

UNIVERSIDADE FEDERAL DO PARANÁ

FLÁVIA KONOWALENKO

**PROBLEMA DO CARTEIRO CHINÊS NÃO-ORIENTADO E MISTO PARA A
OTIMIZAÇÃO DE ROTAS NA CIDADE DE IRATI/PR**

CURITIBA
2012

FLÁVIA KONOWALENKO

**PROBLEMA DO CARTEIRO CHINÊS NÃO-ORIENTADO E MISTO PARA A
OTIMIZAÇÃO DE ROTAS NA CIDADE DE IRATI/PR**

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 Ciências.

Orientadora: Prof. Dra. Deise Maria Bertholdi Costa.

Co-orientador: Prof. Dr. Luiz Fernando Nunes.

CURITIBA
2012

TERMO DE APROVAÇÃO

FLÁVIA KONOWALENKO

PROBLEMA DO CARTEIRO CHINÊS NÃO-ORIENTADO E MISTO PARA A
OTIMIZAÇÃO DE ROTAS NA CIDADE DE IRATI/PR

Dissertação aprovada como requisito parcial à obtenção de grau de Mestre em Ciências, Programa de Pós-Graduação em Métodos Numéricos em Engenharia, área de concentração em Programação Matemática, Setor de Tecnologia, Departamento de Construção Civil e Setor de Ciências Exatas, Departamento de Matemática da Universidade Federal do Paraná, pela seguinte banca examinadora:

Orientadora: Prof^ª. Dr^ª. Deise Maria Bertholdi Costa
PPGMNE e Departamento de Expressão Gráfica, UFPR

Co-Orientador: Prof. Dr. Luiz Fernando Nunes
Departamento de Matemática, UTFPR

Prof^ª. Dr^ª. Angela Olandoski Barboza
Departamento de Matemática, UTFPR

Prof. Dr. Lauro Cesar Galvão
Departamento de Matemática, UTFPR

Curitiba, 01 de Fevereiro de 2012.

“O único defeito das mães é, não serem eternas.”

AGRADECIMENTOS

Considerando que nenhum trabalho é uma realização individual, fica aqui registrado meus sinceros agradecimentos a todas as pessoas e instituições que direta, ou indiretamente contribuíram para a realização deste trabalho.

À Deus por estar presente e guiar nas situações mais difíceis, iluminando e dando forças nos momentos obscuros e de desânimo.

À professora Deise Maria Bertholdi Costa, por confiar-me a oportunidade de ser orientada. Obrigada pelo empenho e dedicação durante a orientação e ainda pelo exemplo que me deixou como uma professora com elevadas qualidades intelectuais e morais.

Ao professor, amigo e co-orientador Luiz Fernando Nunes pela confiança e incentivo desde os primeiros passos, durante o desenvolvimento, até a conclusão do trabalho, sempre amparando e auxiliando nos momentos mais difíceis.

À professora, amiga e praticamente co-orientadora Angela Olandoski Barboza pelo incentivo e auxílio no desenvolvimento do trabalho.

À secretária do PPGMNE Maristela Bandil, que com seu bom humor contagiante sempre estava disposta a ajudar e atender às solicitações prontamente.

Aos professores do PPGMNE, pela oportunidade de ampliar os conhecimentos nas diferentes áreas de pesquisa.

Aos professores integrantes da banca examinadora, pelas valiosas sugestões que contribuíram para a apresentação deste trabalho na sua forma final.

Ao meu marido Sergio e ao meu filho Matheus, pelo apoio, incentivo, paciência, compreensão e carinho. Agradeço por aceitar minha ausência nas épocas de provas e trabalhos e pelo afago nos momentos de desespero.

Aos meus pais Valdir e Marlene (*in memoriam*) por tudo que sou, pelos valiosos ensinamentos que guardo e pratico com grande carinho e consideração.

À minha amiga Paula Francis Benevides, irmã de mestrado, por todo carinho e amizade imensuráveis, pelos momentos de alegria e tristeza, por toda ajuda e principalmente por não ter me deixado desistir no momento mais difícil da minha vida.

À Universidade Federal do Paraná, pela oportunidade de aprimoramento acadêmico e em especial aos colegas do Cesec por todo companheirismo, apoio e auxílio em todos os momentos.

À CAPES REUNI, pelo apoio financeiro.

RESUMO

O Problema do Carteiro Chinês caracteriza-se pela roteirização de arcos e tem como objetivo a cobertura de arcos de um grafo, criando uma rota que passe ao menos uma vez em cada um destes arcos. O problema pode ser subdividido em casos, de acordo com o tipo do grafo analisado. Entre os casos, tem-se o Problema do Carteiro Chinês Não-Orientado, cuja característica é que nenhuma das arestas do grafo associado ao problema possui orientação. Um segundo caso, é denominado Problema do Carteiro Chinês Orientado, quando todos os arcos são orientados, e finalmente, um terceiro caso que é chamado de Problema do Carteiro Chinês Misto, que possui arcos (orientados) e arestas (sem orientação ou duplo sentido). Nesta pesquisa, o estudo do Problema do Carteiro Chinês foi aplicado na área central da cidade de Irati/PR, região representada por um grafo caracterizado como misto, pois possui ruas de mão única e também ruas de mão dupla. Um primeiro estudo foi otimizar a rota percorrida pelo caminhão de coleta de resíduos sólidos. Nesta parte, foi aplicado o caso Misto e para a resolução utilizou-se modelos de programação Linear Inteira. Considerando-se ainda a mesma região da cidade, porém sem levar em conta as orientações das ruas, buscou-se também uma solução para do Problema do Carteiro Chinês Não Orientado. Os resultados encontrados são para a construção da rota para os varredores de ruas, entregadores de correspondências, ou outros serviços que exijam coberturas de arcos e que possam ser executados por funcionários percorrendo as ruas à pé, ou seja, sem o uso de veículos. Para este último caso utilizou-se uma heurística. Os resultados obtidos foram satisfatórios, visto que o melhor valor foi para o problema do grafo misto, através do modelo exato de programação linear inteira, o qual otimizou a rota percorrida pelo caminhão de coleta de resíduos sólidos em 12,67%.

Palavras-chave: Problema do Carteiro Chinês; Algoritmos de construção de rotas; programação linear inteira;

ABSTRACT

The Chinese Postman Problem distinguishes by the routing of arches, targeting to cover all the arches of a graph, passing through them at least once. The problem may have particular characteristics which are presented according to the kind of graph being analyzed. So, the Chinese Postman Problem is called a Non-oriented one when none of the edges of the analyzed problem has one orientation. It is called an Oriented Chinese Postman Problem when all of the arches are oriented and, finally, it's called a Mixed Chinese Postman Problem when it has arches (oriented) and edges (non-oriented or double-guided). The present research was developed at the central area of Irati/PR, a place marked as Mixed, because it has one-way roads and also two-way roads. The objectives were to optimize the route traveled by the solid waste collect truck. It demanded the usage of Full Linear Programming models. Using the same analyzed area, but without considering the way of the roads, it was pursued a solution to the Non-Oriented Chinese Postman Problem, seeking the use of the results by street dustmen, mailmen and other services which require the cover of arches and may be executed by workers wandering the roads on foot (without using vehicles). This last case demanded the usage of a heuristic process. The obtained results were satisfactory, being that the greatest value was given to the mixed graph case, through the exact full linear programming model, which has optimized the route traveled by the solid waste collect truck in 12,67%.

Keywords: Chinese Postman Problem, coverage problem of arcs, Full Linear Programming;

LISTA DE FIGURAS

FIGURA 1: MAPA DO PARANÁ, IRATI EM DESTAQUE.....	19
FIGURA 2: MAPA DO MUNICÍPIO DE IRATI.....	20
FIGURA 3 – MAPA DA ÁREA URBANA, DIVIDIDA EM BAIROS, DO MUNICÍPIO DE IRATI	22
FIGURA 4: MAPA DA REGIÃO CENTRAL DO MUNICÍPIO DE IRATI	23
FIGURA 5: ROTA ATUAL REALIZADA NA COLETA	25
FIGURA 6: PONTES DE KÖNIGSBERG.....	35
FIGURA 7: DOIS CIRCUITOS COM UM VÉRTICE EM COMUM	36
FIGURA 8: CIRCUITO EULERIANO.....	37
FIGURA 9: MAPEAMENTO DAS COORDENADAS GEOGRÁFICAS.....	57
FIGURA 10: FLUXOGRAMA PCCNO – FONTE: O autor (2012).....	64
FIGURA 11: FLUXOGRAMA PCCM	65
FIGURA 12: GRAFO G	88
FIGURA 13: GRAFO EULERIANO.....	88
FIGURA 14 – ROTA ATUAL 1	89
FIGURA 15 – ROTA ATUAL 2	90
FIGURA 16: ROTA ATUAL 3.....	90
FIGURA 17: ROTA ATUAL 4.....	91
FIGURA 18: MATRIZ DE ADJACÊNCIAS	92
FIGURA 19: CUSTO DOS ARCOS	93

LISTA DE TABELAS

TABELA 1: RESULTADOS PCCNO	71
TABELA 2: IDENTIFICAÇÃO DO GRAU DOS NÓS.....	71
TABELA 3: EMPARELHAMENTO INICIAL DOS NÓS	72
TABELA 4: EMPARELHAMENTO FINAL DOS NÓS	73
TABELA 5 – RESULTADOS FORNECIDOS PELO APLICATIVO NEOS.....	78
TABELA 6: RESULTADO NEOS.....	95

LISTA DE ABREVIATURAS E SIGLAS

CONAMA	Conselho Nacional de Meio Ambiente
PL	Programação Linear
PO	Pesquisa Operacional
PI	Programação Inteira
PIM	Programação Inteira Mista
PLIM	Programação Linear Inteira Mista
PNL	Programação Não Linear
PCV	Problema do Caixeiro Viajante
PCC	Problema do Carteiro Chinês
PCCO	Problema do Carteiro Chinês Orientado
PCCNO	Problema do Carteiro Chinês Não Orientado
PCCM	Problema do Carteiro Chinês Misto
PCCV	Problema do Carteiro Chinês com Vento
PCR	Problema do Carteiro Rural
PE	Problema da Empilhadeira
PGR	Problema Geral de Roteamento
GAMS	General Algebraic Modeling System
BL	Busca Local
VB	Visual Basic
ABNT	Associação Brasileira de Normas Técnicas

SUMÁRIO

1. INTRODUÇÃO.....	13
1.1 DESCRIÇÃO E RELEVÂNCIA DO PROBLEMA	13
1.2 JUSTIFICATIVA	14
1.3 OBJETIVOS	15
1.3.1 Geral	15
1.3.2 Específicos	15
1.4 PROBLEMA A SER ESTUDADO	15
1.5 APLICAÇÕES.....	16
1.6 ORGANIZAÇÃO DA DISSERTAÇÃO.....	16
2. A COLETA DE RESÍDUOS SÓLIDOS NA CIDADE DE IRATI IRATI/PR.....	18
2.1 ÁREA ANALISADA NO ESTUDO	23
3. TRABALHOS CORRELATOS.....	26
4. FUNDAMENTAÇÃO TEÓRICA.....	30
4.1 GRAFOS	30
4.2 PROBLEMAS DE ROTEAMENTO DE ARCOS.....	32
4.3 GRAFOS DE EULER	34
4.4 CIRCUITOS EULERIANOS (CE)	35
4.4.1 Problema do Carteiro Chinês (PCC)	39
4.5 HEURÍSTICAS E METAHEURÍSTICAS	47
4.6 PROGRAMAÇÃO LINEAR	48
4.7 PROGRAMAÇÃO INTEIRA (PI)	49
4.7.1 GAMS	51
4.7.2 Sítio NEOS.....	52
5. METODOLOGIA.....	55
5.1 PROBLEMA DE COLETA DE RESÍDUOS SÓLIDOS URBANOS	55
5.2 OBTENÇÃO DOS DADOS PARA O ESTUDO DE CASO.....	56
5.3 CADASTRO DOS PONTOS GEOGRÁFICOS NO <i>GOOGLE EARTH</i>	56
5.4 OBTENÇÃO DAS DISTÂNCIAS ENTRE OS PONTOS GEOGRÁFICOS	58
5.4.1 Veículos	58
5.4.2 Rota Atual do caminhão de coleta de resíduos.....	58
5.5 MODELAGEM EM PROGRAMAÇÃO LINEAR INTEIRA	59
5.6 HEURÍSTICA PARA O PCCNO	60
5.7 ALGORITMOS DE CONSTRUÇÃO DE ROTA	62

5.7.1	<i>Algoritmo Pathwork</i>	62
5.7.2	<i>Algoritmo de Hierholzer</i>	63
5.8	FLUXOGRAMA DA PROPOSTA METODOLÓGICA DESENVOLVIDA PARA O PCCNO	64
5.9	FLUXOGRAMA DA PROPOSTA METODOLÓGICA DESENVOLVIDA PARA RESOLUÇÃO DO MODELO EM PLI.....	65
6.	IMPLEMENTAÇÃO COMPUTACIONAL E ANÁLISE DOS RESULTADOS	66
6.1	TECNOLOGIAS UTILIZADAS.....	66
6.2	DADOS UTILIZADOS	66
6.3	PROBLEMA DO CARTEIRO CHINÊS NÃO ORIENTADO.....	66
6.3.1	<i>Aplicação do Modelo de Programação Linear Inteira</i>	66
6.4	PROBLEMA DO CARTEIRO CHINÊS MISTO.....	74
6.4.1	<i>Aplicação do Modelo de Programação Linear Inteira</i>	74
7.	CONSIDERAÇÕES FINAIS	80
8.	SUGESTÕES PARA TRABALHOS FUTUROS.....	82
9.	REFERÊNCIAS	83

1. INTRODUÇÃO

1.1 DESCRIÇÃO E RELEVÂNCIA DO PROBLEMA

A Associação Brasileira de Normas Técnicas (ABNT) define resíduos sólidos como os resíduos resultantes da comunidade de origem: industrial, comercial, residencial, hospitalar, agrícola, de serviços e de varrição. Normalmente os resíduos apresentam-se sob estado sólido, semi-sólido ou semi-líquido.

Os municípios brasileiros produzem cerca de 200 mil toneladas de lixo/dia, sendo 52%, matéria orgânica, 25% de papel/papelão, 9% estão divididos igualmente entre plástico, metal e vidro e os outros 16%, rejeito (lâmpadas, isopor, papel sanitário, fraldas, etc) (SEMA – desperdício zero, 2009).

A coleta e o transporte dos resíduos sólidos urbanos têm gerado um significativo gasto da verba municipal nas cidades e municípios do mundo inteiro. Em Irati/PR, cidade cuja pesquisa será desenvolvida, esse custo tem um montante entre 7% e 9%, que é considerado elevado em relação à arrecadação total do município (IRATI. Prefeitura Municipal. BOLETIM INFORMATIVO, 2011).

A coleta de lixo e o seu transporte para áreas de tratamento ou destinação final são ações do serviço público municipal, de grande utilidade para a população, pois impedem o desenvolvimento de vetores transmissores de doenças que encontram alimento e abrigo no lixo (D' ALMEIDA, 2002).

Segundo o Panorama dos Resíduos Sólidos no Brasil (2006), o Estado Paranaense possui uma população total de 9.492.790 habitantes, que produz diariamente 7.417 toneladas de resíduos estimando-se um total de 0,781 Kg/hab./dia de resíduos sólidos urbano.

“A instalação do novo Sistema Integrado de Aproveitamento de Resíduos Sólidos (Sipar) será uma grande evolução na questão do tratamento de lixo no Brasil. Até o momento, poucas cidades brasileiras tentaram fugir dos aterros sanitários, aterros controlados ou lixões. Dados do Instituto Brasileiro de Geografia e Estatística (IBGE) mostravam, em 2002, um cenário desolador: 59% dos municípios destinavam seus resíduos para lixões, 17% para aterros controlados e 13% para aterros sanitários. Apenas 2,8% das cidades tinham programas de reciclagem.” (MEIO AMBIENTE/Sipar representa evolução no tratamento de lixo - VINICIUS BOREKI – Acessado em 02/02/2011 às 15:45 hrs.).

Os serviços de limpeza de uma cidade, absorvem entre 7% e 15% dos recursos de um orçamento municipal, dos quais 50% são destinados exclusivamente à coleta e ao transporte de resíduos. Certamente a sua otimização leva a uma economia significativa dos recursos públicos (Carvalho, 2001).

A coleta de lixo doméstico na cidade de Irati/PR é um serviço realizado pela Prefeitura Municipal, assim como a coleta do lixo hospitalar. A coleta de resíduos recicláveis é feita pela Associação dos Catadores, onde as pessoas cadastradas realizam a coleta desses resíduos, os separam e vendem no barracão da associação. Esses resíduos são reciclados por empresas particulares e são transformados em novas embalagens/materiais, poupando matéria prima e energia da natureza.

Os problemas de otimização de rotas surgem em diversos contextos práticos onde há a necessidade de minimizar rotas, como na presente pesquisa, que trata da coleta do lixo urbano e de entrega de correspondências e limpeza das ruas. Grandes quantias em dinheiro podem ser economizadas a cada ano por governos e empresas privadas se houver o planejamento e execução destas operações de minimização das rotas.

Neste contexto, buscou-se para o desenvolvimento deste trabalho, o estudo da otimização do percurso de coleta de resíduos sólidos (com utilização de veículo) e a otimização da rota utilizada por serviços que possam ser realizados sem a utilizações de veículos (à pé) na região central da cidade de Irati/PR. Esse problema de cobertura de arcos, é conhecido como o Problema do Carteiro Chinês. O objetivo é a cobertura de todos os arcos de um grafo de forma a minimizar o custo desse percurso.

1.2 JUSTIFICATIVA

Levando em consideração o montante de gastos que os municípios têm em relação à coleta e o transporte dos resíduos sólidos urbanos, a presente pesquisa visa minimizar a rota percorrida pelo caminhão de coleta do lixo urbano na cidade de Irati/PR, no bairro Centro. Dessa forma, se a distância percorrida for otimizada, os custos diminuirão e proporcionarão economia da verba pública ao município. Ainda, essa otimização irá reduzir o tempo do serviço de coleta.

1.3 OBJETIVOS

1.3.1 Geral

Com a presente pesquisa, pretende-se otimizar a rota de coleta e o transporte dos resíduos urbanos e da limpeza das ruas e entrega de correspondências e faturas, para que haja uma minimização dos gastos dos recursos públicos, gerando, desta forma, economia de verbas da Prefeitura e eventualmente de empresas privadas da cidade de Irati/PR.

1.3.2 Específicos

- Explorar o problema de coleta e transporte dos resíduos sólidos urbanos na cidade de Irati/PR, utilizando dados reais;
- Identificar, desenvolver e comparar propostas que resolvam o problema de otimização do problema de cobertura de arcos através da resolução do Problema do Carteiro Chinês;
- Desenvolver e aplicar algoritmos para a construção de rotas a partir dos resultados obtidos;
- Utilizar dados computacionais, como cadastro de pontos (nós), cálculo das distâncias (arestas) e algoritmos matemáticos, para desenvolver e analisar as soluções encontradas;
- Verificar através da observação e análise de testes, a validação dos resultados e comparar esses resultados com a realidade no caso do problema do carteiro chinês misto, isto é, da coleta de resíduos;
- Analisar e comparar os resultados obtidos pela heurística com os resultados com a aplicação da Programação Linear Inteira.

1.4 PROBLEMA A SER ESTUDADO

No decorrer do desenvolvimento da pesquisa serão utilizadas técnicas de Pesquisa Operacional para modelar e resolver o problema de geração de rotas, com o objetivo de minimizar o custo do percurso da coleta e transporte dos resíduos sólidos urbanos, da entrega de faturas e correspondências e/ou da limpeza das ruas na cidade de Irati/PR, utilizando dados reais.

1.5 APLICAÇÕES

De acordo com Bodin (1978), o Problema do Carteiro Chinês e suas variações têm uma grande gama de aplicações, seja na distribuição de serviços públicos ou privados, na entrega ou coleta de mercadorias e em vários outros problemas que podem ser formulados como tal. Alguns exemplos de aplicações que envolvem roteamento de arcos são as seguintes:

- Coleta de lixo domiciliar;
- Limpeza de ruas usando varredores mecânicos;
- Remoção de neve das vias públicas;
- Pulverização das vias públicas com sais que evitam a formação de gelo;
- Serviço de entrega de cartas e encomendas de correios;
- Inspeções periódicas em linhas elétricas, redes de gasodutos, ou oleodutos;
- Serviços de transporte escolar;
- Distribuição de alguns produtos de consumo em larga escala, como água mineral, refrigerantes, leite, jornais, etc;
- Leitura de medidores de consumo de água, energia, gás, entre outras aplicações.

1.6 ORGANIZAÇÃO DA DISSERTAÇÃO

No primeiro capítulo foi apresentado um breve histórico sobre os resíduos sólidos, a justificativa, os objetivos da pesquisa, o problema a ser resolvido e as aplicações.

No segundo capítulo serão feitas descrições sobre a cidade de Irati/PR, bem como a visualização dos mapas de localização, características da cidade e da população e também a descrição da rota atual do caminhão de coleta de resíduos sólidos na região central da cidade.

No terceiro capítulo serão apresentados resumos sobre trabalhos envolvendo o Problema do Carteiro Chinês.

O quarto capítulo está destinado à análise e revisão de literatura, isto é, as definições e conceitos sobre grafos, problemas de cobertura de arcos, circuitos eulerianos, Programação Linear Inteira, Programação Linear Inteira Mista e as características que distinguem e compõem as várias formas do Problema do Carteiro

Chinês. Também neste capítulo serão feitas observações sobre os aplicativos GAMS e *sítio* NEOS.

