

UNIVERSIDADE FEDERAL DO PARANÁ

GUILHERME VINICYUS BATISTA

**PROPOSTA DE UM MODELO MATEMÁTICO PARA O PROBLEMA DE
ROTEAMENTO EM ARCOS CAPACITADO E PERIÓDICO**

CURITIBA

2014

GUILHERME VINICYUS BATISTA

**PROPOSTA DE UM MODELO MATEMÁTICO PARA O PROBLEMA DE
ROTEAMENTO EM ARCOS CAPACITADO E PERIÓDICO**

Dissertação apresentada ao Programa de Pós-Graduação em Métodos Numéricos em Engenharia, Área de Concentração em Programação Matemática, Departamentos de Construção Civil e de Matemática, Setores de Tecnologia e de Ciências Exatas, Universidade Federal do Paraná, como parte das exigências para a obtenção do título de Mestre em Métodos Numéricos em Engenharia.

Orientador: Prof. Dr. Cassius Tadeu Scarpin

CURITIBA

2014

B333p

Batista, Guilherme Vinicyus

Proposta de um modelo matemático para o problema de roteamento em arcos capacitado e periódico / Guilherme Vinicyus Batista. – Curitiba, 2014. 93f. : il. color. ; 30 cm.

Dissertação (mestrado) - Universidade Federal do Paraná, Setor de Tecnologia, Programa de Pós-graduação em Métodos Numéricos em Engenharia, 2014.

Orientador: Cassius Tadeu Scarpin.

Bibliografia: p. 79-83.

1. Modelos matemáticos. 2. Algoritmos. 3. Otimização combinatória. I. Universidade Federal do Paraná. II. Scarpin, Cassius Tadeu. III. Título.


CDD: 511.8

TERMO DE APROVAÇÃO


GUILHERME VINYCIUS BATISTA

PROPOSTA DE UM MODELO MATEMÁTICO PARA O PROBLEMA DE ROTEAMENTO EM ARCOS CAPACITADO E PERIÓDICO

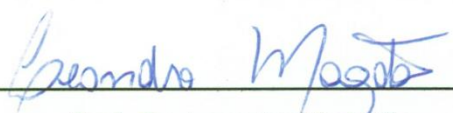
Dissertação aprovada como requisito parcial para obtenção do grau de mestre no Programa de Pós-Graduação em Métodos Numéricos em Engenharia da Universidade Federal do Paraná, pela seguinte banca examinadora:



Prof. Dr. Cassius Tadeu Scarpin.
(Orientador) Membro do PPGMNE/UFPR



Prof.ª Dr.ª Neida Maria Patias Volpi.
Membro do PPGMNE/UFPR



Prof. Dr. Leandro Magatão.
Membro da UTFPR



Prof. Dr. Lauro Cesar Galvão
Membro da UTFPR.

Curitiba, 25 de abril 2014.

Dedico este trabalho a Ana Júlia e
a toda minha família.

AGRADECIMENTOS

Gostaria de agradecer primeiramente a Deus, por minha vida, pelos desafios e pelas pessoas maravilhosas que cruzaram meu caminho ao longo desses anos e influenciaram no que sou hoje.

Ao meu orientador Dr. Cassius Tadeu Scarpin, por vezes meu professor e também colega de turma durante a graduação. Sem dúvida um homem admirável que tenho como exemplo, tem uma didática incrível que torna as coisas mais simples de serem compreendidas, sempre disponível e pronto para ajudar. Além de tudo aceitou a árdua tarefa de me orientar, a qual sou muito grato pela oportunidade de tê-lo como mentor, por ter compartilhado seus conhecimentos e por todo apoio e incentivo ao longo desses anos.

A todos os membros do GTA O, Grupo de Tecnologia Aplicada a Otimização, além de dividirem a estrutura física, tenho muito orgulho de ter participado desse grupo com tantas pessoas focadas em desenvolver pesquisas que façam a diferença. Em especial aos professores Arinei e Gustavo sempre presentes e os quais eu devo muito do meu conhecimento que possibilitou o desenvolvimento desse trabalho.

A minha namorada, Ana Júlia, por todo amor, incentivo, compreensão, ajuda e principalmente por estar caminhando lado a lado comigo nessa jornada sem me deixar desanimar.

A minha família que é meu porto seguro, e os alicerces que me sustentam para enfrentar qualquer desafio.

Aos professores e colegas do PPGMNE por todo ensinamento e troca de conhecimentos durante esse período de muito aprendizado. Ao secretário Jair pela sua disponibilidade e auxílio.

À CAPES REUNI, pelo apoio financeiro.

*“A mente que se abre a uma nova ideia
jamais voltará ao seu tamanho original.”*

Albert Einstein

RESUMO

O Problema de Roteamento em Arcos Capacitado e Periódico (*PCARP*) é uma das variantes dos Problemas de Roteamento em Arcos em que cada arco tem uma demanda geralmente associada a uma frequência ao longo de um horizonte de tempo bem definido. Para solucioná-lo é necessário fazer um planejamento de modo que todas as demandas sejam atendidas da melhor maneira possível sem exceder a capacidade dos carros que farão os atendimentos. É um problema recente e pouco explorado na literatura, porém com grande aplicabilidade em contextos reais como coleta de lixo, atividades de monitoramento, inspeção e manutenção. O objetivo geral é apresentar uma modelagem matemática capaz de resolver um problema que tem a essência de todo *PCARP*, mas apresenta características diferenciadas das já propostas na literatura por causa de fatos como: não necessitar voltar a um depósito ao final de um dia e poder atrasar algum atendimento se necessário. Para isso foi feito um levantamento bibliográfico dos problemas que mais se assemelham e analisadas as diferenças entre os mesmos. Finalmente, o modelo proposto foi implementado, alcançando bons resultados com poucos atrasos em diferentes estruturas de grafos.

Palavras-chave: Problema de Roteamento em Arcos Capacitado e Periódico. Problemas de Roteamento em Arcos. Modelagem Matemática.

ABSTRACT

The Periodic Capacitated Arc Routing Problem (*PCARP*) is a class of Arc Routing Problems in that each arc has a demand usually associated to a frequency over a well defined time horizon. To solve it is necessary to do a planning so that it covers all the demands in the best way possible without exceeding the vehicles capacity at service. It is a recent problem and not intensively studied at the literature, though with a large applicability in various real life applications such as waste collection, network monitoring, inspection and maintenance. The main aim is to present a mathematical formulation capable of solving a problem that has the essence of every *PCARP*, but that presents some distinguished characteristics from those already proposed at the literature. The distinguished characteristics are that the vehicle does not need to come back to the deposit at the end of the day and that the service can be delayed if necessary. For this, it was done a bibliographical research of the nearest problems and analyzed the differences among them. Finally, the proposed model was implemented, reaching good results without delay at several graph structures.

Keywords: Periodic Capacitated Arc Routing Problem. Arc Routing Problems. Mathematical Formulation.

LISTA DE FIGURAS

Figura 1 – Representação de uma malha ferroviária.....	18
Figura 2 – Representação de uma malha ferroviária com pontos definidos.	19
Figura 3 – Atividades básicas da Pesquisa Operacional.....	23
Figura 4 – Transformação do problema das 7 pontes em um grafo por Euler	27
Figura 5 – Representação em um grafo do problema das pontes de <i>Königsberg</i>	27
Figura 6 – Uma simples classificação do <i>PCARP</i>	40
Figura 7 – Sub-horizontes de tempo associado as classes de arcos	48
Figura 8 – Exemplo de problema gerado - arquivo gvb1	58
Figura 9 – Ilustração do resultado do problema gvb1.....	61
Figura 10 – Proposta de problema para adaptação ao modelo proposto por Monroy	64
Figura 11 – Sub-horizontes de tempo e classes criadas para adaptação ao modelo de Monroy	65
Figura 12 – Representação de um problema real	68
Figura 13 – Representação da simplificação do problema real	69
Figura 14 – Rotas dos carros 1 e 2 entre os períodos 1 e 10.....	70
Figura 15 – Rotas dos carros 1 e 2 entre os períodos 11 e 20.....	71
Figura 16 – Rotas dos carros 1 e 2 entre os períodos 21 e 30.....	72
Figura 17 – Rotas dos carros 1 e 2 entre os períodos 31 e 40.....	73
Figura 18 – Rotas dos carros 1 e 2 entre os períodos 41 e 50.....	74
Figura 19 – Rotas dos carros 1 e 2 entre os períodos 51 e 60.....	75

LISTA DE QUADROS

QUADRO 1 – COMPARATIVO ENTRE OS MODELOS MATEMÁTICOS APRESENTADOS	56
--	----

LISTA DE TABELAS

TABELA 1 – RELAÇÃO DOS 10 PROBLEMAS GERADOS, PROPRIEDADES E RESULTADOS	60
TABELA 2 – RELAÇÃO DOS 23 PROBLEMAS ADAPTADOS, PROPRIEDADES E RESULTADOS	62
TABELA 3 – RESULTADO DA ADAPTAÇÃO DE PROBLEMA AO MODELO DE MONROY (2013)	66
TABELA 4 – RESULTADO OBTIDO COM O MODELO PROPOSTO	67

LISTA DE SIGLAS

ARP – Arc Routing Problem

CARP – Capacitated Arc Routing Problem

CPP – Chinese Postman Problem

DCPP – Directed CPP

GRASP – Greedy Randomized Adaptive Search Procedure

ILS – Iterated Local Search

MARP – Multiday Arc Routing Problem

MBCPP – Maximum Benefit Chinese Postman Problem

MCPP – Mixed CPP

OCARP – Open Capacitated Arc Routing Problem

PCARP – Periodic Capacitated Arc Routing Problem

PCC – Problema do Carteiro Chinês

PCCV – Problema do Carteiro Chinês com Vento

PCR – Problemas do Carteiro Rural

PIB – Programação Inteira Binária

PL – Programação Linear

PLI – Programação Linear Inteira

PO – Pesquisa Operacional

PRPV – Problema de Roteamento Periódico de Veículos

UCPP – Undirected CPP

WPP – Windy Postman Problem

SUMÁRIO

1	INTRODUÇÃO	13
1.1	OBJETIVOS	14
1.1.1	Objetivo Geral.....	14
1.1.2	Objetivos Específicos	14
1.2	JUSTIFICATIVA.....	15
1.3	LIMITAÇÕES.....	15
1.4	ESTRUTURA DO TRABALHO.....	16
2	DESCRIÇÃO E ANÁLISE DO PROBLEMA	17
3	REVISÃO DA LITERATURA	21
3.1	FUNDAMENTAÇÃO TEÓRICA	21
3.1.1	Pesquisa Operacional.....	21
3.1.1.1	Programação Linear	23
3.1.1.2	Programação Linear Inteira.....	25
3.1.2	Grafos.....	27
3.2	TRABALHOS CORRELATOS	29
3.2.1	Problema de Roteamento em Arcos.....	29
3.2.1.1	Problema do Carteiro Chinês.....	30
3.2.1.2	Problema do Carteiro Rural	32
3.2.2	Problema de Roteamento em Arcos Capacitados	33
3.2.3	Problema de Roteamento em Arcos Capacitados e Periódicos	38
4	METODOLOGIA.....	43
4.1	MODELOS JÁ PROPOSTOS.....	43
4.1.1	Modelo de Chu, Labadi e Prins (2004)	43
4.1.2	Modelo de Chu, Labadi e Prins (2005)	45
4.1.3	Modelo de Monroy, Amaya e Langevin (2013)	47
4.2	PROPOSTA DE FORMULAÇÃO MATEMÁTICA	49
4.3	SEMELHANÇAS E DIFERENÇAS ENTRE MODELOS	53
5	IMPLEMENTAÇÃO E RESULTADOS	58
5.1	IMPLEMENTAÇÃO.....	58
5.2	RESULTADOS PARA VALIDAÇÃO DO MODELO	60

5.3	COMPARAÇÃO COM UMA ADAPTAÇÃO DO MODELO.....	64
5.4	APLICAÇÃO A UM PROBLEMA REAL.....	68
6	CONCLUSÕES	77
	REFERÊNCIAS.....	79
	APÊNDICES	84

1 INTRODUÇÃO

Muitos pesquisadores têm resolvido problemas logísticos como: localização de facilidades, gerenciamento de armazéns e roteamento de veículos. Esses problemas têm sido solucionados envolvendo tanto decisões estratégicas como táticas e operacionais, o que pode trazer um grande ganho para as empresas envolvidas (HASHEMI DOULABI; SEIFI, 2013).

Um processo que ocorre nas empresas que trabalham com ferrovias é a manutenção preditiva (inspeção dos trilhos) e preventiva. Neste caso, toda malha ferroviária deve ser avaliada para que sejam prevenidos acidentes, como o tombamento de trens. Para isso, existem sofisticados equipamentos capazes de fazer medições específicas para prevenir acidentes. Tais equipamentos se deslocam ao longo da malha e devem estar em locais estratégicos para fazer seu trabalho. A malha é subdividida em trechos, onde cada trecho tem pré-determinado um intervalo de tempo em que devem ocorrer inspeções (periodicidade). Então, torna-se importante definir uma rota para esses equipamentos de forma que as demandas de cada trecho sejam atendidas da melhor maneira possível, fazendo um planejamento de médio e longo prazo.

Problemas de roteamento de veículos procuram resolver situações semelhantes, mais especificamente os problemas de roteamento em arcos. Nesse tipo de problema o principal objetivo é atender determinadas arestas que normalmente são relacionadas às ruas ou vias que fazem parte de uma rede. A coleta de lixo, entrega de correspondências, o despejar de sal em vias com neve, manutenção em redes, entrega e coleta de produtos são exemplos de problemas que podem ser resolvidos dessa forma (CORBERÁN; PRINS, 2010).

Problemas de Roteamento e *Scheduling* de Veículos em que se deve fazer um planejamento a longo prazo também têm grande aplicabilidade a problemas reais e vem trazendo soluções excelentes. São problemas de sucesso nas áreas de Otimização Combinatória, Pesquisa Operacional e Computação Aplicada (TORTELLY JR.; OCHI, 2003). As técnicas de Pesquisa Operacional são amplamente utilizadas com sucesso para solucionar diversos problemas reais através de modelagem matemática e algoritmos (LOPES; RODRIGUES; STEINER, 2013).

Entre os problemas mais estudados envolvendo roteamento em arcos está a gestão de resíduos sólidos. Ghiani *et al.* (2014) afirma que as técnicas de Pesquisa Operacional têm permitido alcançar significantes economias nos processos, assim como ajudar na recuperação dos resíduos com aplicações em partições de território, seleção de dias de coleta, composição da frota e roteamento e *scheduling* dos veículos coletores.

O problema da literatura que engloba planejamento em múltiplos períodos para atendimento em arcos com carros capacitados é chamado de Problema de Roteamento em Arcos Capacitado e Periódico, normalmente aplicados em problemas que envolvem monitoramento, inspeção e manutenção (MONROY; AMAYA; LANGEVIN, 2013).

1.1 OBJETIVOS

O objetivo do trabalho é propor uma solução para o Problema de Roteamento em Arcos Capacitado e Periódico (*PCARP*). Para isso tem-se o objetivo geral e os específicos a seguir.

1.1.1 Objetivo Geral

Propor um modelo matemático em Programação Linear Binária capaz de resolver o Problema de Roteamento em Arcos Capacitado e Periódico com múltiplos veículos, procurando atender, da melhor forma possível, as periodicidades de cada arco.

1.1.2 Objetivos Específicos

Os objetivos específicos são os seguintes:

- Apresentar problemas semelhantes na literatura;
- Propor uma modelagem matemática para o *PCARP*;
- Estabelecer semelhanças e diferenças entre os modelos apresentados na literatura e nesse trabalho sobre o *PCARP*;

- Validar o modelo proposto através de instâncias geradas e de instâncias modificadas da literatura.

1.2 JUSTIFICATIVA

A modelagem matemática que será proposta no desenvolvimento desse trabalho tem grande aplicabilidade em problemas de manutenção que ocorrem periodicamente.

O papel da manutenção é essencial na garantia da qualidade e da produtividade empresarial. Uma manutenção adequada pode garantir que um processo não perca sua capacidade e a política de manutenção é fator determinante do sucesso da produtividade do processo (MARCORIN E LIMA, 2003).

Esse tipo de atividade tem como objetivo manter a segurança e a viabilidade de uma rede de transportes com uma rápida detecção de problemas. É necessário um planejamento cuidadoso que leve em conta a natureza do problema, a localização, o tempo e a frequência dos serviços (MARZOLF; TRÉPANIER; LANGEVIN, 2006).

1.3 LIMITAÇÕES

O Problema de Roteamento em Arcos Capacitado e Periódico é muito recente, logo a literatura não é tão vasta até o momento e muitas das possíveis variações do problema ainda não foram estudadas. Lacomme *et al.* (2002a) são os responsáveis por dar início às pesquisas sobre esse problema, resolvendo-o através de um algoritmo genético. Já a primeira modelagem matemática proposta ocorreu em 2004 no trabalho de Chu, Labadi, e Prins (2004).

Segundo Ghiani *et al.* (2005), diferentemente dos problemas de roteamento em nós, que são os problemas de roteamento mais estudados, a literatura do problema de roteamento em arcos é mais pobre e desorganizada. Este problema também é relatado em Ghiani *et al.* (2014) quando trata-se do problema de gerenciamento de lixo, que é uma das principais aplicações do problema de roteamento em arcos.

Muitos dos problemas de roteamento em arcos e suas variações podem receber nomes diferentes mas tem suas propriedades iguais. Por exemplo, o Problema do Carteiro Rural não deixa de ser um Problema de Roteamento em Arcos Capacitado, o que dificulta a pesquisa, pois sempre há uma relação direta entre os problemas, que apesar de terem nomes diferentes, são trabalhados da mesma forma e aplicados aos mesmos contextos.

1.4 ESTRUTURA DO TRABALHO

A estrutura desse trabalho é composta por 6 capítulos. Neste primeiro capítulo se encontram os objetivos específicos e o objetivo geral bem como uma pequena introdução ao problema que está contextualizado no capítulo dois.

No terceiro capítulo é feita uma fundamentação teórica apresentando algumas das técnicas de Pesquisa Operacional capazes de resolver o problema aqui proposto. Também apresentam-se alguns conceitos que serão utilizados no decorrer desta dissertação, da mesma forma como alguns trabalhos correlatos que tratam desde o Problema de Roteamento em Arcos e suas variantes até chegar ao Problema de Roteamento em Arcos Capacitado e Periódico.

A metodologia está descrita no capítulo quatro, no qual também são descritos os modelos matemáticos que já foram propostos na literatura para o *PCARP* e onde é sugerido um novo modelo. Todos os modelos são analisados e as semelhanças e diferenças entre os modelos são expostas.

Implementação, geração de instâncias, a aplicação a um problema real e os resultados estão listados no quinto capítulo, além de uma pequena comparação matemática com uma adaptação ao modelo matemático que foi proposto mais recentemente em 2013. Finalmente, as conclusões do problema encontram-se no capítulo seis.

2 DESCRIÇÃO E ANÁLISE DO PROBLEMA

A aplicação real do problema que será aqui tratado está associada aos conceitos de manutenção preditiva e preventiva. Além disso, os problemas de manutenção podem estar ligados a intervalos de tempo para garantir a eficiência de um equipamento. A seguir procura-se contextualizar o problema ao qual se dedica este estudo.

O serviço de manutenção é de extrema importância para uma empresa. É um investimento que proporciona uma redução não somente nos custos de reparo de equipamentos como no custo de paradas do processo (MARCORIN; LIMA, 2003). Marcorin e Lima (2003) definem manutenção preventiva como um conjunto de ações que visam prevenir a quebra e está associada à intervenções periódicas geralmente programadas segundo uma frequência. A manutenção preventiva acarreta muitas vezes em intervenções desnecessárias e troca de peças, enquanto as preditivas são definidas da seguinte forma:

A manutenção preditiva caracteriza-se pela medição e análise de variáveis da máquina que possam prognosticar uma eventual falha. Com isso, a equipe de manutenção pode se programar para intervenção e aquisição de peças (custo de manutenção), reduzindo gastos com estoque e evitando paradas desnecessárias da linha de produção (custos de indisponibilidade).

Por ser uma manutenção de acompanhamento, a preditiva exige uma mão-de-obra mais qualificada para o trabalho e alguns aparelhos ou instrumentos de medição. Seu aparente alto custo é plenamente recompensado por seus resultados, situando-se mais próximo do ponto ótimo da relação custo-benefício em equipamentos cuja parada traz grandes prejuízos ao processo e em que o custo do estoque de equipamento/peça também é elevado. (MARCORIN; LIMA, 2003, p. 39).

A malha ferroviária brasileira tem grande extensão e os equipamentos disponíveis para garantir o perfeito estado dos trilhos devem ser colocados em locais estratégicos para fazer seu trabalho de manutenção preditiva. Um caso bem específico é um trem de pequeno porte que mede o vão entre trilhos. Esse trem, que

deve percorrer toda a malha, é um equipamento que necessita de um alto investimento.

Com a possibilidade de locomoção desse pequeno trem sobre os trilhos, faz-se necessário determinar diariamente onde ele deve executar seu trabalho. Porém, existem algumas restrições quanto a sua capacidade diária e os trechos que deve percorrer.

Alguns trechos com alto fluxo de trens necessitam de medições com uma frequência maior, ou seja, é necessário executar esse serviço em intervalos de tempo mais curtos do que em outros. Essa frequência, necessária em cada arco, também é chamada de periodicidade.

Um exemplo que pode ilustrar esse problema é a Figura 1, a qual representa uma malha ferroviária na qual os trechos em azul necessitam ser medidos uma vez a cada 16 dias e os trechos em vermelho uma vez a cada 24 dias.

Uma particularidade desse problema é que o equipamento que se desloca ao longo de toda a malha não necessita voltar a um ponto específico ao final do dia (também conhecido como depósito). O deslocamento do equipamento, em geral, é lento e, durante o intervalo de um dia, o trem inicia seus trabalhos em um local e termina em outro, no qual deve se iniciar o trabalho no próximo dia.

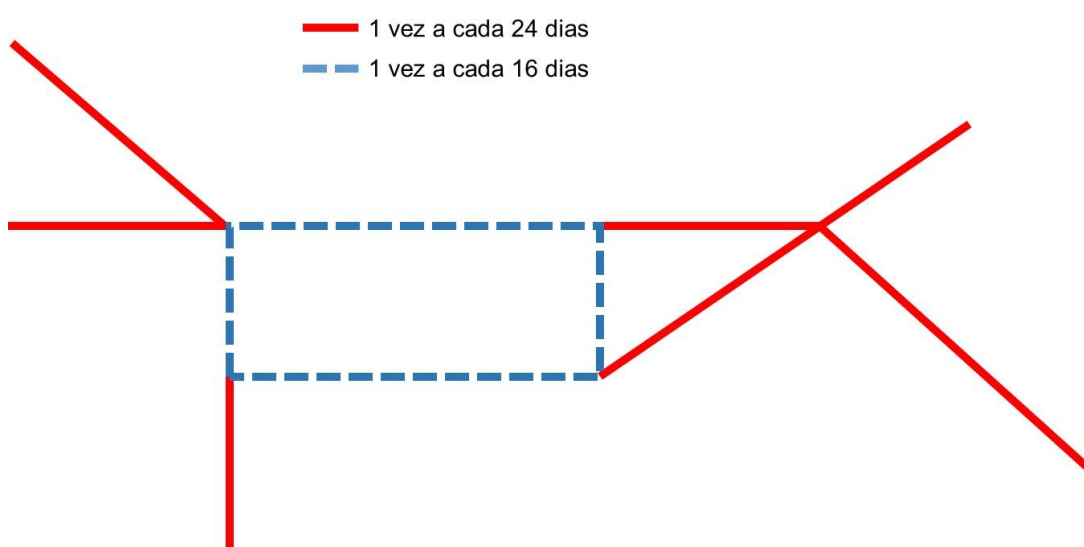


Figura 1 – Representação de uma malha ferroviária.

A capacidade do trem é limitada à quilometragem que ele consegue percorrer em cada dia e em cada trecho, sendo que em alguns trechos os deslocamentos são mais lentos e em outros podem ser mais rápidos. Assim, faz-se necessária a

distribuição de pontos onde o operador e o equipamento podem pernoitar, dado que o trabalho não é executado 24 horas por dia.

A Figura 2 mostra pontos distribuídos ao longo da malha, os quais são definidos previamente pela estrutura física instalada e são onde os carros podem pernoitar.

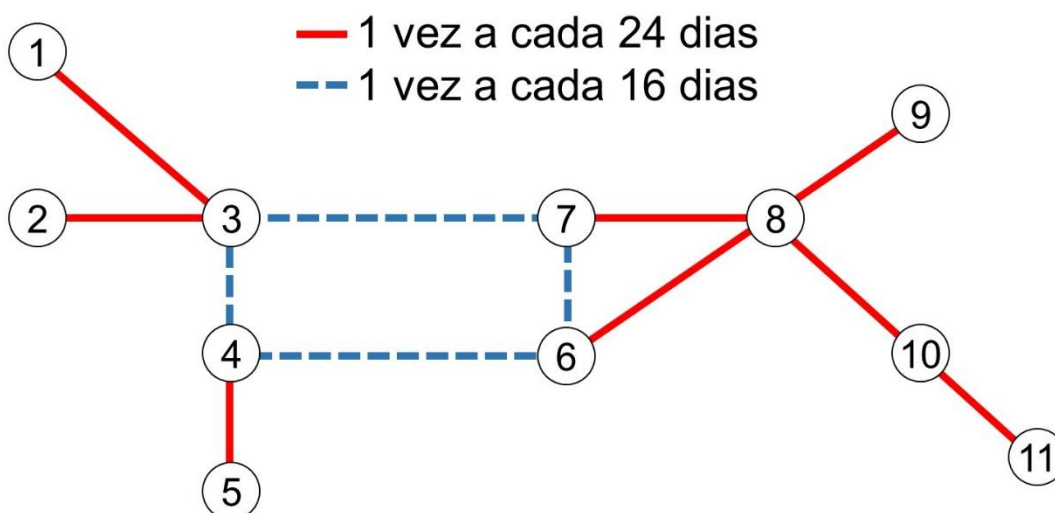


Figura 2 – Representação de uma malha ferroviária com pontos definidos.

