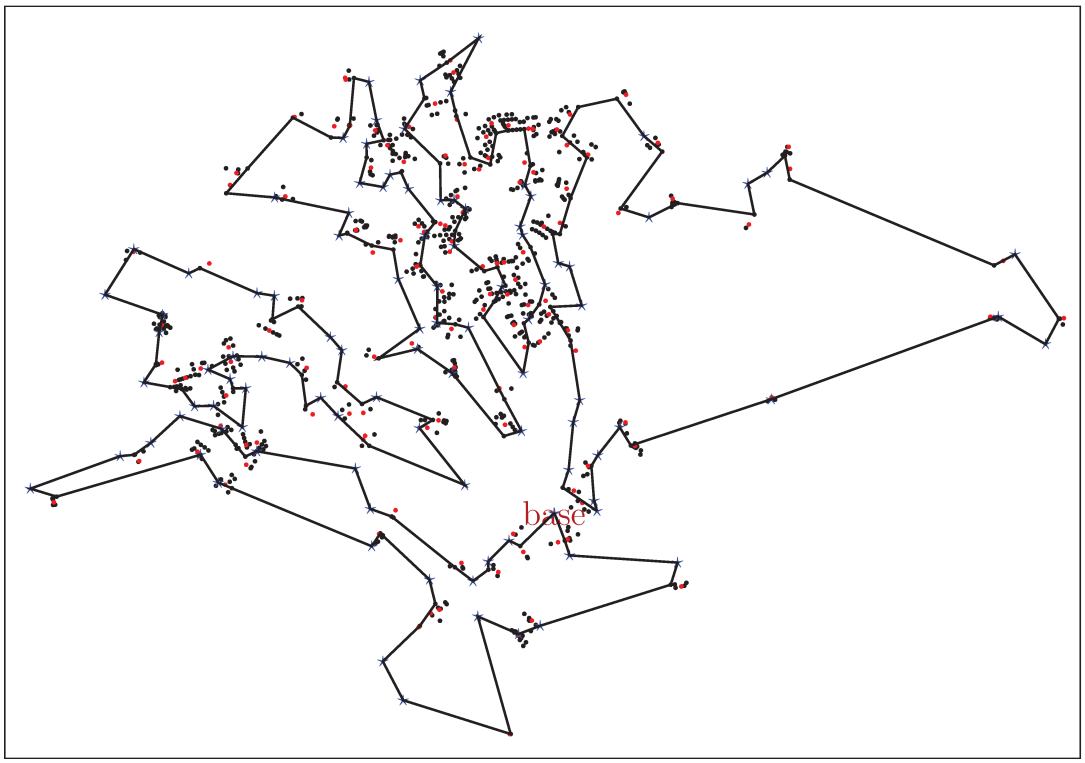


Figura 5.6: Alternativa de rotas para a cidade de Vinhedo: apenas um veículo visita a base

Capítulo 6

Conclusões e Trabalhos Futuros

6.1 Conclusões

No presente trabalho estudamos os aspectos do problema de construção de rotas para o patrulhamento urbano preventivo, no planejamento da segurança pública feita por forças policiais, por meio de uma adaptação do modelo de programação inteira m-PRC. Estudamos e implementamos em código MATLAB um conjunto de heurísticas, que combinadas, buscam uma boa solução para tal problema. Por adaptações das anteriores, novas heurísticas foram criadas, implementadas e comparadas.

Foi possível criar um modelo computacional, ainda dependendo de melhorias, que retorna para um administrador de segurança de uma região bons percursos para que seus respectivos carros de patrulha possam efetuar patrulhamento preventivo de forma ágil e eficiente, atendendo a um conjunto de características desta região. Obtivemos resultados satisfatórios, que atenderam o esperado, para o estudo feito na cidade de Vinhedo, SP, onde algumas rotas foram sugeridas para o serviço de segurança efetuados em patrimônios públicos sob a responsabilidade da Guarda Civil Municipal.

Vários testes foram feitos com as heurísticas apresentadas e podemos dar destaque a heurística Primeiro 1-PRC/Segundo m-PRC que retornou os melhores resultados na maioria dos testes. Por outro lado, a heurística Divide Espaço criada neste trabalho também se destacou.

Este estudo levou-nos a olharmos outros problemas relacionados com a área da segurança pública e privada. Percebemos que há muita pesquisa a ser feita neste campo de atuação. Sabemos que profissionais que atuam no setor de segurança pública e privada têm tarefas variadas e funções a cumprir em seus órgãos, visando um bom atendimento à população de modo geral. Em um estado, ou em um município, temos um número considerável de pessoas atuando nesta área, que necessitam de treinamento e utilizam diversos tipos de materiais, como veículos, fardamentos, armas, equipamentos de comunicação, etc. O gerenciamento eficiente desses recursos e das pessoas exige uma boa administração, para que os diversos serviços sejam executados de forma satisfatória, como por exemplo, ter equipes atuando em prevenção, outras em salvamentos e outras em treinamentos durante um mesmo período.