No quinto capítulo será apresentada a metodologia utilizada na pesquisa, a descrição de como foi feito o estudo exploratório através do aplicativo *Google Earth* e a obtenção dos dados para o estudo de caso.

Implementação computacional e análise dos resultados, bem como as tecnologias utilizadas, e os algoritmos desenvolvidos para a otimização das rotas aplicando os conceitos do Problema do Carteiro Chinês serão apresentados no sexto capítulo.

No sétimo capítulo serão expostas as considerações finais obtidas com o desenvolvimento da pesquisa.

Finalmente o oitavo capítulo apresenta algumas sugestões para o desenvolvimento de trabalhos futuros.

2. A COLETA DE RESÍDUOS SÓLIDOS NA CIDADE DE IRATI IRATI/PR

A cidade de Irati/PR está localizada geograficamente a 25 ° 27' 56" de latitude Sul com interseção com o meridiano 50º 37' 51" de longitude Oeste, conforme destaque na figura 1. Em linha reta, situa-se a 138 km de Curitiba, capital do Estado, e a 1.105 km de Brasília, Distrito Federal. O município de Irati está localizado na zona fisiográfica de IRATI, uma das onze em que o Paraná se divide. Encontra-se na sub-região dos pinhais do Segundo Planalto e possui uma área total de 998,3km² (Site Prefeitura Municipal de Irati, acessado em 05/12/2011).

Na figura 2, é mostrado o mapa do município de Irati/PR, onde se pode notar sua área urbana, destacada em vermelho.

De acordo com dados do IBGE, a população total do Município de Irati é de 56.032 habitantes, pela recontagem no Senso de 2010, sendo a população urbana de 44.782 habitantes e a rural de 11.250 habitantes, representando 80% e 20% respectivamente (SOUZA, 2010).

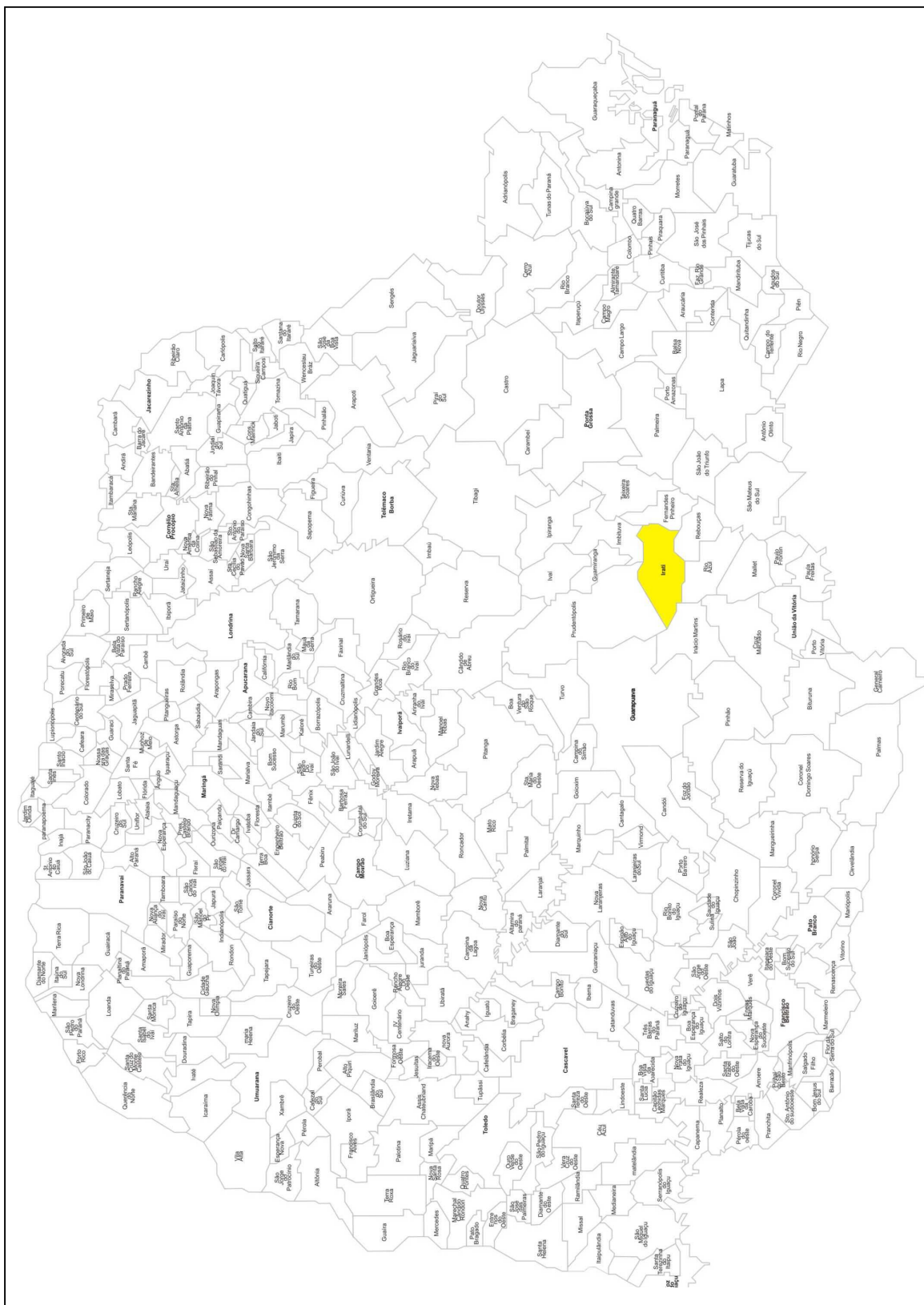


FIGURA 1: MAPA DO PARANÁ, IRATI EM DESTAQUE
FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

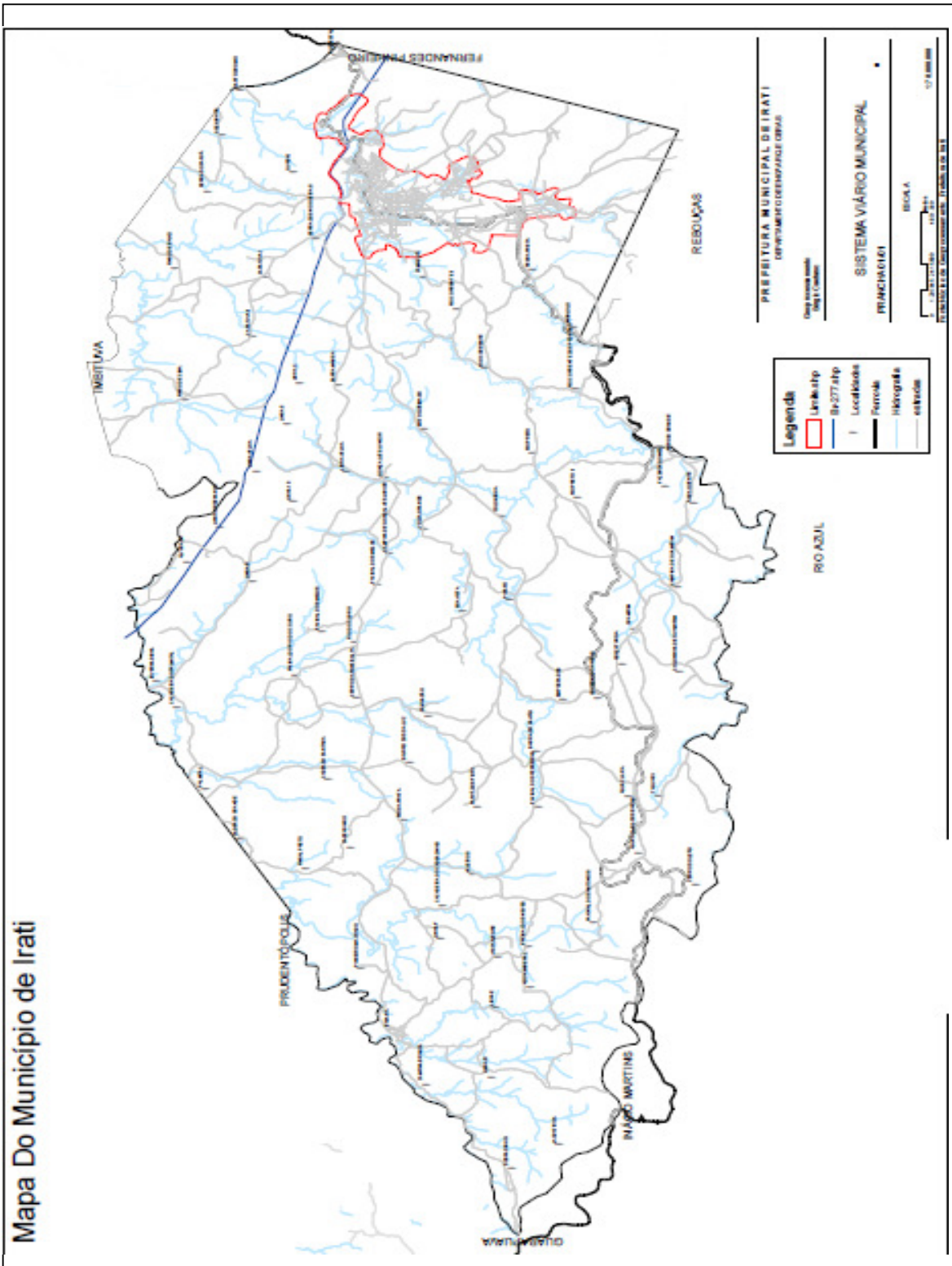


FIGURA 2: MAPA DO MUNICÍPIO DE IRATI
FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

A coleta dos resíduos que compõe a região central da cidade é efetuada através de caminhões coletor-compactadores da própria prefeitura. É diferenciada para os diversos tipos de resíduos, sendo que os caminhões coletam somente os resíduos domiciliares e comerciais. Para os outros tipos de resíduos, tais como de saúde, industrial, agrotóxicos, oficinas e postos de combustíveis, há uma legislação específica (CONAMA – Conselho Nacional de Meio Ambiente) para cada tipo de material, onde o gerador é responsável pela sua destinação.

Esse serviço é realizado de segunda-feira a sábado, sendo oferecido diariamente na área central devido ao grande número de residências e comércio dos diferentes ramos. Nos bairros mais populosos a coleta é realizada 2 a 3 vezes por semana e em bairros com menor número de habitantes, apenas 1 vez por semana, dependendo do trajeto e do volume diário. A figura 3 ilustra a área e população de cada bairro, com dados estimados em 2004.

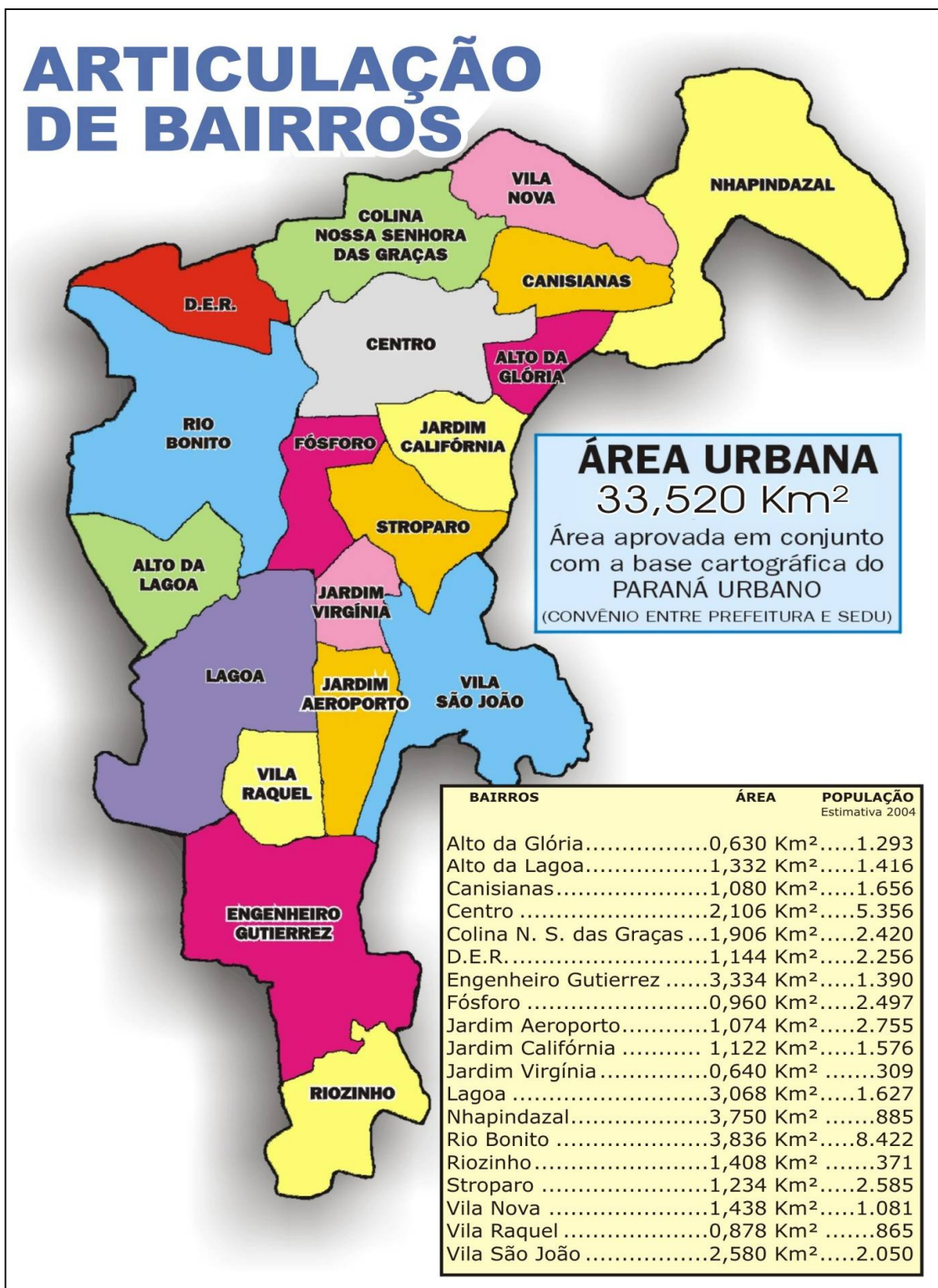


FIGURA 3 – MAPA DA ÁREA URBANA, DIVIDIDA EM BAIRROS, DO MUNICÍPIO DE IRATI

FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

2.1 ÁREA ANALISADA NO ESTUDO

Para o desenvolvimento da presente pesquisa foi utilizada uma área da cidade de Irati/PR, a região central, onde foi feita uma análise para otimização da rota do caminhão de coleta de resíduos sólidos e o percurso de varredores de ruas e entregadores de correspondências. Essa área compreende aproximadamente 2,2km² e pode ser visualizada na figura 4. Essa região é estritamente urbana e desta forma, composta por residências e pontos comerciais.

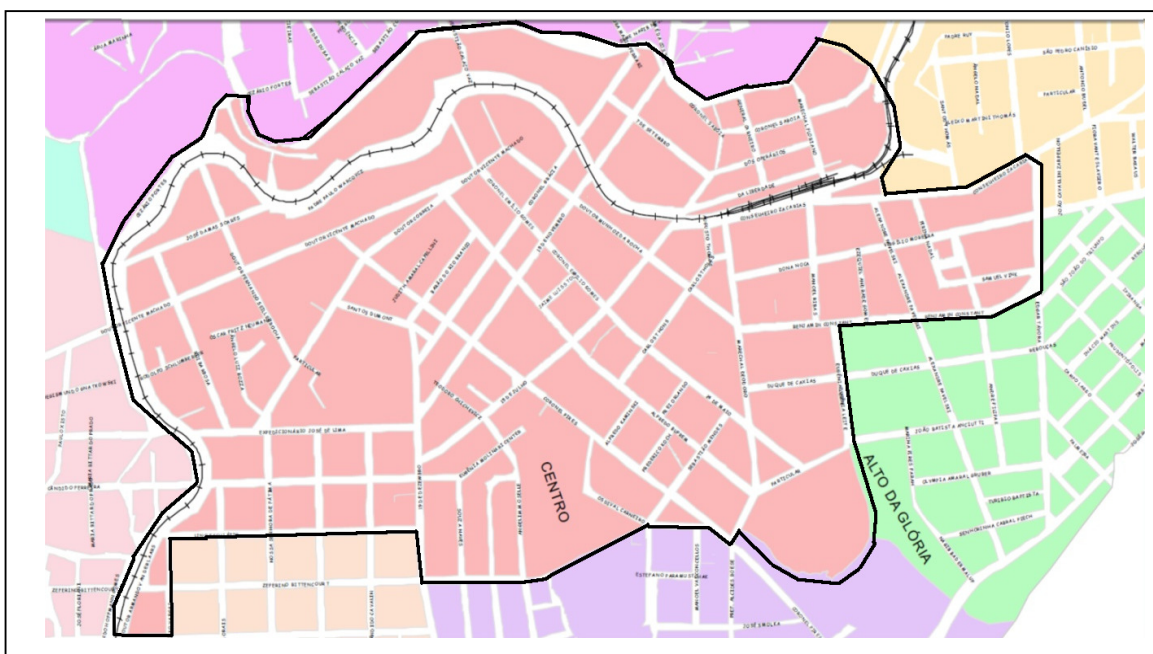


FIGURA 4: MAPA DA REGIÃO CENTRAL DO MUNICÍPIO DE IRATI
FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

A coleta dos resíduos sólidos é baseada no fluxo diário de lixo produzido pela população da cidade citada, sendo que se deve levar em consideração variáveis como o tempo de duração das coletas, a quantidade de quilômetros percorridos, o número de viagens necessárias para que toda a região seja atendida, o número de caminhões que se faz necessário para atender a demanda e quantos funcionários que serão designados para atender às funções. O objetivo da prefeitura é prever a demanda e organizar cada item de forma a minimizar o número de viagens que o caminhão terá que percorrer.

Atualmente, nenhum método tem sido adotado para definir o trajeto a ser percorrido pelo caminhão de coleta e nem pelos entregadores de correspondências.

Assim a responsabilidade dos funcionários, que com base no seu conhecimento e experiência, buscam a melhor/menor rota possível.

De acordo com Pascoal Jr. (2010), o transporte de resíduos é executado por caminhões compactadores com capacidade média de 5 m³ cada. Cada veículo conta com três funcionários, sendo um motorista e dois garis responsáveis pelo recolhimento do material na área urbana. O mesmo autor ainda complementa que, após coletar o material, os caminhões se deslocam até o aterro sanitário pela BR 277, num percurso de 15 km, seguindo 6 km por estrada secundária em bom estado de conservação. O local de transbordo fica localizado em Pinho de Cima, área rural, distante 21 km da sede. O autor observou, por meio de saídas de campo, que a frequência de coleta não está otimizada em certos pontos da área urbana, principalmente na área central, em que o caminhão coletor percorre de quatro a cinco vezes o mesmo roteiro. Alguns fatores que apontaram as dificuldades encontradas pelos funcionários no período em que foi realizada a coleta foram: ruas íngremes e de difícil acesso; trânsito intenso de veículos entre 9h e 15h; resíduos acondicionados em tambores com peso acima de 100 kg, dificultando o trabalho de carregamento dos materiais; latões abertos, ocasionando o acúmulo de água em dias chuvosos; animais soltos (cães), colocando em risco a saúde dos garis; objetos perfuro-cortantes (cacos de vidro) sem acondicionamento adequado, misturados com os resíduos orgânicos; materiais de varrição, poda verde, resíduos de floricultura colocados com os resíduos domiciliares; e materiais acondicionados em sacolas plásticas pelas residências sem o devido cuidado, facilitando o ataque de animais, que espalham o material em vias públicas.

Na figura 5 é apresentada a atual rota percorrida pelo motorista na região central do município, todos os dias da semana. Esse trajeto foi realizado no mês de fevereiro de 2011, percorrendo uma distância de aproximadamente 80 km para coletar os resíduos em todos os pontos comerciais e residências. Essa distância total foi calculada a partir de um ponto inicial de coleta com posterior retorno ao mesmo ponto.