A Figura 2 mostra que um carro pode se deslocar em um dia saindo do ponto três até o ponto sete e ao fazer esse movimento não pode fazer nenhum outro, dado que uma ligação entre dois pontos representa uma possibilidade de trajeto diário. Esses pontos são definidos previamente, assim como as periodicidades de cada trecho. Pode-se notar que as arestas 3-4, 3-7, 4-6, 6-7 tem uma periodicidade de 16 dias, o que significa que eles têm uma demanda de que a cada 16 dias o carro de manutenção os atravesse fazendo medições. Já os demais arcos tem uma periodicidade de 24 dias, o que significa que necessitam ser avaliados preditivamente uma vez durante esses 24 dias.

O horizonte de tempo é definido baseado no arco de maior periodicidade, ou menor frequência. No caso do exemplo da Figura 2 temos um horizonte de tempo de 24 dias. Ou seja, dentro do período de 24 dias os arcos tem que ter sua demanda atendida. Isso implica que os arcos com periodicidade de 16 dias devem ter duas passagens durante esses 24 dias, de forma que o intervalo entre uma passagem e outra seja de, no máximo 15 dias, ou ainda, que em 16 dias o trecho seja percorrido ao menos uma vez.

Como é um problema cíclico, ao final dos 24 dias o carro deve estar no mesmo ponto onde iniciou seu trajeto no dia 1, o que é equivale a dizer que o que deve ocorrer no dia 25 é exatamente o que ocorre no primeiro dia. Arcos com menor periodicidade devem levar em conta a sequência dos dias após o final dos primeiros 24 dias.

Um exemplo de situação a ser evitada seria a seguinte: dado um carro que passou no arco 3-7 (periodicidade de 16 dias) nos dias 3 e 9 atendeu a demanda dos primeiros 24 dias, porém como o dia 3 é equivalente ao dia 27 no próximo ciclo, tem-se que no intervalo entre o segundo atendimento no dia 9 e o terceiro no dia 27 existe diferença de 18 dias, os quais não permitiriam que a demanda do arco fosse atendida.

3 REVISÃO DA LITERATURA

3.1 FUNDAMENTAÇÃO TEÓRICA

O objetivo dessa seção é apresentar possíveis ferramentas capazes de resolver problemas reais e que podem ser uma alternativa para a resolução do problema descrito.

3.1.1 Pesquisa Operacional

As técnicas da Pesquisa Operacional (PO) possuem grande aplicabilidade à problemas nos quais é necessário conduzir e coordenar operações (atividades). As aplicações são encontradas em áreas como manufatura, transportes, construção, telecomunicações, militar, serviços públicos e muitos outros. Dado um determinado problema, a Pesquisa Operacional tem por objetivo identificar uma melhor solução para o mesmo (HILLIER; LIEBERMAN, 2010).

O periódico inglês *Operational Research Quartely*, em 1967, propôs uma definição que explica a finalidade da Pesquisa Operacional, a qual consiste em desenvolver métodos científicos de sistemas complexos para prever e comparar decisões alternativas, ou seja, auxiliar na definição de políticas e determinação de ações de forma científica. De forma simplificada, é um enfoque científico sobre a tomada de decisões (ARENALES ET AL.,2007).

Para Taha (2007) apesar da grande variedade de aplicações, são necessárias algumas diretrizes gerais para a implementação das técnicas PO, definindo as seguintes fases:

1. Definição do problema
2. Construção do modelo
3. Solução do modelo
4. Validação do modelo
5. Implementação do modelo

A definição do problema nada mais é que definir o escopo do mesmo. Nesta fase é necessário descrever as alternativas de decisão, determinar o objetivo do

estudo e identificar quais as leis que regem o problema, ou as limitações que o mesmo apresenta (TAHA, 2007).

O processo de construção de um modelo matemático é definido por Arenales *et al.* (2007, p.3), da seguinte forma:

A partir da observação de fenômenos, processos ou sistemas, que podem ser físicos, químicos, biológicos, econômicos, buscam-se leis que os regem. Essas leis, se passíveis de serem descritas por relações matemáticas, dão origem aos modelos matemáticos. O termo modelo neste texto é usado como objeto abstrato, que procura imitar as principais características de um objeto real para fins de representar o objeto real. (ARENALES *et al.*, 2007, p.3)

Para Puccini e Pizzolato (1990), um modelo é essencial nos estudos de PO, é uma idealização, ou uma visão simplificada da realidade. Hillier e Lieberman (2010) assumem que o modelo matemático deve representar a essência do problema.

Construir um modelo é traduzir a definição do problema em uma formulação matemática, ou simplesmente modelar o problema, etapa que consiste em definir as variáveis e como se relacionam matematicamente para descreve-lo. Vale à pena ressaltar que quando o modelo é muito complexo, pode ser inviável obter uma solução analítica, então se pode simplificar o modelo ou utilizar técnicas não exatas para resolução do problema.

A solução do modelo é a resolução do problema por meio da utilização de algoritmos bem definidos. De maneira complementar, pode ser feita uma análise de sensibilidade que mostra o comportamento da solução quando o modelo tem alguns parâmetros modificados.

O modelo é validado quando, de maneira genérica, apresenta uma solução que faz sentido, ou seja, o modelo apresenta um resultado que atende as expectativas de quem propôs o problema, caso contrário o modelo precisa ser revisado.

Implementar a solução consiste em traduzir os resultados obtidos em instruções operacionais (TAHA, 2007). Hillier e Lieberman (2010) lembram que esta é uma fase crítica, pois é o momento em que os frutos do trabalho são colhidos. É necessário estar muito atento para que eventuais falhas que possam ocorrer sejam corrigidas. A fase de implementação é onde são tomadas as ações necessárias ao analisar a resposta obtida através da modelagem do problema para resolvê-lo.

Arenales *et al.* (2007) salientam que modelos matemáticos auxiliam na tomada de decisões, mas não substitui o responsável por tomá-las.

Puccini e Pizzolato (1990) propuseram um fluxograma simplificado para solucionar problemas por meio de modelagem matemática ou métodos tradicionais representado na Figura 3. Métodos tradicionais costumam ser a forma como um problema é resolvido rotineiramente, ou ainda, por exemplo, um algoritmo que apresenta um resultado após várias tentativas e erros.

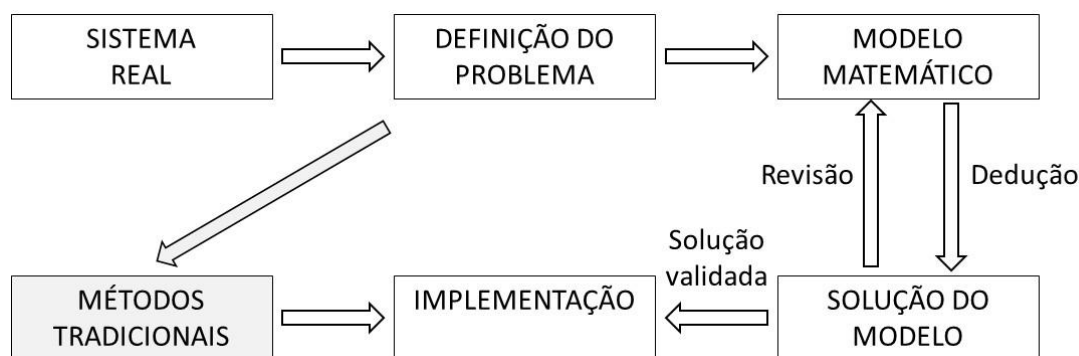


Figura 3 – Atividades básicas da Pesquisa Operacional
 FONTE: Puccini e Pizzolato (1990)

Para resolver os problemas de PO foram criados muitos algoritmos que encontram uma solução de forma eficiente. Em geral, os algoritmos para encontrar uma solução demandam muitos cálculos, e com a evolução computacional nos últimos anos, se tornou cada vez mais viável resolver problemas dessa forma. No mercado existem diversos softwares que resolvem esses problemas, como o LINGO, EXCEL SOLVER, e o CPLEX, sendo utilizados largamente em problemas extensos e desafiadores (HILLIER; LIEBERMAN, 2010).

A programação matemática ou otimização é o ramo da matemática que trabalha com métodos para otimizar (maximizar ou minimizar) uma função objetivo que está sujeita a restrições que envolvem n variáveis, nelas podemos encontrar a Programação Linear e a Programação Linear Inteira (MURTY, 1983).

3.1.1.1 Programação Linear

A Programação Linear (PL) é muito utilizada nos estudos envolvendo PO. Essa ferramenta utiliza um modelo matemático que representa o problema em questão com, necessariamente, funções lineares. Um problema comumente resolvido

assim é o de “... alocar da melhor forma possível (isto é, ótima) recursos limitados para atividades que competem entre si.” (HILLIER; LIEBERMAN, 2010, p.25)

Para Taha (2007), um modelo de PL, como qualquer outro modelo de Pesquisa Operacional, tem três componentes básicos: variáveis de decisão, objetivo e restrições. As variáveis de decisão são determinadas por quem trabalha com o problema, por exemplo, a quantidade a produzir de um determinado item. O objetivo pode ser maximizar ou minimizar uma função, por exemplo, maximizar o lucro na fabricação de um determinado produto. As restrições são fatores que limitam um resultado, como recursos escassos para uma produção.

Puccini e Pizzolato (1990) atribuem três principais características a problema de PL:

- Existe uma função objetivo que deve ser otimizada (maximizando ou minimizando), sendo que essa função contém as variáveis de decisão e é uma função linear.
- Existe outro grupo de equações e/ou inequações lineares que relacionam as relações de interdependência das variáveis de decisão. Essas relações também são conhecidas como restrições.
- Não podem existir variáveis de decisão negativas, isso é, devem ser positivas ou nulas para ser obtida uma solução através do método Simplex. Ressalta-se aqui que a exigência de não negatividade é somente para a resolução do problema pelo método Simplex, podendo ser feitas adaptações quando é necessário que uma variável assumira um valor negativo.

Uma gama muito grande de aplicações pode ser representada na forma de Programação Linear, e existe um “procedimento de solução extraordinariamente eficiente, chamado método Simplex” (HILLIER; LIEBERMAN, 2010, p.26) que está presente em muitos softwares e resolve esses tipos de problemas.

Murty (1983), Puccini e Pizzolato (1990), Hillier e Lieberman (2010) atribuem quatro hipóteses ou propriedades básicas ao problema de PL são elas: proporcionalidade, aditividade, divisibilidade e determinismo.

Proporcionalidade – é uma hipótese que se refere tanto à função objetivo como às restrições. Ela afirma que cada variável de decisão é diretamente proporcional ao coeficiente da variável, por exemplo, para produzir determinado

produto é necessária uma quantidade de recurso por unidade, não importando a quantidade produzida, assim a utilização de economia de escala não tem relevância.

Aditividade – Taha (2007, p. 7) define muito bem essa hipótese da seguinte forma “essa propriedade requer que a contribuição total de todas as variáveis da função objetivo e das restrições seja a soma direta das contribuições individuais de cada variável.”

Divisibilidade – essa propriedade permite que as variáveis de decisão assumam quaisquer valores, isto é, até mesmo valores não inteiros desde que as restrições sejam satisfeitas.

Determinismo – essa propriedade diz que os coeficientes das funções que formam o modelo são determinísticos, ou seja, são constantes conhecidas.

3.1.1.2 Programação Linear Inteira

A Programação Linear permite uma série de aplicações, porém a hipótese de divisibilidade impede a utilização da técnica para muitos outros problemas. Muitas vezes é necessário que as variáveis de decisão assumam apenas valores inteiros. Como, por exemplo, para alocar uma quantidade exata de pessoas ou máquinas para uma determinada atividade, é inviável alocar meia pessoa, o certo seria ou alocar uma pessoa ou nenhuma. Quando ocorre esse tipo de problema, no qual variáveis não podem assumir valores fracionários, estamos trabalhando com um problema de Programação Linear Inteira (PLI) (HILLIER; LIEBERMAN, 2010).

A PLI é também chamada por muitos autores como: Programação Inteira, Programação Discreta ou Programação Inteira Pura. Puccini e Pizzolato (1990, p. 207) definem claramente a área da seguinte maneira: “Programação inteira é a denominação dada a problemas de Programação Linear em que as variáveis só podem assumir valores inteiros”.

Alguns problemas ainda podem trabalhar com os dois tipos de variáveis, inteiras e reais, sendo esse tipo de problema denominado Programação (linear) Inteira Mista (PMI). (ARENALES et al, 2007).

Outra área de aplicação é aquela em que se encontram problemas que envolvem decisões, as quais só podem ter como resposta sim ou não. Com isso, podemos restringir as variáveis de decisão a apenas dois valores 0 e 1, de forma que

0 é uma resposta negativa e 1 uma resposta positiva. Imaginando uma variável x_j de decisão representando a j -ésima decisão, ela seria representada da seguinte forma:

$$x_j = \begin{cases} 1, & \text{Se a decisão for sim} \\ 0, & \text{Se a decisão for não} \end{cases}$$

Quando uma variável só pode assumir valores 0 ou 1, ela é chamada de variável binária, e problemas que envolvem esse tipo de variáveis são ditos problemas de Programação Inteira Binária (PIB). (HILLIER; LIEBERMAN, 2010)

Apesar de parecer que a PLI é mais simples que a PL, a realidade é oposta. Um programa pode resolver rapidamente um problema de PL com milhares de variáveis e restrições, já um problema de programação inteira com meia centena de variáveis demoraria muito mais (PUCCINI; PIZZOLATO, 1990).

A PLI permite ter um número finito de soluções. Porém, números finitos podem ser muito grandes. Considerando um PIB com n variáveis, há 2^n soluções possíveis, por exemplo, um problema com 30 variáveis de decisão tem mais de 1 bilhão de soluções a serem consideradas. (HILLIER; LIEBERMAN, 2010)

Para Arenales *et al.* (2007), os métodos mais bem sucedidos para resolver esses tipos de problema são baseados no método de *branch-and-bound*, e de planos de corte. Esses dois métodos trabalham com um conceito de relaxação linear para obter uma solução inicial. A relaxação linear nada mais é que tratar um PLI como PL, assim o resultado obtido com a PL torna-se um limite superior para um novo problema e assim sucessivamente. O melhor resultado obtido com a PLI nunca será melhor que o obtido com PL, porém pode até ser o mesmo.

Outra alternativa para encontrar soluções de problemas PLI, PIB são a utilização de heurísticas e meta-heurísticas que buscam encontrar pontos ótimos (locais ou global) de um problema. São alternativas às soluções puramente matemáticas (LOPES; RODRIGUES; STEINER, 2013). Entre as meta-heurísticas encontram-se: Algoritmos Genéticos, *Simulated Annealing*, Colônia de Formigas, Busca Tabu, *Iterated Local Search* (ILS), *Greedy Randomized Adaptive Search Procedure* (GRASP), entre outras.

Muitos problemas reais são representados através de grafos, inclusive o objeto de estudo desse trabalho. Então a próxima seção tem por objetivo definir o que são grafos e contar um pouco de sua história.

3.1.2 Grafos

Assad e Golden (1995) atribuem a origem da teoria dos grafos ao clássico problema das sete pontes que atravessam o Rio *Pregel* em *Königsberg* publicado por Euler em 1736. O problema consistia em definir uma única rota partindo de um ponto em que todas as pontes deveriam ser atravessadas uma, e apenas uma, vez e ao final retornasse ao ponto inicial.

Euler fez uma representação do problema onde cada segmento de reta (aresta) era a representação de uma ponte e cada ilha ou margem do rio era representado por um ponto, essa figura passou a ser denominada grafo (ANDRADE,1980). A Figura 4 representa o problema das pontes de *Königsberg* e sua transformação em um grafo e a Figura 5 o grafo final que representa o problema:

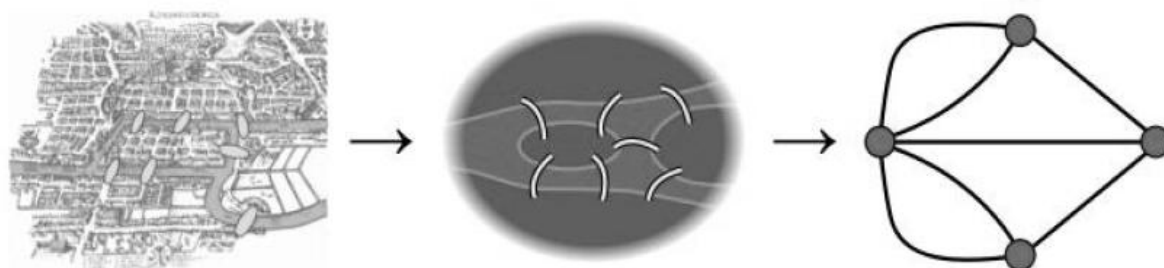


Figura 4 – Transformação do problema das 7 pontes em um grafo por Euler
Fonte: Negreiros *et al.* (2009)

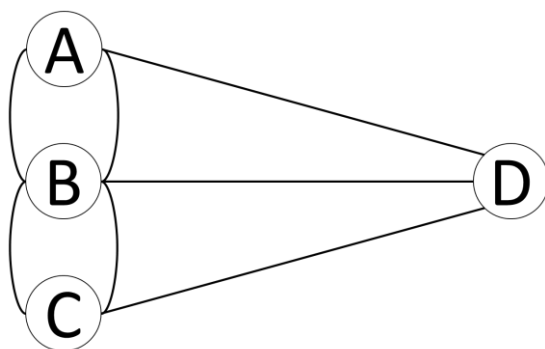


Figura 5 – Representação em um grafo do problema das pontes de *Königsberg*

O problema das Pontes de *Königsberg* foi o primeiro Problema de Roteamento em Arcos estudado (NEGREIROS *et al.*, 2009). Euler mostrou que não havia solução para resolução de tal problema, pois era impossível traçar um caminho sem percorrer mais de uma vez a mesma aresta (ANDRADE,1980).

Segundo Andrade (1980), Euler denominou o grau dos vértices de um grafo, definindo como o número de arestas que tocam um vértice. Por exemplo na Figura 5 o ponto da B é de quinto grau enquanto os pontos A, C e D são de terceiro grau.

Euler também apresentou o teorema que impõe as condições para que possa ocorrer um caminho em um grafo atravessando todas as aresta sem repeti-las. Este teorema diz que um grafo conexo e não orientado possui um ciclo, agora chamado de euleriano, se e somente se, não possuir vértices de grau ímpar. Grafos com essa característica recebem o nome de grafos eulerianos (BOAVENTURA NETTO, 2001).

Para Evans e Minieka (1992), um grafo consiste de basicamente dois elementos, pontos e linhas que ligam esses pontos. Esses pontos também são conhecidos como vértices e essas linhas, que são as ligações entre dois pontos, são as arestas.

Um grafo $G = (X, E)$ é uma estrutura matemática composta por dois conjuntos finitos, onde X é o conjunto dos vértices (também chamados de pontos ou nós) e E o conjunto de arestas ou arcos. (BOAVENTURA NETTO, 2001; GROSS; YELLEN, 2006)

De maneira geral, quando temos um problema com um grafo $G = (X, E)$ com n pontos e m arestas tem-se o conjunto $X = \{x_1, x_2, \dots, x_n\}$ formado por esses n pontos, e o conjunto $E = \{e_1, e_2, \dots, e_m\}$ formado por m arcos ou arestas.

Um arco ou aresta e é uma ligação entre dois pontos, e é normalmente associado a um par de vértices sendo denominado x_{ij} quando formado pelos pontos (x_i, x_j) ou simplesmente $[i, j]$. O que diferencia o arco de aresta, para alguns autores, é se a ordenação dos pontos que os compõe importa ou não.

As arestas podem ser atravessadas em ambos os sentidos, não importando o ponto inicial e o ponto final, de maneira geral, $x_{ij} = x_{ji}$. Já nos arcos, a ordem dos nós que o formam é de extrema importância pois o inverso pode não existir, o que significa que quando o arco for tratado apenas um sentido é válido, ou ainda podem estar associados a diferentes custos dependendo do sentido da travessia.

Quando o conjunto E é formado por pares de vértices em que a ordem não importa (arestas) tem-se um grafo não direcionado, caso contrário é considerado um grafo direcionado composto por arcos (EVANS; MINIEKA, 1992).

Gross e Yellen (2006) apresentam uma série de modelos e aplicações de grafos como em problemas de designação, adjacência geográfica, rodovias entre estados, química molecular, ecossistemas e problemas de *scheduling*.

3.2 TRABALHOS CORRELATOS

Problemas de roteamento podem ser divididos em duas classes: roteamento em nós e roteamento em arcos. O roteamento em nós consiste em encontrar uma ou mais rotas onde um veículo visita alguns ou todos os nós de um grafo e o roteamento em arcos tem por objetivo determinar uma rota que passa por todos ou alguns arcos ou arestas de um grafo (MONROY; AMAYA; LANGEVIN, 2013).

Esses problemas se distinguem basicamente na modelagem em virtude da natureza dos serviços, passagens em arcos ou nós. Segundo Assad e Golden (1995) o roteamento em arcos ocorre quando existe uma alta densidade de pontos em uma linha, sendo convertido de um problema de roteamento em nós para arcos, ou simplesmente quando o interesse é atravessar arcos.

Konowalenko (2012) cita fatores listados na literatura que ampliam a diversidade dos problemas de roteamento como: número de domicílios, tempo para servir um determinado nó ou arco, tamanho da frota, natureza da demanda, localização da demanda, restrições na capacidade dos veículos, natureza do grafo, entre outros.

Corberán e Prins (2010), trazem em seu trabalho um apanhado dos resultados encontrados na literatura durante a década que precede sua publicação sobre os Problemas de Roteamento em Arcos que tiveram origem no Problema das Pontes de *Königsberg* e que tentam encontrar uma ou mais rotas que cubram todas (ou parcialmente) as ligações (arcos ou arestas) de um grafo, satisfazendo algumas restrições e com o menor custo possível. Alguns desses trabalhos serão expostos no decorrer deste capítulo.

3.2.1 Problema de Roteamento em Arcos

O principal objetivo do problema de roteamento em arcos, em inglês *Arc Routing Problems (ARP)*, é determinar o menor custo para atravessar um conjunto de arcos de um grafo, com ou sem restrições (EISELT *et al.* 1995b).

Os *ARPs* são similares aos problemas de roteamento de veículos, exceto pelo fato de que as demandas são distribuídas pelos arcos e não pelos nós. A maior parte dos problemas de roteamento em arcos pode ser convertido em problema de roteamento em nós e vice-versa. Entretanto estas conversões não se tornam viáveis pois, em geral, ocorre um acréscimo no número de variáveis. Por exemplo, um problema de Roteamento em Arcos Capacitado com m arcos, pode ser convertido em um Problema de Roteamento de Veículos com $3m + 1$ nós. Já para converter um problema de roteamento em nós para arcos é necessário associar cada nó com demanda à um arco (ASSAD; GOLDEN, 1995).

Entre os problemas que se encaixam nessa classificação temos o problema das pontes de *Königsberg*, o Problema do Carteiro Chinês e o Problema do Carteiro Rural, incluindo suas variantes.

3.2.1.1 Problema do Carteiro Chinês

O Problema do Carteiro Chinês (PCC), do inglês *Chinese Postman Problem (CPP)*, é um problema em que é necessário encontrar uma rota com menor custo possível de deslocamento para um carteiro, o qual deve partir de um depósito e entregar as cartas ao longo de um conjunto de ruas e voltar ao depósito (TAUBE, 1992).

Wang e Wen (2002) dividem o Problema do Carteiro Chinês em três subproblemas:

- PCC não direcionado (*UCPP – Undirected CPP*)
- PCC direcionado (*DCPP – Directed CPP*)
- PCC misto (*MCCP – Mixed CPP*)

O *UCPP* é um caso especial do *DCPP* quando temos satisfeita uma condição de simetria e, ainda, o *MCCP* é uma combinação do *UCPP* com o *DCPP*. Os autores Wang e Wen (2002) ainda propõe a seguinte modelagem matemática para o *DCPP*, partindo de um grafo $G = (V, A)$:

$$\min z = \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (1)$$

Sujeito às seguintes restrições:

$$\sum_{(i,j) \in A} x_{ij} = \sum_{(j,i) \in A} x_{ji} \quad \forall i \in V \quad (2)$$

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

$$x_{ij} \in Z^+ \quad (4)$$

A variável x_{ij} representa a quantidade de vezes que ocorre o movimento do ponto i para j , fato garantido pela restrição (4). A equação (1) é a função objetivo do modelo que consiste em minimizar os custos C_{ij} associados a cada aresta x_{ij} . O fluxo contínuo está restrito na equação (2) e a garantia de que todo arco será atravessado na inequação (3). A modelagem proposta para o *DCPP* ainda pode ser transformada em uma modelagem de *UCPP* substituindo a restrição (3) pela (5):

$$x_{ij} + x_{ji} \geq 1 \quad \forall (i,j) \in A \quad (5)$$

Existem algumas variações do problema quando se trata do carteiro chinês. Tem-se o Problema do Carteiro Chinês com Vento (PCCV), conhecido como *The Windy Postman Problem (WPP)*, que são problemas em que existem custos diferentes dependendo do sentido que o arco está sendo atravessado. Estes também são associados a grafos não direcionados, direcionados, e mistos. O *WPP* e o *MCP* são os problemas mais estudados na primeira década dos anos 2000 (CORBERÁN; PRINS, 2010).

Um bom exemplo para o PCCV é quando tem-se a subida de uma ladeira em um sentido e a descida se o arco é atravessado no sentido oposto, ou até mesmo quando é feito um deslocamento a favor do vento (KONOWALENKO, 2012).

Além disso existem outras variantes do PCC como o Problema do Carteiro Chinês Hierárquico, onde os arcos são agrupados em *clusters* (classes). Esses *clusters* são classificados em uma relação de precedência que especifica em que ordem eles devem ser atendidas partindo de um depósito e retornando ao mesmo. É um problema aplicado para encontrar caminhos ótimos nos problemas de corte com tocha oxicorte e na remoção de neve quando tem-se ruas com maior prioridade que outras (GHIANI; IMPROTA, 2000).

Outra variante é o Problema do Carteiro Chinês com Benefício Máximo, *The Maximum Benefit Chinese Postman Problem (MBCPP)*. Neste há um benefício ao se atravessar um arco uma vez, quando ele é atravessado mais de uma vez não se torna vantajoso, ou ainda, ao aumentar o número de travessias no mesmo arco os benefícios diminuem (MALANDRAKI; DASKIN, 1993).