Entre os setores envolvidos com segurança estão as polícias e as guardas municipais.

Podemos listar alguns problemas que devem ser resolvidos pelos administradores e que, certamente, podem ser modelados e tratados em pesquisa operacional:

- Escala de serviço.
- Distribuição e alocação de homens e veículos em uma região.
- Alocação de sedes, postos, delegacias e batalhões.
- Distribuição de turnos de serviços, que ocorrem 24 horas por dia.
- Cobertura de regiões ou locais pré-fixados.
- Logística de materiais.

Os problemas acima estão sempre interligados e devem ser tratados levando-se em conta prioridades hierárquicas, índices diversos de ocorrências e atos ilegais, prioridades diversas no atendimento de ocorrência, etc.

Neste trabalho tratamos da distribuição de veículos em rotas de coberturas em locais pré-fixados. Podemos citar um exemplo de problema resolvido em pesquisa operacional, que faz parte desses listados acima. Ele mostra que pesquisas podem ser realizadas com grande sucesso nesta área de atuação, e quando são aplicadas podem facilitar muito o trabalho dos administradores e consequentemente melhoram a qualidade dos serviços prestados por um profissional de segurança, aumentando o seu nível de satisfação e a dos usuários. Trata-se de uma abordagem feita por Silva [27], da Universidade Federal do Paraná, em um problema de escala de funcionários aplicado ao serviço de plantão de soldados da aeronáutica (militares), através de um modelo de Programação Linear Inteira Binária. Tal modelo leva em consideração o número de militares disponíveis, as demandas diárias, o tipo de escala vigente, a necessidade de se ter militares de sobreaviso, a sua hierarquia e, principalmente, as sua preferência com relação aos dias de plantão, visando melhorar o seu nível de satisfação.

6.2 Trabalhos Futuros

Pretendemos continuar a presente pesquisa no doutorado tomando três aspectos diferentes para o problema m-PRC. Primeiramente estudar as características de uma variação deste problema que é apresentada por Motta, Ochi e Martinhon [22], onde uma nova modelagem matemática para o problema, que eles chamam de Problema de Recobrimento de Rotas, é proposta. Na forma original apenas os vértices do conjunto V podem fazer parte da solução, isto é, os vértices opcionais e os obrigatórios. Neste caso não só os vértices de V como também os vértices de W (vértices que devem ser cobertos) poderão fazer parte da rota solução do problema. Sendo assim, $v_i \in W$ também poderá ser utilizado para cobrir a si próprio e a outros vértices $v_j \in W$, $i \neq j$. Também é feito um estudo de regras para a redução do grafo de instância associado, uma nova formulação de Programação Linear

Inteira introduzindo variáveis de fluxo, assim como novas heurísticas: uma para a construção de uma solução factível e outra para a busca local. A fim de que sejam mais eficientes que os convencionais, são compartilhadas as características de duas metaheurísticas que têm se destacado na solução de problemas de otimização combinatória pelos resultados obtidos: GRASP *(Greedy Randomized Adaptive Search Procedures)* e VNS *(Variable Neigborhood Search)*.

Pretendemos ainda sugerir uma nova heurística, que combina o processo de varredura com a divisão do espaço em setores. Com isso, melhorar a heurística Divide-Espaço que não consegue controlar os valores N_k para exemplares que não têm distribuição uniforme e a base não está centralizado no espaço geográfico. Iremos comparar seus resultados com os melhores resultados obtidos até o momento. Ainda neste linha, pretendemos amadurecer a idéia de construir um conjunto de rotas onde apenas uma viatura visite a base, estudar a possibilidade de resolver um problema com mais de uma base e tornar as implementações interativas, para que seja, utilizadas por administradores de segurança interessados.

Por fim, e levando em conta o que é apresentado por M. Gendreau, G. Laporte e F. Semet [14], pretendemos encontrar desigualdades válidas para o m-PRC a fim de propor um algoritmo exato.

Referências Bibliográficas

[1] K. Altinkemer, B. Gavish, *Parallel savings based heuristics for the delivery problem*, Operations Research, Vol. 39, No. 3. (May – Jun., 1991), pp. 456–469.

[2] E. M. Arkin, M. M. Halldórsson, R. Hassin, *Approximating the tree and tour covers of a graph*. Information Processing Letters 1993; 47: 275–282.

[3] E. M. Arkin, R. Hassin, *Approximation algorithms for the geometric covering salesman problem*. Discrete Appl. Mathematics 1994; 55: 197–218.

[4] E. Balas, A. Ho, *Set covering algorithms using cutting planes, heuristics, and subgradient optimization: A computational study*, Mathematical Programming 12 (1980) 37–60.