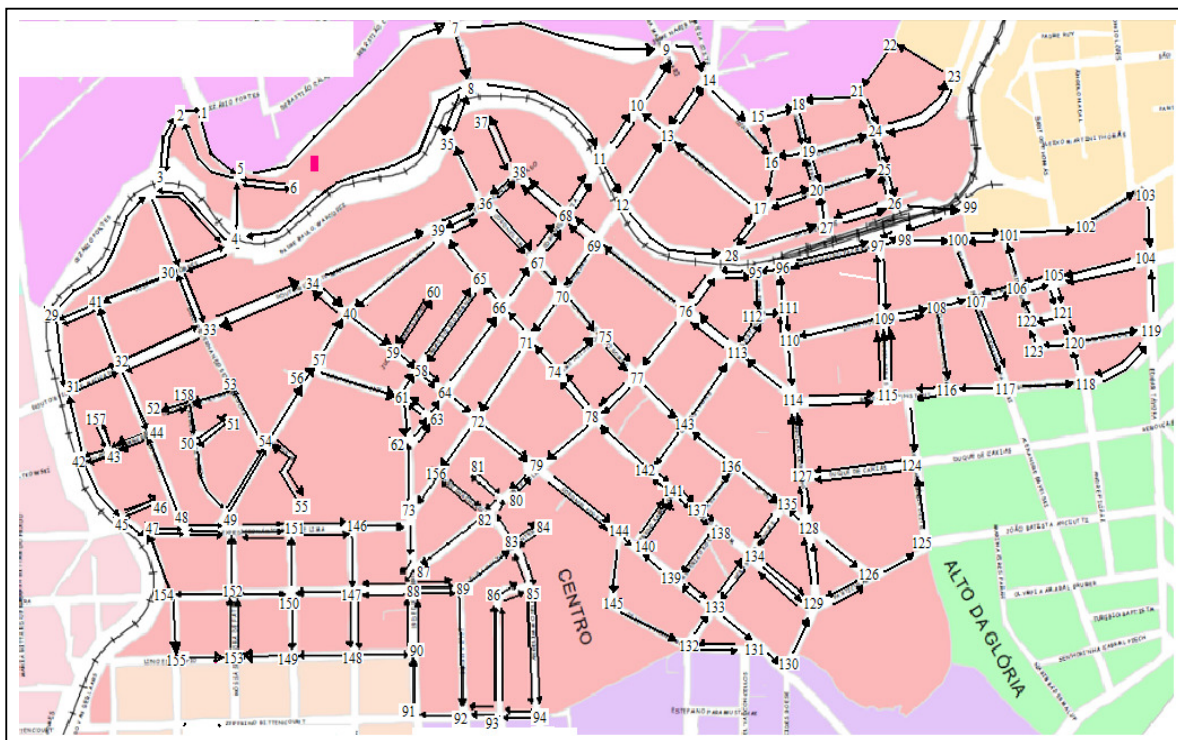


FIGURA 5: ROTA ATUAL REALIZADA NA COLETA
 FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

Pode-se notar que uma grande parte da distância percorrida pelo caminhão de resíduos não é feita pela coleta dos materiais em si e sim pelo trajeto de deslocamento da garagem ao ponto inicial de coleta, para levar os resíduos até o aterro sanitário e para voltar à garagem, totalizando nesse trajeto 21 km (ida e volta).

A dificuldade para se definir o trajeto a ser realizado pelo caminhão de coleta incide diretamente no custo da coleta dos resíduos sólidos para os cofres públicos. Portanto, a definição de uma rota que minimize esse trajeto se faz fundamental para reduzir o custo operacional total com esta função.

3. TRABALHOS CORRELATOS

Serão abordados neste capítulo alguns trabalhos, entre artigos, dissertações e teses, relacionados ao problema de cobertura de arcos, ou seja, coleta de resíduos, Problema de Roteamento de Veículos, Problema do Carteiro Chinês, modelos exatos, heurísticas e metaheurísticas para a resolução do problema de roteamento de arcos.

Detofeno (2009) apresenta uma metodologia para o problema de cobertura de arcos via solução aproximada, para otimizar a rota do serviço de coleta de resíduos sólidos urbanos na cidade de Joinville/SC. Utilizou uma combinação de técnicas da área de Pesquisa Operacional, como a heurística de Teitz e Bart para a obtenção das p -medianas necessárias, onde foram designados os pontos a cada mediana através do algoritmo de Gillet e Jonhson, que foi adaptado pelo autor. O Algoritmo do Carteiro Chinês foi utilizado para se obter o roteamento em cada grupo de atendimento. O autor verificou resultados satisfatórios, obtendo um ganho aproximado de 7,83%.

Xavier (2010) em sua pesquisa propõe uma heurística para a minimização do consumo de combustível em rotas de coleta de lixo urbano. O problema caracteriza-se como sendo não-linear, o que dificulta, a elaboração de um algoritmo eficaz para sua solução. O autor utilizou técnicas de programação linear associadas a métodos heurísticos. O método foi validado em um problema-teste e aplicado a uma situação real de cobertura de arcos, onde foi obtida uma redução de 5% no consumo de combustível.

Souza (2009) adaptou o Algoritmo do Carteiro Chinês para um problema real, em um bairro da cidade de Volta Redonda/RJ, aplicando-o na otimização da rota da coleta de resíduos domésticos. Apresentou comparação dos resultados obtidos através da metodologia proposta com uma rota viável escolhida aleatoriamente, concluindo que as vantagens eram significativas e iriam além da mera economia de recursos.

Pascoal Jr. *et al.* (2010) realizaram uma pesquisa para avaliar o roteiro de coleta de resíduos sólidos domiciliares na região central da cidade de Irati/PR. Utilizaram técnicas de rede em ambiente de Sistemas de Informações Geográficas, aplicando ferramentas de roteirização para a obtenção dos resultados. O aplicativo

utilizado foi o *Spring* versão 5.1.3. Os resultados obtidos mostraram uma redução de até 30,84%.

Paes (2004) desenvolveu um estudo sobre a otimização de rotas para a coleta do lixo doméstico, utilizando um algoritmo híbrido baseado na metaheurística *Grasp*. Para tanto, desenvolveu um procedimento de busca local objetivando encontrar um ótimo local que tivesse melhor resultado que a solução construída, chamada pelo autor de Fase de Melhoria. O algoritmo mostrou resultados satisfatórios em um tempo computacional aceitável.

Smiderle; Steiner e Wilhelm (2004) propuseram uma solução para o trabalho de leitura dos medidores das contas de água dos clientes efetuado por uma empresa de saneamento básico do município de Pato Branco/PR. A metodologia utilizada apresentou duas fases: a primeira formando os grupos de atendimento para cada leiturista, onde foi utilizada a meta-heurística Algoritmo Genético e posteriormente para refinar a resposta foi aplicado a heurística clássica de Teitz e Bart e, a segunda, propondo a melhor rota a ser seguida, de modo a minimizar a distância a ser percorrida pelos leituristas, utilizando o algoritmo do Carteiro Chinês.

Costa *et al.* (2001) apresentaram uma metodologia para a otimização do serviço de entrega de correspondências realizado pela Empresa de Correios e Telégrafos (ECT) a partir da aplicação de algumas técnicas da Pesquisa Operacional. O problema foi tratado como um problema de roteamento de veículos. Inicialmente foram definidas áreas de atendimento dos carteiros, chamados distritos postais, e em seguida, foi estabelecido o roteiro de entrega das correspondências, considerando as distâncias reais entre os pontos de entrega. Foram realizados vários testes computacionais, variando-se os algoritmos e parâmetros iniciais, e as respostas comparadas através das distâncias totais e de tempos computacionais, determinando os algoritmos com melhores desempenhos.

Fiório (2006) propõe uma estratégia de solução para a construção otimizada de distritos postais. A metodologia é apresentada em duas fases distintas. A primeira compreende o processo de obtenção dos grafos cíclicos e, a segunda trata do processo de montagem de distrito postal, através da meta-heurística *Simulated Annealing*, com formulações suporte utilizando o Problema do Carteiro Chinês e Carteiro Chinês Capacitado. O objetivo principal do trabalho, de acordo com o autor,

foi obter, de forma rápida e eficiente, o distrito postal otimizado, oferecendo agilidade no processo de distribuição domiciliária de objetos postais.

Lacerda (2003) relatou a análise do uso de um Sistema de Informação Geográfica – SIG como ferramenta para roteirização de veículos de coleta de resíduos sólidos domiciliares. O software utilizado foi o TransCAD, que é um sistema específico para profissionais de transportes, permitindo desenvolver rotas utilizando-se algoritmos que incluem um procedimento de roteirização em arcos. O objetivo foi minimizar a extensão total a ser percorrida pelos coletores. O estudo de caso foi realizado na cidade de Ilha Solteira/SP. Os resultados obtidos com a rotina demonstraram reduções percentuais de até 41% em termos de distância percorrida e de 68% no tempo total do percurso em relação ao serviço atual.

Simonetto e Borenstein (2006) apresentaram a concepção, modelagem e implementação de um sistema de apoio à decisão aplicado ao planejamento operacional da coleta seletiva de resíduos sólidos. Para seu desenvolvimento, foi utilizada a combinação de técnicas advindas da Pesquisa Operacional, quais sejam: a simulação de eventos discretos e algoritmos/heurísticas para o problema da alocação e roteamento de veículos. Na formulação matemática do problema de roteamento de veículos utilizou-se primeiramente o modelo decisório de Fisher e Jaikumar de 1981. As alterações básicas no modelo original se fizeram necessárias devido às características peculiares do problema de coleta seletiva de resíduos. Para a avaliação foram utilizados dados da coleta seletiva de Porto Alegre/RS. Especificamente para o estudo de caso realizado obteve-se uma melhoria de 8,82%, em média, para os percursos e uma redução de 17,89% no número de viagens.

Sherafat (2004) desenvolveu um estudo dos problemas de roteamento com o objetivo de determinar um circuito de custo mínimo para cobrir um conjunto de arcos ou nós de um grafo classificado como misto, sujeito a algumas restrições. Ressaltou que com raras exceções, todos os problemas já formulados a respeito de roteamento de arcos são *NP*-completos. Portanto, para os problemas de maior porte existem apenas soluções aproximadas. Os resultados computacionais confirmaram a eficiência do método na obtenção de soluções próximas do valor ótimo para problemas razoavelmente grandes.

Gomes (2009) apresentou um estudo geral sobre o Problema do Carteiro Chinês nas versões simétrica, orientada e mista. Para apresentar as soluções

exatas das versões do problema, foram utilizadas as implementações exatas dos algoritmos de Sherafat com adaptações, e de Edmonds & Johnson adaptada de Burkard & Derigs. Foi exposto o *software* idealizado para gerar grafos das instâncias e soluções exatas, que pode ser usado como demonstração efetiva da importância do problema, assim como um potencial analista de rotas de problemas semelhantes, ou até mesmo como verificador de instâncias das mais complicadas ou de razoável porte do Problema do Carteiro Chinês.

Godinho Filho *et al.* (2005) propuseram em seu trabalho, após a estruturação e análise de uma revisão bibliográfica, um algoritmo para auxiliar na escolha de métodos adequados para se resolver o Problema do Carteiro Chinês. Essa escolha foi baseada nas características do problema e nos métodos de solução, a saber: orientação do grafo, conectividade do grafo, o grafo ser ou não Euleriano, porte do problema, complexidade e objetivo das soluções. O algoritmo desenvolvido foi ilustrado por meio de sua utilização na escolha de métodos para resolução do Problema do Carteiro Chinês em caso real, na cidade de São Carlos/SP. Os resultados se mostraram satisfatórios para os objetivos da pesquisa.

Apaydin e Gonullu (2007) propuseram um estudo para a otimização da rota de coleta de resíduos domiciliares na cidade de Trabzon, Turquia. A cidade possui aproximadamente 185.000 habitantes. As rotas foram geradas por um sistema de informações geográficas, onde foram cadastrados 777 pontos na cidade. Quando foram obtidas as rotas otimizadas, essas foram comparadas as rotas originais, resultando em um ganho de 24,7% nas distâncias e 44,3% no tempo da coleta dos resíduos sólidos.

4. FUNDAMENTAÇÃO TEÓRICA

4.1 GRAFOS

A maioria das definições e termos relacionados à Teoria de Grafos, usada nesse trabalho é padrão e pode ser encontrada em vários trabalhos de roteamento de arcos, como por exemplo, na pesquisa de Christofides (1981). Entretanto, serão definidos os termos usados com maior freqüência no decorrer do trabalho, ou que não são considerados como sendo padrões.

Uma rede é representada por um grafo $G = (N, A, E)$, em que $N = \{x_1, x_2, \dots, x_n\}$ representa o conjunto de nós (ou vértices), $A = \{a_1, a_2, \dots, a_r\}$ o conjunto de arcos (com direcionamento) e $E = \{e_1, e_2, \dots, e_m\}$ o conjunto de arestas (sem direcionamento). A cardinalidade de cada um desses conjuntos é representada por $n = |N|$, $r = |A|$ e $m = |E|$ respectivamente o conjunto, definido por $L = A \cup E$, pode ter a denominação de *links* do grafo. Portanto, um *link* $l \in L$ pode ser um arco, ou uma aresta.

Um *link* pode ser descrito pelo par de nós (x_i, x_j) que indicam seus nós terminais. Quando se trata de uma aresta, a ordem de nós terminais, nesta notação, é irrelevante. Porém, no caso de arcos a ordem é do nó inicial para o final. Sempre que for conveniente, é possível considerar uma aresta (x_i, x_j) como um par de arcos contrariamente orientados (x_i, x_j) e (x_j, x_i) . Dois nós conectados por um *link* são chamados de adjacentes. A cada *link* (x_i, x_j) de um grafo pode ser associado um custo d_{ij} . Uma matriz $D = [d_{ij}]$ é a matriz de custos associada ao grafo, onde d_{ij} é o custo do *link* $(x_i, x_j) \in L$ e $d_{ij} = \infty$ se $(x_i, x_j) \notin L$.

Se $E = \phi$ então G é um grafo orientado; se $A = \phi$, ele é um grafo não-orientado; e se $E \neq \phi$ e $A \neq \phi$, o grafo é denominado de grafo misto.

Quando uma malha urbana ou uma rede rodoviária é representada por um grafo, os arcos representam os trechos de ruas de mão única e as arestas, os de mão dupla. Os nós são os cruzamentos entre as ruas. Todavia, um grafo é uma estrutura mais genérica do que uma malha urbana.

Um grafo completo é um grafo orientado (ou não-orientado) em que para quaisquer nós $x_i \in N$ e $x_j \in N$ o arco $(x_i, x_j) \in A$ ou a aresta $(x_i, x_j) \in E$.

Em um grafo não-orientado, para cada nó x_i define-se o grau $g(x_i)$ como o número de arestas que incidem no nó x_i . Quando o grafo é orientado, define-se como o grau de entrada, $g_e(x_i)$, o número de arcos cujos nós finais são o nó x_i . Analogamente, o grau de saída $g_s(x_i)$, é o número de arcos, cujos nós iniciais são o nó x_i . É trivial verificar que a soma de graus de entrada de todos os nós de um grafo é igual a soma dos graus de saída.

Um caminho é uma seqüência de *links*, respeitando sua orientação, onde o nó final de um arco ou aresta é o inicial do próximo. Desta forma um caminho $C(x_i, x_j)$ começa em um nó x_i e termina em x_j , onde x_i e x_j não são necessariamente adjacentes. Considera-se x_i e x_j como sendo os nós inicial e final do caminho, respectivamente, e conectados nesta ordem pelo caminho. Uma cadeia é uma seqüência de *links*, também ligando dois nós não necessariamente adjacentes, sem considerar-se a orientação dos arcos.

Um circuito é um caminho em que o nó inicial coincide com o nó final.

Um grafo é dito fortemente conexo, se para qualquer par ordenado de nós x_i, x_j existe pelo menos um caminho que conecta x_i a x_j . Essa definição implica que num grafo fortemente conexo, dois nós quaisquer são mutuamente acessíveis. Um grafo é dito conexo, ou fracamente conexo, se para qualquer par de nós (x_i, x_j) existe pelo menos uma cadeia que conecta x_i a x_j . Se pelo menos para um par de nós, tal cadeia não existe, então o grafo é dito desconexo.

Dado um grafo $G = (N, L)$, um grafo parcial G_p de G é o grafo (N, L_p) com $L_p \subset L$. Portanto, um grafo parcial é um grafo com o mesmo número de nós, porém com apenas um subconjunto próprio de *links* do grafo original.

Um subgrafo G_s de G é o grafo (N_s, L_s) com $N_s \subset N$ e $L_s = \{(x_i, x_j) | (x_i, x_j) \in L, x_i \in N_s, x_j \in N_s\}$. Portanto, um subgrafo é um grafo contendo um subconjunto de nós, porém com todos os *links* que conectam estes nós no grafo original.

Num grafo $G = (N, L)$, um **componente** é definido como um subgrafo $K = (N_k, L_k)$, em que K é conexo e não existe nenhum *link* (x_i, x_j) tal que $x_i \in N_k$ e $x_j \in N \setminus N_s$. Se G é conexo, então ele é formado por um único componente; um grafo desconexo é formado por mais de um componente.

Um circuito que passa por todos os nós de um grafo, sem que repita o mesmo nó mais de uma vez, é denominado de Circuito Hamiltoniano. Nem todo grafo contém um circuito hamiltoniano; porém quando possui, é chamado de Grafo Hamiltoniano.

Um circuito que passa por todos os *links* de um grafo, sem que repita o mesmo *link* mais de uma vez, é denominado de Circuito Euleriano. Também, nem todo grafo contém um circuito euleriano, e quando possui, ele é chamado de Grafo Euleriano, ou Grafo Unicursal. Um circuito que passa por todos os *links* de um grafo, pelo menos uma vez, é denominado de Circuito de Carteiro. Todo grafo fortemente conexo contém um circuito de carteiro.

4.2 PROBLEMAS DE ROTEAMENTO DE ARCOS

Paes (2004) afirma que o objetivo dos problemas de roteamento em arcos (PRA's) é determinar um circuito de custo mínimo em um subconjunto de arcos de um grafo, tal que todos os arcos deste subconjunto sejam atravessados ao menos uma vez, podendo o circuito ter ou não restrições. Neste caso, pode-se estar interessado em rotear um veículo de cada vez ou m veículos ao mesmo tempo.

Estes problemas surgem em diversos contextos práticos onde há a necessidade de otimizar a rota como, por exemplo, na entrega de cartas, na coleta do lixo doméstico, na varrição e lavagem de ruas, na inspeção de redes elétricas, de água ou gás, na fiscalização de linhas de ônibus (Canales e Morales, 2002), entre outros. Para Sherafat (2004) é um serviço dispendioso que envolve diversas questões combinatórias como roteirização, distritamento, localização de facilidades e atribuição de cada veículo a uma das facilidades.

Grandes quantias não somente em dinheiro, mas também em tempo são gastas a cada ano por governos e empresas privadas no planejamento e execução destas operações.

Outra forma de classificar os Problemas de Roteamento é basicamente separando os clássicos problemas de roteamento em grafos das demais variantes – mais tipicamente voltada para o atendimento de casos práticos particulares. Goldberg (2005) afirma que nesse sentido os problemas de roteamento em geral poderiam ser classificados em duas formas: Roteamento em Grafos e Roteamento de Veículos propriamente ditos. A classe geral dos problemas de Roteamento em

Grafos seria constituída pelas subclasses: Problema de Roteamento em Nós (associados aos ciclos Hamiltonianos) e Problema de Roteamento em Arcos (associados aos ciclos Eulerianos).

Goldbarg (2005) afirma que os problemas de roteamento de veículos abordam basicamente a determinação de sequências de visitas que objetivem atender uma determinada função objetivo. As visitas podem tanto estar associadas às ligações (arestas) ou aos pontos de visita (nós) do grafo que representa as possíveis conexões entre os pontos de visita (ou pontos de ligações entre as arestas).

Os problemas de roteamento de veículos estão entre os mais complexos da área de otimização combinatória. Pelo grande número de variáveis, diversidade de restrições e objetivos apresentados impõe-se o exame de uma cuidadosa taxonomia para seu melhor entendimento. Uma das taxonomias clássicas é a proposta por Bodin e Golden (1981) *apud* Goldbarg (2005). Segundo os autores o Problema de Roteamento de Veículos pode ser classificado seguindo os seguintes critérios:

1. Tempo para servir um determinado nó ou arco;
2. Número de domicílios;
3. Tamanho da frota de veículos;
4. Tipo de frota disponível;
5. Natureza da demanda e parâmetros;
6. Localização da demanda;
7. Grafo de substrato;
8. Restrições na capacidade de veículos;
9. Tempo de roteamento;
10. Custos;
11. Operações;
12. Objetivo;

Restrições nas capacidades dos arcos.

4.3 GRAFOS DE EULER

Uma trilha no grafo G é chamada trilha de *Euler* se ela incluir toda aresta desse grafo G .

Denomina-se *tour* um passeio fechado em G que inclui toda aresta de G pelo menos uma vez.

O *tour de Euler* em G é um *tour* que inclui cada aresta de G exatamente uma vez. Assim, um *tour* de Euler é uma trilha de Euler fechada.

O grafo G é chamado *grafo de Euler* se tiver um *tour* de Euler.

Seja G um grafo cujo grau de todo vértice é pelo menos dois. Então G contém um ciclo.

Somente se o grau de todo vértice de G for par, o grafo conexo G é um grafo de Euler.

Seja G um grafo conexo e sejam i e j vértices distintos em G . Então existe uma trilha de Euler em G de i e j a j e i se e somente se os graus de i e de j forem ímpares e os graus de todos os outros vértices em G , pares.

De acordo com Micoletti e Hruschka Jr. (2006) o nome Euler para esse tipo especial de trilhas, *tour* e grafo deve-se ao matemático suíço Leonhard Euler (1707-1783), que foi a primeira pessoa a resolver um problema conhecido como “As Pontes de Königsberg”. Tal problema consiste em saber se um indivíduo pode, a partir de um determinado ponto, passar em cada uma das sete pontes exatamente uma vez e voltar ao ponto de origem. A figura 6 mostra um mapa do trecho da cidade de Königsberg, mostrando o rio Pregel e as sete pontes.

Euler equacionou o problema a ser resolvido com o seguinte questionamento: “É possível percorrer o diagrama a partir de qualquer um dos pontos A, B, C ou D, usando os arcos apenas uma vez, e voltar ao ponto de início?” A resposta é não, pois o grafo não contém a trilha de Euler e não há possibilidade de percorrer as sete pontes apenas uma vez voltando ao ponto de partida.

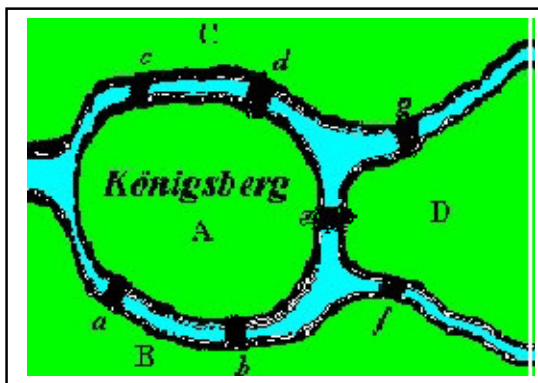


FIGURA 6: PONTES DE KÖNIGSBERG
 FONTE: www.mat.uc.pt/~alma/escolas/pontes, PAES (2004)

O Problema das Pontes de Königsberg inspirou o estudo do problema do carteiro, em que o objetivo é determinar um caminho de comprimento mínimo cobrindo cada arco ao menos uma vez. O problema foi relatado de forma simplificada por Guan (1962) *apud* Paes (2004): “Um carteiro tem de cobrir sua rota e depois retornar ao Posto de Correio. O problema é encontrar a menor distância a ser percorrida pelo carteiro”.