No trabalho de Archetti *et al.* (2013) *The Team Orienteering Arc Routing Problem*, utiliza-se a mesma ideia do MBCPP, porém além de existir a lista dos arcos requeridos (clientes conhecidos a priori), existe uma lista de outros clientes em potencial que podem ou não ser atendidos através de uma frota que é delimitada por um tempo máximo de viagem.

Em Corberán e Prins (2010) apenas um subconjunto de arcos do grafo deve ser atendido e cada arco ainda pode ser associado a um custo ao servi-lo e a um custo ao atravessá-lo, sendo que existe uma lista decrescente de lucros ao servir o arco.

Outro caso especial é o Problema do Carteiro Chinês com Janelas de Tempo, onde são adicionadas restrições que interferem em o quão cedo e o quão tarde um arco pode ser atendido.

A partir desse problema básico chamado Problema do Carteiro Chinês as pesquisas cresceram motivadas por aplicações em problemas reais nas últimas décadas (CORBERÁN; MOTA; SANCHIS, 2006)

3.2.1.2 Problema do Carteiro Rural

O Problema do Carteiro Rural (PCR), *Rural Postman Problem (RPP)*, é semelhante ao PCC, porém ao invés de um grafo inteiro ser atendido, apenas um subconjunto de arcos, chamado de arcos requeridos, devem ser atravessados ou servidos e tem recebido grande atenção na literatura (CORBERÁN; PRINS, 2010).

Fernández *et al.* (2003) define o PCR da seguinte maneira: dado um grafo $G = (V, E)$, um conjunto de arcos requeridos $E_r \subseteq E$ e comprimentos de arestas $c_{ij} \geq 0$ para $\{i, j\} \in E$, é necessário encontrar um caminho de comprimento mínimo em E que atravesse cada arco de E_r ao menos uma vez.

Para Eiselt *et al.* (1995a) a maior parte dos problemas reais eram usualmente modelados como PCR. Há muitas aplicações que envolvem vários veículos e deve haver restrições que delimitem capacidade, distância ou tempo viajado. Cita, ainda,

alguns trabalhos modelados dessa forma em: limpeza de ruas, remoção de neve, coleta de lixo, entrega de cartas, rotas de ônibus escolares, leituristas (por exemplo os responsáveis por fazer leitura de energia elétrica e/ou água).

Segundo Corberán e Prins (2010) alguns autores demonstram que o PCR é um caso especial do *MBCPP*, o autor também apresenta algumas variações vinculadas a estrutura do grafo, como: Problema do Carteiro Rural com Vento (semelhante ao *WPP*), problemas não direcionados, direcionados e mistos. E variações conforme a sequência em que os arcos são atendidos.

Ghiani *et al.* (2005) apresenta uma heurística para resolver uma variante chamada de Problema do Carteiro Rural Periódico onde deve ser obtida uma rota com custo mínimo que deve levar em conta múltiplos períodos e cada arco deve ser visitado um determinado número de vezes ao longo desses dias.

3.2.2 Problema de Roteamento em Arcos Capacitados

Dror (2000) apresenta uma questão simples: “O que aconteceria se introduzisse uma coleta ou entrega ao atravessar um arco?”. Obviamente a questão só seria importante se um veículo tivesse uma capacidade bem definida.

Frequentemente essa capacidade é associada a carga máxima em um veículo, mas também pode ser vinculado à máxima capacidade de trabalho ou distância percorrida (CORBERÁN; PRINS, 2010).

Chamado em inglês pelo nome de *Capacitated Arc Routing Problem (CARP)*, o Problema de Roteamento em Arcos Capacitados é caracterizado por ter uma demanda não negativa associada a cada arco do grafo. Dado que cada veículo de uma frota tem uma determinada capacidade W , esses veículos devem atravessar os arcos coletando ou entregando as determinadas demandas sem exceder sua capacidade, problema este proposto por Golden e Wong em 1981.

Segundo Lacomme, Prins e Ramdane-Chérif (2002a) o problema *CARP* básico da literatura envolve apenas períodos simples, um dia, e um modelo não direcionado onde cada arco de uma rede pode ser atravessado em qualquer direção.

O modelo matemático proposto por Dror (2000) parte de um grafo $G = (V, A)$ com R o conjunto dos arcos requeridos em A , utilizando os seguintes parâmetros:

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

- W = a capacidade do veículo k
- C_{ij} = custo do arco $(i, j) \in A$
- K = o número de carros
- x_{ijk} = o número de vezes que um veículo k atravessa o arco $(i, j) \in A$
- y_{ijk} = variável binária que assume o valor 1 se o veículo k descarrega ou coleta ao longo do arco $(i, j) \in R$, ou 0 caso contrário

O problema é modelado da seguinte maneira:

$$\min Z = \sum_{(i,j) \in A} \sum_{k=1}^K C_{ij} x_{ijk} \quad (6)$$

Sujeito às seguintes restrições:

$$\sum_{p \in A} x_{pik} = \sum_{p \in A} x_{ipk} \quad \forall i \in V, \forall k = 1, 2, \dots, K \quad (7)$$

$$\sum_{k=1}^K y_{ijk} = 1 \quad \forall (i, j) \in R \quad (8)$$

$$x_{ijk} \geq y_{ijk} \quad \forall (i, j) \in R, k = 1, 2, \dots, K \quad (9)$$

$$\sum_{(i,j) \in R} q_{ij} y_{ijk} \leq W \quad \forall k = 1, 2, \dots, K \quad (10)$$

$$M \sum_{i \notin S, j \in S} x_{ijk} \geq \sum_{(j,p) \in A[S] \cap R} x_{jpk} \quad \forall S \subseteq N, 1 \notin S, A[S] \cap R \neq \emptyset, k = 1, 2, \dots, K \quad (11)$$

$$y_{ijk} \in \{0,1\}, \forall (i, j) \in R, k = 1, 2, \dots, K \quad (12)$$

$$x_{ijk} \in Z^+, \forall (i, j) \in A, k = 1, 2, \dots, K \quad (13)$$

A função objetivo (6) procura minimizar o custo representado pela distância total percorrida pelos veículos. Importante perceber que os arcos podem ser atravessados mais de uma vez. A restrição (7) garante a conservação do fluxo dos veículos ao longo do grafo. A equação (8) faz com que cada arco seja atendido uma vez durante todo o trajeto por um único veículo k . Um carro só irá entregar ou receber algo no arco $(i, j) \in R$ se esse arco for atravessado, isso é garantido pela restrição (9).

A capacidade de cada veículo está delimitada na restrição (10). A restrição (11) evita a formação de subciclos e ainda garante que o depósito representado pelo nó 1 esteja incluído na rota, M é um número tão grande quanto o número máximo de vezes que um arco pode ser atravessado. Em (12) temos as variáveis binárias que definem a entrega ou coleta no arco, e em (13) a variável x_{ijk} assumindo um valor inteiro que representa a quantidade de vezes que o arco foi atravessado.

Golden e Wong (1981) apresentam o primeiro estudo sobre o *CARP*. Um problema muito rico em teoria e aplicabilidade que era negligenciado até então. No seu artigo foi apresentado o problema, uma formulação matemática, sua relação com outros problemas da literatura, sua complexidade computacional *NP-Hard* e um algoritmo para as primeiras soluções do problema.

Existem diversas maneiras de solucionar o *CARP* na literatura. Algumas delas são apresentadas em Hertz, Laporte e Mittaz (2000) utilizando um algoritmo baseado em Busca Tabu, Beullens *et al.* (2003) se basearam em *guided local search*, Belenguer e Benavent (2003) utilizando planos de corte, Lacomme, Prins e Sevaux (2006) utilizando Algoritmos Genéticos e o trabalho de Longo, Aragão e Uchoa (2006) transformando o problema em um Problema de Roteamento de Veículos Capacitado.

Eiselt *et al.* (1995a) afirmaram na época, que o *CARP* é provavelmente o mais importante problema na área de roteamento em arcos e, ainda que o PCR é um caso especial de *CARP* onde $W = |E_r|$, $q_{ij} = 1$ se $(i, j) \in E_r$ e $q_{ij} = 0$ se $(i, j) \notin E_r$.

Golden e Wong (1981) associam o *CARP* a outros problemas ainda, segundo eles se $q_{ij} > 0$ para todo $(i, j) \in E$, o *CARP* é reduzido a um Problema do Carteiro Chinês Capacitado (CCPP) e se há apenas um veículo de capacidade $W \geq \sum_i \sum_j q_{ij}$ tem-se o CPP.

Golden, DeArmon e Baker (1983) propuseram 25 instâncias para realizar experimentos computacionais com algoritmos para essa classe de problemas chamando de Problema do Carteiro Chinês Capacitado. Segundo Corberán e Prins (2010) essas instâncias são referências clássicas, porém as mais utilizadas são apenas 23 pois 2 das 25 instâncias contém inconsistências, mais especificamente as instâncias 8 e 9.

Tagmouti, Gendreau e Potvin (2007), em seu trabalho de roteamento em arcos com custos dependentes de tempo, ressaltam a importância de um serviço ser feito no momento certo, como no caso da manutenção em rodovias durante o inverno em

idades que nevam. Levam em conta a capacidade dos carros e propõe uma transformação do *ARP* em um problema de roteamento em nós, solucionando através de um esquema de geração de colunas.

Lacomme, Prins e Ramdane-Chérif (2002b) apresentam uma versão estendida do *CARP* onde existem turnos proibidos, ou seja, existe uma relação de precedência entre os arcos que podem proibir um movimento ao sair de um arco e entrar em outro. Além disso apresentam a possibilidade de um carro exceder em 10% sua capacidade quando necessário (possível em problemas de coleta de lixo, por exemplo).

Para resolver o problema Lacomme, Prins e Ramdane-Chérif (2002b) propõem uma modelagem matemática e um algoritmo de planos de corte, onde inicialmente são desconsideradas as restrições que evitam subciclos (*subtours*) e a cada iteração quando ocorre um subciclo a restrição que evitará o ocorrido é inserida. Isso ocorre até a solução ótima ser encontrada, quando possível. Fazem testes em 11 instâncias, onde também testam um Algoritmo Genético Híbrido¹ que encontrou 5 soluções ótimas e um desvio médio de 4,75% dos custos obtidos com o Algoritmo de Planos de Corte.

Bartolini, Cordeau e Laporte (2013) desenvolveram um algoritmo exato capaz de resolver o *CARP* e outra variante chamada *CARP with Deadheading Demand*, que obtido quando tem-se associado um consumo de capacidade adicional ao atravessar arcos que não devem ser servidos. Problema aplicado a situações em que há um consumo de energia dependendo da atividade na aresta.

Em Belenguer *et al.* (2010) encontra-se uma variação para o problema chamada *Split-Delivery CARP*. Os autores ressaltam que uma aresta não precisa ser servida por apenas um veículo. Citam o exemplo de um caminhão que está fazendo coleta de lixo e tem toda sua capacidade completa no meio do atendimento a uma rua, logo outro caminhão poderia terminar a tarefa. Para solução, utilizam uma metaheurística chamada de *Multistart Evolutionary Local Search* que é uma generalização dos métodos *GRASP*, *ILS* e *Evolutionay Local Search*.

Arc Routing and Scheduling Problem with Transshipment é uma variante do *CARP* encontrada em Rosa *et al.* (2002). Basicamente, uma demanda é coletada por uma frota de veículos e levada até um depósito. Nesse depósito a demanda é

¹ O autor considera Algoritmo Genético Híbrido por usar um operador de mutação

processada e novamente transportada por outros veículos para uma facilidade/localidade final. O modelo também é utilizado na coleta de lixo, os autores encontraram uma solução através da metaheurística Busca Tabu.

Hashemi Doulabi e Seifi (2013) utilizam os conceitos de *CARP* para tomar uma decisão estratégica que é determinar a localização de múltiplos depósitos em grafos. Eles mostram economias significativas nos custos dos problemas.

Uma característica relevante ao problema tratado aqui nesse trabalho é a questão da falta de um depósito v_0 , essa é também uma característica que ocorre no chamado Problema de Roteamento em Arcos Capacitado e Aberto (*Open Capacitated Arc Routing Problem – OCARP*) explorada em Usberti, França e França (2011) e bem definida na tese de Usberti (2012, p.3):

O *OCARP* é similar ao *CARP*, porém, no primeiro problema, as rotas não estão restritas a ciclos. Logo tanto rotas abertas quanto fechadas são permitidas. Nos dois problemas existem as arestas requeridas, que devem ser atendidas, e as arestas não-requeridas, que são utilizadas como atalhos entre uma aresta requerida e outra. Considerando que uma rota se inicia em um nó v_s e termina em um nó v_t , no *CARP* tem-se que $v_s = v_t = v_0$ para todas as rotas, sendo assim um caso particular do *OCARP*. Conseqüentemente, o *OCARP* pode ser enxergado como uma generalização do *CARP* e, após uma exaustiva busca, constatou-se que esse problema nunca foi formalmente reportado na literatura, apesar de relevantes problemas práticos poderem ser modelados como um *OCARP*. (USBERTI, 2012, p.3).

Entre as aplicações Usberti (2012) e Usberti, França e França (2011) estão as já tratadas na literatura como em rotas de leituristas de água e energia e o problema de caminho de corte. É também proposta a modelagem matemática, método exato, uma heurística construtiva *GRASP* baseada em uma heurística de varredura de caminhos combinadas com algoritmos de inserção e uma metaheurística de reconexão evolutiva por caminhos. Fung, Liu e Jiang (2013) também tratam do problema propondo um algoritmo memético com resultados de qualidade superior ao clássico Algoritmo Genético.

Outras variantes do trabalho são *CARP* com demandas estocásticas disponível em Laporte, Musmanno e Vocaturo (2010). Em Amaya *et al.* (2007) *CARP* com Pontos de Refil, onde um veículo servindo um arco deve ser reabastecido *in loco*

por um segundo veículo, como no caso de pinturas de rodovias em Quebec (Canadá) onde os veículos são reabastecidos quando estão trabalhando.

3.2.3 Problema de Roteamento em Arcos Capacitados e Periódicos

O *CARP* é amplamente utilizado para resolver o problema da coleta de lixo urbano como uma decisão de nível operacional, tomada diariamente. Porém em alguns problemas é necessário tomar uma decisão de nível tático, onde envolva um horizonte de tempo maior com múltiplos períodos, sujeito a restrições que envolvem determinadas frequências. Lacomme, Prins e Ramdane-Chérif (2002) apresentam este problema, suas variantes e um Algoritmo Genético para resolvê-lo.

Outro nome dado para o problema foi Problema de Planejamento em Roteamento em Arcos em Lacomme, Prins e Ramdane-Chérif (2002c) devido ao fato de se considerar diversos dias. Com isso, foi possível resolver simultaneamente os problemas de planejamento alocando *CARPs* diariamente. Lacomme, Prins e Ramdane-Chérif (2002a) fazem seu trabalho baseado na coleta de lixo urbano da cidade de Troyes na França e o resolvem com um Algoritmo Genético.

Alguns conceitos começaram a ser tratados com Eglese (1994) com o roteamento de veículos responsáveis por não permitir que haja acúmulo de neve nas vias dos países com invernos rigorosos. Nesse problema, o autor fez um planejamento para apenas um dia, no qual algumas vias deveriam ser tratadas mais de uma vez no intervalo de poucas horas.

Christofides e Beasley (1984) levam em conta a importância do período no roteamento de veículos, visam reduzir os custos de distribuição ao longo de alguns dias. O problema é solucionado designando rotas para cada dia através da resolução de um problema de p -medianas e então resolvendo um Problema do Caixeiro Viajante. Cada cliente deve ser atendido mais de uma vez durante o horizonte de tempo em uma combinação de dias, sempre limitado pela capacidade dos carros.

As companhias de lixo reúnem várias estatísticas que visam prever a taxa de lixo a ser produzida diariamente em cada região. Sabem que nos grandes centros e em regiões com muitos prédios a produção de lixo é maior que em outros bairros onde existem apenas casas. Logo, os centros necessitam de coletas mais frequentes que as outras áreas residenciais, por isso é necessário um planejamento baseado em

múltiplos períodos. Baseado nisso Lacomme, Prins e Ramdane-Chérif (2005, p. 539) definem o *PCARP* da seguinte forma:

O problema de roteamento de veículos capacitado e periódico em arcos consiste em selecionar para cada rua um número de tratamento diário ou igual a sua frequência (decisão tática), e então formar uma rota em cada dia de forma que as ruas designadas para esse determinado dia sejam atendidas (decisão operacional). Obviamente, o custo total no final do período depende da combinação das decisões de designação e roteamento (LACOMME; PRINS; RAMDANE-CHÉRIF, 2005, p. 539, tradução nossa).

De maneira geral o *PCARP* é uma extensão natural do *CARP* e ao invés do problema ser resolvido para apenas um dia ele é expandido para múltiplos períodos (MONROY; AMAYA; LANGEVIN, 2013).

Resolver o *PCARP* implica em determinar simultaneamente as decisões em nível tático e operacional durante um horizonte de tempo. Infelizmente, em muitas aplicações desse problema, existem algumas complicações, como por exemplo: as demandas podem ser flutuantes de acordo com os dias ou espaçadas em restrições entre dias de serviço (LACOMME *et al.*, 2005).

Lacomme, Prins e Ramdane-Chérif (2002a) comparam o *PCARP* com um gerenciamento de produção, onde tem-se a fase de decisões no nível tático, a qual é designado um plano de produção, enquanto o resultado de apenas um dia (resultado do problema *CARP*) corresponde a uma decisão operacional, o qual é a sequência dos serviços a serem executados nesse único dia.

Na prática o custo de servir um arco depende da demanda. Baseado nisso Lacomme e Prins (2005) propõe uma classificação para o *PCARP* dividindo inicialmente em duas classes, que pode ser encontrada na Figura 6.

A CLASSE A representa o conjunto dos problemas em que a demanda não resulta de acumulação diária de algum fator, por exemplo, muitos serviços de prevenção, inspeção e monitoramento como a pulverização de herbicidas em trilhos de trem: o tempo gasto por metro percorrido e a demanda ao longo do tempo é fixa.

Já a CLASSE B envolve problemas como a coleta de lixo, onde a demanda pela coleta aumenta conforme vão se passando os dias. Além disso, podem ser subdivididos em problemas acíclicos (subclasse B1) ou cíclicos (subclasse B2). Em B1,

o horizonte de tempo é bem definido e a demanda é flutuante ao longo do tempo, as vezes é necessário passar muitas vezes e em outros casos não, como no caso da remoção de plantas em rodovias, visto que no inverno as plantas não crescem. Em B2 o intervalo de tempo se repete continuamente, por exemplo, na coleta de lixo que se repete de semana em semana.

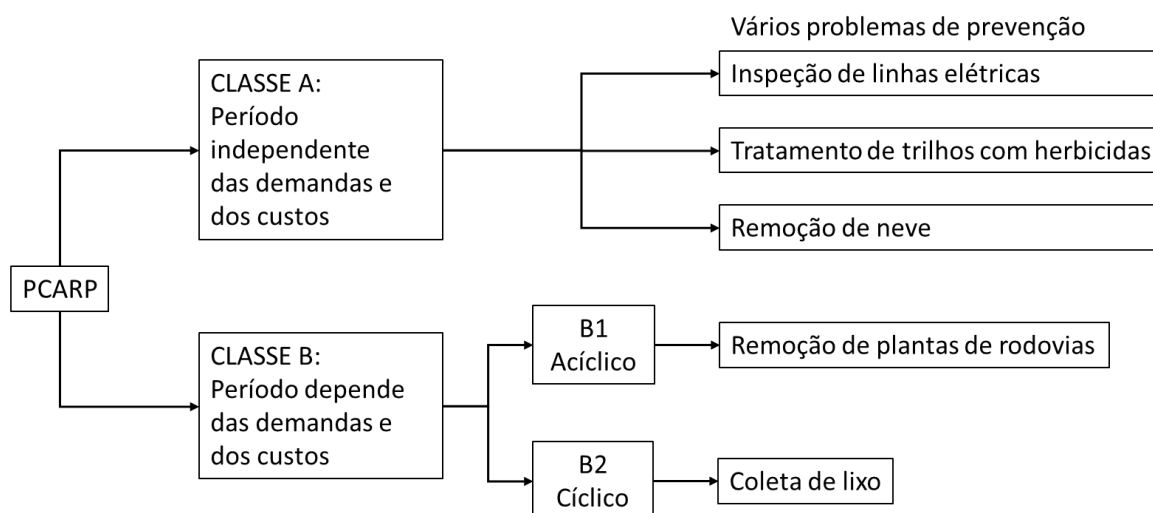


Figura 6 – Uma simples classificação do *PCARP*
Fonte: Lacomme, Prins e Ramdane-Chérif (2005)

Monroy, Amaya e Langevin, (2013) trabalham com o problema do monitoramento da malha rodoviária que é realizado periodicamente, CLASSE A conforme Figura 6. As ruas ou rodovias são divididas em classes de acordo com a necessidade de vigilância e a hierarquia das vias. Cada categoria tem sua demanda (número de passagens) durante um horizonte de tempo já definido. O objetivo é designar um conjunto de rotas satisfazendo as frequências de cada classe de vias em cada sub período de tempo sem exceder a capacidade do veículo.

Eglese (1994), em seu trabalho *“Routeing winter gritting vehicles”* trata do roteamento de veículos que despejam sal nas rodovias durante o inverno para descongelar as vias e evitar acidentes. O problema até sua data de publicação não foi chamado de *PCARP*, mas possui algumas das características que envolvem esse tipo de problema. Apesar de ser um planejamento para apenas um dia, existiam intervalos para atendimentos.

O problema de Eglese (1994) levava em conta a capacidade dos veículos delimitada pela quantidade de sal que cada carro é capaz de carregar e despejar nas vias. As ruas são divididas em duas classes, a primeira classe é formada pelas vias que deveriam ser atendidas a cada 2 horas e na outra classe as vias com uma

demanda de 4 horas. Se o despejar de sal ocorresse muito cedo, o vento poderia espalhar todo sal despejado e se fosse feito muito tarde talvez não pudesse ser feito com o aumento do fluxo de veículos.

Neste trabalho de Eglese (1994) haviam múltiplos depósitos, garagens onde os carros ficavam aguardando a ordem de executar o serviço. Tinha como objetivo reduzir os custos fixos com veículos (reduzindo o número dos mesmos), reduzir a distância percorrida e o número de dias no ano em que eram requeridos esses veículos. O autor resolveu seu problema em basicamente duas fases, inicialmente dividindo em zonas e resolvendo o Problema de Carteiro Chinês relaxado, para então fazer melhorias com um algoritmo de *Simulated Annealing* usado para minimizar o número de rotas e punir algumas restrições que eram excedidas. Eglese (1994) conseguiu reduzir o número de veículos e o número de depósitos usados atendendo as demandas.

Uma decisão importante a ser tomada antes de trabalhar com o problema é decidir como serão tratadas as restrições de periodicidades e isso não deixa de ser uma decisão estratégica a ser tomada por um especialista.

As restrições correspondentes a periodicidade podem assumir basicamente duas formas e devem estar disponíveis ao formular o problema. A primeira forma consiste em definir um mínimo espaço de tempo em que cada tarefa (u) deve ser executada ou um espaço de tempo máximo entre dois serviços. Outra forma é utilizar uma combinação de dias em que pode ser executado o serviço. Se o horizonte de tempo for de np dias pode haver um conjunto para cada tarefa, $comb(u)$, onde é designado os dias em que o arco deve ser atendido. Por exemplo, se um arco necessita ser atendido todos os dias, esse conjunto é representada pelo conjunto $(1,2,\dots,np)$, ou se for apenas no primeiro e terceiro dia do horizonte de tempo, o conjunto seria formado por $(1,3)$ (LACOMME; PRINS; RANDAME-CHÉRIF, 2002).

A forma da combinação de dias é utilizada por Chu, Labadi e Prins (2005) e também por Lacomme, Prins e Ramdane-Chérif (2002) e é defendida pelos autores por existirem duas grandes vantagens: as restrições de espaço são satisfeitas implicitamente e os cálculos de demanda podem ser feitos antecipadamente para os problemas da CLASSE B.

Chu, Labadi e Prins (2006) aplicam dois algoritmos para resolver o *PCARP*. O primeiro é uma Heurística de Melhor Inserção e o segundo é chamado *Scatter Search* baseado em busca local. Ainda lembram que uma solução factível deve ter suas

necessidades atendidas o número de vezes que forem necessárias (frequências), cada viagem deve iniciar e terminar no depósito, cada tarefa deve ser designada no máximo uma vez em um dia e ainda a capacidade do veículo deve ser respeitada.

Chen *et al.* (2009) trabalha com múltiplos períodos utilizando um conceito de Problema de Roteamento em Arcos em Múltiplos dias, do inglês *Multiday Arc Routing Problem (MARP)*. Ele agrupa os problemas de roteamento aqui citados numa categoria de Problemas de Roteamento em Arcos Determinísticos e introduz os conceitos de Problema de Roteamento em Arcos Probabilístico e *MARP* para resolver um problema no contexto de entrega de pequenos pacotes. Em ambos problemas, considera-se uma probabilidade de atravessar determinado arco em algum período.

Kansou e Yassine (2009) combinaram um algoritmo da colônia de formigas com uma heurística de inserção para resolver o *PCARP* conseguindo resultados robustos e com uma performance rápida. Cada serviço podia acontecer em uma combinação de dias que satisfaziam as frequências. Chamou seu problema de Problema de Roteamento em Arcos Capacitado e Periódico Misto em virtude da natureza do grafo.

Lacomme, Prins e Ramdane-Chérif (2002c) tem um *PCARP* com dois objetivos, minimizar a frota de veículos e minimizar os custos das viagens. Para resolver este problema utilizaram um Algoritmo Genético híbrido.

O equivalente ao *PCARP* no roteamento em nós é o Problema de Roteamento Periódico de Veículos (PRPV), Tortelly Jr. e Ochi (2003) citam algumas das heurísticas já utilizadas na literatura para resolver essa classe de problemas e propõe uma nova metaheurística híbrida baseada nos conceitos de *GRASP* com Busca Tabu para chegar a uma solução.

Em Francis e Smilowitz (2006) encontra-se uma variação do PRPV onde existe a escolha de serviços. É um problema onde a frequência de visita é definida pelo modelo. O problema presume que benefícios podem ser ganhos ao visitar algum ponto mais frequentemente que um mínimo requerido.