[5] M. L. Brown, L. U. S. Fintor, *Screening mammography services with mobile units: results from the National Survey of Mammography Facilities*, Radiology 1995; 195: 529–32.

[6] G. Clarke, J. W. Wright, *Scheduling of vehicles from a central depot to a number of delivery points*, Operations Research, Vol. 12, No. 4. (Jul. – Aug., 1964), pp. 568–581.

[7] G. A. Croes, *A method for solving traveling-salesman problems*, Operations Research 6 (1958) 791–812.

[8] J. R. Current, H. Pirkul, E. Rolland, *Efficient algorithms for solving the shortest covering path problem*, Transpotation Science 1994; 28: 317–27.

[9] J. R. Current, D. A. Schilling, *The median tour and maximal covering problems*, European Journal of Operational Research 1994; 73: 114–126.

[10] Disponível em: `http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsplib.html`.

[11] M. L. Fisher, *Vehicle routing*, In: M.O. Ball, T.L. Magnanti, C.L. Monma, G.L. Nemhauser, editors. Network routing, Handbooks in Operations Research and Management Science, Amsterdam: North-Holland, 1995: 1–33.

[12] F. Foord, *Gambia: evaluation of the mobile health care service in West Kiang district*, World Health Statistics Quarterly 1995; 48: 18–22.

[13] M. Gendreau, A. Hertz, G. Laporte, *New insertion and postoptimization procedures for the traveling salesman problem*, Operations Research, Vol. 40, No. 6. (Nov. – Dec., 1992), pp. 1086–1094.

[14] M. Gendreau, G. Laporte, F. Semet, *The covering tour problem*, Operations Research, Vol. 45, No. 4. (Jul. – Aug., 1997), pp. 568–576.

[15] B. E. Gillet, L. R. Miller, *A heuristic algorithm for the vehicle dispatch problem*, Operations Research, Vol. 22, No. 2. (Mar. – Apr., 1974), pp. 340–349.

[16] M. Hachicha, M. J. Hodgson, G. Laporte, F. Semet, *Heuristics for the multi-vehicle covering tour problem*, Computers and Operations Research 27 (2000) 29–42.

[17] D. Hanselman, B. Littlefield, *Matlab 6: curso completo*. Prentice Hall 2003.

[18] M. J. Hodgson, G. Laporte, F. Semet, *A covering tour model for planning mobile health care facilities in Suhum district, Ghana*, Journal of Regional Science 1998; 38: 621–38.

[19] M. Labbé, G. Laporte, *Maximizing user convenience and postal service efficiency in post box location*, Belgian Journal of Operations Research, Statistics and Computer Science 1986; 26: 21–35.

[20] G. Laporte, *The vehicle routing problem: an overview of exact and approximate algorithms*, European Journal of Operational Research 1992; 59: 345–58.

[21] J. K. Lenstra, A. H. G. Rinnooy Kan, *Some simple applications of the travelling salesman problem*, Operational Research Quarterly (1970–1977), Vol. 26, Nº 4, Part 1.(Nov., 1975), pp. 717–733.

[22] L. C. S. Motta, L. S. Ochi, C. A. Martinhon, *O problema de recobrimento de rotas: formulação, testes de redução e soluções aproximadas via GRASP*, $XXXII$ Simpósio Brasileiro de Pesquisa Operacional Viçosa, 18–20/10/2000.

[23] G. L. Nemhauser, L. A. Wolsey, *Integer and combinatorial optimization*, Wiley-Interscience Series in Discrete Mathematics and Optimization, 1999.

[24] J. R. Oppong, MJ. Hodgson, *Spatial accessibility to health care facilities in Suhum district, Ghana*, The Professional Geographer 1994; 46: 199–209.

[25] V. N. Puchkin, *Heurística: a ciência do pensamento criador*, Zahar editores 1976 2ºed.

[26] C. S. ReVelle, G. Laporte, *New directions in plant location*, Studies in Locational Analysis 1993; 5: 31–58.

[27] T. C. L. Silva, *Determinação de escalas de plantão para militares considerando preferências e hierarquia*, Pesquisa Operacional, v. 24, n. 3, p. 373–391, Setembro a Dezembro de 2004.

[28] J. C. Simms, *Fixed and mobile facilities in dairy practice*, Veterinary Clinics of North America - Food Animal Practice 1989; 5: 591–601.

[29] W. Swaddiwudhipong, C. Chaovakiratipong, P. Nguntra, P. Lerdlukanavonge, S. Koonchote, *Effect of a mobile unit on changes in knowledge and use of cervical cancer screening among rural Thai womem*, International Journal of Epidemiology 1995; 24: 493–8.

[30] L. A. Wolsey, *Integer programming*, Wiley-Interscience Series in Discrete Mathematics and Optimization, 1998.