4.4 CIRCUITOS EULERIANOS (CE)

Dado um grafo não-orientado , fortemente conexo, um circuito que contém todas as arestas do grafo, sem que repita a mesma aresta mais de uma vez, é denominado de Circuito Euleriano. E como foi analisado por Sherafat (2004), nem todo grafo contém um circuito euleriano; quando possui, ele é chamado de grafo euleriano, ou grafo unicursal. O teorema básico sobre a existência de um circuito euleriano, em um grafo não-orientado, é o seguinte:

TEOREMA – *Um grafo fortemente conexo contém um circuito euleriano, se, e somente se, o grafo não tem nenhum nó de grau ímpar.*

Prova:

Seja G um grafo euleriano. Logo ele contém um circuito euleriano. Em cada ocorrência de vértice desse caminho existe uma aresta que chega nesse vértice e outra que sai desse vértice. Como toda aresta faz parte do caminho, isto é, nenhuma aresta fica fora do caminho, necessariamente o número de arestas incidentes em cada vértice é par.

Como todo vértice possui grau par, então na construção de um caminho sempre é possível chegar e sair de um vértice passando por arestas diferentes, que ainda não tenham sido utilizadas. Assim, é possível sair do vértice v e retornar a ele sem repetição de arestas (Lema 1). Seja C_1 um circuito contendo v construído de maneira arbitrária. Logo, se C_1 contém todas as arestas de G , temos um circuito euleriano. Senão, retiramos de G todas as arestas que fazem parte de C_1 . No grafo resultante G' , todos os vértices também possuem grau par e necessariamente um deles faz parte de C_1 , senão o grafo não seria conexo. Recomeçemos o mesmo processo com o grafo G' , partindo de um vértice comum com C_1 , obtendo assim um novo circuito C_2 .

Para ilustrar o teorema, a figura 7 mostra que dois circuitos que têm um vértice em comum podem formar um circuito único, chegando no vértice comum em um dos dois circuitos, continuando o percurso no outro circuito. Continuando esse processo, necessariamente obteremos um circuito único que contém todas as arestas de G . Logo o grafo é euleriano.

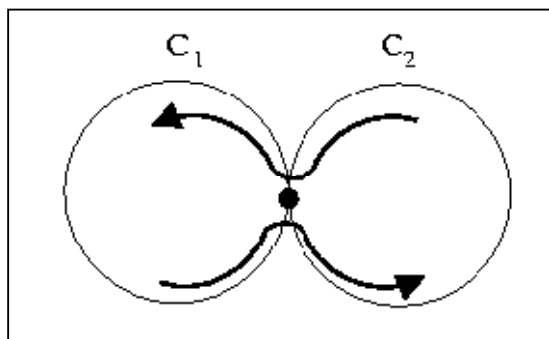


FIGURA 7: DOIS CIRCUITOS COM UM VÉRTICE EM COMUM

FONTE: https://docs.google.com/viewer?a=v&q=cache:r5dqcm6te6qj:www.din.uem.br/~ademir/grafos/arquivos/e1-grafoeuler_hamilton.pdf

Um aspecto muito interessante dessa prova é que ela sugere um algoritmo para identificar um circuito euleriano. Considere por exemplo o grafo ilustrado na figura 8(a).

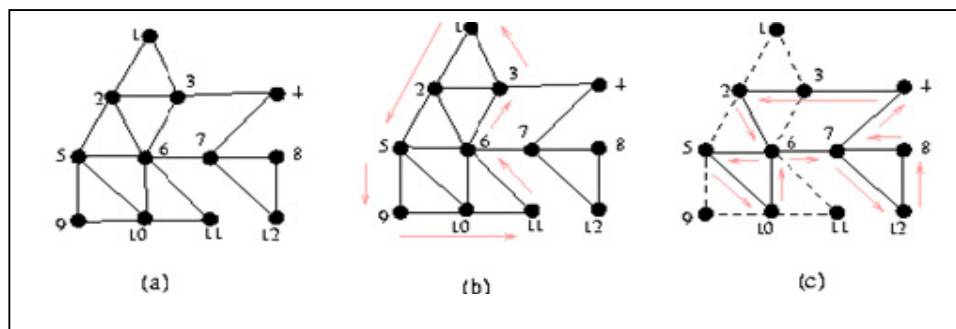


FIGURA 8: CIRCUITO EULERIANO

FONTE: https://docs.google.com/viewer?a=v&q=cache:r5dgcm6te6qj:www.din.uem.br/~ademir/grafos/arquivos/e1-grafoeuler_hamilton.pdf

Começa-se pelo vértice 1 e escolhe-se aleatoriamente uma aresta nunca visitada. Seja o próximo vértice, o extremo desse arco. Procura-se outros arcos não visitados e repete-se o procedimento até retornar-se ao vértice 1, quando não há mais arcos a usar. A figura 8(b) mostra um circuito obtido, que consiste na seqüência 1, 2, 5, 9, 10, 11, 6, 3 e 1. Como sobram arestas não percorridas, deve-se recomeçar a partir de um vértice desse circuito.

Supondo que o vértice 6 foi o escolhido, pode-se obter, como ilustrado na figura 8(c), o circuito 6, 7, 12, 8, 7, 4, 3, 2, 6, 5, 10, 6. Combinando esse circuito com o anterior, obtém-se um novo circuito 1, 2, 5, 9, 10, 11, 6, 7, 12, 8, 7, 4, 3, 2, 6, 5, 10, 6, 3, 1.

Como esse circuito cobre todos os arcos do grafo, não é preciso continuar o processo: o circuito euleriano já está definido.

Esse algoritmo é conhecido como o algoritmo de Hierholzer e será descrito em detalhes na seção 5.8.

A necessidade da condição estabelecida no teorema acima pode ser verificada, tendo-se em vista que qualquer circuito euleriano deve usar uma aresta para entrar em cada nó, e uma outra para partir do mesmo. Considerando-se que todas as arestas devem ser percorridas uma única vez, então deve haver um número par de arestas incidentes a cada nó, isto é, o grau de todo nó deve ser par.

A suficiência da condição poderá ser estabelecida usando uma prova construtiva. É trivial verificar que se G é fortemente conexo e todos os seus nós são de grau par, nele é possível a construção de um circuito euleriano.

Uma forma de construir um circuito euleriano é começar o traçado em qualquer nó arbitrário x_i , percorrer uma aresta não utilizada para chegar a um outro nó x_j , e a partir deste prosseguir para um outro nó, repetindo-se o processo. Como o grau de todos os nós é par, então cada vez que se entra num nó, deve haver uma aresta não utilizada, disponível para sair. Portanto, o processo terminaria forçosamente no mesmo nó de partida x_i , quando se completa um circuito C . Se C contiver todas as arestas, então um circuito euleriano é obtido; caso contrário, identifica-se o grafo parcial de G_p , formado pelas arestas não utilizadas do grafo G . Pode-se verificar que todos os nós de G_p também seriam de grau par. Escolhe-se um nó x_k que é terminal de uma aresta de G_p , tal que x_k esteja contido em C . Tal nó existe, caso contrário, G não seria conexo. A partir de x_k será construído um novo circuito C' , o qual será acoplado a C , a partir de x_k , formando um único circuito. Este procedimento será continuado, até a obtenção de um circuito euleriano completo no grafo G .

Edmonds (1973) utilizou uma abordagem baseada em construção de uma arborescência, para construir um circuito euleriano. Seja qual for o método escolhido, quando o grafo é euleriano, a obtenção de um circuito euleriano é uma tarefa trivial.

Uma propriedade notável de um grafo, quando não é euleriano, é a de possuir um número par de nós de grau ímpar. Este fato pode ser explicado pelo número total de incidências nos nós do grafo: se o grafo possui m arestas e cada aresta está em contato com dois nós, então a soma de graus de todos os nós é $2m$, que é um número par. Seja g_p a soma de graus dos nós de grau par, e g_i a soma para os nós de grau ímpar. Portanto:

$$g_p + g_i = 2m \quad (1)$$

Se g_p é um número par, então g_i deve ser um número par.

4.4.1 Problema do Carteiro Chinês (PCC)

O PCC é um problema de otimização que objetiva cobrir com uma rota todos os arcos de um grafo, minimizando a distância total percorrida. A rota do carteiro distingue-se do circuito euleriano por nesta ser permitida, se necessária, a repetição de arestas. Pode ser aplicado a grafos não-direcionados (ruas de mão dupla), direcionados (ruas de mão única) ou mistos (algumas ruas de mão dupla e outras de mão única).

Em todos os casos, de acordo com Sherafat (2004), foi estabelecida a propriedade de Unicursalidade como o centro de atenção. Um grafo conexo $G(V, A)$ é dito unicursal (ou Euleriano) se existe em G um ciclo ou circuito contendo todos os arcos e/ou arestas exatamente uma única vez. As condições necessárias e suficientes para unicursalidade são as seguintes:

- a) Se G é não-orientado, todos os nós devem ter grau par i , isto é, um número par de arestas incidentes a cada nó;
- b) Se G é orientado, o número de arcos entrando e saindo em cada nó devem ser iguais;
- c) Se G é misto, cada nó deve ser incidente a um número par de arcos, orientados ou não; e mais ainda para cada $S \subset V$, a diferença entre o número de arcos orientados de S para $S \setminus V$ e o número de arcos orientados de $V \setminus S$ para S deva ser o menor ou igual ao número de arestas que ligam S e $V \setminus S$.

Esta última é chamada de condição de conjuntos balanceados e a que torna complexa qualquer abordagem para a solução deste caso.

Para Sampaio (2007) deve-se considerar o caso de um carteiro responsável pela correspondência de uma área da cidade. O carteiro deverá sempre começar o percurso em uma esquina (nó) inicial, e passar por todas as ruas (arestas) e retornar ao nó inicial. Este caracteriza um dos mais antigos problemas da teoria dos grafos, que é a determinação de um passeio sobre um grafo G que contenha todas aresta de G exatamente uma vez. Tal circuito é denominado circuito euleriano, pelo fato de Euler ter sido o primeiro a reportar um estudo sobre a sua determinação, no ano de 1736. Assim, para se realizar um ciclo euleriano em um grafo, este deverá ser modificado de modo a tornar de grau par todos os seus nós de grau ímpar. Para

isso, é necessário combinar dois a dois todos os seus nós de grau ímpar. Esse problema de combinação é chamado de “Pairwise Matching”, e foi resolvido por Edmonds (1973).

De maneira geral, o PCC somente pode ser resolvido em tempo não polinomial. Contudo, os problemas específicos onde $A = \emptyset$ e onde $E = \emptyset$ podem ser resolvidos em tempo não polinomial utilizando algoritmos de emparelhamento perfeito (algoritmo de Edmonds e algoritmo Húngaro, respectivamente) e de caminho mais curto sobre grafos (tipicamente o algoritmo de Dijkstra) (GOLDBARG, 2005).

O Algoritmo do Carteiro Chinês pode ser encontrado na literatura tendo a seguinte sequência:

Passo 1: Ler o grafo $G(V, A)$.

Passo 2: Se todos os nós em G , o grafo original, possuem grau par então determinar um ciclo euleriano em G e Fim.

Passo 3: Organizar um grafo k_n da seguinte forma:

Passo 4: Reunir todos os vértices de grau ímpar no grafo k_n e

Passo 5: Associar a cada par de vértices i e j no grafo, uma aresta (i, j) com peso igual ao caminho mais curto que liga i a j no grafo G .

Passo 6: Determinar o 1-*matching* mínimo em k_n , M^* . Para cada aresta pertencente a M^* associar uma nova aresta em G no caminho mínimo que ela representa, obtendo um grafo G_a .

Passo 7: Determinar a solução do carteiro chinês que é representada por um ciclo euleriano em G_a .

4.4.1.1 Problema do Carteiro Chinês Orientado (PCCO)

Se $N' = \emptyset$, $A' = A$ e $E = \emptyset$, tem-se o caso orientado do Problema de Carteiro Chinês. Nesse caso, um circuito de carteiro de custo mínimo precisa ser construído num grafo completamente orientado (SHERAFAT, 2004).

A condição necessária e suficiente para existência de um circuito euleriano num grafo orientado é que, além de ser fortemente conexo, o grafo deve ser simétrico, isto é, para cada nó o grau de entrada deve ser igual ao grau de saída. Quando em alguns nós o número de arcos de entrada diverge do número de arcos

de saída, o grafo não é unicursal e, para torná-lo assim é necessário acréscimo de cópias apropriadas de alguns arcos.

Beltrami e Bodin (1974) mostram que um grafo euleriano de custo mínimo pode ser construído resolvendo um trivial Problema de Transporte. Nesse caso, os nós com excesso de entrada serão considerados como *suprimento* e os com excesso de saída como *demanda*. A solução do Problema de Transporte indica qual nó de suprimento deva ser associado em qual demanda. As cópias dos arcos devem ser acrescentadas ao grafo, ao longo dos caminhos mínimos que ligam os nós de suprimento aos de demanda na solução do Problema de Transporte.

O Algoritmo para PCCO é apresentado a seguir:

Passo 1: Tomar G , com as informações de distância, e testar sua conectividade.

Passo 2: Se for conexo, transformar as informações de distância para informações de fluxo, com os limites de fluxo máximo e mínimo permitidos por arco, e aplicar um algoritmo de fluxo em redes de custo mínimo.

Passo 3: Após o passo e obtenção de G' , onde os resultados de fluxo indicam a quantidade de vezes que cada arco será repetido em G' , aplicar o resultado encontrado para a formação da rota euleriana.

A complexidade computacional desse algoritmo é da ordem $O(mn^2)$. Edmonds e Johnson (1973) apresentam um algoritmo alternativo que corresponde à solução de fluxo de custo mínimo no grafo, cuja complexidade é de $O(n^3)$. Lin e Zhao (1988) apresentam uma abordagem diferente, desenvolvida na base do Teorema de Folga Complementar da Programação Linear, que tem a idéia do algoritmo baseada nas formulações primal e dual do problema. A complexidade desta é de $O(kn^2)$, onde k depende da estrutura do grafo. Eles demonstram que para os grafos esparsos k é menor que m e n , portanto para estes o método funciona melhor que os outros mencionados acima.

A formulação matemática para o PCCO é a seguinte:

$$\text{Min } \sum_{(i,j) \in A} l_{ij} x_{ij} + \sum_{(i,j) \in A'} l_{ij} x_{ij} \quad (6)$$

$$\text{Sujeito à: } \sum_{j:(i,j) \in A} x_{ij} + \sum_{j:(i,j) \in A'} x_{ij} = \sum_{j:(j,i) \in A} x_{ji} + \sum_{j:(j,i) \in A'} x_{ji}, \forall i \quad (7)$$

$$x_{ij} + x_{ji} \geq 1, \forall (i,j) \in A' \quad (8)$$

$$x_{ij} \geq 0, \forall (i,j) \in A \quad (9)$$

$$x_{ij} \text{ int.}, \forall (i,j) \in A \quad (10)$$

$$x_{ij} \text{ int.}, \forall (i,j) \in A' \quad (11)$$

onde: l_{ij} representa o custo do arco ij e x_{ij} representa o arco ij .

A equação (6) representa a função objetivo envolvendo o custo de cada variável a ser minimizada, a restrição (7) implica que $G(V, B)$ será um grafo Euleriano simétrico (número de arestas entrando em cada vértice é igual ao número de arestas saindo), a restrição (8) implica na passagem de pelo menos uma vez em cada aresta de E e as restrições (9) e (10) garantem que todas as variáveis pertencentes à A e A' sejam inteiras e positivas (XAVIER *et. al.*, 2004).

Um problema interessante associado ao PCCO consiste em determinar um *subgrafo euleriano de custo máximo* num dado grafo orientado. Richey e Parker (1991) têm demonstrado que este é um problema *NP-hard* no caso genérico e sugerem um algoritmo que resolve o problema num caso particular que foi estudado.

4.4.1.2 Problema do Carteiro Chinês Misto (PCCM)

Um grafo misto contém tanto arcos orientados como arcos não orientados. Assim como nas redes direcionadas, nas redes mistas também o grafo deve ser totalmente conectado para que o PCC tenha solução.

A condição para que uma rede mista tenha um roteiro Euleriano é que esta seja par e balanceada. Uma rede par é aquela na qual o número total de arcos (direcionados ou não) de cada um de seus nós é par, ou seja, a rede não deve ter nós de grau ímpar. Já uma rede balanceada é aquela na qual, para todo nó genérico S , a diferença entre o número de arcos direcionados que saem deste nó e o número de arcos direcionados que entram neste nó deve ser menor ou igual ao número de arcos não direcionados que ligam S a qualquer nó (NORBERT & PICARD, 1991;

EISELT *et al.*, 1995; GODINHO FILHO, 2005). Estas duas condições são versões modificadas do Teorema de Euler para o caso de redes mistas.

Esta formulação é obtida a partir de $A' = \phi$, $A' = A$ e $E' = E$. O PCCM é a versão do PCC que mais se aproxima à realidade das malhas urbanas, não obstante, é a mais difícil do ponto de vista da solução. Papadimitriou (1976) demonstrou que PCCM é NP-completo (não existem soluções com complexidade polinomial, os resultados são obtidos via soluções/algoritmos aproximadas). Portanto, não existem soluções exatas com complexidade polinomial para este caso. O caminho natural é via soluções aproximadas. Entretanto, poucos métodos aproximados foram desenvolvidos para o caso.

Caso o grafo misto seja Euleriano, o problema passa a ser determinar o roteiro Euleriano neste grafo. Eiselt *et al.* (1995) sugerem uma metodologia em três etapas para se encontrar um roteiro Euleriano em um grafo misto Euleriano:

Passo 1: Atribuir direção a alguns arcos de tal forma que o grafo se torne simétrico, isto é, em cada nó, o número de arcos que chegam é igual ao número de nós que saem.

Passo 2: Atribuir direção aos arcos restantes.

Passo 3: uma vez que o grafo esteja completamente direcionado, encontrar o circuito Euleriano utilizando um algoritmo para grafos direcionados.

Caso o grafo não seja Euleriano deve-se proceder a uma abordagem semelhante à descrita para os casos não direcionados e direcionados, isto é, duplicar um número suficiente de arcos de tal forma que o grafo se torne Euleriano. Muitos autores usam a programação inteira para determinar esse aumento do grafo misto a um custo mínimo, porém os resultados computacionais são apenas esboçados e, portanto, comparações entre as abordagens são difíceis. A única conclusão a respeito de todos estes modelos é que eles são ineficientes computacionalmente e que somente problemas de pequeno e médio porte podem ser resolvidos (PEARL & LIU, 1995 e EISELT *et al.*, 1995).

Para o grafo misto $G(V, E, A)$ e seu respectivo grafo aumentado $G(V, B)$, considerar $c_j \geq 0$ o custo associado para passar no arco direcionado $b_j \in B$, e x_j o número de vezes que é necessário passar por b_j . O problema do carteiro chinês misto pode ser formulado como:

$$\text{Min } \sum_{(i,j) \in A} l_{ij} x_{ij} + \sum_{(i,j) \in A'} l_{ij} x_{ij} \quad (12)$$

$$\text{Sujeito à: } \sum_{j:(i,j) \in A'} x_{ij} + \sum_{j:(i,j) \in A} x_{ij} = \sum_{j:(j,i) \in A} x_{ji} + \sum_{j:(j,i) \in A'} x_{ji}, \forall i \quad (13)$$

$$x_{ij} + x_{ji} \geq 1, \forall (i,j) \in A' \quad (14)$$

$$x_{ij} \geq 0, \forall (i,j) \in A \quad (15)$$

$$x_{ij} \text{ int.}, \forall (i,j) \in A \quad (16)$$

$$x_{ij} \text{ int.}, \forall (i,j) \in A' \quad (17)$$

Sobre as restrições, segundo Xavier (2004), pode ser verificado que a equação (12) trata-se da função objetivo a ser minimizada, a restrição (13) garante que o grau de cada aresta do vértice aumentado seja zero, a restrição (14) implica que $G(V, B)$ será um grafo Euleriano simétrico (número de arestas entrando em cada nó é igual ao número de arestas saindo), a restrição (15) implica na passagem de pelo menos uma vez em cada aresta de E e as restrições (16) e (17) impõem que as variáveis pertencentes à A e A' sejam inteiras e positivas.

Generalizando, devido à sua complexidade, o PCC pode ser resolvido somente em tempo não polinomial. Contudo, os problemas específicos onde $A = \emptyset$ e onde $E = \emptyset$ podem ser resolvidos em tempo não polinomial utilizando algoritmos de emparelhamento perfeito (algoritmo de Edmonds e algoritmo Húngaro, respectivamente) e de caminho mais curto sobre grafos (tipicamente o algoritmo de Dijkstra) (PAES, 2004).

Como o PCCM é um problema *NP*-Completo e, portanto, encontrar a solução ótima apresenta grande dificuldade, a literatura específica sobre o tema tem proposto soluções heurísticas para encontrar soluções aproximadas para o problema. Algumas heurísticas foram apresentadas com modificações por Pearn & Liu (1995). Esses novos algoritmos conseguiram alguns resultados computacionais melhores do que os algoritmos originais.