Vidal *et al.* (2012) trata de um PRPV com múltiplos depósitos. Lembra que nem todos os problemas da literatura recebem o mesmo grau de atenção apesar da aplicabilidade a situações reais como: suprimento de matéria prima, coleta e distribuição de comida e operações de manutenção. Os autores conseguem excelentes resultados utilizando um algoritmo baseado em Algoritmos Genéticos.

4 METODOLOGIA

4.1 MODELOS JÁ PROPOSTOS

Entre os modelos propostos para o *PCARP*, encontra-se na literatura três de alta relevância que tratam do *PCARP*. Os dois primeiros são proposto por Chu, Labadi e Prins (2004, 2005) e outro proposto por Monroy, Amaya e Langevin (2013). Ao longo dessa seção, serão apresentados cada um desses modelos e sua aplicação.

4.1.1 Modelo de Chu, Labadi e Prins (2004)

Chu, Labadi e Prins (2004, 2005) partem da seguinte proposição: O *PCARP* é definido em múltiplos períodos num horizonte de tempo H com p períodos (“dias”) e um grafo não direcionado $G = (X, E)$ onde X é o conjunto de n nós (cruzamentos) e E um conjunto de m arcos (ruas). Um subconjunto R de t arcos, chamados tarefas, deve ser servido. Cada tarefa $[i, j]$ tem uma frequência f_{ij} (número de serviços requeridos em H). Vale ressaltar que os autores trabalham com um problema que está inserido num contexto de coleta de lixo, sempre que for enunciado que o arco está sendo servido é o mesmo que dizer que está coletando o lixo do mesmo arco.

No primeiro modelo publicado por Chu, Labadi e Prins (2004) cada arco $[i, j]$ tem um custo para ser atravessado de C_{ij} e uma produção diária q_{ij} para cada dia p . $\Delta_{min}[i, j]$ e $\Delta_{max}[i, j]$ são, respectivamente, o espaçamento mínimo e máximo entre dois serviços consecutivos no arco $[i, j]$.

O nó 1 é o depósito que contém uma frota de V veículos idênticos limitados a uma capacidade C . O espaço de tempo entre serviços pode ser diferente para cada serviço em cada arco.

As variáveis de decisão são as seguintes:

- $x_{ijvp} = 1$ se o arco é atravessado na direção $[i, j]$ pelo veículo v no dia p
- $l_{ijvp} = 1$ se o arco $[i, j]$ é servido pelo veículo v no dia p
- $W_{ijvp}^r = 1$ se a produção diária do arco $[i, j]$ no dia r é coletado pelo veículo v no dia p

As variáveis assumem 0 no caso contrário e a modelagem segue a seguinte função objetivo e as restrições:

$$\min Z = \sum_{i=1}^n \sum_{j=1}^n \sum_{v=1}^V \sum_{p=1}^P C_{ij} x_{ijvp} \quad (14)$$

Sujeito às seguintes restrições:

$$\sum_{j=1}^n x_{ijvp} = \sum_{j=1}^n x_{jivp} \quad \forall v = 1, \dots, V, \forall p = 1, \dots, P, \forall i \in X \quad (15)$$

$$x_{ijvp} + x_{jivp} > l_{ijvp} \quad \forall [i, j] \in R, \forall v = 1, \dots, V, \forall p = 1, \dots, P \quad (16)$$

$$\sum_{v=1}^V l_{ijvp} \leq 1 \quad \forall [i, j] \in R, \forall p = 1, \dots, P \quad (17)$$

$$\sum_{p=1}^P \sum_{v=1}^V l_{ijvp} = f_{ij} \quad \forall [i, j] \in R \quad (18)$$

$$\sum_{p=1}^P \sum_{v=1}^V W_{ijvp}^r = 1 \quad \forall [i, j] \in R, \forall r = 1, \dots, P \quad (19)$$

$$W_{ijvp}^r \leq l_{ijvp} \quad \forall [i, j] \in R, \forall r, p = 1, \dots, P, \forall v = 1, \dots, V \quad (20)$$

$$W_{ijvp}^r \geq l_{ijvp} - \sum_{v=1}^V \sum_{k=r}^P W_{ijvk}^r - \sum_{v=1}^V \sum_{k=1}^{p-1} W_{ijvk}^r \quad (21)$$

$$\forall [i, j] \in R, \forall p = 1, \dots, P, \forall r = p + 1, \dots, P, \forall v = 1, \dots, V$$

$$W_{ijvp}^r \geq l_{ijvp} - \sum_{v=1}^V \sum_{k=r}^{p-1} W_{ijvk}^r \quad (22)$$

$$\forall [i, j] \in R, \forall p = 1, \dots, P, \forall r = 1, \dots, P, \forall v = 1, \dots, V$$

$$\sum_{[i,j] \in R} \sum_{r=1}^P q_{ijr} W_{ijvp}^r \leq C \quad \forall p = 1, \dots, P, \forall v = 1, \dots, V \quad (23)$$

$$\sum_{k=p}^{p+\Delta_{\min}(i,j)-1} l_{ijvp} \leq 1 \quad \forall [i, j] \in R, \forall v = 1, \dots, V, \forall p = 1, \dots, P - \Delta_{\min}(i, j) - 1 \quad (24)$$

$$\sum_{k=p}^{p+\Delta_{\max}(i,j)-1} l_{ijvp} \geq 1 \quad \forall [i, j] \in R, \forall v = 1, \dots, V, \forall p = 1, \dots, P - \Delta_{\max}(i, j) - 1 \quad (25)$$

$$\sum_{i \in S} \sum_{j \notin S} (x_{ijvp} + x_{jivp}) \geq l_{rsvp} \quad (26)$$

$$\forall p = 1, \dots, P, \forall v = 1, \dots, V, \forall S \subseteq \{2, \dots, n\}, \forall [r, s] \in R | r, s \in S$$

$$W_{ijvp}^r, x_{ijvp}, x_{jivp}, l_{ijvp} \in \{0, 1\} \quad (27)$$

$$\forall [i, j] \in R, \forall p = 1, \dots, P, \forall r = 1, \dots, P, \forall v = 1, \dots, V$$

A restrição (27) garante que todas as variáveis sejam binárias. O problema consiste em minimizar o custo total das viagens (14). Restrição (15) garante a continuidade de viagens, o fluxo dos veículos. Um serviço em um arco só pode ocorrer se esse arco for atravessado devido à restrição (16). A restrição (17) garante que cada arco pode ser servido ao máximo uma vez por dia e por apenas um carro. Em (18) tem-se a garantia de que os arcos serão servidos quantas vezes forem necessárias dado por f_{ij} que é a frequência de cada arco $[i, j]$. Em (19) a produção de um dia é coletada em uma viagem em apenas uma direção.

A restrição (20) diz que a produção diária não pode ser recolhida se não houver coleta no mesmo dia no arco. As restrições (21) e (22) garantem que não haja coleta entre os dias r e o dia p se a produção diária do dia r é coletada no dia p para os casos $r > p$ e $r \leq p$, respectivamente.

A capacidade de cada veículo não pode ser excedida em cada dia p e em cada viagem v (23). Restrições (24) e (25) asseguram um valor mínimo e máximo de dias entre duas coletas consecutivas de um arco. E a restrição (26) previne a criação de subciclos. Benavent *et al.* (2000) *apud*. Chu, Labadi e Prins (2005) utilizou uma equação similar para resolver o *CARP*, mostrando a sua eficiência.

4.1.2 Modelo de Chu, Labadi e Prins (2005)

O segundo modelo proposto por Chu, Labadi e Prins (2005) utiliza a mesma proposição do modelo anterior, porém neste a forma como cada serviço será atendido é abordada de uma forma diferente.

Tem-se um conjunto com combinações de dias permitidos $comb(i, j)$ em que os serviços podem ser executados e nc como o número total de distintas combinações de dias. Combinações essas que são indexadas de 1 até nc e definidas por uma matriz A , 0-1, de dimensões $nc \times P$, onde $A_{kp} = 1$ se a combinação k contém o dia p .

O nó 1 é o depósito que contém uma frota de V veículos idênticos limitados a uma capacidade W . A demanda Q_{ijkp} deve ser atendida por qualquer combinação k de dias e qualquer dia p é conhecido.

As variáveis de decisão utilizadas são as seguintes:

- $x_{ijvp} = 1$ se o arco $[i, j]$ é atravessado de i para j pelo veículo v no dia p
- $l_{jvkp} = 1$ se a tarefa $[i, j]$ é servida de i para j pelo veículo v no dia p da combinação k
- $Z_{ijk} = 1$ se a tarefa $[i, j]$ usa a combinação k

Todas as variáveis assumem valor zero caso contrário.

Para cada tarefa $[i, j]$, cada combinação permitida em $comb(i, j)$ é um conjunto de combinação de índices. Por convenção ainda tem-se que $Q_{ijkp} = 0$ se e somente se $k \notin comb(i, j)$ ou $A_{kp} = 0$. O problema é formulado da seguinte maneira:

$$\min z = \sum_{[i,j] \in R} \sum_{v=1}^V \sum_{p=1}^P C_{ij} \cdot (x_{ijvp} + x_{jivp}) \quad (28)$$

Sujeito às seguintes restrições:

$$\sum_{k \in comb(i,j)} Z_{ijk} = 1 \quad \forall [i, j] \in R \quad (29)$$

$$\sum_{[i,j] \in R} x_{ijvp} = \sum_{[i,j] \in R} x_{jivp} \quad \forall v = 1, \dots, V, \forall p = 1, \dots, P, \forall i \in X \quad (30)$$

$$x_{ijvp} \geq \sum_{k \in comb(i,j)} l_{jvkp} \quad \forall [i, j] \in R, \forall v = 1, \dots, V, \forall p = 1, \dots, P \quad (31)$$

$$x_{jivp} \geq \sum_{k \in comb(i,j)} l_{jvkp} \quad \forall [i, j] \in R, \forall v = 1, \dots, V, \forall p = 1, \dots, P \quad (32)$$

$$\sum_{v=1}^V (l_{jvkp} + l_{jivkp}) = A_{kp} Z_{ijk} \quad \forall [i, j] \in R, \forall k \in comb(i, j), \forall p = 1, \dots, P \quad (33)$$

$$\sum_{[i,j] \in R} \sum_{k \in comb(i,j)} Q_{ijkp} \times (l_{jvkp} + l_{jivkp}) \leq W \quad \forall p = 1, \dots, P, \forall v = 1, \dots, V \quad (34)$$

$$\sum_{i \in S} \sum_{j \notin S} x_{ijvp} \geq l_{rsvkp} + l_{srvkp} \quad (35)$$

$$\begin{aligned}
& \forall S \subseteq \{s, \dots, n\}, \forall [r, s] \in E_r(S), \forall p = 1, \dots, P, \forall k \in \text{comb}(r, s), \forall v = 1, \dots, V \\
& x_{ijvp}, x_{jivp}, z_{ijk}, l_{jvkp}, l_{jivp} \in \{0, 1\} \\
& \forall [i, j] \in R, \forall p = 1, \dots, P, \forall v = 1, \dots, V, \forall k \in \text{comb}(i, j)
\end{aligned} \tag{36}$$

A função objetivo dada pela equação (28) é de minimizar o custo total das viagens. Apenas uma combinação de dias pode ser designado para cada tarefa (29). As restrições de fluxo de veículos são impostas na restrição (30). A restrição (31) e (32) não permite que um arco seja servido se ele não for atravessado. Se uma tarefa ocorre no dia p significa que ela pertence à combinação selecionada para a tarefa, cada tarefa ainda só pode ser atendida por um carro e em apenas uma direção graças a restrição (33). A restrição (34) garante que a capacidade de cada veículo seja respeitada para cada dia p e cada viagem v .

A equação (35) evita a formação de subciclos na solução ótima pois:

(...) se $l_{rsvkp} + l_{srvpk} = 1$ então o veículo v deve atravessar ao menos uma vez o corte $(S, X|S)$, desde que ambos nós finais do arco requerido $[r, s]$ estejam em S . Isso não previne a formação isolada de ciclos constituídos apenas por arcos soltos. Felizmente, como os custos são não negativos, isso nunca acontecerá quando a solução ótima for alcançada. (CHU; LABADI; PRINS, 2005, p. 245, tradução nossa).

Finalizando em (36) temos que todas as variáveis são binárias.

4.1.3 Modelo de Monroy, Amaya e Langevin (2013)

O modelo proposto por Monroy, Amaya e Langevin (2013) contempla um problema em que os arcos são divididos em classes. Essas classes servem como base para determinar as necessidades de cada arco ao longo do horizonte de tempo. Sua modelagem é feita para grafos direcionados, as notações que utiliza para formular seu problema são as seguintes:

- Ω_c : conjunto de arcos da classe c
- T : conjunto dos conjuntos Ω_c
- e_c : conjunto dos subperíodos (sub horizontes) associados a arcos da classe c
- $O(n)$: conjunto dos arcos deixando o nó $n \in N$
- $I(n)$: conjunto dos arcos entrando no nó $n \in N$
- S : subconjunto de arcos
- $N(S)$: conjunto dos nós incidentes nos arcos de S

V : conjunto de rotas
 w_j : conjunto de rotas correspondentes ao subperíodo j
 f_j : número de serviços requeridos no subperíodo j
 p_a : peso do arco a

Em suas instâncias, utiliza um horizonte de tempo de 14 dias e 3 tipos de classes. A classe 1 é formada pelos arcos que tem necessidades de ao menos uma passagem trabalhando nos dias 1 até 5, 6 até 7, 8 até 12 e 13 até 14, a classe 2 tem demandas entre os dias 1 até 7 e 8 até 14, e a classe 3 é formada pelos arcos que necessitam de passagem entre o dia 1 até o dia 14. A Figura 7 a seguir ilustra essa situação:

	Classe de arcos	Segunda – Sexta (1)	Sábado – Domingo (1)	Segunda – Sexta (2)	Sábado – Domingo (2)
Subperíodos	CLASSE 1	w_1	w_2	w_3	w_4
	CLASSE 2	w_5		w_6	
	CLASSE 3	$w_7 = V$			

Figura 7 – Sub-horizontes de tempo associado as classes de arcos

Fonte: Monroy, Amaya e Langevin (2013)

Sua formulação consiste em definir k rotas que atendam as demandas, as variáveis de decisão utilizam notação de arcos e são as seguintes:

- x_a^k assume o valor 1 se o arco a é servido pela rota k ou então 0 caso contrário
- y_a^k contabiliza o número de vezes que o arco a é atravessado pela rota k (sem ser servido)

Quando um arco é servido ele ocupa um valor c_a da capacidade Q do veículo, e quando é atravessado um valor t_a . O problema é modelado da seguinte forma:

$$\max z = \sum_{k \in V} \sum_{a \in R} p_a x_a^k \quad (37)$$

Sujeito às seguintes restrições:

$$\sum_{a \in O(n)} (x_a^k + y_a^k) - \sum_{a \in I(n)} (x_a^k + y_a^k) = 0 \quad \forall k \in V, n \in N \quad (38)$$

$$\sum_{a \in R} (c_a x_a^k) + \sum_{a \in R} (t_a y_a^k) \leq Q \quad \forall k \in V \quad (39)$$

$$M \sum_{a \in I(N(S))} (x_a^k + y_a^k) \geq \sum_{a \in O(N(S))} x_a^k + y_a^k \quad \forall S \subset A, 1 \notin N(S), k \in V \quad (40)$$

$$\sum_{k \in w_j} x_a^k \geq f_j \quad \forall j \in e_c, a \in \Omega_c, c = 1, \dots, |T| \quad (41)$$

$$x_a^k \in \{0,1\} \quad \forall a \in R, k \in V \quad (42)$$

$$y_a^k \in Z^+ \quad \forall a \in A, k \in V \quad (43)$$

A função objetivo maximiza o número de serviços levando em conta a classe de cada arco e seu respectivo peso, quanto mais importante o arco maior o peso. A restrição (38) garante a conservação do fluxo dos veículos. A capacidade de cada rota é dimensionada pela restrição (39). Sendo que a capacidade total de qualquer rota é obtida pela adição dos custos c_a e t_a dos arcos usados. A restrição (40) é referente a restrições de conectividade. M é um número tão grande quanto o número máximo de vezes que um arco pode ser atravessado. Se um arco em S é atravessado ou servido, ao menos um arco entrante no conjunto $N(S)$ deve ser atravessado ou servido. As restrições relacionadas ao número de serviços por cada período e classe de arcos é dada na equação (41). E as restrições (42) e (43) definem as variáveis. (MONROY; AMAYA; LANGEVIN, 2013, p. 695, tradução nossa).

4.2 PROPOSTA DE FORMULAÇÃO MATEMÁTICA

A formulação parte de um grafo não direcionado $G = (X, E)$ com n pontos, logo, $X = \{x_1, x_2, \dots, x_n\}$, e m arestas com $E = \{e_1, e_2, \dots, e_m\}$ que devem ser percorridos por nk carros definidos pelo conjunto $K = \{1, 2, \dots, nk\}$. Cada arco e é formado por um par de nós e é chamado de $x_{ij} = (x_i, x_j)$ ou $[i, j]$ e é associado a um custo c_{ij} .

Eglese (1994) considerou a capacidade dos veículos a serem usados e as taxas de despejo de sal nas rodovias transformando em uma restrição que levava em conta a máxima distância que poderia ser feita durante uma rota. Para simplificar o problema aqui tratado a capacidade de cada carro também foi convertida na máxima distância que um carro pode se deslocar, ou seja, 1 arco por dia.

A demanda ou periodicidade de cada arco é expressa na quantidade máxima de períodos em que o arco deve ser atendido ao menos uma vez $MP(x_{ij})$. Tem-se ainda o horizonte de tempo H , formado por np períodos $H = \{1, 2, \dots, np\}$, onde cada período é simbolizado pela letra p .

O modelo é baseado em Programação Linear Binária, logo para a resolução do problema foram criadas três variáveis binárias apresentadas abaixo:

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

As variáveis p_{ijp} e f_{ikp} foram criadas para que o problema tenha uma viabilidade. Cada variável p_{ijp} tem associado a ela um custo PU_{ij} , que é uma punição caso a periodicidade de um arco não seja atendida. Isso permite que na modelagem proposta haja um atraso no atendimento de alguns arcos, visto que a quantidade deve ser a menor possível.

O trabalho de Eglese (1994), no uso do algoritmo *Simulated Annealing*, utilizou um sistema de punição sempre que um veículo excedia o limite de distância estabelecido, e quando um veículo passava mais vezes que o necessário em algum arco e cita:

O uso da abordagem da função de penalidade também permite que as duas restrições sejam modeladas como restrições relaxadas, ou seja, estamos preparados para considerar rotas que só quebram o número de vezes ou restrições de distância por uma pequena quantia. Isto é desejável, como as velocidades médias e taxas de disseminação de sal só pode ser especificadas aproximadamente. (EGLESE, 1994, p. 238, tradução nossa).

A variável f_{ikp} permite que o carro folgue em um determinado dia, pois algumas vezes é possível obter uma solução com menos deslocamentos se o carro esperar para atravessar um arco. Ou simplesmente quando essa variável assumir valor 1, pode significar que as periodicidades podem ser atendidas em menor tempo, ou seja, o horizonte de tempo pode diminuir.

A função objetivo é dada pela equação (44):

$$\min Z = \sum_{[i,j] \in E} \sum_{k=1}^{nk} \sum_{p=1}^{np} c_{ij} * x_{ijkp} + \sum_{[i,j] \in E} \sum_{p=1}^{np} PU_{ij} * p_{ijp} \quad (44)$$

E o modelo está sujeito às seguintes restrições:

$$\sum_{[i,j] \in E} x_{ijkp} + f_{jkp} - \sum_{[i,j] \in E} x_{jik,p+1} - f_{jk,p+1} = 0 \quad \forall j \in X, \forall k \in K, \forall p \in H \quad (45)$$

$$\sum_{[i,j] \in E} x_{ijkp} + \sum_{[i,j] \in E} x_{jikp} + \sum_{i=1}^n f_{ikp} = 1 \quad \forall p \in H, \forall k \in K \quad (46)$$

$$\sum_{k=1}^{nk} \left(x_{ijkp} + x_{jikp} + x_{ijk,p+1} + x_{jik,p+1} + \dots \right. \\ \left. + x_{ijk,p+MP(x_{ij})-1} + x_{jik,p+MP(x_{ij})-1} \right) + p_{ijp} \geq 1 \quad (47)$$

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

$$x_{ijkp}, p_{ijp}, f_{ikp} \in \{0,1\} \quad \forall [i,j] \in E, \forall k \in K, \forall p \in H \quad (48)$$

A restrição (45) garante o fluxo diário dos carros permitindo a folga, sendo que, como serão formados ciclos, ao fim do período np o carro deve retornar para o mesmo local em que estava no dia 1, ou seja, o período $np + 1$ é igual ao dia 1 e assim por diante. Isso vale para todo o horizonte de tempo e para todas as restrições.

Ainda em (45) tem-se que cada carro que chega em um ponto num determinado dia somado a possibilidade do veículo já estar parado nesse ponto, deve ser igual ao movimento desse mesmo carro partindo desse mesmo ponto no dia seguinte ou permanecer parado onde está.

Nos problemas reais podem ocorrer imprevistos e outros problemas que atrasem o atendimento à alguns arcos. A possibilidade de um carro folgar na modelagem permite que quando ocorra algum problema dessa natureza a folga compense o tempo perdido.

A restrição (46) garante que todos os carros terão alguma designação para o dia p , além disso, garante implicitamente que a capacidade de cada carro não seja excedida, visto que todas as variáveis são binárias. Tem-se basicamente que em cada

dia cada veículo deve executar um movimento, podendo deslocar-se de i para j , ou de j para i , ou simplesmente ficar parado em um ponto i .

Já a restrição (47) é a mais complexa. Ela se refere à periodicidade que deve ser atendida em cada trecho. Na restrição (47) é possível que as variáveis p_{ijp} assumam valor 1, ou seja, um arco pode ser atrasado em um dia ao custo de ser punido com um custo PU_{ij} . Contudo, se a necessidade for de apenas uma passagem durante todo o horizonte de tempo, a restrição pode ser simplificada para apenas um dia p , $p = 1$, o que abrange todo o horizonte de tempo e a variável p_{ijp} pode ser descartada, como feito nos problemas aqui propostos.

Em (47) cada periodicidade dos arcos é analisada com a finalidade de verificar se a demanda nos intervalos de tempo são atendidas. Cada intervalo de $MP(x_{ij})$ dias é avaliado de forma que se o arco é atravessado ao menos uma vez, pelo menos uma variável x_{ijkp} assumirá valor 1, se isso não ocorrer obrigatoriamente a variável p_{ijp} implicará em uma punição na função objetivo.

A restrição (49) abaixo representa o que foi feito nesse trabalho sempre que os arcos necessitavam de apenas uma passagem durante todo o horizonte de tempo. É apenas uma simplificação com a finalidade de reduzir a quantidade de restrições a serem processadas para resolução do problema, se todos os arco que têm $MP(x_{ij}) = np$ estiverem em um conjunto R , a restrição que obriga o arco a ser atendido assume a seguinte forma

$$\sum_{k=1}^{nk} \left(\begin{array}{l} x_{ijk1} + x_{jik1} + x_{ijk2} + x_{jik2} + \dots \\ + x_{ijk,MP(x_{ij})} + x_{jik,MP(x_{ij})} \end{array} \right) \geq 1 \quad \forall [i, j] \in R \quad (49)$$

Lacomme, Prins e Ramdane-Chérif (2002a) consideram complicado trabalhar com restrições de espaço de tempo como está sendo trabalhado aqui, porém em seu trabalho o horizonte de tempo era muito curto e é considerado vantajoso utilizar uma metodologia designando uma combinação de dias para cada tarefa e resolver o problema.

Finalmente, a equação (48) determina que todas as variáveis sejam binárias.

4.3 SEMELHANÇAS E DIFERENÇAS ENTRE MODELOS

Os quatro modelos apresentados aqui procuram resolver o Problema de Roteamento em Arcos Capacitado e Periódico, porém cada modelo tem suas particularidades que serão detalhadas e comparadas nessa seção.

Avaliando inicialmente a função objetivo encontramos três situações distintas, os modelos propostos por Chu *et al.* (2004, 2005) visam reduzir o custo das viagens minimizando os deslocamentos. Monroy, Amaya e Langevin (2013) procuram maximizar o número de passagens em arcos que tem uma maior prioridade e o modelo aqui proposto procura minimizar dois objetivos que são: minimizar os deslocamentos (semelhante aos dois primeiros modelos) e, além disso, minimizar a quantidade de eventuais atrasos ao atender a demanda de algum arco.

Os modelos de Chu e o proposto trabalham com grafos não direcionados, diferente do modelo proposto por Monroy que trata de grafos direcionados. Os três modelos mais antigos tem um depósito sempre no nó 1 e o modelo proposto não tem um depósito explícito pelo fato de não precisar retornar ao final de um dia a algum ponto específico.

O número de variáveis de decisão e suas abordagens são distintos em cada problema, mas tem a semelhança de que devem mostrar o deslocamento que deve ser feito em qual período e por qual carro. Em Monroy (2013), é subentendido qual carro deve fazer o deslocamento e quando.

Chu (2004, 2005) e Monroy (2013) levam em conta o deslocamento servindo efetivamente um arco ou apenas atravessando. O modelo aqui proposto não aborda essa hipótese porque a duração dos deslocamentos, trabalhando e apenas atravessando, são muito próximas, justificando até o fato de que o local de pernoite se encontra pré-definido. A inclusão de mais variáveis tornariam o modelo mais complexo de ser resolvido.

Nos modelos de Chu a variável que mostra o deslocamento entre os arcos (apenas atravessar) é associada a variável que diz se o arco será servido ou não, ou seja, os deslocamentos são feitos e então relacionados a atividade de servir ou não um arco.

Já no modelo de Monroy (2013) as únicas duas variáveis de decisão do modelo não se relacionam da mesma forma. O fato de atravessar o arco ou servir o arco é diretamente associado a cada uma das variáveis. Sendo que quando um arco

é servido a variável é binária e quando o arco é atravessado é uma variável inteira, ou seja, o arco pode ser atravessado quantas vezes forem necessárias. Este fato é interessante pois, como o objetivo é maximizar o número de arcos servidos, ele basicamente não pode atender diversas vezes o arco prioritário, tendo sempre que variar arcos em suas rotas.

Chu (2004) tem mais uma variável que representa quando uma demanda, produção diária, é coletada e também relacionada com a variável referente ao arco ser servido ou não. Já em Chu (2005) tem-se a variável que designa em que combinação de dias um serviço será executado.

Tanto nos modelos de Chu como no proposto se trabalha apenas com variáveis binárias. Já o modelo de Monroy (2013) tem uma variável binária e uma inteira além de utilizar notação de arcos, diferente dos outros modelos que usam notação de nós.

Nenhum dos três modelos tem variáveis como as outras duas do modelo proposto, onde é possível atrasar um serviço ao custo de uma punição ou, simplesmente, deixar de executar uma tarefa por ser vantajoso.

As restrições de fluxo de veículos seguem a lógica básica dos primeiros modelos de roteamento em arcos (PCC e *CARP*). Tudo que chega em um ponto deve sair desse mesmo ponto. Porém existem algumas pequenas diferenças: nos modelos de Chu a restrição é feita para exclusivamente um dia e um carro, em Monroy a restrição é feita baseada em uma rota que implicitamente será feita em algum dia e por algum carro específico, levando em conta se o deslocamento será feito servindo ou apenas atravessando um arco.

Já a restrição de fluxo proposta tem características diferentes pelo fato de que ela é feita de um dia para o outro e é considerada a possibilidade do carro não efetuar nenhum deslocamento. Essa maneira como está estruturada a restrição de fluxo combinada com as restrições de demanda irão prevenir a formação de subciclos nesse modelo. É importante ressaltar ainda que o deslocamento do último período deve ocorrer de forma que termine no mesmo ponto em que foram iniciados os deslocamentos no primeiro dia.

A capacidade dos carros é tratada da mesma forma nos três primeiros modelos propostos: sempre que uma demanda é atendida ela ocupa um pouco da capacidade dos carros. No caso dos modelos de Chu a capacidade só é ocupada quando um arco é servido e em Monroy existe uma ocupação diferente da capacidade

para quando o arco é atravessado ou servido. A capacidade do veículo nunca pode ser excedida. A questão de que existe uma produção diária nos modelos de Chu está vinculada ao intervalo de tempo entre atendimentos.

No modelo proposto, a capacidade está implícita na distância que o carro pode percorrer durante o dia. Isso faz com que essas limitações sejam conhecidas a priori para poder resolver o problema.

As restrições de demanda estão baseadas em uma frequência nos modelos de Chu (2004) e Monroy (2013). Em Chu (2004) essa frequência é dada em intervalos de tempo e em Monroy (2013) ela deve ser maior que um valor pré-estabelecido. Chu (2005) apenas exige que cada arco seja atendido por uma combinação de dias.

No modelo proposto a demanda está implícita nas restrições de periodicidade vinculadas ao fato de que é necessário garantir que os carros façam algo em cada dia, podendo até mesmo ficar parado. Porque diferente dos outros modelos, não existe uma variável que diferencie o fato de atravessar e servir.

As restrições de periodicidade são as que tem mais diferenças entre os modelos.

Chu (2004) e o modelo proposto aqui tratam de periodicidades individuais para cada arco; são modelos que se assemelham nessa questão. Chu (2004) trabalha com um intervalo mínimo para que um arco seja servido duas vezes seguidas, bem como um intervalo máximo de dias que um arco pode ficar sem ser servido. O modelo proposto contabiliza apenas o valor máximo de dias que um arco pode ficar sem ser atravessado ou, ainda, o número de dias em que deve ocorrer ao menos uma passagem.

Chu (2005) tem a periodicidade implícita ao designar um arco para ser atendido em uma combinação de dias. Os arcos de Monroy tem suas periodicidades atendidas ao serem designados para uma rota que está vinculada ao intervalo de tempo em que a frequência mínima é estabelecida. É importante ressaltar que as frequências estão associadas a hierarquia de cada arco.

As restrições que evitam a formação de subciclos são muito complexas e crescem exponencialmente conforme a quantidade nós de um problema. No modelo proposto não ocorrem subciclos porque já está pré-determinado quantos movimentos cada carro é capaz de fazer e as restrições de fluxo obrigam que os carros sigam uma sequência de eventos, não podendo simplesmente aparecer em um ponto qualquer para formar um novo ciclo. Nos outros modelos as restrições para evitar a formação

de subciclos trabalham com subconjuntos de nós ou arcos que não podem assumir determinados valores.

O QUADRO 1 traz um resumo das principais características dos modelos aqui apresentados. Procurou-se listar características de cada modelo como: função objetivo, forma do grafo, como funcionam os depósitos, quantas variáveis de decisão são utilizadas e sua definição, a notação utilizada, como funcionam as restrições de fluxo, as restrições da capacidade dos carros, as restrições referentes às demandas, periodicidade, e como é evitada a formação de subciclos, bem como o número de conjuntos de restrições.

Características	Chu 2004	Chu 2005	Monroy 2013	Modelo Proposto
Função objetivo	Minimizar custos de deslocamento	Minimizar custos de deslocamento	Maximizar o número de passagens	Minimizar custos de deslocamento e atrasos
Grafos	Não direcionados	Não direcionados	Direcionados	Não direcionados
Depósito	Presente no nó 1	Presente no nó 1	Presente no nó 1	Sem depósito
Tipos de variáveis de decisão	Três binárias	Três binárias	Uma binária e uma inteira	Três binárias
Notação utilizada	Modelagem por nós	Modelagem por nós	Modelagem por arcos	Modelagem por nós
Restrições de fluxo	Por dia e por carro	Por dia e por carro	Para cada rota; Considera atravessar e servir	De um dia para o outro; Considera a possibilidade de folga
Restrições de capacidade dos carros	Devido ao acúmulo de tarefas executadas	Devido ao acúmulo de tarefas executadas	Devido ao acúmulo de tarefas executadas	Implícita
Restrições de demanda	Limitado a frequência de cada arco	Cada arco deve ser atribuído a uma combinação de dias	Deve ser maior ou igual a frequência de cada arco	Demanda vinculada a periodicidade, e todo carro deve fazer algo em um dia
Restrições de Periodicidade	Intervalo mínimo e máximo entre serviços	Implícitos na designação de uma combinação de dias	Implícitos na designação de uma rota e vinculados a frequência definida	Intervalo máximo sem passagens

QUADRO 1 – COMPARATIVO ENTRE OS MODELOS MATEMÁTICOS APRESENTADOS **continua**
 FONTE: O autor (2014)

Características	Chu 2004	Chu 2005	Monroy 2013	Modelo Proposto
Outras restrições	Restrições evitando subciclos e garantindo a presença do depósito nas rotas	Restrições evitando subciclos e garantindo a presença do depósito nas rotas	Restrições evitando subciclos e garantindo a presença do depósito nas rotas	Subciclo implícitas em uma sequência de eventos
Número conjuntos de restrições	13	8	6	4

QUADRO 1 – COMPARATIVO ENTRE OS MODELOS MATEMÁTICOS APRESENTADOS *conclusão*

FONTE: O autor (2014)

5 IMPLEMENTAÇÃO E RESULTADOS

5.1 IMPLEMENTAÇÃO

Para validar o modelo aqui proposto, inicialmente foram criados 10 pequenos problemas em que era possível encontrar uma solução de maneira intuitiva, ou ainda, partindo de uma solução desejada o problema era criado.

Essa criação foi feita definindo a quantidade de carros, o horizonte de tempo, a estrutura de cada grafo e as respectivas periodicidades de cada arco. Todos esses problemas encontram-se nos Apêndices, para ilustrar brevemente encontra-se um exemplo na Figura 8 abaixo:

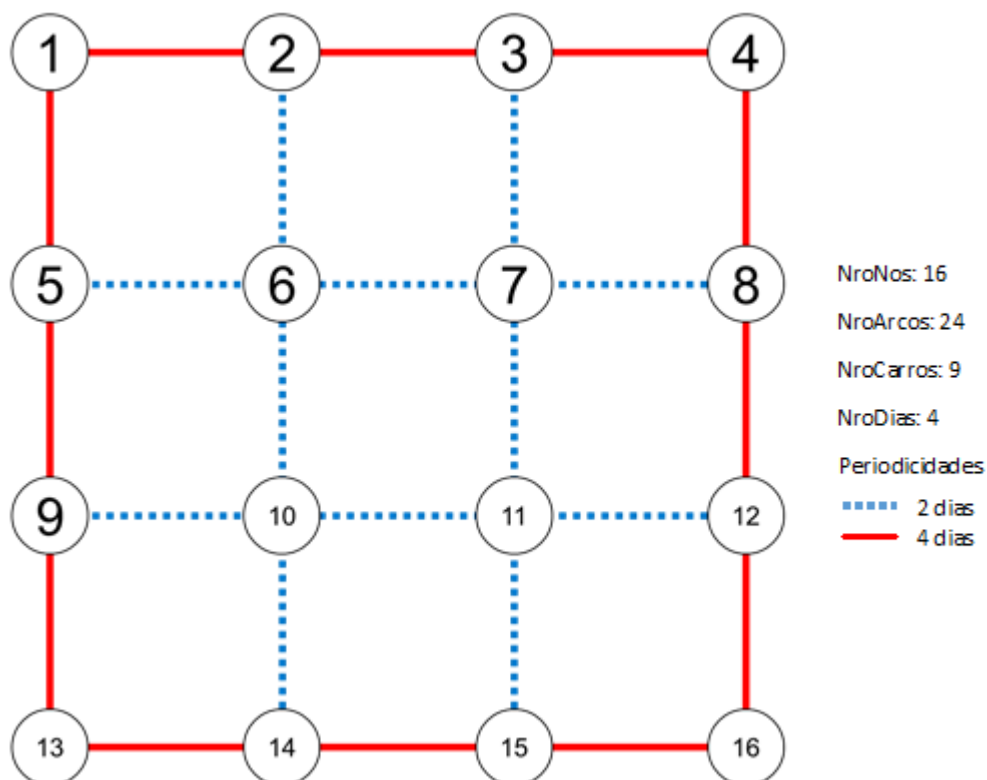


Figura 8 – Exemplo de problema gerado - arquivo gvb1

No exemplo da Figura 8 tem-se um problema com 16 nós, 24 arcos, 9 carros e um horizonte de tempo de 4 dias, e 2 periodicidades diferentes. Se analisada a figura podemos ver que ela é formada por 9 pequenos quadrados, uma das possíveis soluções seria um carro servindo cada quadrado de maneira sincronizada para que todas as periodicidades sejam atendidas, por exemplo se todos os carros partissem

do canto superior direito de “seu quadrado” e fizessem uma rota no sentido horário voltando ao mesmo ponto as periodicidades seriam atendidas. A mesma lógica foi utilizada para criação dos outros 9 problemas.

A fim de diversificar os problemas analisados, foram aproveitadas as estruturas dos grafos de 23 instâncias propostas no trabalho de Golden, DeArmon e Baker (1983), que são referência para muitos trabalhos como em Monroy, Amaya, e Langevin (2013). As instâncias foram inicialmente propostas para o Problema de Roteamento em Arcos Capacitado e adaptadas conforme as necessidades dos trabalhos para o *PCARP*.

Como existem algumas diferenças entre o trabalho de Monroy, Amaya, e Langevin (2013) e o problema aqui proposto, diferenças já apresentadas na seção anterior, as novas adaptações que foram feitas nas instâncias são as seguintes:

- Foram mantidos apenas os custos e as ligações entre os nós e os grafos foram considerados não direcionados;
- O número de periodicidades variaram entre 2, 3 ou 4;
- O número de carros que trabalham em cada grafo foi definido aleatoriamente entre 1 a 6;
- A necessidade de cada arco foi definida aleatoriamente, se um grafo tem 3 periodicidades, foram definidos arcos que necessitavam de 3 passagens, 2 passagens e 1 passagem ao longo do horizonte de tempo e assim sucessivamente;
- O número de necessidades de cada problema é a soma das necessidades de cada arco (número de atendimentos no arco);
- O horizonte de tempo é dado pelo número de necessidades do problema multiplicado por 1,2 arredondado para cima, tudo dividido pelo número de carros e também arredondado para cima;
- A periodicidade de cada arco é dado pelo horizonte de tempo, dividido por sua necessidade e arredondado para cima;
- Logo, a necessidade de cada arco também pode ser dada pela periodicidade de cada arco dividido pelo horizonte de tempo e arredondado para cima. Por exemplo, se a periodicidade de um arco é de 11 dias e o horizonte de tempo é de 20 dias, a necessidade desse arco é de 2 passadas durante o horizonte de tempo.

Para resolução do problema foi desenvolvido um programa em VisualBasic.NET em que a partir dos dados apresentados era gerado o modelo matemático do problema para ser rodado no *software* CPLEX 12.4 utilizando a configuração *default* do *solver*.

5.2 RESULTADOS PARA VALIDAÇÃO DO MODELO

As 10 instâncias criadas, os pequenos problemas, foram resolvidas pelo *software* CPLEX 12.4 em um computador Intel Core i7 com 2 GHz de processamento e 4 Gb de memória RAM e sistema operacional 64 bits. Os resultados foram satisfatórios e estão expostos na TABELA 1 bem como algumas características do problema.

TABELA 1 – RELAÇÃO DOS 10 PROBLEMAS GERADOS, PROPRIEDADES E RESULTADOS

Problema	Nós	Arestas	Period.	Carros	Neces.	Dias	Solução	Atrasos	Folgas	Tempo
gvb1	16	24	2	9	36	4	36*	0	0	2,40
gvb2	16	24	3	3	36	12	36*	0	0	364,85
gvb3	16	24	3	6	36	6	36*	0	0	338,99
gvb4	16	24	2	4	32	8	32*	0	0	69672,00
gvb5	6	7	2	3	18	6	18*	0	0	0,08
gvb6	6	7	2	1	8	8	8*	0	0	0,03
gvb7	12	24	2	7	42	6	42*	0	0	4,34
gvb8	8	12	2	2	16	8	16*	0	0	0,39
gvb9	14	23	3	10	60	6	60*	0	0	1,53
gvb10	11	12	2	1	16	24	24*	0	0	239,06

* Solução ótima

Nas informações da TABELA 1 encontram-se as características dos problemas estruturados como: número de nós, número de arestas, número de periodicidades (Period.), a quantidade de carros, o número de necessidades (Neces.), o horizonte de tempo (Dias), a solução obtida, quantos atrasos houveram, quantas folgas houveram e o tempo de processamento em segundos.

Nota-se que todas chegaram em soluções ótimas sem atrasos e sem folgas, o que já era esperado visto que esses problemas foram criados com esse objetivo. Um fator importante dos problemas é seu tempo de resolução, que oscilou entre 0,03 segundos até 69672 segundos.

Se comparados os problemas gvb1, gvb2, gvb3 e gvb4 vê-se que apesar da estrutura dos grafos serem parecidas, pequenas modificações nas propriedades dos problemas podem tornar o problema mais complexo de ser resolvido.

A Figura 9, ilustra o resultado do problema proposto na Figura 8 (arquivo gvb1), como era esperado cada um dos carros ficou responsável por atender um dos pequenos quadrados que compõe o grafo e de maneira sincronizada não permitiram que não ocorressem atrasos em nenhum dos arcos.

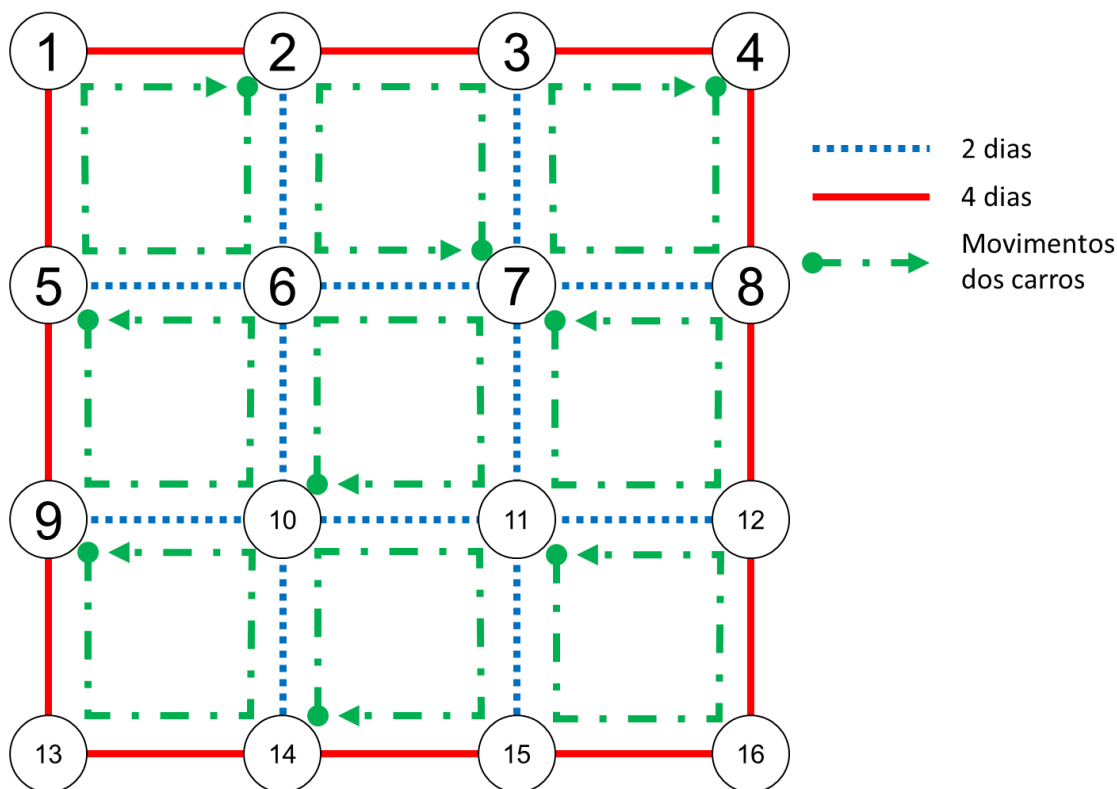


Figura 9 – Ilustração do resultado do problema gvb1

Na Figura 9 os 12 arcos que tem uma necessidade de duas passagens ao longo dos 4 dias com um periodicidade de 2 dias para ser servido são atendidos perfeitamente pois:

- Os arcos 2-6, 7-3, 6-10, 7-11, 14-10, 11-15 são atendidos nos dias 1 e 3
- Os arcos 5-6, 6-7, 8-7, 9-10, 10-11, 11-12 são atendidos nos dias 2 e 4

Os demais arcos que necessitam de apenas uma passagem durante os 4 dias também são servidos de maneira adequada.

As 23 instâncias propostas por Golden e Wong (1981) e adaptadas ao problema tratado nesse trabalho, foram modeladas e executadas também no *software* CPLEX 12.4, porém em computadores com processador Intel Core i5 com processamento de 3,1 GHz com 8 Gb de memória RAM em um sistema operacional de 64 bit.

Como esses problemas apresentavam características que tornam os problemas mais complexos de serem resolvidos, foi estabelecido um tempo limite para obtenção de uma solução ótima que foi de 24 horas ou 86400 segundos.

As soluções bem como as características dos problemas estão apresentadas na TABELA 2 abaixo:

TABELA 2 – RELAÇÃO DOS 23 PROBLEMAS ADAPTADOS, PROPRIEDADES E RESULTADOS

Problema	Nós	Arestas	Period.	Carros	Neces.	Dias	Solução	Atrasos	Folgas	Tempo
gdb1	12	22	2	1	32	38	4450	4	0	39000 ¹
gdb2	12	26	3	1	44	53	11622	11	1	70423 ¹
gdb3	12	22	2	2	31	19	340	0	3	36906 ¹
gdb4	11	19	3	2	34	21	3555	3	1	49595 ¹
gdb5	13	26	4	2	53	32	1750	1	1	86400 ²
gdb6	12	22	2	3	32	14	456	0	3	86400 ²
gdb7	12	22	3	3	37	45	518	0	5	55767 ¹
gdb8	27	46	4	3	67	27	12395	12	0	86400 ²
gdb9	27	51	3	4	78	24	13454	13	2	86400 ²
gdb10	12	25	4	4	70	21	734	0	6	86400 ²
gdb11	22	45	2	1	50	60	64437	64	2	31510 ¹
gdb12	13	23	3	1	33	40	536	0	0	86400 ²
gdb13	10	28	2	2	33	20	620	0	5	86400 ²
gdb14	7	21	3	2	32	20	145*	0	7	4,21
gdb15	7	21	4	2	48	29	140	0	7	63135 ¹
gdb16	8	28	2	3	40	24	182	0	28	35301 ¹
gdb17	8	28	3	3	52	21	168	0	4	77693 ¹
gdb18	9	36	4	3	81	33	415	0	6	86400 ²
gdb19	8	11	3	4	29	12	125	0	17	47566 ¹
gdb20	11	22	4	4	53	16	2280	2	1	67903 ¹
gdb21	11	33	2	5	46	12	212	0	10	38181 ¹
gdb22	11	44	4	5	88	22	433	0	17	86400 ²
gdb23	11	55	2	6	82	17	336	0	15	86400 ²

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

² Problemas que rodaram até o tempo limite estabelecido

* Solução ótima

A TABELA 2 apresenta as mesmas informações contidas na TABELA 1 e já mencionadas anteriormente. Porém esses problemas listados são de maior complexidade que os primeiros já apresentados.

Analisando a TABELA 2 percebe-se que apenas um dos 23 problemas obteve solução ótima, o problema gdb14 que continha apenas 7 nós, e 21 arestas que tiveram suas 32 necessidades atendidas por 2 carros, o que significa que cada uma dos 3 tipos diferentes de periodicidades foram respeitadas.

Apesar da limitação de tempo para obtenção da solução ótima verificou-se que 12 dos 23 problemas esgotaram a memória do computador antes mesmo de exceder o tempo estabelecido, então a solução considerada foi o *bound* com melhor resultado até o momento.

De maneira geral, os problemas obtiveram soluções com poucos atrasos exceto o problema gdb11 onde havia apenas um carro para atender 50 necessidades em 60 dias, onde então houveram 64 punições que influenciaram diretamente na função objetivo. Foi adotada uma política de punir em 1000 a função objetivo sempre que um arco atrasava enquanto o custo para atravessar um arco não passavam de dezenas, na maior parte das vezes.

As folgas têm grande importância na resolução do problema, 20 dos 23 problemas apresentaram folgas em seus itinerários. Pode parecer contraditório folgar e ser punido no mesmo problema (fato que ocorre em 6 dos 23 problemas), mas a folga pode evitar outras punições, além de que como é necessário que o trajeto termine no mesmo ponto que começou, poderia não ser possível se o carro não fizesse paradas.

Os carros folgando podem significar que o trajeto pode ser feito em menos tempo do que o horizonte de tempo definido. Por exemplo, se temos um problema com um único carro e esse carro folgou 10 vezes, o horizonte de tempo poderia ser reduzido em 10 dias e as folgas seriam eliminadas podendo, assim, fazer um ciclo menor.

Como é um problema de manutenção preditiva, seria interessante a redução do horizonte de tempo eliminando as folgas, pois poderia ser feito mais trabalho em menos tempo com as periodicidades respeitadas e podendo até reduzi-las. Vale ressaltar que o problema proposto por Monroy, Amaya, e Langevin (2013) procura maximizar o número de passagens nos arcos.

De maneira geral, quando se tem muitas folgas em um problema pode-se afirmar que alguns recursos podem estar ociosos e poderiam ser aproveitados de maneira melhor. Quando existem muitos atrasos é necessário aumentar os recursos para atender os serviços.

Nada impediria que um arco fosse punido durante todo o horizonte de tempo, exceto os arcos com necessidade de 1 passagem que foram modelados de forma que fossem sempre atendidos para reduzir o número de restrições. Se algum arco fosse

punido tantas vezes seria relevante avaliar a quantidade de carros disponíveis e procurar aumentar a frota de veículos.

5.3 COMPARAÇÃO COM UMA ADAPTAÇÃO DO MODELO

Para efeitos comparativos, o problema proposto na Figura 10 foi adaptado à modelagem mais recente da literatura proposta por Monroy, Amaya, e Langevin (2013).