4.4.1.3 Problema do Carteiro Chinês com Vento (PCCV)

Existe uma formulação semelhante a PCC, com $N' = \emptyset, A = \emptyset$ e $E' = E$, mas com um grau de liberdade a mais: cada aresta pode ser percorrida em ambos os sentidos com custos desiguais. Esta situação pode ocorrer quando, por exemplo, um

dos sentidos é a subida de uma ladeira, e o outro a descida no sentido contrário; ou, um dos sentidos é a favor do vento, e o outro contra (SHERAFAT, 2004).

O PCCV foi introduzido inicialmente por Minieka (1979) e recentemente recebeu atenção de alguns pesquisadores. É considerada uma versão interessante do PCC, não pela sua real aplicação, mas porque todos os três casos anteriores de PCC podem ser derivados desta simples generalização. Por exemplo, um PCCM pode ser formulado como PCCV; basta fixar para as arestas custos iguais, para ambos os sentidos, e para os arcos considerar o custo normal no sentido da orientação do arco, e custo igual a ∞ no sentido contrário.

Guan (1984) provou que computacionalmente PCCV é equivalente a PCCM, portanto é NP-completo. Ao mesmo tempo, ele sugeriu uma solução aproximada para o caso: em cada aresta o custo é convertido para o valor médio das duas orientações. O problema resultante é resolvido como PCC, usando algoritmo de emparelhamento de custo mínimo. O grafo euleriano obtido é decomposto arbitrariamente em sub-ciclos não orientados. Em seguida, cada sub-ciclo é orientado escolhendo a orientação que produz o menor custo. Pearn e Li (1994) mostraram que esta é uma ∞ -aproximação para o PCCV. Isto é, no pior caso, a razão (Solução de Guan) / (Solução Ótima) tende a ∞ .

Pearn e Li (1994) desenvolveram uma versão melhorada para o algoritmo de Guan, a qual reduziu em média 4,2% o custo, em relação ao algoritmo de Guan original. Os autores sugeriram, também, uma nova estratégia para o algoritmo de Win, a qual denominaram de algoritmo de *Win Reverso*. A nova estratégia demonstrou ser mais eficiente que a versão original, apenas quando o grafo é *próximo a ímpar* (predominantemente composto de nós de grau ímpar) e com “forte ventania” (diferença acentuada entre custos das duas orientações das arestas). Os testes computacionais foram realizados em grafos de até 50 nós e 377 arestas.

Uma estrutura poliedral de PCCV foi estudada por Grötschel e Win (1992), na qual foi desenvolvido um algoritmo de planos de corte para o problema, baseado numa descrição linear parcial do poliedro. A solução foi testada em grafos de até 264 nós e 489 arestas. Dos 36 casos testados, em 31 o método encontra uma solução ótima e inteira; quando o algoritmo falha em achar valores inteiros, eles recomendam uma estratégia de arredondamento para os valores finais.

4.4.1.4 Problema do Carteiro Rural (PCR)

O *Problema de Carteiro Rural* constitui uma classe de problemas de roteamento de arcos em que $N' = \emptyset, A' \subset A$ e/ou $E' \subset E$. Ao contrário de todas as versões de PCC, em PCR apenas um subconjunto de *links* necessita de serviço. Portanto é uma formulação mais genérica e mais realista para os problemas de distribuição que contém o PCC como caso particular. A exemplo do PCC, o PCR pode ser formulado num grafo não-orientado, num grafo orientado – PCRO, ou num grafo misto – PCRM. Ao contrário do PCC, o PCR é NP-hard, mesmo nos casos orientado e não-orientado (SHERAFAT, 2004).

4.4.1.5 Problema da Empilhadeira (PE)

O problema da empilhadeira é um caso particular de PCR. Definido num grafo $G(N, A, E)$, com $A' = A, E \neq \emptyset, E' = \emptyset$ e $N' = \emptyset$. O problema é determinar, num grafo misto, o menor circuito que inclui todos os arcos, enquanto nenhuma aresta é requerida. Os arcos podem ser vistos como os movimentos a serem executados por uma empilhadeira, cada um exatamente uma vez, numa direção específica. Se os custos dos arcos forem nulos, o problema se reduz a um PCV, portanto, PE é um problema NP-hard. Frederickson *et al.* (1978) propuseram duas heurísticas em que é preciso que o grafo G satisfaça duas condições:

- a) Cada nó é incidente por pelo menos um arco de A , e
- b) Os custos aplicados às arestas satisfazem a desigualdade triangular.

Se G não satisfaz estas duas propriedades, ele pode ser transformado num grafo equivalente que satisfaça as mesmas. As duas heurísticas são denominadas de *Largearcs* e *Smallarcs*. A primeira apresenta resultados melhores, quando o custo total dos arcos é grande, comparado com o custo total numa solução ótima. A segunda funciona melhor numa situação inversa.

A complexidade de ambas as heurísticas é $O(\max\{|N|^3, |A|^3\})$. O método *Largearcs* produz uma rota r para o PE, cujo comprimento $dL(r)$ satisfaz a equação (18):

$$dL(r) \leq 3z^* - 2d(A) \quad (18)$$

Onde z^* é o valor da solução ótima de PE e $d(A)$ é a soma de custos de todos os arcos de A . Para o método Smallarcs, o comprimento $dS(r)$ da rota gerada satisfaz a equação (19):

$$dS(r) \leq \frac{1}{2}(3z^* + d(A)) \quad (19)$$

Uma forma para melhorar o pior caso desses algoritmos é a aplicação de ambos e escolhendo o melhor resultado. Desta forma, o pior caso se reduz para um valor representado na equação (20):

$$d(r) \leq \frac{9z^*}{5}. \quad (20)$$

4.4.1.6 Problema Geral de Roteamento (PGR)

Uma formulação possível para os problemas de roteamento é quando $N' \neq \emptyset$, $A' \subset A$ e $E' \subset E$, conhecida como o problema geral de roteamento. A diferença entre PGR e PCRM é o fato de que no PGR, além de nós implicitamente requeridos, isso é, os nós que incidem aos *links* requeridos, eventualmente outros nós precisam ser servidos. É o caso mais genérico entre todas as formulações não-capacitadas de problemas de roteamento. Portanto, todos os casos acima estudados podem ser formulados como casos especiais do PGR. O PGR foi proposto e estudado inicialmente por Orloff (1974).

Embora possam ser formulados problemas reais de distribuição na forma do PGR, a sua importância não é devida a uma possível aplicação direta; se houver algum método eficiente para este caso, isso significaria a existência de uma ferramenta comum para a solução de uma variedade de formulações particulares do problema. Entretanto, não houve progresso significativo na solução desse problema.

Vale notar também que o PGR não é precisamente um Problema de Roteamento de Arcos, embora estes possam ser formulados como tal.

4.5 HEURÍSTICAS E METAHEURÍSTICAS

Metaheurísticas são procedimentos destinados a encontrar uma boa solução, eventualmente a ótima, consistindo na aplicação em cada passo, de uma heurística subordinada, a qual tem que ser modelada para cada problema específico (CHAVES, 2003). De acordo com o autor, a principal característica das

metaheurísticas é a capacidade que estas possuem de escapar de ótimos locais, isto é, melhor solução para a função objetivo em uma vizinhança do ponto, oferecendo certa flexibilidade às restrições da função objetivo.

"Uma metaheurística é um conjunto de conceitos que podem ser usados para definir os métodos heurísticos que podem ser aplicados a um vasto conjunto de problemas diferentes. Em outras palavras, uma metaheurística pode ser vista como um quadro geral de algoritmos que podem ser aplicados a diferentes problemas de otimização com relativamente poucas modificações para torná-los adaptados a um problema específico" (METAHEURISTICS NETWORK, 2011).

Heurísticas baseiam-se na melhoria do movimento a ser executado, buscando um ótimo local. Porém não são limitadas e podem oferecer a mesma solução quando iniciadas em um mesmo ponto inicial. As metaheurísticas suprem esse tipo de deficiência, baseando-se na melhoria da solução, deixando temporariamente um ótimo local para procurar um ótimo global, isto é, uma solução que fornece o melhor valor para a função objetivo considerando toda a região factível, mesmo que essa ação motive a perda temporária do valor da função objetivo.

No capítulo 5 será detalhado o algoritmo desenvolvido para a resolução do Problema do Carteiro Chinês Não-Orientado via heurística. Buscou-se através da proposta otimizar com uma busca local a paridade dos nós que eram de grau ímpar. Fez-se uma busca gulosa entre as combinações possíveis entre esses nós para que fosse encontrada a melhor combinação que resultasse um menor custo.

4.6 PROGRAMAÇÃO LINEAR

A Programação Linear (PL) é uma das mais importantes e mais usadas técnicas de Pesquisa Operacional. A simplicidade dos modelos matemáticos envolvidos e a disponibilidade de técnicas de soluções programáveis através de modelos exatos, heurísticos e metaheurísticos facilitam sua aplicação. A técnica do Método Simplex é amplamente utilizada, pois possui habilidade para modelar os complexos problemas de decisão, e capacidade de produzir soluções rapidamente, dependendo das características do problema. A descrição do Método Simplex pode ser encontrada em Zionts (1974).

Os problemas de PL são compostos por:

- a) Uma função linear formada pelas variáveis de decisão, chamada de função objetivo, cujo valor deve ser otimizado;

b) Relações de interdependência entre as variáveis de decisão que se expressam por um conjunto de equações ou inequações lineares, chamadas de restrições do modelo;

c) Variáveis de decisão que devem ser positivas ou nulas;

As equações (21), (22) e (23) mostram a formulação para um problema de PL:

$$\text{Maximizar (minimizar): } z = \sum_{j \in N} c_j x_j, \quad N = \{1, \dots, n\} \quad (21)$$

$$\text{Sujeito a: } \sum_{j \in N} a_{ij} x_j (\leq, = \text{ ou } \geq) b_i, \quad i \in M = \{1, 2, \dots, m\} \quad (22)$$

$$x_j \geq 0, \quad j \in N \quad (23)$$

No qual c_j , a_{ij} e b_i em (21), (22) e (23) são constantes conhecidas para todo i e j , e x_j (23) são variáveis não negativas.

Afirma Barboza (2005) que as restrições do problema podem ser transformadas em equações adicionando-se uma variável de folga (não negativa) x_{n+i} , se a i -ésima desigualdade é do tipo \leq e subtraindo uma variável de folga (não negativa), x_{n+k} se a k -ésima desigualdade é do tipo \geq . Considerando que ao serem acrescentadas as variáveis de folga, obtém-se um total de $m + n$ variáveis, pode-se escrever o problema na forma matricial como mostram as equações (24), (25) e (26):

$$\text{Minimize (ou maximize): } z = cx \quad (24)$$

$$\text{Sujeito à: } Ax = b \quad (25)$$

$$x \geq 0 \quad (26)$$

No qual c em (24) é um vetor linha de ordem $(m + n)$, A em (25) é uma matriz $m \times (m + n)$, x em (26) é um vetor coluna de ordem $(m + n)$ e b em (25) é um vetor coluna de ordem m .

4.7 PROGRAMAÇÃO INTEIRA (PI)

Alguns problemas reais requerem o uso de variáveis que assumem somente valores inteiros. Quando isso acontece, tem-se um problema de Programação Inteira

(PI), que tem a forma da modelagem que envolve as equações de (27) à (30) a seguir:

$$\text{Minimize (ou maximize): } z = g_0(x_1, x_2, \dots, x_n) \quad (27)$$

$$\text{Sujeito a: } g_i(x_1, x_2, \dots, x_n) (\leq \text{ ou } = \text{ ou } \geq) b_i, i \in M = \{1, 2, \dots, m\} \quad (28)$$

$$x_j \geq 0, j \in N = \{1, 2, \dots, n\} \quad (29)$$

$$x_j \text{ inteira, } j \in I \subseteq N \quad (30)$$

No qual $x_j, j \in N$ são as variáveis, $g_i, i \in M \cup \{0\}$ são funções das variáveis x_1, x_2, \dots, x_n e $b_i, i \in M$ são constantes conhecidas. Se $I = N$, isto é, todas as variáveis são inteiras, então o problema é dito de PI. Caso contrário, se $I \subset N$, então chama-se de problema de Programação Inteira Mista (PIM).

Em muitos dos problemas abordados em PI, as funções $g_i, i \in \{0\} \cup M$ são lineares e o modelo pode então ser descrito como mostra as equações de (31) à (34) a seguir:

$$\text{Minimize (ou maximize): } z = \sum_{j \in N} c_j x_j, N = \{1, \dots, n\} \quad (31)$$

$$\text{Sujeito à: } \sum_{j \in N} a_{ij} x_j (\leq, \geq \text{ ou } =) b_i, i \in M = \{1, \dots, m\} \quad (32)$$

$$x_j \geq 0, j \in N \quad (33)$$

$$x_j \text{ inteira, } j \in I \subseteq N \quad (34)$$

Onde c_j, a_{ij} e b_i são constantes conhecidas para todo i e j , e x_j são as variáveis não negativas. Se $I = N$, isto é, todas as variáveis são inteiras, então temos um problema de Programação Linear Inteira (PLI). Se $I \subset N$, então o problema é de Programação Linear Inteira Mista (PLIM) (BARBOZA, 2005).

Muitos modelos práticos de PLI restringem algumas das variáveis inteiras para valores "0" ou "1" e, neste caso, tem-se um problema de Programação Linear Inteira Binária (PLIB). Estas variáveis são usadas para as decisões sim ("1") e não ("0").

4.7.1 GAMS

A linguagem de modelagem GAMS (*General Algebraic Modeling System*) foi utilizada para modificar o modelo em Programação Inteira para as modelagens dos problemas do Carteiro Chinês Misto e Não-Orientado.

A linguagem GAMS é utilizada na Pesquisa Operacional (PO), para o desenvolvimento e otimização de modelos matemáticos lineares e combinatórios, até mesmo em problemas considerados *NP-Hard* (SILVA, 2009).

A codificação de modelos ocorre por meio do desenvolvimento das linguagens de modelagem (LMs). Dentre as LMs que se destacaram a década de 80 e início da década de 90, podem ser citados: CML (Conversational Modeling Language), a LPM (System for Constructing Linear Programming System), a LAMP (Language for Interactive General Optimization), LINGO (Language for Interactive General Optimization) e o GAMS (General Algebraic Modeling System). Essas LMs vêm adquirindo maiores significâncias para os modeladores, tendo em vista que os problemas de estudo de caso analisados estão se tornando cada vez mais complexos, e as LMs permitem que os modeladores dediquem cada vez mais tempo para solucionar problemas referentes ao modelo, e não a implementação computacional (SILVA, 2009).

O GAMS é uma LMs que foi projetada para o desenvolvimento e solução de modelos de programação matemática complexa e eliminar os erros, pois os programas que realizaram as operações de dados eram acessíveis apenas ao especialista que o desenvolveu e não aos analistas responsáveis pelo projeto (ROSENTHAL, 2011). As principais vantagens na utilização do GAMS são:

- a) Fornecer uma linguagem de alto nível para uma representação compacta de modelos extensos e complexos;
- b) Permitir mudanças na especificação dos modelos de forma simples e segura;
- c) Permitir relações algébricas enunciadas de forma não ambígua;
- d) Permitir descrições de modelos independentes dos algoritmos de solução;

e) Simplificar a preparação de dados de entrada e relatórios de saída e transformar automaticamente os dados para a forma requerida pelos pacotes de programação matemática.

O GAMS é um compilador de idéias tiradas da teoria de banco de dados (quando se faz manipulação dos dados) mesclando essas idéias para atender às necessidades dos modeladores da teoria de programação matemática (para descrição e solução dos problemas). GAMS também disponibiliza um conjunto de *Solvers*, que são pacotes com opções de várias técnicas de solução de problemas de programação matemática, como por exemplo, Programação Linear (PL), Programação Inteira Mista (PIM), Programação Não Linear (PNL), entre outras, que podem ser utilizados de acordo com a escolha do modelador.

4.7.2 *Sítio* NEOS

O *sítio* NEOS disponibiliza um conjunto de solucionadores de problemas de otimização. O usuário pode fazer sua opção de escolha do solver e da linguagem usada para definir o problema de otimização (NEOS, 2011).

A utilização do servidor é através do *sítio* do NEOS. A ferramenta de submissão ao NEOS é de alta velocidade. O usuário verifica os formatos disponíveis para o tipo de problema que tem interesse em resolver, prepara o arquivo usando a sintaxe de sua preferência e apresenta o trabalho ao servidor. Após o processamento, o resultado é enviado ao usuário por *e-mail*.

Para a resolução de problemas de Programação Inteira Mista através do aplicativo NEOS, o usuário pode escolher o *solver* que julgar conveniente para o seu tipo de problema. Entre os *solvers* disponíveis tem-se: Cbc, feaspump, GLPK, Gurobi, Minto, MOSEK, qsopt_ex, scip, SYMPHONY e XpressMP. Para a solução dos Problemas do Carteiro Chinês Misto e Não-Orientado, neste trabalho foram utilizados os programas que permitiam a entrada através da linguagem de programação GAMS. Esses programas são: Cbc, GLPK, Gurobi, MOSEK, e XpressMP.

O Cbc é um *open source* (fonte aberta) para que o usuário possa ler, redistribuir e modificar o código-fonte para que o software evolua. Para melhorá-lo, os usuários o adaptam e corrigem os *bugs* (erros).

Os GLPK é um pacote destinado para a resolução de problemas de grande escala em programação linear, programação inteira mista e outros problemas relacionados à pesquisa operacional. É um conjunto de rotinas escritas em ANSI C e organizado na forma de uma biblioteca acessível. O *solver* GLPK suporta a *linguagem de modelagem MathProg GNU*, que é um subconjunto da linguagem AMPL. O pacote GLPK inclui os seguintes componentes principais: primal e dual métodos simplex, método primal-dual de pontos interiores, método *branch-and-cut*, tradutor para *GNU MathProg*, interface de programação de aplicativos (API) e stand-alone LP/MIP solver.

O Gurobi *Optimization* é um software de alto desempenho e robusto, que utiliza tecnologias recentes de programação linear, quadrática e inteira-mista. O progresso computacional em programações linear, quadrática e inteira-mista nos últimos 20 anos tem sido notável, permitindo aplicações científicas que seriam inacessíveis a poucos anos atrás. O fundador desse desenvolvimento foi Gurobi. A Gurobi *Optimization* foi o primeiro a oferecer licenças gratuitas para a comunidade acadêmica.

O *solver* MOSEK possui uma implementação extremamente eficiente de um algoritmo de ponto interior para a solução de problemas de programação linear, programação inteira e programas convexos não-lineares. Se destaca por empregar uma ótima tecnologia de pontos interiores, ter uma boa base para a identificação de problemas lineares e por disponibilizar otimização pelo método simplex primal e dual para problemas lineares. Foi desenvolvido por MOSEK ApS C/O Symbion Science Park em Copenhagen, Dinamarca.

Finalmente, o *solver* XpressMP desenvolveu e implantou aplicativos de otimização para resolver problemas do mundo real mais rápido do que nunca, levando a melhores decisões de negócios. O Xpress *Optimization Suite* permite que profissionais da pesquisa operacional, analistas e consultores encontrem as melhores estratégias para alcançar um determinado resultado. Matematicamente identifica os pontos de operações mais vantajosas sob qualquer desafios enfrentados em uma organização de dados. É um programa que permite ao usuário lidar com problemas de grande escala e complexos, com milhões de variáveis e restrições. O Xpress é conhecido por oferecer um desempenho excepcional em relação aos desafios dos clientes mais exigentes em todo o mundo também por

resolver problemas recentes em jogos de *benchmarking*. O seu desempenho no mercado bateu a concorrência medindo o custo implícito das políticas atuais e outras limitações, ajudando a alterar processos ou remover essas limitações, levando a uma maior otimização em um melhor tempo. Oferece um suporte de 64-bit para otimização e modelagem, que permite aos usuários resolver problemas de larga escala de modelos com mais de 2 bilhões de coeficientes, reduzindo o tempo de solução de forma significativa.

5. METODOLOGIA

5.1 PROBLEMA DE COLETA DE RESÍDUOS SÓLIDOS URBANOS

A otimização da rota realizada pelo caminhão de coleta e transporte de resíduos sólidos, pelos funcionários dos serviços de limpeza das ruas e pelos carteiros na entrega de correspondências tem uma significativa importância financeira para os cofres públicos. No problema em estudo, cada caminhão faz sua rota com uma equipe de três funcionários, sendo um motorista e mais dois, que coletam o lixo das lixeiras e os depositam no caminhão. Nessas circunstâncias, além da despesa com o combustível e desgaste do caminhão, devem ser somados os salários das pessoas envolvidas nesse contexto. Já o serviço de limpeza para varrição das ruas e entrega de correspondências, que não utiliza veículos, envolve um investimento de pessoal.

Contudo, a minimização do percurso, hora em estudo, vem trazer uma economia financeira significativa, não somente para a prefeitura municipal da cidade de Irati/PR, bem como para empresas que necessitem minimizar rotas percorridas pelos funcionários, em relação ao tempo de duração da coleta do lixo, entrega de correspondências e limpeza de ruas. Para a minimização são contabilizados: combustível, desgaste do veículo e mão de obra.

O levantamento das informações sobre a rota de coleta dos resíduos sólidos na cidade de Irati/PR realizou-se através de dados fornecidos pelos funcionários do pátio de obras da prefeitura municipal da referida cidade, conforme anexo II. Não foram obtidos dados reais sobre o tempo e distâncias de percursos dos varredores de ruas, nem de funcionários que trabalham com entrega de correspondências. Os resultados obtidos pela heurística serão comparados com resultados obtidos pelo modelo de Programação Inteira.