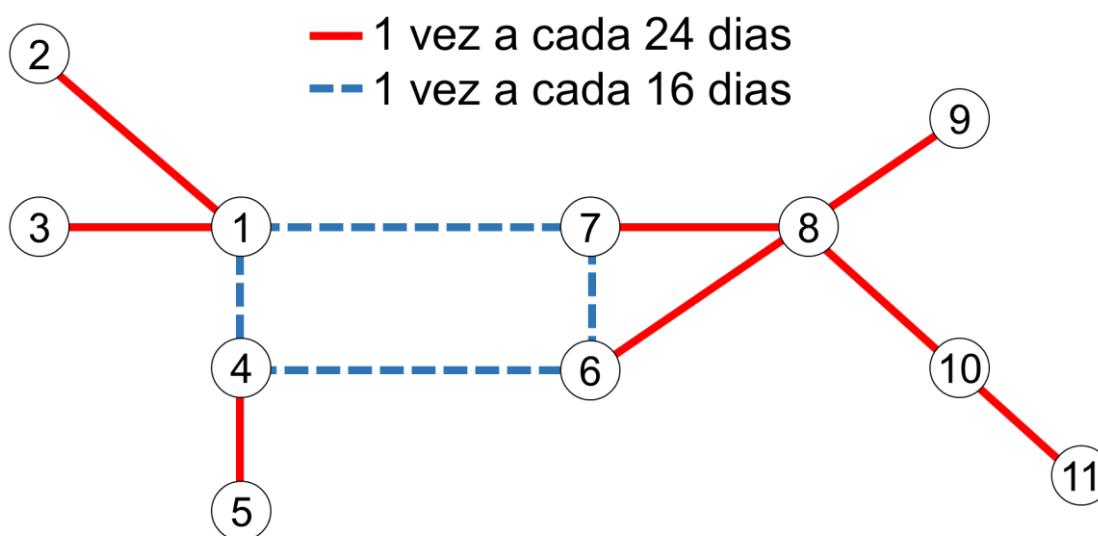


Figura 10 – Proposta de problema para adaptação ao modelo proposto por Monroy

Contudo, como os problemas abordados são diferentes nos dois modelos, foram necessárias algumas adaptações para ver a solução que o modelo adaptado apresentaria. A adaptação feita teve as seguintes considerações:

- Foi considerado um horizonte de tempo de 24 dias que foram subdivididos em 2 grupos de dias baseado nos dias de cada periodicidade. Então obteve-se o primeiro grupo envolvendo os dias entre 1 e 16, e o outro grupo com os dias 17 a 24
- O grafo foi considerado não direcionado.
- A função objetivo foi definida para minimizar o custo das viagens
- Como a modelagem proposta por Monroy necessita de uma depósito, foi escolhido o ponto 1 pois ele está conectado a arcos com maior número de necessidades (1-7, 1-4)

- Foram definidas duas classes de arcos. A primeira classe, Classe 1, necessita de 2 passagens no horizonte de tempo em w_1, w_2 que são formados respectivamente, pelos dias 1 a 16 e 17 a 24. A Classe 2 necessita apenas uma passagem no intervalo w_3 que engloba todo o horizonte de tempo. A Figura 11 abaixo exemplifica as classes:

	Dias 1 - 16	Dias 17 - 24
Classe 1	w_1	w_2
Classe 2	w_3	

Figura 11 – Sub-horizontes de tempo e classes criadas para adaptação ao modelo de Monroy

- Fez-se necessário a criação de duas rotas, a primeira com uma capacidade de 16 que englobava os 16 primeiros dias, e outra com capacidade de 8 que englobava os dias 17 até 24, que no futuro formarão uma rota única que deve atender de maneira aproximada as periodicidades
- As restrições que evitam subciclos foram simplificadas apenas considerando o os subconjuntos de arcos que fizessem com que o depósito sempre fosse incluído nas rotas. No trabalho proposto por Monroy, era feita uma simplificação semelhante sendo que se a rota apresentasse subciclos então eram acrescentadas restrições que evitavam as mesmas

As rotas ótimas obtidas foram as seguintes: 1 - 7 - 6 - 8 - 9 - 8 - 10 - 11 - 10 - 8 - 6 - 4 - 1 - 2 - 1 - 3 - 1 e 1 - 7 - 8 - 7 - 6 - 4 - 5 - 4 - 1, fazendo uma correspondência ao problema real, adaptamos cada deslocamento a um dia. De maneira geral as variáveis que assumiram valor 1 estão listadas na TABELA 3.

Observa-se que a solução obtida é interessante, porém não satisfaria a resolução do problema, pois ao avaliar os arcos que tem necessidade maior que 1 passagem ao longo do horizonte de tempo ocorre uma falha no atendimento, percebe-se que:

- O arco 1-7, destacado em amarelo na TABELA 3, é atendido nos dias 1 e 17, dentro do limite máximo da periodicidade pois o arco é atravessado exatamente 16 dias depois da primeira passagem. Contando a característica de ser um problema cíclico ele seria atravessado no dia 25 novamente que seria após 8 dias após a última passagem.
- O arco 6-4, em azul, é atendido nos dias 11 e 21, e futuramente no 35 também dentro das periodicidades
- O arco 1-4, em verde, é atendido nos dias 12 e 24, e futuramente no 36 também dentro das periodicidades

- O arco 7-6, em vermelho, é atendido nos dias 1 e 20, o que não atenderia a periodicidade pois entre os dias 2 a 19 não é atravessado, teria um novo atendimento no dia 25 porém extrapolou da primeira vez

TABELA 3 – RESULTADO DA ADAPTAÇÃO DE PROBLEMA AO MODELO DE MONROY (2013)

Variável	Ponto de partida	Ponto de chegada	Rota	Correspondência ao dia	Valor
X_1_7_1	1	7	1	1	1
X_7_6_1	7	6	1	2	1
X_6_8_1	6	8	1	3	1
X_8_9_1	8	9	1	4	1
Y_9_8_1	9	8	1	5	1
X_8_10_1	8	10	1	6	1
X_10_11_1	10	11	1	7	1
Y_11_10_1	11	10	1	8	1
Y_10_8_1	10	8	1	9	1
Y_8_6_1	8	6	1	10	1
X_6_4_1	6	4	1	11	1
X_4_1_1	4	1	1	12	1
Y_1_2_1	1	2	1	13	1
X_2_1_1	2	1	1	14	1
X_1_3_1	1	3	1	15	1
Y_3_1_1	3	1	1	16	1
X_1_7_2	1	7	2	17	1
X_7_8_2	7	8	2	18	1
Y_8_7_2	8	7	2	19	1
X_7_6_2	7	6	2	20	1
X_6_4_2	6	4	2	21	1
X_4_5_2	4	5	2	22	1
Y_5_4_2	5	4	2	23	1
X_4_1_2	4	1	2	24	1

De fato, no modelo proposto, se não fosse possível atender os arcos dentro da periodicidade, ocorreria uma punição no arco 7-6 por atrasos e a solução seria aceita, porém a rota ótima obtida com o modelo aqui proposto é a seguinte: 6 - 4 - 1 - 7 - 8 - 10 - 11 - 10 - 8 - 7 - 6 - 4 - 5 - 4 - 1 - 2 - 1 - 3 - 1 - 7 - 6 - 8 - 9 - 8 - 6, ou seja, as variáveis assumiram os seguintes valores, segundo a TABELA 4.

Com o resultado é possível perceber que não ocorreu atraso em nenhum arco pois:

- O arco 6-4, destacado em azul na TABELA 4, foi servido nos dias 1 e 11, e seria servido no dia 25 novamente. Uma diferença de 10 e 14 dias respectivamente.

- O arco 4-1, em verde, nos dias 2 e 14, e 26 no início do próximo ciclo. Diferenças de 12 dias em ambos os casos.
- O arco 1-7, em amarelo, nos dias 3, 19 e 27. Diferença de 16 dias e 8 dias respectivamente
- O arco 7-6, em vermelho, nos dias 10, 20 e 34. Diferença de 10 e 14 dias respectivamente

TABELA 4 – RESULTADO OBTIDO COM O MODELO PROPOSTO

Variável	Ponto de partida	Ponto de chegada	de Dia	Carro
X_6_4_1_1	6	4	1	1
X_4_1_1_2	4	1	2	1
X_1_7_1_3	1	7	3	1
X_7_8_1_4	7	8	4	1
X_8_10_1_5	8	10	5	1
X_10_11_1_6	10	11	6	1
X_11_10_1_7	11	10	7	1
X_10_8_1_8	10	8	8	1
X_8_7_1_9	8	7	9	1
X_7_6_1_10	7	6	10	1
X_6_4_1_11	6	4	11	1
X_4_5_1_12	4	5	12	1
X_5_4_1_13	5	4	13	1
X_4_1_1_14	4	1	14	1
X_1_2_1_15	1	2	15	1
X_2_1_1_16	2	1	16	1
X_1_3_1_17	1	3	17	1
X_3_1_1_18	3	1	18	1
X_1_7_1_19	1	7	19	1
X_7_6_1_20	7	6	20	1
X_6_8_1_21	6	8	21	1
X_8_9_1_22	8	9	22	1
X_9_8_1_23	9	8	23	1
X_8_6_1_24	8	6	24	1

Não houve nenhuma punição e o modelo atingiu os objetivos propostos de uma forma melhor que de Monroy. Outro fator considerável é que o modelo de Monroy teve uma solução mais rápida computacionalmente para esse experimento. Ao rodar os modelos no *software* CPLEX 12.4 em um computador Intel Core i7 com 2 Ghz de processamento e 4 Gb de memória RAM e sistema operacional 64 bits, o modelo de Monroy foi resolvido em menos de 1 segundo (importante ressaltar da simplificação feita para evitar subciclos e que nenhum subciclo foi formado na primeira tentativa)

enquanto o modelo aqui proposto demora pouco menos de 30 segundos para chegar em uma solução com o valor ótimo. Porém, esse valor só é confirmado após 330 segundos. Considerando que é uma solução para um planejamento a longo prazo, esses valores se tornam irrelevantes ao comparar com 24 dias.

5.4 APLICAÇÃO A UM PROBLEMA REAL

A modelagem proposta pode solucionar problemas reais. a Figura 12 ilustra um problema real de uma empresa ferroviária no Brasil na qual deve ser feito um planejamento de 180 dias que se repetem formando um ciclo. Dois carros devem percorrer os trilhos fazendo manutenção preditiva para evitar futuros problemas.

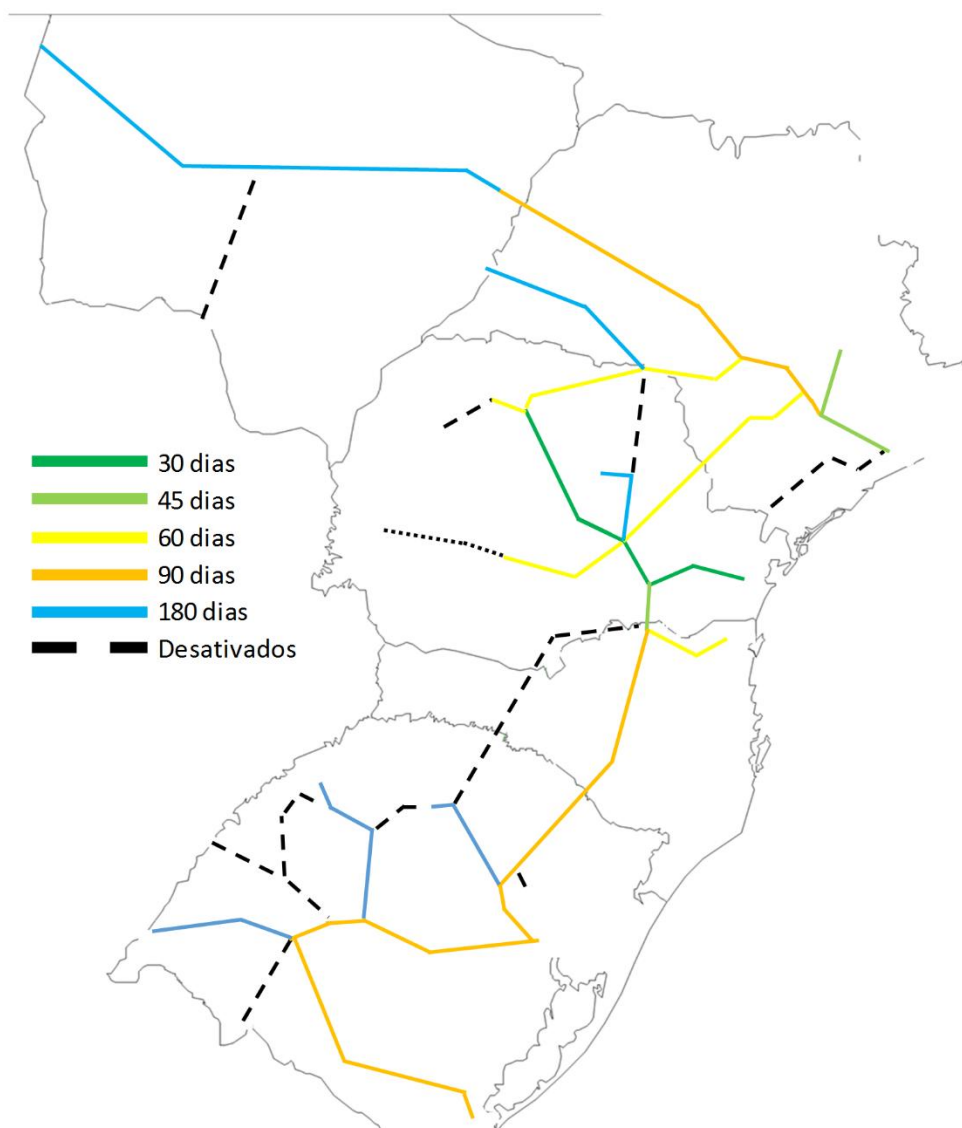


Figura 12 – Representação de um problema real

Na representação nota-se que há 5 tipos de periodicidades diferentes e, como o Brasil é um país de dimensões continentais, as distâncias entre um ponto e outro são relativamente grandes, podendo demorar muitos dias para ir de um ponto a outro. O mapa pode ser representado por um grafo, nele existem diversas cidades que são atravessadas pelas arestas e seriam candidatas a pontos do mesmo.

Um problema tão grande como o representado pela Figura 12 ao ser modelado exigiria um grande número de variáveis, pois são 180 dias e um número muito grande de arcos.

Alguns dos resultados computacionais aqui apresentados mostram que a memória do computador se esgota conforme a complexidade do problema e a solução obtida pode não ser de agrado. Afim de resolver o problema ilustrado, foi realizada uma simplificação no grafo que está ilustrado na Figura 13.

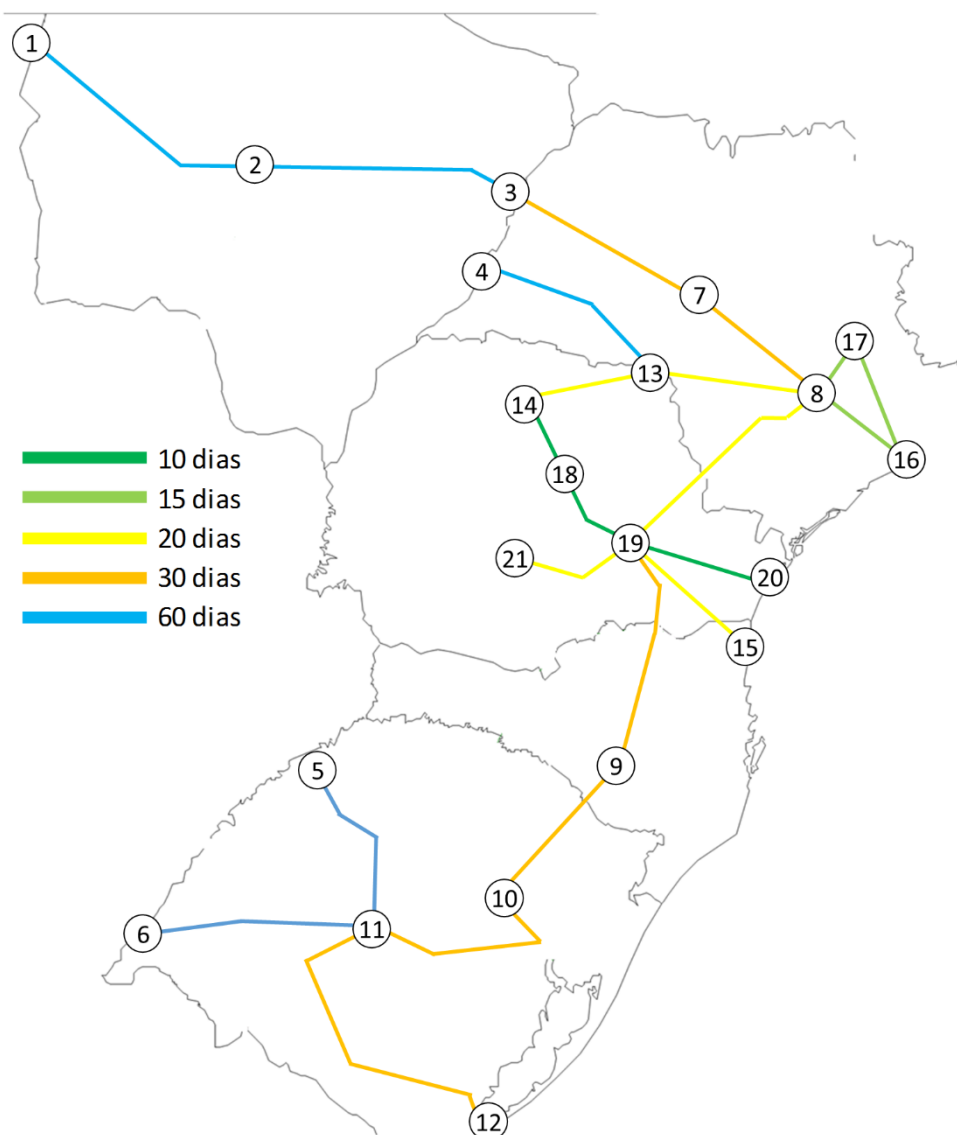


Figura 13 – Representação da simplificação do problema real

A simplificação representada na Figura 13 foi feita da seguinte maneira: as periodicidades foram divididas por 3 e os pontos foram distribuídos de forma que a duração para sair de um ponto e chegar a outro dura em média 3 dias. Essa simplificação reduz consideravelmente o número de variáveis e restrições, porém resulta em uma perda de detalhamento dos movimentos diários. Ao final quando obter-se um resultado, é simples transportá-lo para o problema real.

A solução obtida será representada a partir da Figura 14 até a Figura 19. Cada figura tem representada a rota dos dois carros para 10 dias de execução. Foi escolhida a representação de 10 dias por ser a menor periodicidade a ser atendida.

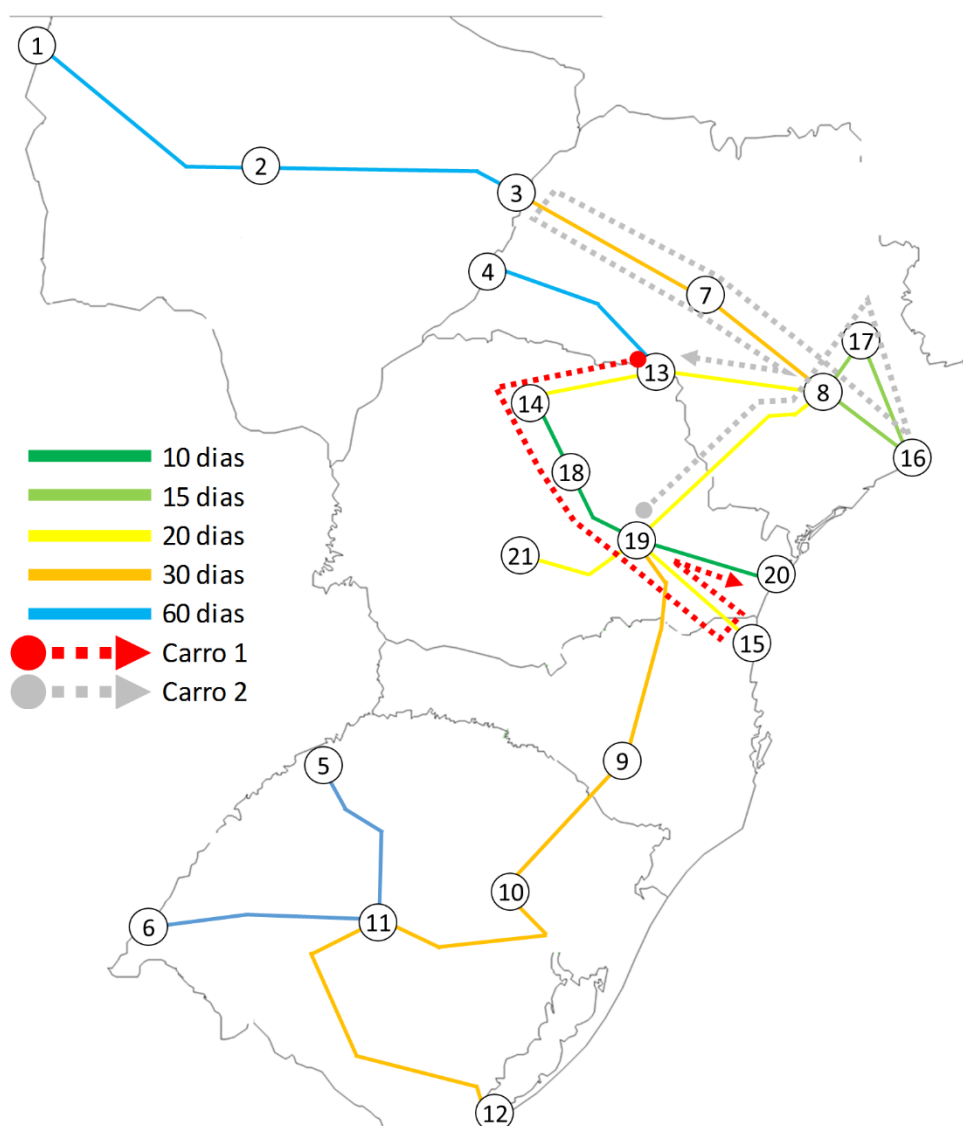


Figura 14 – Rotas dos carros 1 e 2 entre os períodos 1 e 10

Na Figura 14 temos representados os movimentos que os carros devem fazer entre o dia 1 e o dia 10, a solução encontrada foi a seguinte:

Carro 1: 13 - 14 - 18 - 19 - 15 - 19 - 20 - 20 - 20 - 20 - 20

Carro 2: 19 - 8 - 17 - 16 - 8 - 8 - 7 - 3 - 7 - 8 - 13

O que significa que o Carro 1 ficou parado no ponto 20 nos dias 7,8,9,10, e o Carro 2 no dia 5 ficou parado no ponto 8.

O problema foi modelado e executado no CPLEX 12.4 com a mesma configuração dos problemas *gdb* com uma limitação de 24 horas no tempo de processamento. A solução obtida foi completamente aceitável onde não ocorreu nenhuma punição ou atraso, apenas algumas folgas.

Em algumas das ilustrações é possível notar que existem menos de 10 deslocamentos ao longo do horizonte de tempo, o que significa que houveram folgas.

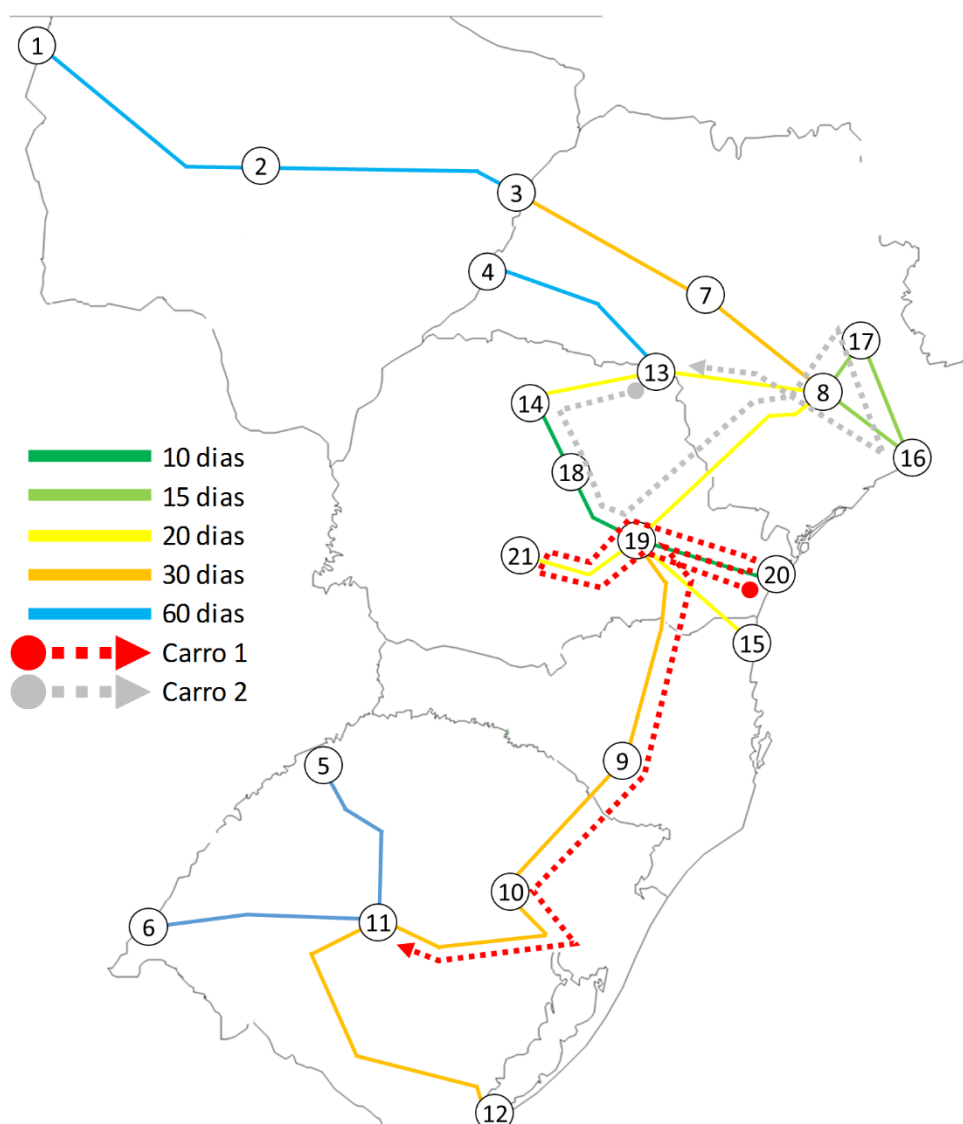


Figura 15 – Rotas dos carros 1 e 2 entre os períodos 11 e 20

Na Figura 15 representados os movimentos entre os dias 11 e 20 os quais foram:

Carro 1: 20 - 19 - 19 - 19 - 21 - 19 - 20 - 19 - 9 - 10 - 11

Carro 2: 13 - 14 - 18 - 19 - 8 - 17 - 16 - 16 - 16 - 8 - 13

Onde o Carro1 ficou parado nos dias 12 e 13 no ponto 19, e o Carro 2 no ponto 16 nos dias 17,18.

Nota-se que a maior parte das necessidades estão localizadas na região central do mapa, representados pelos estados do Paraná e São Paulo e as demais regiões tem menos necessidades de passagens.

Como são dois carros o que se espera como resultado é que enquanto um esteja cuidando da região mais “crítica” (com maior número de necessidades), outro carro se desloque para atender as regiões mais afastadas.

Enquanto o Carro 1 em vermelho se deslocou para o sul do mapa, o Carro 2 em cinza manteve sua rota na região central.

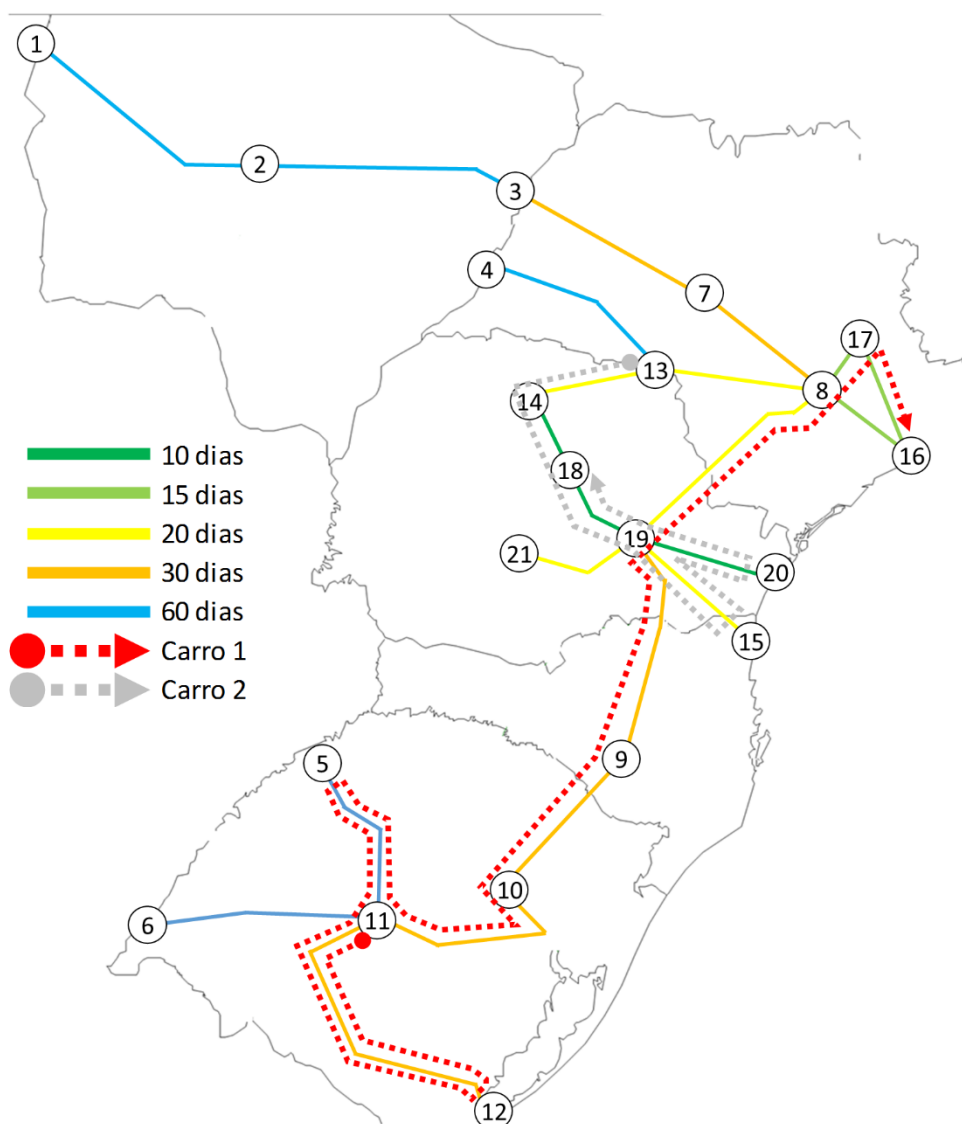


Figura 16 – Rotas dos carros 1 e 2 entre os períodos 21 e 30
Na Figura 16 os movimentos entre os dias 21 e 30 foram:

Carro 1: 11 - 12 - 11 - 5 - 11 - 10 - 9 - 19 - 8 - 17 - 16

Carro 2: 13 - 14 - 18 - 19 - 15 - 19 - 19 - 20 - 20 - 19 - 18

O carro 2 ficou parado nos pontos 19 e 20 nos dias 26 e 28 respectivamente. Analisando graficamente parece que o Carro 1 se deslocou muito mais do que o Carro 2, de fato é verdade, mas não apenas pelo fato de duas folgas do Carro 2.

Vale lembrar que os deslocamentos são feitos baseados na duração de tempo para se deslocar de um ponto ao outro e não na sua distância. A distância interfere, porém o tempo tem uma relação direta com a velocidade em que cada trecho pode ser servido, isso faz com que em algumas regiões com tráfego menor de veículos e que necessitam de menos manutenção preditiva sejam atravessados mais rápido que outros. Ainda na Figura 16 foram finalizados os serviços dos 30 primeiros dias, a metade do horizonte de tempo.

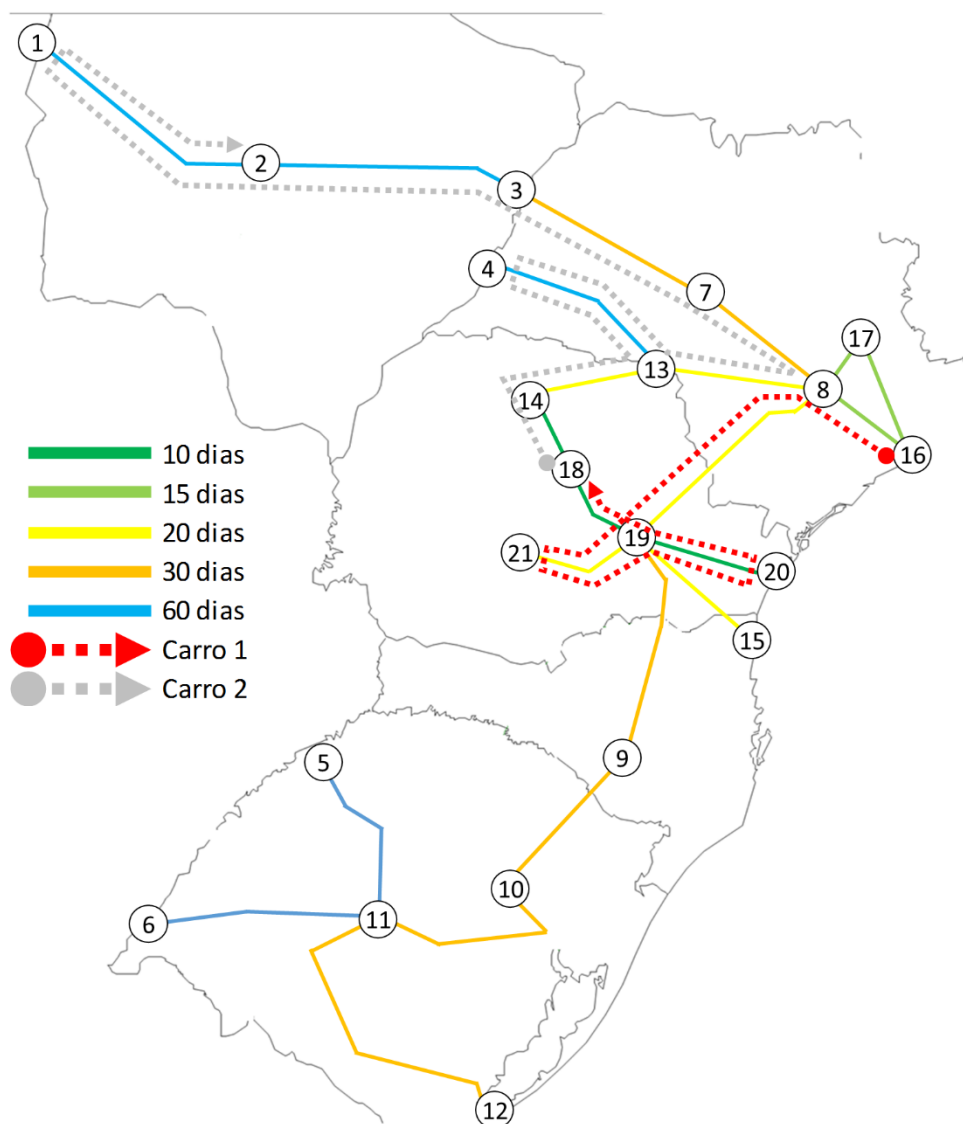


Figura 17 – Rotas dos carros 1 e 2 entre os períodos 31 e 40

Na Figura 17 entre os dias 31 e 40 temos os movimentos:

Carro 1: 16 - 8 - 19 - 21 - 21 - 21 - 21 - 19 - 20 - 19 - 18

Carro 2: 18 - 14 - 13 - 4 - 13 - 8 - 7 - 3 - 2 - 1 - 2

O Carro 1 ficou parado no ponto 21 entre os dias 34 a 36. É o início da segunda metade das avaliações dos trilhos, o que significa que os arcos com periodicidade de 30 devem começar a ser medidos pela segunda vez pelo menos, e é o que de fato acontece pois, por exemplo o arco 3-7 que foi medido logo no início da rota, tem novamente medições feitas no início dessa segunda metade do horizonte de tempo.

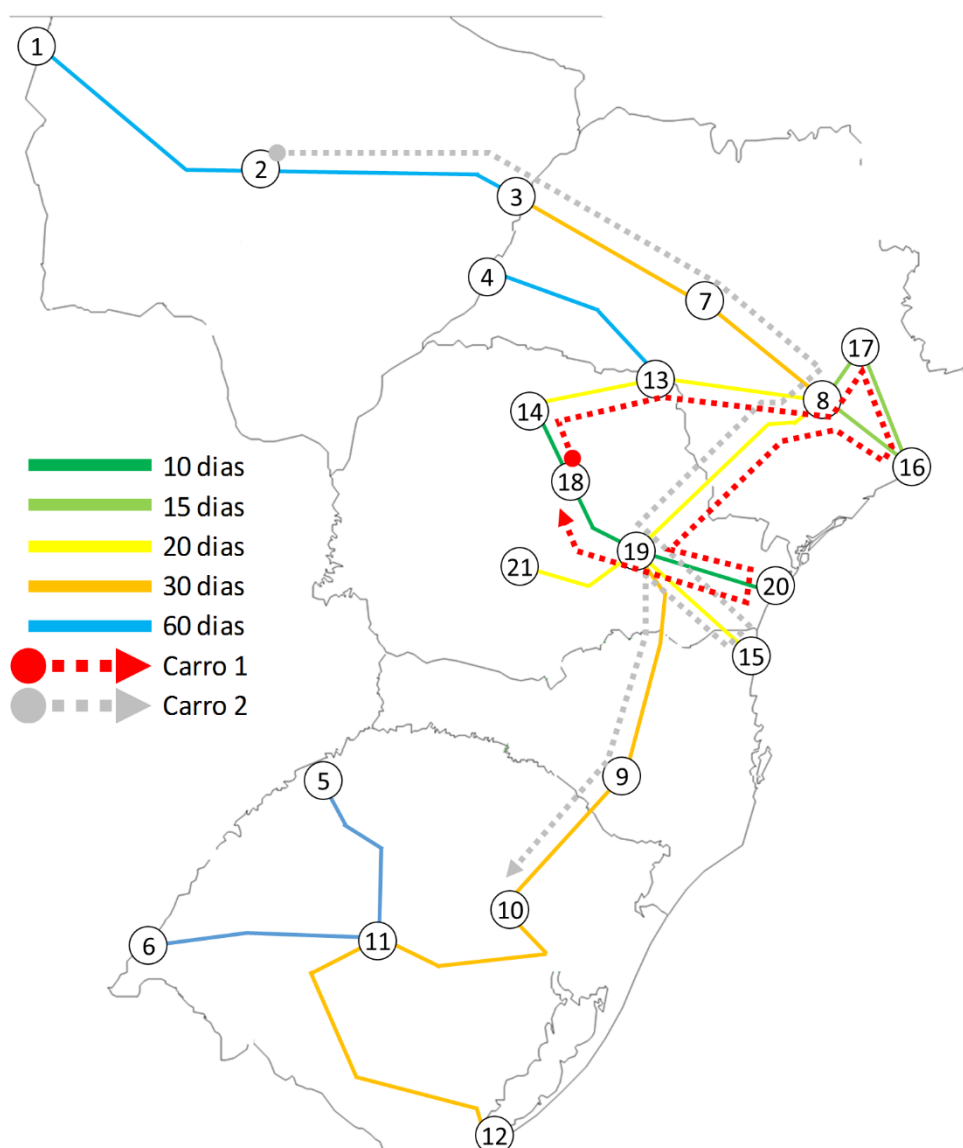


Figura 18 – Rotas dos carros 1 e 2 entre os períodos 41 e 50

Os movimentos entre os dias 41 e 50 representados na Figura 18 são:

Carro 1: 18 - 14 - 13 - 8 - 17 - 16 - 8 - 19 - 20 - 19 - 18

Carro 2: 2 - 3 - 7 - 8 - 19 - 15 - 15 - 15 - 19 - 9 - 10

Onde o Carro 2 ficou parado no ponto 15 nos dias 46, 47. Novamente percebe-se que um carro fica concentrado na região centralizada enquanto outro atravessa o mapa indo aos pontos com menor necessidade.

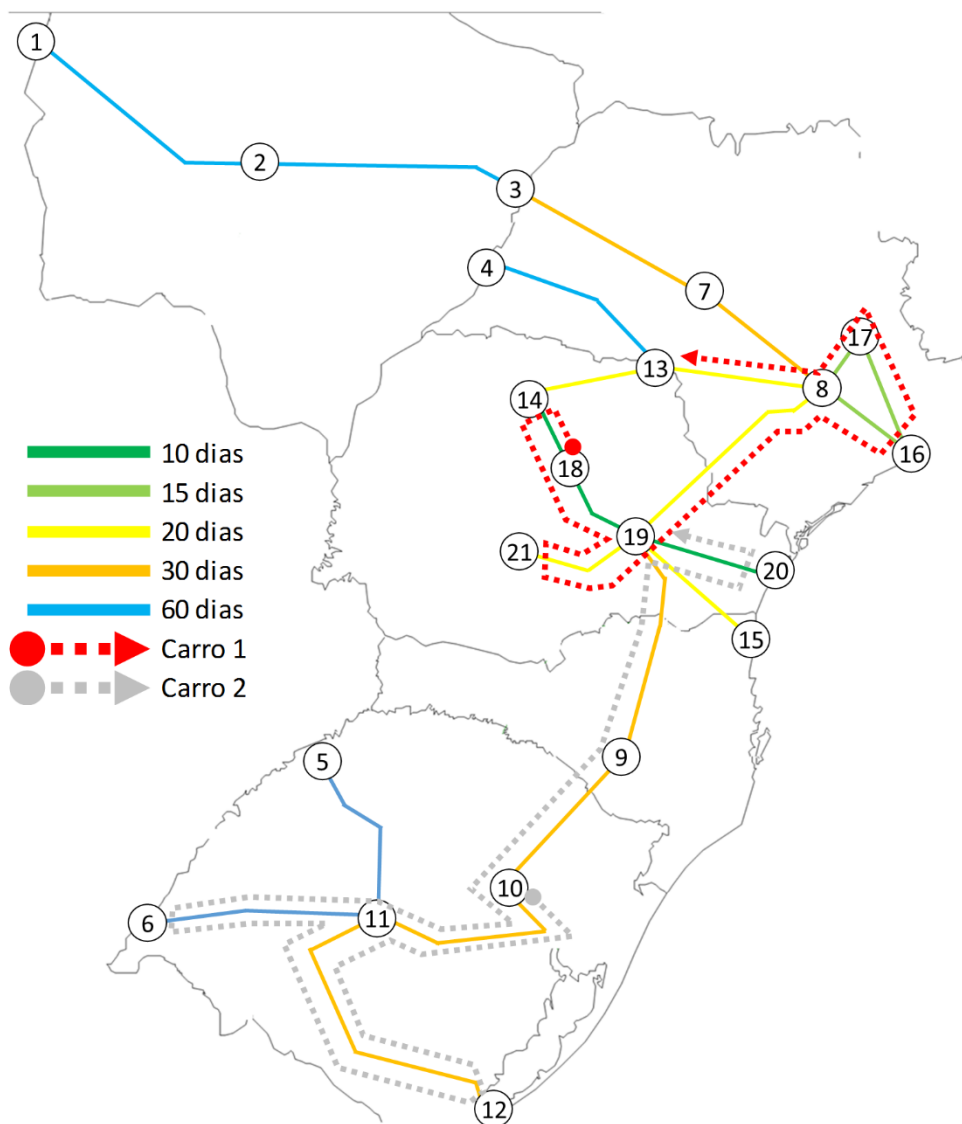


Figura 19 – Rotas dos carros 1 e 2 entre os períodos 51 e 60

A Figura 19 representa os últimos 10 movimentos dos carros que foram feitos entre os dias 51 a 60:

Carro 1: 18 - 14 - 18 - 19 - 21 - 19 - 8 - 16 - 17 - 8 - 13

Carro 2: 10 - 11 - 12 - 11 - 6 - 11 - 10 - 9 - 19 - 20 - 19

O Carro 1 iniciou seu trajeto e finalizou no ponto 13 e o mesmo acontece com o Carro 2 que foi no ponto 19. Ao longo de todos os trajetos é possível perceber que todas as periodicidades são aceitas.

Os arcos em verde, que tem a maior periodicidade, estão sendo atravessados em todas as figuras, os arcos em azul são atravessados apenas uma vez. Nota-se

que os arcos em azul são bem intercalados no horizonte de tempo, por exemplo, na região sul existem basicamente dois arcos em azul, como a região é visitada duas vezes, cada vez que ocorre a visita um dos arcos é visitado.

Ao todo o carro 1 folgou 9 vezes e o carro 2 folgou 7 vezes, o que representa na realidade representam 27 dias e 21 dias respectivamente, essas folgas são extremamente importantes no problema real pois podem ocorrer eventuais paradas nas máquinas por falha ou manutenção e as folgas acabariam compensando essas paradas.

A simplificação realizada permitiu que o problema apresentasse uma solução muito boa onde a segurança da via seria mantida através da manutenção preditiva. Essa simplificação acaba agrupando os arcos em periodicidades próximas e um analista deve fazer a correspondência ao problema real desagrupando os arcos.

6 CONCLUSÕES

Os problemas de roteamento em arcos têm grande aplicabilidade e podem ser adaptados conforme as necessidades. Abrangem um campo de atuação ainda não tão explorado como os problemas de roteamento por pontos. Entre os problemas reais que podem ser resolvidos dessa forma encontram-se: coleta de lixo, manutenção, monitoramento e prevenção de rodovias, ferrovias, linhas elétricas, remoção de neve de ruas, entre outros.

No que se trata do *PCARP* é um problema ainda pouco explorado na literatura, é um derivado dos problemas *CARP* que tem uma natureza *NP-Hard* e quanto mais restrições tiver, seja quanto à quantidade, capacidade, ou tempo, mais difícil é encontrar uma solução. Porém é um problema que quando se encontram soluções podem trazer muitos ganhos para as empresas que o utilizarem.

É muito relevante fazer um planejamento de serviços que não se restrinjam a apenas um dia e abranjam um horizonte de tempo mais longo, ou seja, é possível planejar bem todos os movimentos a serem tomados a fim de evitar que ocorram problemas ou que seja feito um “balé” sincronizado para atender todas as necessidades que já são conhecidas a priori e que virão a ocorrer no futuro. Planejar visando um longo horizonte de tempo é, muitas vezes, necessário para manter um sistema estável e sem falhas.

A evolução dos *ARPs* ao longo do tempo é evidente e a cada período apresentam-se novas restrições e novas formas de modelar e tratar cada problema. Aqui foram apresentados 4 modelos para o *PCARP*, cada um com uma finalidade e um objetivo que o tornam diferentes entre si, mas que tratam do mesmo problema.

O que torna o modelo proposto diferente dos demais é a flexibilidade em poder atrasar algum arco ao custo de uma punição ou, simplesmente, parar o carro para atender melhor às necessidades. Além da maneira como as restrições de fluxo de veículos foram feitas e a forma com que os subciclos são evitados.

O problema aqui proposto foi modelado e executado no CPLEX 12.4 em computadores com processadores Intel Core i5 e i7 com memórias de 8 Gb e 4 Gb, respectivamente, e em um sistema operacional de 64 bits. Notou-se que instâncias pequenas obtiveram solução exata em tempos aceitáveis conforme TABELA 1 (p. 60). Já problemas com uma estrutura mais complexa podem ser processados até a

memória do computador se esgotar e não se obtêm uma solução ótima (TABELA 2 p. 62). Porém, se o programa for limitado a 24 horas de iterações conseguem-se soluções aceitáveis sem que ocorram punições.

Tanto tempo de execução é compensado pelo fato de se estar planejando para uma grande quantidade de dias que se repetirão formando ciclos. Assim, 24 horas de execução não são relevantes quando se trata de um planejamento que será feito para meses.

Algumas adaptações que podem ser feitas decorrem da necessidade de que o carro trabalhe continuamente sem parar, neste caso não é necessário utilizar as variáveis f_{ikp} , ou, se não for permitido que atrasos ocorram, eliminam-se as variáveis p_{ijp} . Entretanto, essas adaptações podem fazer com que o problema torne-se infactível e, como foi explicado, essas variáveis permitem ao problema uma certa flexibilidade. Como, por exemplo, se houverem muitas folgas, é possível reduzir o horizonte de tempo de forma que os arcos sejam ainda mais monitorados em menos tempo.

Da mesma maneira se algum problema desses apresentar uma solução com muitos atrasos, significa que a capacidade de trabalho foi excedida, que vale a pena avaliar a quantidade de carros que estão trabalhando e talvez adicionar carros a frota para evitar esses atrasos.

Entre as sugestões para trabalhos futuros podemos citar:

- desenvolver heurísticas e comparar soluções para escolher um método apropriado para resolução do problema;
- podem-se desenvolver modelos para definir a real necessidade de equipamentos para malha, de forma que todos os serviços sejam atendidos sem que ocorram ociosidade ou atrasos;
- pode ser definida uma técnica de agrupamento de arcos para definir quais conjuntos de arcos devem ser atendidos em um mesmo dia;
- incluir uma modelagem levando em conta a possibilidade de atravessar o arco sem servi-lo;
- usar o conceito de *Split-Delivery* para atender aos arcos de modo que um arco possa ser atendido por mais de um veículo, contanto que dividido o serviço cubram 100% da aresta;
- e considerar paradas obrigatórias para manutenção dos veículos.

REFERÊNCIAS

- AMAYA, A.; LANGEVIN, A.; TRÉPANIÉ, M. The capacitated arc routing problem with refill points. **Operations Research Letters**, v. 35, n. 1, p. 45–53, 2007. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S016763770600006X>>. Acesso em: 28/8/2013.
- ANDRADE, M. A. D. DE. **A criação no processo decisório : o grafo como opção metodológica**. Rio de Janeiro, 1980.
- ARCHETTI, C.; SPERANZA, M. G.; CORBERÁN, Á.; SANCHIS, J. M.; PLANA, I. The Team Orienteering Arc Routing Problem. **Transportation Science**, p. 1–16, 2013.
- ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. **Pesquisa Operacional para cursos de Engenharia**. 6ª ed. ed. Rio de Janeiro: Elsevier, 2007.
- ASSAD, A. A.; GOLDEN, B. L. Arc Routing Methods and Applications. **Handbooks in Operations Research and Management Science, volume 8, Network Routing**. v. 8, p.375–483, 1995.
- BARTOLINI, E.; CORDEAU, J.; LAPORTE, G. An Exact Algorithm for the Capacitated Arc Routing Problem with Deadheading Demand. **Operations Research**, v. 61, n. 2, p. 315–327, 2013.
- BELENGUER, J.; BENAVENT, E. A cutting plane algorithm for the capacitated arc routing problem. **Computers & Operations Research**, v. 30, p. 705–728, 2003. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054802000461>>. Acesso em: 20/1/2014.
- BELENGUER, J.-M.; BENAVENT, E.; LABADI, N.; PRINS, C.; REGHIOUI, M. Split-Delivery Capacitated Arc-Routing Problem: Lower Bound and Metaheuristic. **Transportation Science**, v. 44, n. 2, p. 206–220, 2010. Disponível em: <<http://pubsonline.informs.org/doi/abs/10.1287/trsc.1090.0305>>. Acesso em: 13/3/2014.
- BEULLENS, P.; MUYLDERMANS, L.; CATTRYSSE, D.; VAN OUDHEUSDEN, D. A guided local search heuristic for the capacitated arc routing problem. **European Journal of Operational Research**, v. 147, n. 3, p. 629–643, 2003. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S037722170200334X>>. Acesso em: 28/8/2013.
- BOAVENTURA NETTO, P. O. **Grafos : teoria, modelos, algoritmos**. 4ª ed. ed. São Paulo: Edgard Blucher, 2001.
- CHEN, S.; GOLDEN, B.; WONG, R.; ZHONG, H. Arc-Routing Models for Small-Package Local Routing. **Transportation Science**, v. 43, n. 1, p. 43–55, 2009.

Disponível em: <<http://transci.journal.informs.org/cgi/doi/10.1287/trsc.1080.0255>>. Acesso em: 28/8/2013.

CHRISTOFIDES, N.; BEASLEY, J. E. The Period Routing Problem. **Networks**, v. 14, p. 237–256, 1984.

CHU, F.; LABADI, N.; PRINS, C. THE PERIODIC CAPACITATED ARC ROUTING PROBLEM LINEAR PROGRAMMING MODEL, METAHEURISTIC AND LOWER BOUNDS. **Journal of Systems Science and Systems Engineering**, v. 13, n. 4, p. 423–435, 2004.

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

CHU, F.; LABADI, N.; PRINS, C. A Scatter Search for the periodic capacitated arc routing problem. **European Journal of Operational Research**, v. 169, p. 586–605, 2006.

CORBERÁN, A.; MOTA, E.; SANCHIS, J. M. A comparison of two different formulations for arc routing problems on mixed graphs. **Computers & Operations Research**, v. 33, p. 3384–3402, 2006.

CORBERÁN, A.; PRINS, C. Recent Results on Arc Routing Problems : An Annotated Bibliography. **Networks**, p. 50–69, 2010.

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

EGLESE, R. W. Routeing winter gritting vehicles. **Discrete Applied Mathematics**, v. 48, n. 3, p. 231–244, 1994. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/0166218X92000035>>. .

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

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

EVANS, J. R.; MINIEKA, E. **Optimization algorithms for networks and graphs**. 2nd ed. 1992.

FERNÁNDEZ, E.; MEZA, O.; GARFINKEL, R.; ORTEGA, M. ON THE UNDIRECTED RURAL POSTMAN PROBLEM: TIGHT BOUNDS BASED ON A NEW FORMULATION. **Operations Research**, v. 51, n. 2, p. 281–291, 2003.

FRANCIS, P.; SMILOWITZ, K. Modeling techniques for periodic vehicle routing problems. **Transportation Research Part B**, v. 40, p. 872–884, 2006.