Para que os dados geográficos pudessem ser obtidos foram feitas marcações no aplicativo *Google Earth*. Estes dados foram então transformados em coordenadas cartesianas para cumprir com os objetivos propostos para a pesquisa. Os dados puderam ser utilizados tanto na resolução do problema do carteiro chinês misto, bem como no problema do carteiro chinês não-orientado.

Após o levantamento dos dados, no caso da coleta e transporte dos resíduos sólidos, calculou-se o custo gerado na coleta e transporte dos resíduos. Para a

resolução do problema, utilizou-se Programação Linear Inteira, nos casos misto e não-orientado e uma heurística para o problema não-orientado. Para a aplicação da coleta do lixo, deve-se obrigatoriamente usar o caso do problema do carteiro chinês misto. Devido à existência de ruas com mão única, o resultado do problema do carteiro chinês misto gera custos maiores que para o caso não-orientado, que considera todas as ruas como sendo de mão dupla. Os resultados obtidos foram comparados com os roteiros realizados no mês de Fevereiro de 2011, que somavam uma distância total de percurso igual a 32725,54 metros, para a coleta de resíduos sólidos.

5.2 OBTENÇÃO DOS DADOS PARA O ESTUDO DE CASO

O *Google Earth* é um aplicativo desenvolvido e distribuído pela empresa que leva a logomarca *Google*, cuja função é disponibilizar um modelo tridimensional do globo terrestre. Este modelo é construído a partir da captura de imagens via satélite, obtidas de diversos ângulos de visão, imagens aéreas (através de aeronaves) e sistemas de informações geográficas *3D*. Esse aplicativo foi utilizado para a obtenção das coordenadas geográficas de latitude e longitude do mapa da região central da cidade de Irati/PR.

De acordo com os dados fornecidos pelos funcionários da prefeitura municipal sobre as delimitações entre os bairros da referida cidade, foi realizado através do aplicativo *online Google Earth* um mapeamento de todas as esquinas (nós) que compunham a região central da cidade (grafo). Também foram observadas as características, em relação ao sentido das ruas, isto é, se a rua é de mão única (arco: direcionado) ou dupla (aresta: não direcionada).

5.3 CADASTRO DOS PONTOS GEOGRÁFICOS NO *GOOGLE EARTH*

A figura 9 ilustra as coordenadas geográficas de todas as esquinas (nós do grafo) obtidas pelo *Google Earth*. A utilização dessas coordenadas fornece um maior grau de precisão e assim, resultados mais significativos à pesquisa de estudo de caso.

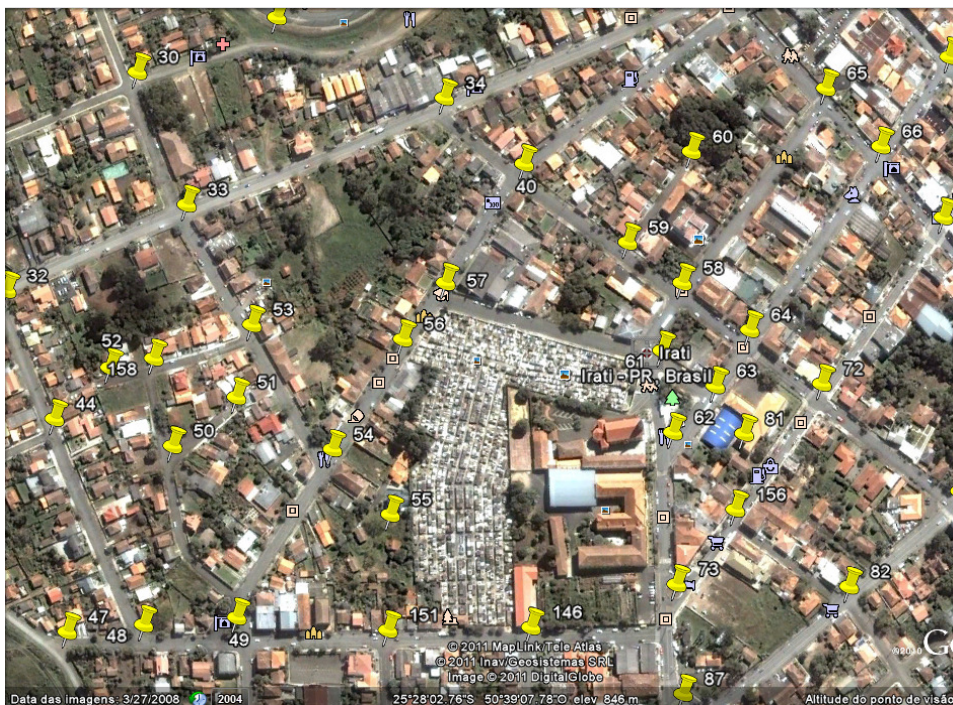


FIGURA 9: MAPEAMENTO DAS COORDENADAS GEOGRÁFICAS
 FONTE: GOOGLE EARTH 6.0.0 BETA (2011)

Após o cadastramento de todos os pontos geográficos, realizou-se a conversão desses pontos para o plano cartesiano. As coordenadas geográficas obtidas no *Google Earth* são dadas em latitude e longitude, referência para localização global, que foram transformadas em coordenadas cartesianas, que traduzem as distâncias métricas a partir de determinada referência.

Para que pudesse ser feita essa conversão foi utilizada uma planilha do aplicativo *Microsoft Excel*, na qual inseriu-se os dados de latitude e longitude (transformadas em latitude decimal e longitude decimal), para que pudessem ser transformadas em coordenadas cartesianas X e Y. As equações (35) e (36) foram utilizadas para fazer esta conversão.

$$X = (\text{Raio da Terra}) * \text{sen}\left(\frac{(90 + \text{latitude decimal}) * \pi}{180}\right) * \cos\left(\frac{(360 - \text{longitude decimal}) * \pi}{180}\right) \quad (35)$$

$$Y = (\text{Raio da Terra}) * \text{sen}\left(\frac{(90 + \text{latitude decimal}) * \pi}{180}\right) * \text{sen}\left(\frac{(360 - \text{longitude decimal}) * \pi}{180}\right) \quad (36)$$

Após todas as coordenadas estarem devidamente codificadas, foram determinadas as distâncias entre todos os pontos, que é a distância real entre as quadras da região central da cidade de Irati/PR.

5.4 OBTENÇÃO DAS DISTÂNCIAS ENTRE OS PONTOS GEOGRÁFICOS

Utilizou-se o cálculo das distâncias euclidianas entre os nós (quadras), usando-se a fórmula (37):

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (37)$$

onde (x_1, y_1) e (x_2, y_2) são as coordenadas dos pontos.

Para a análise e validação dos dados dessa pesquisa, as distâncias serão consideradas geograficamente sobre uma superfície plana, isto é, desconsiderando a altitude da região estudada.

5.4.1 Veículos

Para o caso da coleta e transporte dos resíduos sólidos são utilizados os veículos (caminhões compactadores) disponíveis. A frota é constituída de 8 veículos que atendem toda a área urbana do município de Irati/PR, em dias e horários alternados, pré agendados pelos funcionários.

5.4.2 Rota Atual do caminhão de coleta de resíduos

De acordo com o relatório do anexo II, fornecido pelos funcionários da prefeitura municipal, foi construída a rota atual percorrida pelo caminhão de coleta de resíduos, e calculado manualmente o custo desse percurso. O grafo que constitui essa rota é composto por 158 (cento e cinqüenta e oito) nós e 451 quadras que são percorridos pelo caminhão de coleta de resíduos, distribuídos em uma área de aproximadamente 697500 m², somando uma distância de 32725,54 m.

Um dos objetivos dessa pesquisa é tanto minimizar as distâncias percorridas pelo caminhão de coleta e transporte de resíduos quanto as distâncias percorridas pelos funcionários responsáveis pela limpeza de ruas e entregadores de correspondências, que devem passar por todas as ruas (arcos) da região central da cidade de Irati/PR.

5.5 MODELAGEM EM PROGRAMAÇÃO LINEAR INTEIRA

Para a resolução do problema do Carteiro Chinês Misto, que é o estudo de otimização da rota do caminhão de coleta de resíduos sólidos, foi desenvolvido um modelo em PLI, em linguagem GAMS. O fluxograma da metodologia para a construção do modelo em PLI para o PCCM está descrito na seção 5.9. Parte da matriz de adjacências e_{ij} e os dados utilizados para a construção da matriz de custos d_{ij} , estão disponíveis nos apêndices 1 e 2. O modelo desenvolvido está descrito a seguir:

Índices:

i, j e k representam os nós;

Conjunto:

$CNós$ o conjunto formado pelos nós do grafo misto;

Parâmetros:

$e_{ij} \begin{cases} 1, & \text{se o nó } i \text{ tem ligação com o nó } j, \text{ e} \\ 0, & \text{caso contrário.} \end{cases}$

d_{ij} distâncias do nó i ao nó j ;

M número inteiro maior que 1000;

Variáveis:

a_{ij} número de vezes que o arco do nó i para o nó j aparece na solução;

Função objetivo:

Para este modelo assume-se que o resultado ótimo é aquele que minimiza a distância total percorrida pelo caminhão de coleta de resíduos sólidos. É determinado pela soma do produto das distâncias pela variável a_{ij} :

A distância total é calculada de acordo com a equação (38):

$$\sum_{i \in CNós} \sum_{j \in CNós} d_{ij} a_{ij}$$

(38)

Restrições:

a) O número de arcos que chegam em um determinado nó deve ser igual ao número de arcos que saem deste nó. Essa restrição (39) garante a continuidade da rota:

$$\sum_{i,k \in \text{CNós}} e_{ik} a_{ik} - \sum_{j,k \in \text{CNós}} e_{kj} a_{kj} = 0 \quad (39)$$

b) A restrição (40) garante que a rota final contenha todos os arcos, pelo menos uma vez:

$$e_{ij} a_{ij} + e_{ji} a_{ji} \geq 1, \forall i, j \in \text{CNós}. \quad (40)$$

c) As restrições (41) e (42) garantem que se o parâmetro e_{ij} é zero, então a variável a_{ij} será também zero. Isto é, a variável a_{ij} terá valor diferente de zero apenas se existir adjacências entre os nós i e j :

$$a_{ij} \geq e_{ij} - 1, \forall i, j \in \text{CNós} \quad (41)$$

$$a_{ij} \leq e_{ij} \cdot M, \forall i, j \in \text{CNós} \quad (42)$$

$$a_{ij} \text{ inteira e positiva, } \forall i, j \in \text{CNós}. \quad (43)$$

Essa modelagem foi transcrita para a linguagem GAMS e pode ser visualizada no apêndice 4.

5.6 HEURÍSTICA PARA O PCCNO

Na sequência será apresentada a heurística desenvolvida para o Problema do Carteiro Chinês Não-Orientado. Essa heurística possui a característica de fazer emparelhamento de nós de um grafo não direcionado. Este método é constituído de duas partes: Num primeiro momento é construído um emparelhamento inicial e em seguida, são efetuados melhoramentos na solução inicial, usando a heurística 2-Opt (Busca Local). O método é aplicado da seguinte maneira:

Passo 1: Considere o conjunto de cardinalidade par que contém todos os nós de grau ímpar do grafo associado ao problema do carteiro chinês não orientado.

Passo 2: Utilizando o algoritmo de Floyd, calcula-se a matriz das distâncias mínimas D , entre todos estes nós ímpares, utilizando-se para isto o grafo completo (com os nós de graus pares e ímpares do grafo associado ao problema). Encontra-se também a matriz T para a recuperação das trajetórias, isto é, a matriz que permite encontrar o menor caminho entre cada par de nós de grau ímpar.

Passo 3: Inicia-se a construção de um emparelhamento inicial, que é construído da seguinte forma: Procura-se o menor valor na matriz D , obtendo-se assim, o par de nós ímpares de menor distância entre si (suponha que este valor esteja na posição d_{ij}). Inclui-se este par de nós ímpares em uma lista de pares de nós de graus ímpares e atribui-se valor infinito para as posições d_{ij} e d_{ji} na matriz simétrica D . Repete-se o processo até que todos os nós de grau ímpar estejam emparelhados. Este emparelhamento inicial não é necessariamente o de menor custo, pois foi construído através de um método do tipo guloso.

Passo 4: Para a melhoria da solução inicial, considere que o par ordenado (N_{i1}, N_{i2}) é o i -ésimo par de nós ímpares da solução inicial. Os segundos índices (neste caso 1 para o primeiro nó do par e 2 para o segundo), indicam o primeiro e o segundo nó do i -ésimo par de nós emparelhados.

Passo 5: Agora para cada par (N_{j1}, N_{j2}) , $j \neq i$ da lista citada, calcule o custo de emparelhamento dos seguintes pares de nós obtidos com as trocas de nós efetuadas em (N_{i1}, N_{i2}) e (N_{j1}, N_{j2}) , isto é calcule:

$$C_1^j = \text{Custo dos emparelhamentos } (N_{i1}, N_{j2}) \text{ e } (N_{j1}, N_{i2}) \text{ e,}$$

$$C_2^j = \text{Custo dos emparelhamentos } (N_{i1}, N_{j1}) \text{ e } (N_{i2}, N_{j2}).$$

Passo 6: Se o custo mínimo entre todos os novos emparelhamentos obtidos $\text{Min}\{C_1^j, C_2^j\}, \forall j \neq i$ for menor que o custo do emparelhamento inicial de (N_{i1}, N_{i2}) e (N_{j1}, N_{j2}) , efetue a troca entre os pares de nós que fornece o maior economia.

Passo 7: Repita o procedimento substituindo i por k ($\forall k \neq i$), isto é, partindo do par (N_{k1}, N_{k2}) e (N_{j1}, N_{j2}) , $j \neq k$, até que nenhuma melhoria possa ser realizada.

A expectativa é de que o algoritmo termine em um mínimo local ou eventualmente global do espaço de busca.

5.7 ALGORITMOS DE CONSTRUÇÃO DE ROTA

Para a construção de rotas a partir do resultado obtido pela resolução do problema de Programação Inteira ou pela heurística, dois algoritmos foram utilizados.

5.7.1 Algoritmo *Pathwork*

O algoritmo *Pathwork* foi desenvolvido para encontrar a rota, a partir do resultado dado na forma de número de vezes que cada arco é percorrido. Esse resultado especifica os arcos que são utilizados e a quantidade de vezes. A rota não é visível nesta forma de resultado. A descrição do algoritmo é mostrada a seguir:

Passo1:

Escolha um nó inicial *nó_inicial*;

$i = 1$

Passo 2:

$Nó_atual = nó_inicial$;

$Retalho_i = nó_inicial$

Faça até que o $nó_atual$ seja igual ao $nó_inicial$;

Insira na rota o primeiro nó com ligação ao $nó_atual$, e denomine-o $denó_inserido$;

$Retalho\ i = retalho\ i\ e\ nó_inserido$;

$Nó_atual = Nó_inserido$

Fim

Passo3

Se existem arcos ainda não percorridos, então

Faça $nó_inicial$ igual ao primeiro nó do arco ainda não utilizado

$i = i+1$

Volte ao passo2;

Fim

Faça o status de todos os retalhos igual a "não_usado";

Passo4

Para todo retalho i e j com status "não_usado", faça:

Para todo retalho j , diferente do retalho i , também com status “não_usado”, faça:

Se o retalho i possui em sua rota o nó_inicial do retalho j , então insira o retalho j na posição onde se encontra este nó no retalho i ;

Faça o status do retalho j igual a “usado”;

Fim

Se o número de retalhos com status “não_usado” for maior que um, então esses retalhos restantes possuem três nós.

Troque a ordem dos nós de cada um desses retalhos, da seguinte forma: se a ordem é “ $a - b - a$ ”, troque pela forma “ $b - a - b$ ” e volte ao passo 4;

A rota_final é igual ao retalho com status não usado.

5.7.2 Algoritmo de Hierholzer

Seja V o conjunto de vértices $V = \{v_1, v_2, \dots, v_n\}$ do grafo G e seja $E = \{a_1, a_2, \dots, a_m\}$ o número de arestas. Suponha-se que um caminho de um vértice v_1 até v_k é representado por uma lista $[v_1, a_1, \dots, a_{k-1}, v_k]$, que alterna vértices e arestas. A seguir uma descrição do algoritmo de Hierholzer, supondo-se que o grafo é euleriano:

Passo 1: Ler o grafo $G = (V, E)$;

Passo 2: Identificar os nós ímpares em um grafo $G' = (V', E')$, que inicialmente contém somente o nó inicial em uma matriz C . Enquanto $E' \neq \emptyset$, insira os vértices $d(v_i) > 0$ em G' . Assim, C' possui um circuito em G' que contém v_i e G' é a matriz $G' - \{as\ arestas\ contidas\ em\ C'\}$;

Passo 3: Em C substituir o vértice v_i pelo circuito C' .

5.8 FLUXOGRAMA DA PROPOSTA METODOLÓGICA DESENVOLVIDA PARA O PCCNO

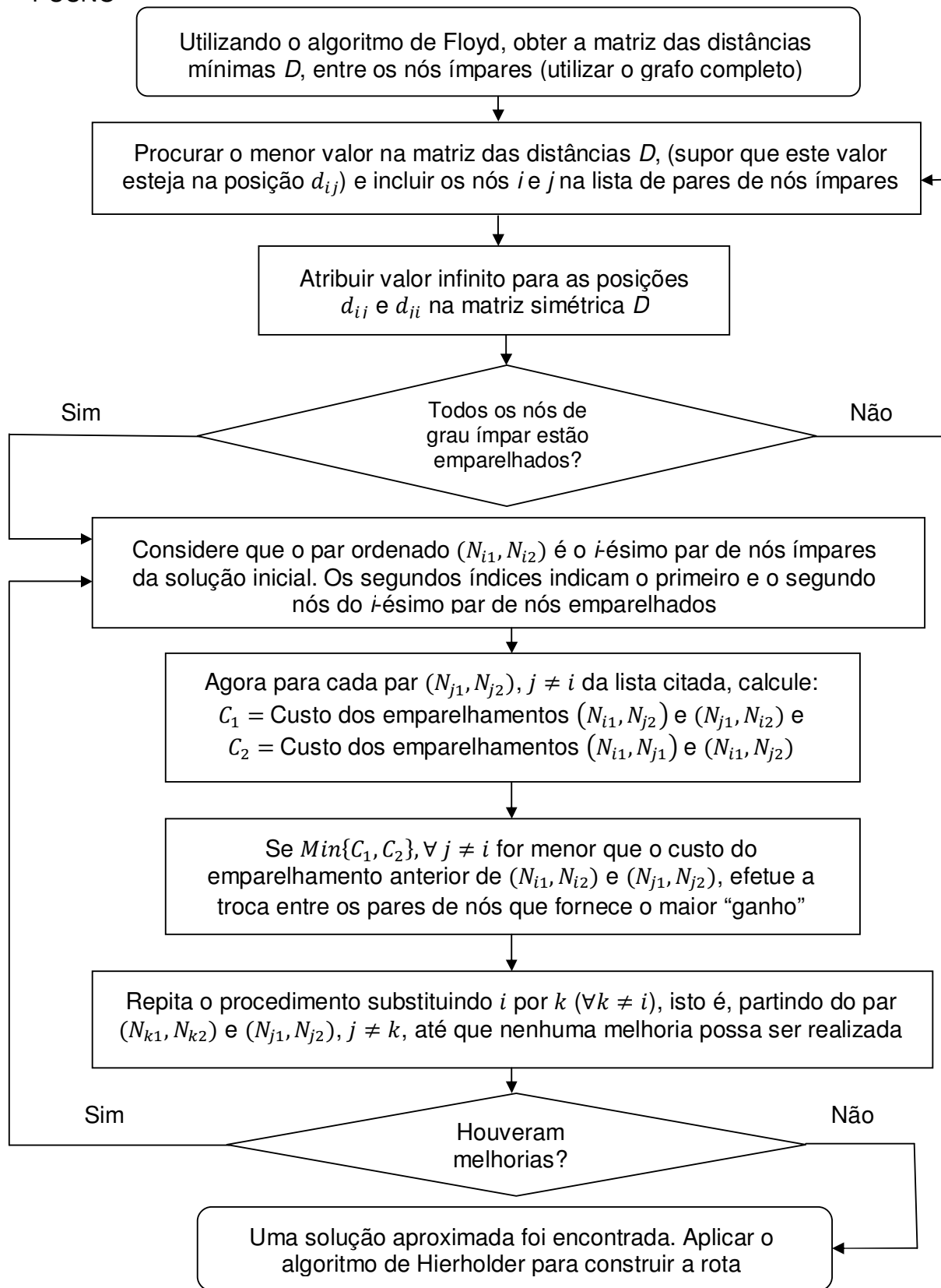


FIGURA 10: FLUXOGRAMA PCCNO – FONTE: O autor (2012)

5.9 FLUXOGRAMA DA PROPOSTA METODOLÓGICA DESENVOLVIDA PARA RESOLUÇÃO DO MODELO EM PLI

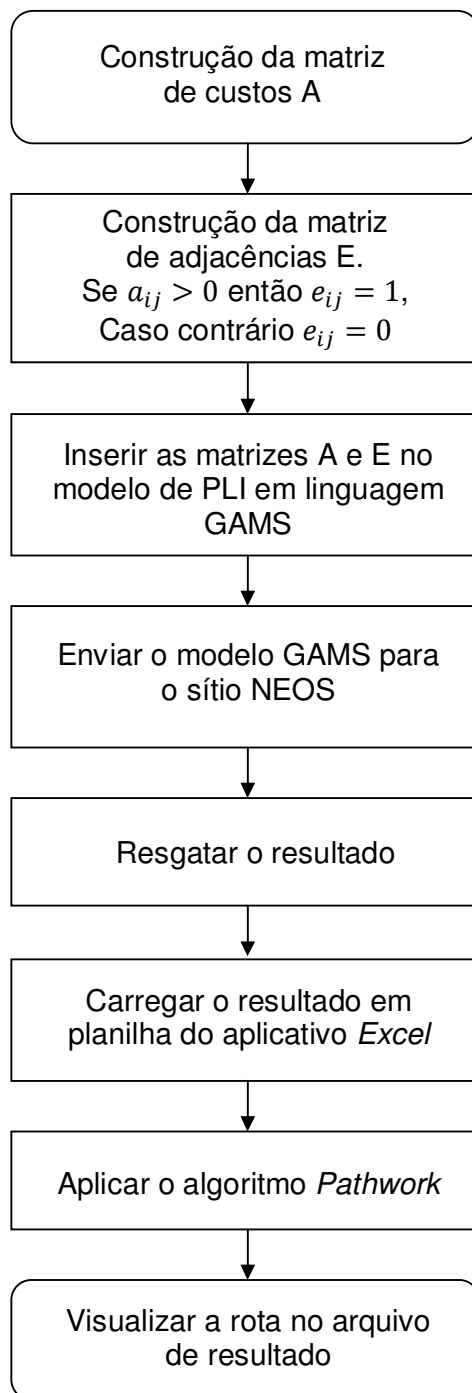


FIGURA 11: FLUXOGRAMA PCCM
FONTE: O autor (2012)

6. IMPLEMENTAÇÃO COMPUTACIONAL E ANÁLISE DOS RESULTADOS

6.1 TECNOLOGIAS UTILIZADAS

Os algoritmos propostos foram implementados em um microcomputador Intel(R) Core (TM)2 Duo CPU T5550 @1.83GHz (2 CPUs), com 1918MB RAM.