FUNG, R. Y. K.; LIU, R.; JIANG, Z. A memetic algorithm for the open capacitated arc routing problem. **Transportation Research Part E: Logistics and Transportation Review**, v. 50, p. 53–67, 2013. Elsevier Ltd. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S1366554512000932>>. Acesso em: 28/8/2013.

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

GHIANI, G.; LAGANÀ, D.; MANNI, E.; MUSMANNO, R.; VIGO, D. Operations research in solid waste management: A survey of strategic and tactical issues. **Computers & Operations Research**, v. 44, p. 22–32, 2014.

GHIANI, G.; MUSMANNO, R.; PALETTA, G.; TRIKI, C. A heuristic for the periodic rural postman problem. **Computers & Operations Research**, v. 32, n. 2, p. 219–228, 2005. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0305054803002132>>. Acesso em: 28/8/2013.

GOLDEN, B. L.; DEARMON, J. S.; BAKER, E. K. Computational experiments with algorithms for a class of routing problems. **Computers & Operations Research**, v. 10, n. 1, p. 47–59, 1983. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0305054883900266>>. Acesso em: 13/3/2014.

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

GROSS, J. L.; YELLEN, J. **Graph Theory and its applications**. 2nd ed. New York, 2006.

HASHEMI DOULABI, S. H.; SEIFI, A. Lower and upper bounds for location-arc routing problems with vehicle capacity constraints. **European Journal of Operational Research**, v. 224, n. 1, p. 189–208, 2013. Elsevier B.V. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0377221712004705>>. Acesso em: 28/8/2013.

HERTZ, A.; LAPORTE, G.; MITTAZ, M. A Tabu Search heuristic for the Capacitated Arc Routing Problem. **Operations Research**, v. 48, n. 1, p. 129–135, 2000.

HILLIER, F. S.; LIEBERMAN, G. J. **Introdução à Pesquisa Operacional**. 8ª ed. ed. Porto Alegre, 2010.

KANSOU, A.; YASSINE, A. Ant Colony System for the Periodic Capacitated Arc Routing Problem Problem. Proc. 2009 International Network Optimization Conference. **Anais...** p.1–7, 2009.

KONOWALENKO, F. **PROBLEMA DO CARTEIRO CHINÊS NÃO-ORIENTADO E MISTO PARA A OTIMIZAÇÃO DE ROTAS NA CIDADE DE IRATI / PR**, 2012. Universidade Federal do Paraná.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. Evolutionary Algorithms for Multiperiod Arc Routing Problems. In: IPMU 2002 (Ed.); 9th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based systems. **Anais...** p.1–8, 2002a. Annecy, France: ESIA-University of Savoie.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. General arc routing problems solved by a cutting plane algorithm and a genetic algorithm. In: IFAC (Ed.); 15th Triennial World Congress. **Anais...**, 2002b. Barcelona, Spain.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. Planning Problems in Arc Routing. In: F. U.-E. de Valencia (Ed.); Actes de PMS 2002 (EURO Working Group on Project Management and Scheduling). **Anais...** p.232–235, 2002c. Valencia, Espanha.

LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIF, W. Evolutionary algorithms for periodic arc routing problems. **European Journal of Operational Research**, v. 165, p. 535–553, 2005.

LACOMME, P.; PRINS, C.; SEVAUX, M. A genetic algorithm for a bi-objective capacitated arc routing problem. **Computers & Operations Research**, v. 33, n. 12, p. 3473–3493, 2006. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0305054805000730>>. Acesso em: 28/8/2013.

LAPORTE, G.; MUSMANNO, R.; VOCATURO, F. An Adaptive Large Neighbourhood Search Heuristic for the Capacitated Arc-Routing Problem with Stochastic Demands. **Transportation Science**, v. 44, n. 1, p. 125–135, 2010. Disponível em: <<http://transci.journal.informs.org/cgi/doi/10.1287/trsc.1090.0290>>. Acesso em: 28/8/2013.

LONGO, H.; DE ARAGÃO, M. P.; UCHOA, E. Solving capacitated arc routing problems using a transformation to the CVRP. **Computers & Operations Research**, v. 33, n. 6, p. 1823–1837, 2006. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0305054804003223>>. Acesso em: 28/8/2013.

LOPES, H. S.; RODRIGUES, L. C. D. A.; STEINER, M. T. A. **Meta-Heurísticas em Pesquisa Operacional**. Curitiba PR, 2013.

MALANDRAKI, C.; DASKIN, M. S. Theory and Methodology The maximum benefit Chinese postman problem and the maximum benefit traveling salesman problem. **European Journal of Operational Research**, v. 65, n. 1093, p. 218–234, 1993.

MARCORIN, W. R.; LIMA, C. R. L. Análise dos Custos de Manutenção e de Não-manutenção de Equipamentos Produtivos. **Revista de Ciência & Tecnologia**, v. 11, n. 22, p. 35–42, 2003.

MARZOLF, F.; TRÉPANIER, M.; LANGEVIN, A. Road network monitoring: algorithms and a case study. **Computers & Operations Research**, v. 33, n. 12, p. 3494–3507, 2006. Disponível em:

<<http://linkinghub.elsevier.com/retrieve/pii/S0305054805000717>>. Acesso em: 20/1/2014.

MONROY, I. M.; AMAYA, C. A.; LANGEVIN, A. The periodic capacitated arc routing problem with irregular services. **Discrete Applied Mathematics**, v. 161, p. 691–701, 2013.

MURTY, K. G. **Linear programming**. New York, 1983.

NEGREIROS, M.; COELHO JUNIOR, W. R.; PALHANO, A. W. D. C.; et al. O PROBLEMA DO CARTEIRO CHINÊS, ALGORITMOS EXATOS E UM AMBIENTE MVI PARA ANÁLISE DE SUAS INSTÂNCIAS: SISTEMA XNÊS. **Pesquisa Operacional**, v. 29, n. 2, p. 323–363, 2009.

PUCCINI, A. DE L.; PIZZOLATO, N. D. **Programação Linear**. 2ª ed. ed. Rio de Janeiro, 1990.

ROSA, B. DE; IMPROTA, G.; GHIANI, G.; MUSMANNO, R. The Arc Routing and Scheduling Problem with Transshipment. **Transportation Science**, v. 36, n. 3, p. 301–313, 2002.

TAGMOUTI, M.; GENDREAU, M.; POTVIN, J.-Y. Arc routing problems with time-dependent service costs. **European Journal of Operational Research**, v. 181, n. 1, p. 30–39, 2007. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0377221706004644>>. Acesso em: 28/8/2013.

TAHA, H. A. **Pesquisa Operacional**. 8ª ed. ed. São Paulo: Pearson Prentice Hall, 2007.

TAUBE, J. DE M. **O problema do Carteiro Chinês**, 1992. UNICAMP.

TORTELLY JR., A. C.; OCHI, L. S. Uma metaheurística híbrida GRASP+Tabu para o problema de roteamento periódico em uma frota de veículos. XXXV Simpósio Brasileiro de Pesquisa Operacional. **Anais...** p.1–11, 2003. Natal-RN.

USBERTI, F. L. **Métodos Heurísticos e Exatos para o Problema de Roteamento em Arcos Capacitado e Aberto**, 2012. Universidade Estadual de Campinas.

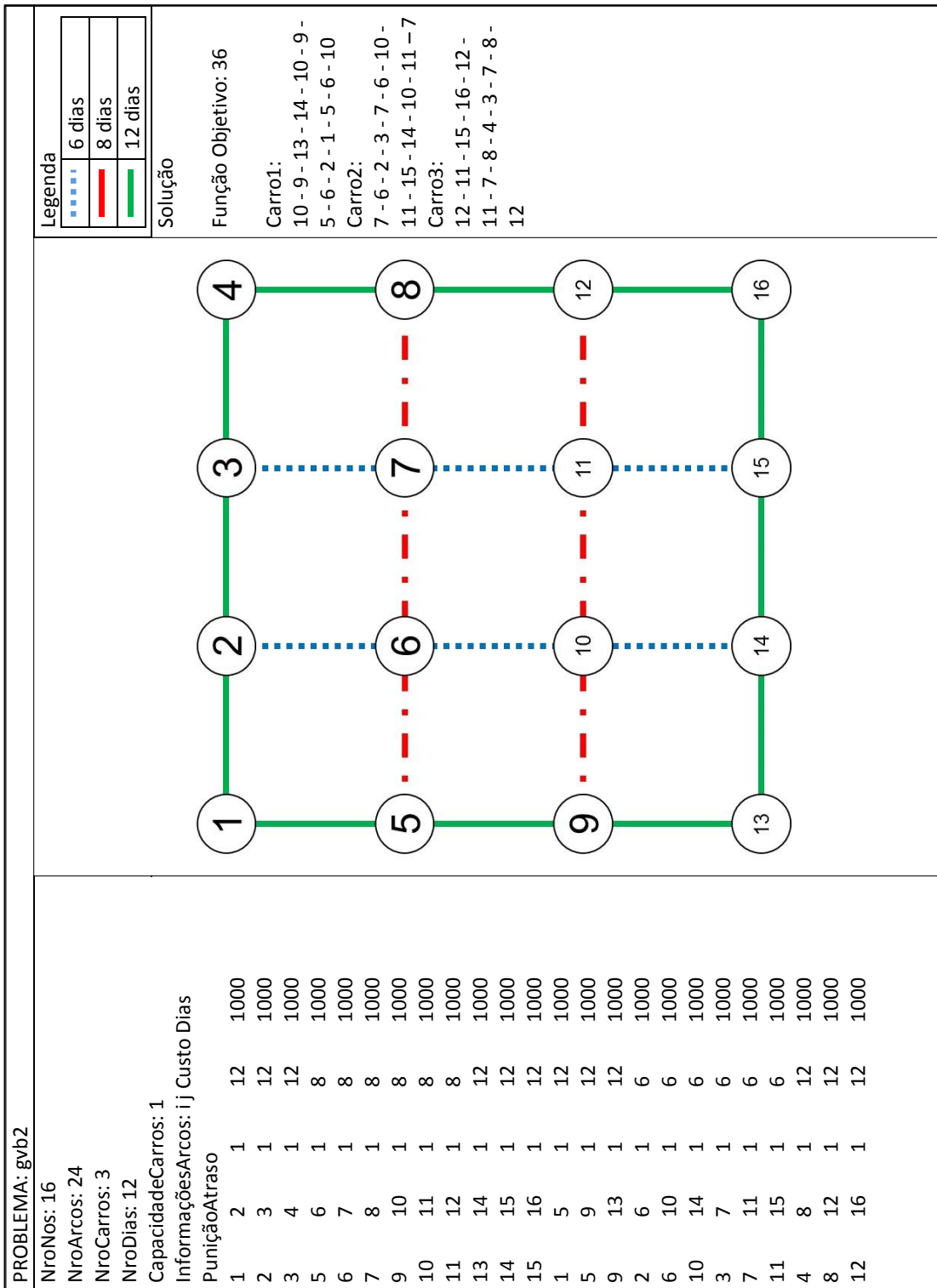
USBERTI, F. L.; FRANÇA, P. M.; FRANÇA, A. L. M. The open capacitated arc routing problem. **Computers & Operations Research**, v. 38, n. 11, p. 1543–1555, 2011. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0305054811000244>>. Acesso em: 28/8/2013.

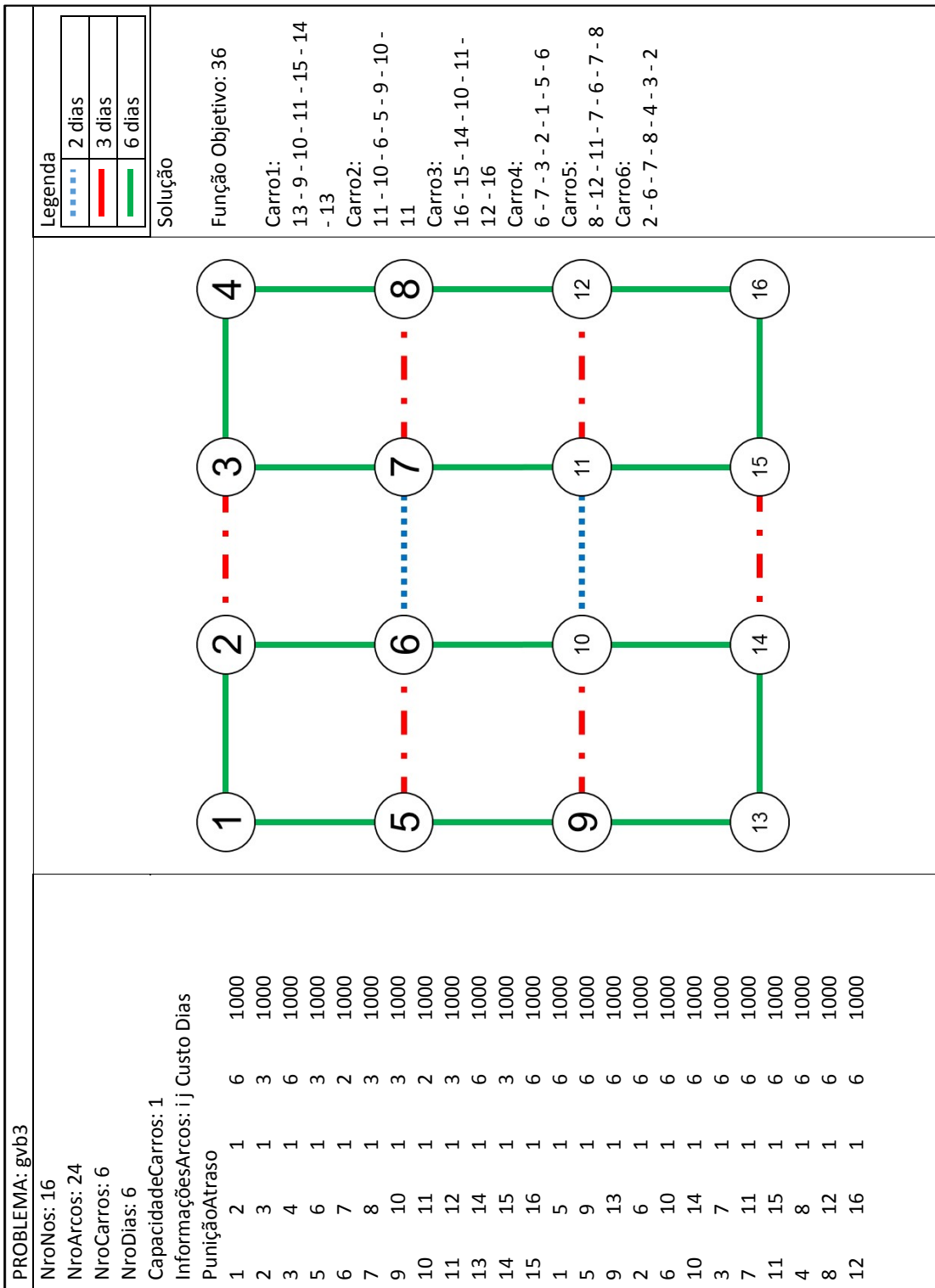
VIDAL, T.; CRAINIC, T. G.; GENDREAU, M.; LAHRICHI, N.; REI, W. A Hybrid Genetic Algorithm for Multidepot and Periodic Vehicle Routing Problems. **Operations Research**, v. 60, n. 3, p. 611–624, 2012.

WANG, H.-F.; WEN, Y.-P. Time-Constrained Chinese Postman Problems. **Computers and Mathematics with Applications**, v. 44, p. 375–387, 2002.

APÊNDICES

<p>PROBLEMA: gvb1</p> <p>NroNos: 16 NroArcos: 24 NroCarros: 9 NroDias: 4 CapacidadeCarros: 1 InformaçõesArcos: i j Custo Dias PuniçãoAtraso</p> <table style="width: 100%; border-collapse: collapse;"> <tr><td>1</td><td>2</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>2</td><td>3</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>3</td><td>4</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>5</td><td>6</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>6</td><td>7</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>7</td><td>8</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>9</td><td>10</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>10</td><td>11</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>11</td><td>12</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>13</td><td>14</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>14</td><td>15</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>15</td><td>16</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>1</td><td>5</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>5</td><td>9</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>9</td><td>13</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>2</td><td>6</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>6</td><td>10</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>10</td><td>14</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>3</td><td>7</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>7</td><td>11</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>11</td><td>15</td><td>1</td><td>2</td><td>1000</td></tr> <tr><td>4</td><td>8</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>8</td><td>12</td><td>1</td><td>4</td><td>1000</td></tr> <tr><td>12</td><td>16</td><td>1</td><td>4</td><td>1000</td></tr> </table>	1	2	1	4	1000	2	3	1	4	1000	3	4	1	4	1000	5	6	1	2	1000	6	7	1	2	1000	7	8	1	2	1000	9	10	1	2	1000	10	11	1	2	1000	11	12	1	2	1000	13	14	1	4	1000	14	15	1	4	1000	15	16	1	4	1000	1	5	1	4	1000	5	9	1	4	1000	9	13	1	4	1000	2	6	1	2	1000	6	10	1	2	1000	10	14	1	2	1000	3	7	1	2	1000	7	11	1	2	1000	11	15	1	2	1000	4	8	1	4	1000	8	12	1	4	1000	12	16	1	4	1000		<p>Legenda</p> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="border: 1px solid black; width: 20px; height: 10px; background-color: #e0e0e0;"></td><td style="padding: 2px;">2 dias</td></tr> <tr><td style="border: 1px solid black; width: 20px; height: 10px; background-color: #e0e0e0;"></td><td style="padding: 2px;">4 dias</td></tr> </table> <p>Solução</p> <p>Função Objetivo: 36</p> <p>Carro1: 11 - 15 - 16 - 12 - 11</p> <p>Carro2: 2 - 6 - 5 - 1 - 2</p> <p>Carro3: 10 - 6 - 7 - 11 - 10</p> <p>Carro4: 9 - 13 - 14 - 10 - 9</p> <p>Carro5: 7 - 3 - 2 - 6 - 7</p> <p>Carro6: 4 - 8 - 7 - 3 - 4</p> <p>Carro7: 14 - 10 - 11 - 15 - 14</p> <p>Carro8: 7 - 11 - 12 - 8 - 7</p> <p>Carro9: 5 - 9 - 10 - 6 - 5</p>		2 dias		4 dias
1	2	1	4	1000																																																																																																																										
2	3	1	4	1000																																																																																																																										
3	4	1	4	1000																																																																																																																										
5	6	1	2	1000																																																																																																																										
6	7	1	2	1000																																																																																																																										
7	8	1	2	1000																																																																																																																										
9	10	1	2	1000																																																																																																																										
10	11	1	2	1000																																																																																																																										
11	12	1	2	1000																																																																																																																										
13	14	1	4	1000																																																																																																																										
14	15	1	4	1000																																																																																																																										
15	16	1	4	1000																																																																																																																										
1	5	1	4	1000																																																																																																																										
5	9	1	4	1000																																																																																																																										
9	13	1	4	1000																																																																																																																										
2	6	1	2	1000																																																																																																																										
6	10	1	2	1000																																																																																																																										
10	14	1	2	1000																																																																																																																										
3	7	1	2	1000																																																																																																																										
7	11	1	2	1000																																																																																																																										
11	15	1	2	1000																																																																																																																										
4	8	1	4	1000																																																																																																																										
8	12	1	4	1000																																																																																																																										
12	16	1	4	1000																																																																																																																										
	2 dias																																																																																																																													
	4 dias																																																																																																																													



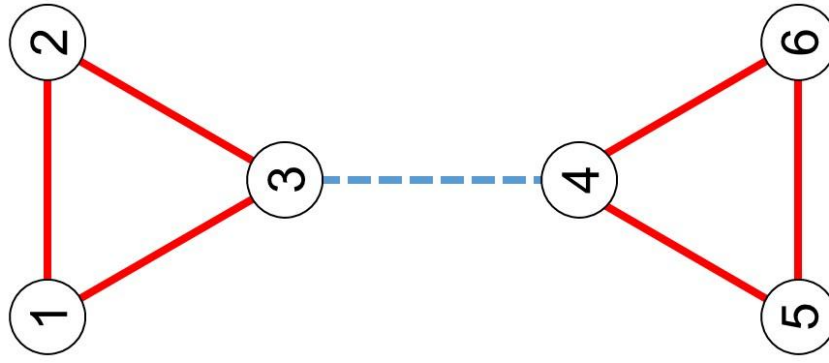


PROBLEMA: gvb4 NroNos: 16 NroArcos: 24 NroCarros: 4 NroDias: 8 CapacidadeCarros: 1 InformaçõesArcos: ij Custo Dias PuniçãoAtraso		Legenda <table border="1"> <tr> <td>.....</td> <td>4 dias</td> </tr> <tr> <td>-----</td> <td>8 dias</td> </tr> </table>	4 dias	-----	8 dias
.....	4 dias					
-----	8 dias					
1 2 1 8 1000 2 3 1 8 1000 3 4 1 8 1000 5 6 1 8 1000 6 7 1 4 1000 7 8 1 8 1000 9 10 1 8 1000 10 11 1 4 1000 11 12 1 8 1000 13 14 1 8 1000 14 15 1 8 1000 15 16 1 8 1000 1 5 1 8 1000 5 9 1 8 1000 9 13 1 8 1000 2 6 1 8 1000 6 10 1 4 1000 10 14 1 8 1000 3 7 1 8 1000 7 11 1 4 1000 11 15 1 8 1000 4 8 1 8 1000 8 12 1 8 1000 12 16 1 8 1000		Solução Função Objetivo: 32 Carro1: 13 - 14 - 15 - 11 - 10 - 6 - 5 - 9 - 13 Carro2: 11 - 7 - 3 - 2 - 1 - 5 - 9 - 10 - 11 Carro3: 10 - 6 - 7 - 8 - 12 - 16 - 15 - 14 - 10 Carro4: 3 - 4 - 8 - 12 - 11 - 7 - 6 - 2 - 3				

PROBLEMA: gvb5

NroNos: 6
NroArcos: 7
NroCarros: 3
NroDias: 6
CapacidadeCarros: 1
InformaçõesArcos: ij Custo Dias
PuniçãoAtraso

1	2	1	3	1000
1	3	1	3	1000
2	3	1	3	1000
3	4	1	1	1000
4	5	1	3	1000
4	6	1	3	1000
5	6	1	3	1000



Legenda

.....	1 dia
.....	3 dias

Solução

Função Objetivo: 18

Carro1:

2 - 3 - 1 - 2 - 3 - 1 - 2

Carro2:

4 - 6 - 5 - 4 - 6 - 5 - 4

Carro3:

3 - 4 - 3 - 4 - 3 - 4 - 3

PROBLEMA: gvb6

NroNos: 6

NroArcos: 7

NroCarros: 1

NroDias: 8

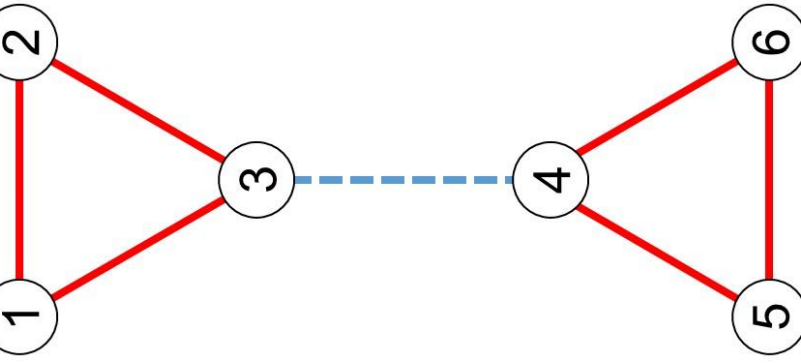
CapacidadeCarros: 1

InformaçõesArcos: ij Custo Dias

Punição	Atraso	1	2	1	8	1000
1	3	1 <td>8<td>1000</td></td>	8 <td>1000</td>	1000		
2	3	1 <td>8<td>1000</td></td>	8 <td>1000</td>	1000		
3	4	1 <td>4<td>1000</td></td>	4 <td>1000</td>	1000		
4	5	1 <td>8<td>1000</td></td>	8 <td>1000</td>	1000		
4	6	1 <td>8<td>1000</td></td>	8 <td>1000</td>	1000		
5	6	1 <td>8<td>1000</td></td>	8 <td>1000</td>	1000		

Legenda

.....	4 dias
.....	8 dias



Solução

Função Objetivo: 8

Carro1:

2 - 3 - 4 - 6 - 5 - 4 - 3 - 1
- 2

<p>PROBLEMA: gvb7</p> <p>NroNos: 12 NroArcos: 24 NroCarros: 7 NroDias: 6 CapacidadeCarros: 1 InformaçõesArcos: ij Custo Dias PuniçãoAtraso</p>	<p>Legenda</p> <table border="1"> <tr> <td>.....</td> <td>3 dias</td> </tr> <tr> <td>.....</td> <td>6 dias</td> </tr> </table>	3 dias	6 dias	<p>Solução</p> <p>Função Objetivo: 42</p> <p>Carro1: 7 - 1 - 6 - 12 - 5 - 6 - 7</p> <p>Carro2: 4 - 11 - 5 - 4 - 11 - 5 - 4</p> <p>Carro3: 10 - 11 - 12 - 7 - 8 - 9 - 10</p> <p>Carro4: 2 - 3 - 9 - 2 - 3 - 9 - 2</p> <p>Carro5: 1 - 8 - 2 - 1 - 8 - 2 - 1</p> <p>Carro6: 3 - 10 - 4 - 3 - 10 - 4 - 3</p> <p>Carro7: 12 - 5 - 6 - 7 - 1 - 6 - 12</p>
.....	3 dias					
.....	6 dias					

PROBLEMA: gvb8

NroNos: 8

NroArcos: 12

NroCarros: 2

NroDias: 8

CapacidadeCarros: 1

InformaçõesArcos: ij Custo Dias

Punição	Atraso	1	2	3	4	5	6	7	8	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	
1	2	1	8																	
2	3	1	4																	
3	4	1	8																	
4	5	1	8																	
5	6	1	4																	
1	6	1	8																	
2	6	1	4																	
2	7	1	8																	
3	7	1	8																	
3	5	1	4																	
5	8	1	8																	
6	8	1	8																	

Legenda

.....	4 dias
.....	8 dias

Solução

Função Objetivo: 16

Carro1:

4 - 5 - 8 - 6 - 1 - 2 - 7 - 3
- 4

Carro2:

6 - 2 - 3 - 5 - 6 - 2 - 3 - 5
- 6