Os dados necessários para a implementação dos problemas de programação inteira mista foram obtidos do *Google Earth 6.0.0 Beta*. O grafo analisado na pesquisa para o caso do Problema do Carteiro Chinês Misto (onde há duplicação das arestas) foi composto por 158 nós e 451 arcos/arestas.

O algoritmo *Pathwork* foi implementado no VBA (*Visual Basic for Application*), do aplicativo computacional *Microsoft Excel Office 2007* (APÊNDICE 8).

O algoritmo via heurística proposto para a resolução do percurso do Carteiro Chinês não-orientado foi implementado no *software Matlab 5.3* (APÊNDICE 7).

6.2 DADOS UTILIZADOS

Os dados utilizados para a otimização das rotas para o problema do carteiro chinês compreende a região central da cidade de Irati/PR. As ruas foram esquematizadas em um grafo composto por 158 nós (esquinas) e 249 arestas (grafo sem a duplicação das arestas). O custo para cada aresta foi calculado a partir da equação (37) das distâncias euclidianas.

6.3 PROBLEMA DO CARTEIRO CHINÊS NÃO ORIENTADO

O algoritmo do Carteiro Chinês para redes não direcionadas foi desenvolvido com o objetivo de resolver o problema de cobertura de arcos (ruas de mão dupla), para o percurso de rotas que não dependam da utilização de veículos, como por exemplo, varredores de ruas, carteiros, medidores e entregadores de faturas de água e energia, entre outros.

6.3.1 Aplicação do Modelo de Programação Linear Inteira

Primeiramente, o problema do Carteiro Chinês Não-Orientado foi modelado em Programação Linear Inteira na linguagem GAMS. Esse modelo foi enviado ao sítio *Neos Server* para a resolução, usando os aplicativos de otimização: CBC,

GLPK, Gurobi, Mosek e XPress. Os resultados fornecidos após a aplicação desses otimizadores são limitados ou pela memória dos computadores ou pelo número máximo de iterações. Todos geraram uma solução inteira factível que não são necessariamente o custo ótimo para o problema. Um dos resultados gerados pode ser visualizado no apêndice 4. A partir do resultado fornecido pelo NEOS, encontrou-se a rota para cada aplicação usando-se o algoritmo *Pathwork*. Os resultados estão a seguir:

CBC

- Função Objetivo: 29356,2800m
- Tempo computacional: 0,010 segundos:
- Rota:

1-2-3-30-41-30-3-4-30-33-53-54-53-158-52-158-50-51-50-49-151-146-73-156-73-62-63-62-61-58 -65-66-67-68-69-76-77-78-79-144-79-80-81-80 -82-87-73-146 -147-148-149-150-152-154-155-153-152-153-149-148-90 -88-89-83-89-92-93-86-85-86-93-94-85-83-84-83-82-87-88-90-91-92-89-88-147-150-151-150-152-49-48-44 -32-41-29-31-42-43-157-43-44-43-42-45-46-45-47-154-47-48 -49-54-55-54-56-57-40-34-39-65-39-36-35-4-5-6-5-7-8-11-12-69-70-71-72-79-80-82-156-72-64-63-61-57-40-39-36-38-37-38-68-11-10-13-17-20-27-28-95-96-111-112-95-112-113-114-127-128 -135-136-143-113-76-28-17-28-95-96-97-109-115-124-127-128-129-134-138-137-141-142-143-77-75-70-67-36-35-8-7-9-10-9 -14-15-16-15-18-19-20 -25-26-27-28 -12-13-14-16-17-16-19-24-21-18 -21-22-23-99-23-24-25-26-98-100-101-106 -122-106-107-108 -116-117-118-119-120-121-122-123-120-118-117-107-100 -101-102-103-104-119-104-05-121-105-106-107-100-98-99-98-97-109-110-111-110-114-115-116-108 -109-115-124-25-126 -128 -126-129 -130-131-132-133 -139 -138 -139 -140 -144-145-132-131-133-134-135-136-137-141-140-141-142-78-74-75-74-71-66-64-58-59-60-59-40-34-33-32-31-29-3-2-5-1.

GLPK

- Função Objetivo: 28486,2400m
- Tempo computacional: 0,012 segundos
- Rota:

1-5-6-5-2-3-29-41-29-31-32-33-30 -33-53-54-49 -54-55-54-56-57-56-57-61-63-64-72-
 156 -73-156-82-87-88-89 -92-91-90-148-147-88-90 -148-149-150 -147-146-151-150-
 152-153-149 -153-155-154 -47-48-49-152-154-47-45-46 -45-42-31-42-43-157-43-44-
 48-44-32-41-30 -3-4-35-36-38-37-38-68 -11-12-69-68-11-8-7-8-35-36-39-4-40-39-65-
 58-61-62-73-87-88-89-83-85-86-93-92-93-94-85-83-84-83-82-80-81-80-79-72-64-63-
 62-73-146-151-49-50 -51-50-158-52-158-53-33-34-40 -57-40-59-60 -59-58-64-66-65-
 66-71-70 -67-68-67-36-67-66-71-72-71-74-75-74-78-79 -144-140 -141-142-78-77-76-
 28-12-13-17-20-19-18-21-22-23-24-25-26-98-97-96-111-112-111-110 -109-108 -107-
 100-98-97-109 -115-114-113-76-69-70 -75-77-143-113 -112-95-28-95 -96 -111 -110-
 114-127-124-125-126-128 -135-136- 137-141-142-143 -136 -137-138 -134-133 -131-
 132-133 -139 -138 -139 -140-144-145 -132-131-130 -129-128-127-128-135-134-129-
 126 -125-124-115-116-108 -116-117-118 -119 -120-123 -122-121-120 -118-117-107-
 106-122-106-101-106-105-121-105 -104-119-104-103-102-101-100-98-99-23-22-21-
 24-19-20-25-26-27-20-27-28-17-16-14-15-16-19-18-15-14-13-10-11-10-9-14-9-7-5-4-
 30-3-2-1- 126-125-124-115-116-108-116-117-118-119-120-123-122-121-120-118-
 117-107-106-122-106-101-106-105-121-105-104-119-104-103-102-101-100 -98 -99-
 23-22-21-24-19-20 -25-26-27-20-27-28-17-16-14-15-16-19-18-15-14-13-10-11-10-9-
 14-9-7-5-4-30-3-2-1

Gurobi

- Função Objetivo: 30362.9300m
- Tempo computacional: 0.006 segundos
- Rota:

1-2-3-30-33-32-31-29-3-4-30-41-29-41-32-33-34-33-53-54-55-54-49-48-44-32-44-43-157-43-42-31-42-45-46 -45-47-48 -47-154-152-154-155 -153-152 -49 -151-146 -151-150-149-153-149 -148-147-146-73-62-63-62-61-57-40 -34-39-36-67-70-75-70-69-68-67-66-64-63 -61-58-65-58-59-60 -59-40 -57-56-54-53-158 -52-158 -50-51-50-49-152-150-147-88-89-92-93-92-91-90-148-90 -88-87-73-156-72-71-66-65-39 -40 -59-58-64-72-79-144-79 -72-156-82-87-82-83-89-83-85-94-93-86-85-83-84-83-82-80-81-80-79-78-142-78-74-78-77-75-74-71-70 -67-68-38-37-38-36-35-4-5-7-9 -14-9-10 -13-12-28-27-20-19-16-14-13-17-20-25-24-19-18-21-24-23-24-25-26-27-26-98-97-96-95-28-95-96-111-110 -109 -97-98-99 -23-22-21-18 -15-14-15-16-17-28 -76-77-143 -136 -135-128 -135-134-133 -131-132-145 -144-140 -141-140 -139 -133-132-131-130-129-128-126-125-124-127-124-115-109 -108-107-100-98-100-101-100-107-106-105-104-105-121-122-123-120 -119-104-103 -102-101-106 -122-121-120 -118-119 -118 -117-107-108-116-117-116-115-114-110 -111-112-95-112-113-143-142-141-137-136-137-138-139-138-134-129-126-128-127-114-113-76-69 -12-11-68-11-10-11-8-35-8-7-5-6-5-2-1-5-1

Mosek

- Função Objetivo: 28915.6500m
- Tempo computacional: 0.012 segundos
- Rota:

1-5-2-3-30-41-29-3-4-5-6-5-7-8-11-68-38 -37-38-36-35-36-39-34-33-30-41-32-31-29-31-42-43-157-43-42-45-46-45-47-154-47-48-49-151-146-151-150-147-148-90-91-92-93-92-89 -83-85-94-93-86-85-83-84-83-82-87-88-90-148-149 -153-149-150-152-154-155-153-152-49-50-51-50-158-52-158-53-33-32-44-43-44-48-49 -54-53-54-55-54-56-57-40-34-40 -39-65-58-61-57-40-59-60-59-58-64-63-61-62-63-62-73-87-73-146-147-88-89-83-82-80-81-80-79-72-64-66-67-36-39 -65-66-71-74-78-142-143-142-141-137-138 -134-129-126-128-127-114-115 -124-127-128 -129 -130 -131-132-133-139 -138-139 -140-144-145-132-131-133-134-135 -136 -143-77-75 -74-71-72-79-78-77-75-70-67-68-69-70 -71-72-156 -73-156-82-80 -79 -144-140-141-137-136-135-128-126-125-

124-115-109-97-96-95-28-12-69 -76-113-143-77-76-28-17-13-12-11-10-9-14-15-18-
 19-24-21-22-23-24-25 -26-27-28-95-112-111-96-95-112-113-114-110-111-110-109-
 108 -116-117-107-100 -98-26-98 -97-109 -115-116 -117-118-119 -104-119-120 -121-
 105-121-122-106 -107-108-116-117-118 -120-123 -122-106 -105 -104-103 -102-101-
 106-101-100-98-99-23-22-21-18-15-16-17-20-25-26-27-20-19-16-14-13-10-9-7-8-35-
 4-30-3-2-1

Xpress

- Função Objetivo: 29749.5700m
- Tempo computacional: 0.007 segundos
- Rota:

1-5-2-3-30-41-29-3-4-5-6-5-7-8-11-68-38-37-38-36 -35-36-39-34-33-30-41-32-31-29-
 31-42-43-157-43-42-45-46-45-47-154-47-48-49-151-146-151-150-147-148-90-91-92-
 93-92-89-83-85-94-93 -86-85-83-84-83-82-87-88 -90-148-149-153-149-150-152-154-
 155-153-152-49-50-51-50-158-52-158-53-33 -32-44-43-44-48-49-54-53-54-55-54-56-
 57-40-34-40-39-65-58 -61-57-40-59-60-59-58-64-63-61-62-63-62-73-87-73-146-147-
 88-89-83-82-80-81-80-79-72-64-66-67-36-39 -65-66-71-74-78-142-143-142-141-137-
 138 -134-129-126-128 -127-114-115 -124-127-128 -129 -130 -131-132-133-139-138-
 139-140 -144-145 -132-131-133 -134-135 -136-143-77-75-74-71-72-79-78-77-75-70-
 67-68-69-70-71-72-156 -73-156-82-80 -79 -144-140 -141-137-136-135-128-126-125-
 124-115-109-97-96 -95-28-12-69 -76-113-143 -77-76-28-17-13-12-11-10-9-14-15-18-
 19-24-21-22-23-24-25-26-27-28 -95 -112-111-96 -95-112-113-114-110-111-110-109-
 108-116-117-107-100 -98 -26-98-97-109 -115-116 -117-118-119 -104-119 -120 -121-
 105 -121-122-106 -107-108 -116 -117-118-120 -123-122-106-105 -104-103-102-101-
 106-101-100-98-99-23-22-21-18-15-16-17-20-25-26-27-20-19-16-14-13-10-9-7-8-35-
 4-30-3-2-1

Os resultados obtidos para a modelagem em Programação Linear Inteira do problema do carteiro chinês não-orientado foram organizados na tabela 1, para melhor visualização e interpretação dos resultados.

TABELA 1: RESULTADOS PCCNO

Linguagem de Programação	Tempo de Execução	Resultado da Função Objetivo
Cbc	0,010 segundos	29356,2800m
GLPK	0,012 segundos	28486,2400m
Gurobi	0,006 segundos	30362,9300m
Mosek	0,012 segundos	28915,6500m
Xpress	0,007 segundos	29749,5700m

FONTE: O autor (2012)

Antes da aplicação da heurística, foi necessária a identificação dos nós de grau ímpar. A tabela 2 mostra o grau de cada um dos nós do grafo associado ao centro da cidade de Irati (PR), onde o número 1 (um) indica que o nó é de grau ímpar par e 0 (zero) que o nó é de grau par.

TABELA 2: IDENTIFICAÇÃO DO GRAU DOS NÓS

Nó	Nó	Nó	Nó	Nó	Nó	Nó							
Status	Status	Status	Status	Status	Status	Status							
1	0	24	0	47	1	70	0	93	1	116	1	139	1
2	1	25	1	48	1	71	0	94	0	117	1	140	1
3	0	26	1	49	1	72	0	95	1	118	1	141	1
4	0	27	1	50	1	73	0	96	1	119	1	142	1
5	1	28	1	51	1	74	1	97	1	120	0	143	0
6	1	29	1	52	1	75	1	98	0	121	1	144	1
7	1	30	0	53	1	76	0	99	0	122	1	145	0
8	1	31	1	54	0	77	0	100	1	123	0	146	1
9	1	32	0	55	1	78	0	101	1	124	1	147	0
10	1	33	0	56	0	79	0	102	0	125	0	148	1
11	0	34	1	57	1	80	1	103	0	126	1	149	1
12	0	35	1	58	0	81	1	104	1	127	1	150	0
13	0	36	0	59	1	82	0	105	1	128	0	151	1
14	0	37	1	60	1	83	0	106	0	129	0	152	0
15	1	38	1	61	0	84	1	107	0	130	0	153	1
16	0	39	0	62	1	85	1	108	1	131	1	154	1
17	0	40	0	63		86	0	109	0	132	1	155	0
18	1	41	1	64	0	87	1	110	1	133	0	156	1
19	0	42	1	65	1	88	0	111	1	134	0	157	1
20	0	43	1	66	0	89	1	112	1	135	1	158	1
21	1	44	1	67	0	90	1	113	0	136	1		
22	0	45	1	68	0	91	0	114	0	137	1		
23	1	46	1	69	0	92	1	115	0	138	1		

FONTE: O autor (2012)

Após a construção do conjunto com os nós com cardinalidade ímpar, foi feito o emparelhamento inicial, com a duplicação de arcos. A tabela 3 a seguir, mostra os resultados obtidos, sendo que na primeira e na segunda coluna tem-se os nós e na terceira coluna o comprimento do arco duplicado. Vale lembrar que os custos obtidos podem incluir um percurso com outros nós intermediários, isto é, a duplicação de um nó i para um nó j pode incluir arcos que liguem outros nós. Estes percursos não afetam a cardinalidade dos nós intermediários, pois esses nós recebem um arco que chega neste e outro que sai.

TABELA 3: EMPARELHAMENTO INICIAL DOS NÓS

Nó Inicial	Nó Final	Custo	Nó Inicial	Nó Final	Custo
157	43	22,65269576	122	101	77,48675114
111	110	24,10469174	49	48	80,46392537
144	140	29,22427688	75	74	81,40040533
138	137	30,01291603	85	84	89,20245245
112	95	32,62666584	6	5	89,43948126
26	25	32,66045082	135	127	97,27064296
121	105	34,982452	96	28	97,44271941
158	50	36,30377088	148	146	100,6656012
63	62	40,85881337	151	149	101,1772443
142	141	44,24364274	41	29	117,5198542
8	7	47,4397238	139	132	123,1653243
46	45	48,00334133	118	117	123,2196024
81	80	54,23895305	53	52	128,7172681
119	104	57,5046277	57	34	147,4853415
154	47	61,72710631	100	97	153,8595091
60	59	63,75990814	126	124	186,5252417
18	15	67,23655545	156	89	219,8931824
116	108	69,32514864	27	21	228,1277195
42	31	70,65583093	153	51	246,459351
38	37	71,16238614	65	35	253,6147603
93	92	73,69972623	136	131	335,1842281
90	87	73,96448689	55	44	341,3944499
10	9	74,40765828	23	2	1662,72102

FONTE: O autor (2012)

O comprimento total dos arcos duplicados (soma dos valores da terceira coluna) é de 6243,23 m.

Após a aplicação dos passos que fazem a melhoria do emparelhamento inicial, foram obtidos os resultados melhorados para a duplicação dos arcos. Esses resultados podem ser observados na tabela 4 onde, na primeira e na segunda

colunas tem-se os nós e na terceira coluna o comprimento ou custo do arco duplicado.

TABELA 4: EMPARELHAMENTO FINAL DOS NÓS

Inicial	Final	Custo	Inicial	Final	Custo
157	43	22,65269576	122	101	77,48675114
111	110	24,10469174	49	55	150,908536
144	140	29,22427688	75	74	81,40040533
138	139	84,032823	85	84	89,20245245
112	96	75,86660201	6	5	89,43948126
26	25	32,66045082	135	127	97,27064296
121	105	34,982452	95	28	50,28342734
158	50	36,30377088	148	146	100,6656012
63	62	40,85881337	151	149	101,1772443
142	141	44,24364274	41	29	117,5198542
8	35	55,22990003	131	132	124,541693
46	45	48,00334133	118	117	123,2196024
81	80	54,23895305	53	52	128,7172681
119	104	57,5046277	57	34	147,4853415
154	47	61,72710631	100	97	153,8595091
60	59	63,75990814	126	124	186,5252417
2	7	566,252464	156	87	102,0331153
116	108	69,32514864	18	27	104,6485941
42	31	70,65583093	153	51	246,459351
38	37	71,16238614	65	10	362,287226
93	92	73,69972623	136	137	72,06039228
90	89	146,5109509	48	44	110,0219885
15	9	196,069414	23	21	269,9326761

FONTE: O autor (2012)

O comprimento total dos arcos duplicados, isto é, a soma dos valores da terceira coluna é igual a 5046,22 m.

Para obtenção do Circuito Euleriano no grafo obtido com os arcos duplicados, aplicou-se o algoritmo de Hierholzer descrito na seção 4.3.1, obtendo-se a seguinte rota:

1-2-5-6-5-2-3-29-31-42-45-46-45-47-154-155-153-152-49-54-55-54-56-57-40-34-40-39-65-66-65-58-61-63-62-63-64-58-59-60- 59-40-57-61-62-73-87-73-156-72-64-66-67- 68-69-76-113-143-142-78-79-144-140-144-145-132-133-139-140-141-142-141-137-138 -139 -138 -134-133 -131-132-131-130-129 -134-135 -128-129-126-125-124-125-126-128-127-124-115-109-110-111-96 -111-110-114-127-128-135-136-137-136-143-77-78-74-75-70-67-36-38-37-38-68-11-68-67-66-71-72-79-80-81-80-82-156-73-146-151-49-152-150-151-150 -147-146-147-148-147- 88 -87-82-83 -85-83-84-83-89-

88- 89-92-93-94-85-86-93-92-91-90-88-90-148-149-150-149-153-152-154-47-48-44-43-157- 43-42-31-32-41-29-41-30-3-4-30-33-32-44-48-49-50-158-52-158-53-158-50-51-50-49-54-53-33-34-39-36-35-4-5-7-8-35-8-11-10-11-12-69-70-71-74-75-77-76-28-95 -112-111-112-113-114-115-116 -108-109 -97-98 -100-107-108 -116-117-118-117-107-106 -122 -123-120 -118-119 -120-121-105 -121-122 -106-101-106-105-104-119-104-103-102-101-100-98 -97-96-95 -28 -12-13-14-9-14-15-18-21-22-23-22-21-24-19-18-19-20-19-16-15-14-16-17-28-27-20-27-26-25-26-98 -99 -23-24-25-20-17-13-10-9-7-5-1.

Para que pudesse ser calculado o custo da solução obtida, precisou-se de um limite inferior para o problema do carteiro chinês não-orientado. Para esse limite inferior utilizou-se a soma dos comprimentos das arestas do grafo original (sem as duplicações obtidas pelo algoritmo de emparelhamento). Este valor é de: 23457,60 metros.

O custo da solução obtida pelo método heurístico proposto é igual ao limite inferior acrescido da soma dos comprimentos das arestas duplicadas, isto é: 23457,60 metros acrescido de 5046,22 metros, resultando em um valor final de 28503,82 metros. É este o comprimento da rota total que deve percorrer o carteiro chinês.

Sabe-se que o valor ótimo, embora desconhecido, está entre o limite inferior 23457,60 metros e o comprimento total da rota obtido 28503,82 metros.

Isto significa que o erro máximo que está sendo cometido entre o valor ótimo e o valor heurístico é de no máximo 5046,22 metros que equivale a no máximo 17,8%.

6.4 PROBLEMA DO CARTEIRO CHINÊS MISTO

O algoritmo do Carteiro Chinês para redes mistas foi desenvolvido com o objetivo de resolver o problema de cobertura de arcos (ruas de mão dupla ou única), para o percurso de rotas que utilizem veículos como a coleta de lixo.

6.4.1 Aplicação do Modelo de Programação Linear Inteira

O modelo para o problema do Carteiro Chinês Misto é o mesmo utilizado para o Problema do Carteiro Chinês Não-Orientado. A diferença entre os dois modelos

está nas tabelas de adjacências e custos, que passam a serem matrizes não simétricas, isto é, para um arco de um nó i para um nó j , pode-se não ter um arco do nó j para o nó i . Também foi modelado na linguagem GAMS. Esse modelo foi enviado ao sítio Neos *Server* para a resolução, usando os aplicativos de otimização: CBC, GLPK, Gurobi, Mosek e XPress. Vale lembrar que os resultados fornecidos após a aplicação desses otimizadores são limitados ou pela memória dos computadores ou pelo número máximo de iterações. Todos geraram uma solução inteira factível que não são necessariamente o custo ótimo para o problema. Um exemplo de resultado pode ser visualizado no apêndice 4. A partir do resultado fornecido pelo NEOS, encontrou-se a rota para cada aplicação usando-se o algoritmo *Pathwork*. Os resultados estão a seguir:

CBC

- Função Objetivo: 29400.3900 m
- Tempo computacional: 0.013 segundos:
- Rota:

1-2-3-30-33-34-40-39-34-40-57-40-59-60-59-58-64-66-65-39-36-35-4-3-30-33-53-54-49-54-55-54-56-57-61-58-65-39-36-35-8-7-8-11-12-69-70-75-74-75-77-143-113-114-110-111-110-109 -108 -107-100 -101-102-103 -104-119 -118 -120-121-105-106-107-108 -116-117-107-117-118 -120-123 -122-121-105 -104 -119 -120-123-122-106-101-100-98 -97-109 -115 -114-127-124-115-116-115-124-125-126-128-126-129-130-131-132-133 -139-138 -139 -140 -144 -145 -132-131-133-134-135-136-137-141-142-143-136 -135-128-127-128-129-134-138-137-141-142-78-79-144-140-141-142-78-77-76-69 -68-11-10-9 -10-13 -17-20 -27-26-27-28 -95-112-113-76-28-95-96-95-112-111-96-97-98 -26-25-26-25-20 -19-16-15 -18 -19-24 -21-22-23-99 -98-26-25-24-23-22-21-18-15-14-13-12-28-17-16-14-9 -7-5-6-5-4-30 -41-29 -41-32-33-53-158-52-158-50-51-50-49-151-146-147-150 -151-146-73-62-63 -61-62-63 -64-72-79-78 -74-71-66-67-68-38-37-38-36-67-70-71-72-156-73-87-82-156-82-80-79-80-81-80-82-83-84-83-85-94-93-86-85-83-89-88-87-88-89-92-93-92-91-90-88-147-148-90-148-149-153-152-150-149-153-155 -154-47-48-49-152-154 -47-45-46 -45-42-43 -157-43-44-48-44-32-31-42-31-29-3-2-5-1.

- GLPK
- Função Objetivo: 28579.1300 m
- Tempo computacional: 0.057 segundos:
- Rota:

1-2-3-30-33-34-40-39-34-40-57-40-59-60-59-58-64-66-65-39-36-35-4-3-30-33-53-54-49-54-55-54-56-57-61-58-65-39-36-35-8-7-8-11-12-69-70-75-74-75-77-143-113-114-110 -111-110-109 -108 -107-100 -101-102-103 -104 -119-118-120-121-105-106-107-108 -116-117-107-117-118 -120-123 -122-121-105 -104 -119 -120-123-122-106-101-100 -98 -97-109 -115-114-127-124-115-116-115-124-125-126-128-126-129-130-131-132-133-139 -138 -139-140 -144-145 -132-131-133 -134-135 -136-137-141-142-143-136-135-128 -127-128-129-134-138-137-141-142-78-79-144-140-141-142-78-77-76-69 -68 -11-10-9 -10-13-17-20-27-26-27-28-95-112-113-76-28-95-96-95-112-111-96-97-98-26-25-26-25-20-19-16-15-18-19-24-21-22-23-99-98-26-25-24-23-22-21-18-15-14-13-12-28-17-16-14-9-7-5-6 -5-4-30 -41-29-41-32-33-53-158-52-158-50-51-50-49-151-146-147-150-151-146 -73-62-63- 61-62-63-64-72-79-78-74-71-66-67-68 -38 -37-38-36 -67-70-71-72-156-73-87-82-156-82-80-79-80-81-80-82-83-84-83-85-94-93-86-85- 83 -89 -88 -87-88-89 -92-93-92-91-90-88-147-148-90-148-149-153-152-150-149-153 -155-154 -47-48 -49 -152-154-47-45-46-45-42-43-157-43-44-48-44-32-31-42-31-29-3-2-5-1.

- Gurobi
- Função Objetivo: 29308.8200 m
- Tempo computacional: 0.118 segundos:
- Rota:

1-2-3-30-33-34-40-39-34-40-57-40-59-60-59-58-64-66-65-39-36-35-4-3-30-33-53-54-49-54-55-54-56-57-61-58-65-39-36-35-8-7-8-11-12-69-70-75-74-75-77-143-113-114-110 -111-110-109 -108 -107-100 -101-102 -103 -104-119-118-120-121-105-106-107-108 -116-117-107-117-118 -120 -123 -122-121-105-104-119 -120 -123-122-106-101-100 -98 -97-109 -115-114-127-124-115-116-115-124-125-126-128-126-129-130-131-132-133 -139 -138 -139 -140 -144-145-132-131-133 -134-135-136-137-141-142-143-136 -135-128-127-128-129-134-138-137-141-142-78-79-144-140-141-142-78-77-76-69 -68 -11-10 -9 -10-13-17-20-27-26-27-28 -95-112-113-76-28-95-96-95-112-111-96-97-98-26-25-26-25-20-19-16-15-18-19-24-21-22-23-99-98-26-25-24-23-22-21-18-15-

14-13-12-28 -17-16-14-9-7-5-6-5-4-30-41-29 -41-32-33-53-158-52-158-50 -51-50-49-151-146-147-150 -151-146-73-62-63 -61-62-63 -64-72-79-78-74-71-66 -67-68-38-37-38-36-67-70-71-72-156-73-87-82-156-82-80-79-80 -81-80-82-83-84-83-85-94-93-86-85-83-89 -88-87-88-89 -92-93-92-91-90 -88 -147-148-90 -148-149-153-152-150-149-153-155-154-47-48-49 -152-154-47-45-46-45 -42-43-157-43 -44-48 -44-32-31-42-31-29-3-2-5-1.

- Mosek
- Função Objetivo: 29039.5700 m
- Tempo computacional: 0.055 segundos:

- Rota:

1-2-3-30-33-34-40-39-34-40-57-40-59-60-59-58-64-66-65-39-36-35-4-3-30-33-53-54-49-54-55-54-56-57-61-58-65-39-36-35-8-7-8-11-12-69-70-75-74-75-77-143-113-114-110-111-110-109-108-107-100 -101-102-103 -104-119 -118 -120 -121-105 -106-107-108 -116 -117-107-117-118-120 -123 -122-121-105-104-119 -120 -123-122-106-101-100-98 -97-109 -115-114-127-124-115-116-115-124-125-126-128 -126-129-130-131-132-133-139-138 -139-140 -144-145 -132-131-133 -134-135 -136 -137-141-142-143-136-135-128-127-128-129-134-138 -137-141-142-78-79-144-140-141-142-78-77-76-69 -68-11-10-9 -10-13-17-20-27-26-27-28 -95-112-113-76-28 -95-96 -95-112-111-96-97-98-26-25-26-25-20-19-16-15-18-19-24-21-22-23-99-98-26-25-24-23-22-21-18-15-14-13-12-28-17-16-14-9-7-5-6-5-4-30 -41-29 -41-32-33-53-158-52-158-50 -51-50-49-151-146-147-150 -151-146-73-62-63-61-62-63-64-72-79 -78 -74-71-66-67-68 -38-37-38-36-67-70-71-72-156 -73-87-82-156-82-80-79-80-81-80-82-83-84-83-85-94-93-86-85-83-89-88-87-88-89-92-93-92-91-90 -88-147-148-90 -148 -149 -153-152-150 -149-153-155-154-47-48-49 -152-154-47-45-46-45-42-43 -157-43 -44-48 -44-32-31-42-31-29-3-2-5-1.

- Xpress
- Função Objetivo: 30011.2400 m
- Tempo computacional: 0.080 segundos:
- Rota:

1-2-3-30-33-34-40-39-34-40-57-40-59-60-59-58-64-66-65-39-36-35-4-3-30-33-53-54-49-54-55-54-56-57-61-58-65-39-36-35-8-7-8-11-12-69-70-75-74-75-77-143-113-114-110-111-110-109 -108-107-100 -101-102-103 -104-119 -118 -120 -121-105-106-107-108 -116-117-107-117-118 -120 -123-122-121-105 -104-119 -120-123 -122-106-101-100 -98-97-109 -115-114-127-124-115-116-115-124-125-126-128-126-129-130 -131-132-133-139 -138-139 -140 -144-145-132-131-133 -134-135 -136 -137-141-142-143-136-135-128 -127-128-129-134-138-137-141-142-78-79-144-140-141-142-78-77-76-69-68-11-10 -9-10 -13-17-20 -27-26-27-28 -95-112-113-76 -28-95-96-95-112-111-96-97-98-26-25-26-25-20-19-16-15-18-19-24-21-22-23-99-98-26-25-24-23-22-21-18-15-14-13-12-28 -17-16-14-9 -7-5-6-5-4-30 -41-29-41-32-33-53-158-52-158-50-51-50-49-151-146-147-150 -151-146-73-62-63 -61-62-63 -64-72-79 -78-74-71-66-67-68-38-37-38-36-67-70-71-72-156-73-87-82-156 -82-80-79-80-81-80-82-83-84-83-85-94-93-86-85-83-89 -88-87-88-89-92-93-92-91-90 -88-147-148 -90-148-149 -153-152-150 -149-153-155 -154-47-48-49 -152-154-47-45-46 -45 -42-43-157-43-44-48-44-32-31-42-31-29-3-2-5-1.

Os dados foram organizados em forma de tabela, para melhor visualização e interpretação dos resultados e verificação de qual linguagem forneceu melhor resultado da função objetivo otimizada, comparando com a rota atual percorrida pelo caminhão de coleta de resíduos é de 32725,54 metros:

TABELA 5 – RESULTADOS FORNECIDOS PELO APLICATIVO NEOS

Linguagem de Programação	Tempo de Execução	Resultado da Função Objetivo
Cbc	0.013 segundos	29400,3900m
GLPK	0.057 segundos	28579,1300m
Gurobi	0.118 segundos	29308,8200m
Mosek	0,055 segundos	29039,5700m
Xpress	0.080 segundos	30011,2400m

FONTE: O autor (2012)

Pode-se verificar que o melhor resultado foi fornecido pela linguagem de programação GLPK, com um valor da função objetivo igual a 28579,13m, que comparado com o valor da rota atual realizada pelo caminhão de coleta de resíduos oferece uma economia de 12,87%.

7. CONSIDERAÇÕES FINAIS

Considerando a dificuldade de se resolver problemas de grande escala de forma exata, soluções aproximadas podem ser uma boa alternativa com um tempo computacional aceitável. Ainda que os valores obtidos possam não ser exatos, é possível a obtenção de limites inferiores para se avaliar os erros que estão sendo cometidos. O objetivo dessa pesquisa foi em sua totalidade alcançado, pois a metodologia desenvolvida foi implementada e foram obtidos resultados adequados na otimização tanto da rota do caminhão de coleta de resíduos sólidos, quanto no percurso para entrega de correspondências, leitura e entrega de faturas de água e energia elétrica ou ainda na limpeza das ruas da cidade de Irati/PR.

No caso do problema não direcionado, o método heurístico apresentou um custo apenas 0,062% maior que o melhor resultado encontrado usando-se PLI no Sítio NEOS, com a aplicação do otimizador GLPK. Desta forma é possível concluir que a heurística pode ser aplicada para o problema abordado com vantagens, uma vez que é um algoritmo de simples implementação que fornece resultados em um tempo computacional ínfimo, em torno de 1 segundo. Apesar do resultado ser adequado e o tempo computacional despendido na utilização dos otimizadores do sítio NEOS ser atrativo, sua utilização não é ideal porque depende da mudança de dados na modelagem GAMS e também, quando enviado o modelo para resolução, pode-se eventualmente ter que esperar pela solução numa fila de usuários.

A melhor solução do percurso otimizado para o Problema do Carteiro Chinês Misto, obtida com o otimizador GLPK, minimizou a rota atual em 12,67%. O resultado pode gerar uma economia significativa, com relação às despesas com combustível, desgaste dos veículos, tempo de coleta e mão de obra.

Quando comparados os resultados para o modelo de PLI para o PCCNO e o PCCM, pode-se observar que para o problema misto, o melhor resultado ficou apenas 0,325% acima do resultado para o caso não orientado. Esse percentual corresponde a aproximadamente 92,9m.

Se as metodologias desenvolvidas forem aplicadas à cidade de Irati/PR, pode-se conseguir uma redução nos custos, tanto para o problema não-orientado como para o misto, o que pode gerar uma economia significativa para a prefeitura do município. Menor distância percorrida pelo caminhão de coleta de resíduos sólidos proporciona redução na manutenção e despesas com combustível dos veículos.

Com a otimização é possível também uma melhor alocação do pessoal, podendo inclusive reduzir o número de funcionários, envolvidos nessas atividades podendo-se dessa forma direcioná-los para outras competências.

8. SUGESTÕES PARA TRABALHOS FUTUROS

Visando o aprimoramento dos resultados obtidos no presente estudo, são propostas algumas sugestões para trabalhos futuros:

- a) Redistribuir a frota de caminhões de modo a atender todo o município, organizando os horários e as rotas que cada veículo fará em determinado dia e horário;
- b) Utilizar maior quantidade de restrições como, por exemplo, a quantidade de resíduos coletados em cada caminhão de lixo, os dias da semana, o tempo gasto no trajeto, a capacidade de cada veículo, entre outros;
- c) Simular ganhos financeiros com a implementação do trabalho na região analisada, assim como para as demais regiões da referida cidade;
- d) Desenvolver outras técnicas envolvendo heurísticas e metaheurísticas para a comparação em relação à eficiência dos resultados.

9. REFERÊNCIAS

APAYDIN, O.; GONULLU, M. T.; **Route optimization for solid waste collection: Trabzon (Turkey) case study**. Global NEST Journal, V.9, nº1, Istanbul-Turkey, 2007.

BARBOZA, A.O. (2005). **Simulação e Técnicas da Computação Evolucionária Aplicadas a Problemas de Programação Linear Inteira Mista**. Tese de Doutorado, Curitiba, PR: UTFPR.

BELTRAMI, E. J.; BODIN, L. D.; **Networks and Vehicle Routing for Municipal Waste Collection**. Networks 4, 65-94, 1974.

BOREKI, V.; MEIO AMBIENTE/ Sipar representa evolução no tratamento de lixo <http://www.mp.pr.gov.br/modules/conteudo/conteudo.php?conteudo=1267> Acessado em 02/02/2011 às 15:45 hrs.

CHRISTOFIDES, N.; CAMPOS, V.; CORBERÁN, A.; MOTA, E.; **An Algorithm for the Rural Postman Problem**. Imperial College Report IC.ºR.81.5, London, 1981.

COSTA, D. M. B.; STEINER, M. T. A.; CARNIERI, C.; ZAMBONI, L. V. S.; SILVA, A. C. L. **Técnicas da Pesquisa Operacional Aplicada na Otimização dos Serviços Postais**. Revista Gestão e Produção, v.8, nº 1, Abril, 2001.

D'ALMEIDA, Maria Luiza Otero (coord.) **Lixo Municipal: Manual de Gerenciamento Integrado** Brasília: CEMPRE, 2002, 2ª ed.

DETOFENO, T. C. **Otimização de rotas e coleta de resíduos sólidos urbanos, utilizando técnicas de Pesquisa Operacional**. Dissertação de Mestrado do Programa de Pós-Graduação em Métodos Numéricos em Engenharia – Área: Programação Matemática. Curitiba, 2009.

EDMONDS, J., JOHNSON, E. L., **Matching, Euler Tours and the Chinese Postman Problem**, Math. Program. 5, 1973.

ELISELT, H. A.; GENDREAU, M.; LAPORTE, G. **Arc Routing Problems, Part I: The Chinese Postman Problem**, Operations Research, Vol. 43, Nº 2, 1995.

FIÓRIO, R. C. **Uma Abordagem Heurística para o Problema de Otimização de Distrito Postal**. Dissertação de Mestrado da Universidade Federal do Espírito Santo, Vitória/ES, Junho/2006.

FREDERICKSON, G. N.; HECHT, M. S., KIM, C. E., **Approximation Algorithms for Some postman Problems**. SIAM J. Comput. 7, 178-193, 1978.

GODINHO FILHO, M.; JUNQUEIRA, R. Á. R.; **Um Algoritmo para Auxiliar na Escolha de Métodos de Solução para o Problema do Carteiro Chinês: uma Proposta e Aplicação em uma Grande Cidade do Interior Paulista**. XII SIMPEP – Bauru/SP, de 07 a 09 de Novembro de 2005.

GOLDBARG, Marco Cesar. **Otimização combinatória e programação linear: modelos e algoritmos**. 2ª Edição – Rio de Janeiro, 2005.

GOMES, M. J. N.; COELHO Jr., W. R.; PALHANO, A. W. C.; COUTINHO, E. F.; CASTRO, G. A.; GOMES, F. J. N.; BARCELLOS, G. C.; RESENDE, B. F.; PEREIRA, L. W. L.; **O Problema do Carteiro Chinês, Algoritmos Exatos e um Ambiente MVI para Análise de suas Instâncias: Sistema XNÊS**. Revista Pesquisa Operacional, v.29, nº2, maio a agosto de 2009.

GROTSCHER, M., WIN, Z., **A Cutting plane Algorithm for the Windy Postman Problem, Mathematical Programming**, Vol. 55, 1992.

GUAN, M. K., **On the Windy Postman Problem**. Discrete Appl. math. 9, 1984.

Hamilton Carlos Massaro dos Santos; Paulo Morelato França; **Meta-heurística para Programação da Produção com Tempo de Preparação Dependentes da Sequência**; Artigo Scielo, UNICAMP – Campinas/SP, Dezembro/1995. Acessado em 07/04/2011, às 15:08h; <http://www.scielo.br/pdf/gp/v2n3/a01v2n3.pdf> .

IBGE – **INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA**. http://www.censo2010.ibge.gov.br/dados_divulgados/index.php?uf=41 - Acessado em 25/02/2011 às 19:08h.

IRATI, Prefeitura Municipal. **Secretaria Municipal de Obras e Serviços Urbanos: Secretaria de Planejamento**. 2011.

LACERDA, M. G.; **Análise de Uso de Sig no Sistema de Coleta de Resíduos Sólidos Domiciliares em uma Cidade de Pequeno Porte**. Dissertação de Mestrado. Universidade Estadual Paulista – UNESP, Ilha Solteira/SP, 2003.

LIN, Y., ZHAO, Y., **A New Algorithm for the Directed Chinese Postman Problem**, Comput. Opns. Res. Vol.15, No. 6, 1988.

METAHEURISTICS NETWORK. Metaheuristic. Disponível em: <http://www.metaheuristics.net/index.php?main=1&sub=11> . Acesso em 14 de março de 2011.

MINIEKA, E., The **Chinese Postman Problem for Mixed Networks**, Management Sciences, Vol 25, N°7, 1979.

MORO, L. F. L. **Técnicas de Otimização Mista-Inteira para o Planejamento e Programação de Produção em Refinarias de Petróleo**. Tese de Doutorado. Escola Politécnica da Universidade de São Paulo, SP, Brasil, 2000.

NEOS – Disponível em: <http://neos-server.org/> ; acessado em 16/12/2011, às 15:23hrs.

NICOLETTI, M. C.; FIGUEIRA, L. B.; HRUSCHKA JR., E. R.; **Transferring neural network based knowledge into an exemplar-based learner, Neural Computing & Applications**, vol. 1, pp. 10-20, 2007.

NUNES, L. F. **Algoritmos Genéticos aplicados na abordagem de um problema real de roteamento de veículos**. Curitiba, 1998. Dissertação de do Programa de Pós-Graduação em Métodos Numéricos em Engenharia, UFPR, 1998.

ORLOFF, C. S., **A Fundamental Problem in Vehicle Routing**. Networks 4, 35-64, 1974.

PAES, F. G.; ARICA, G. M. **Um Tratamento Grasp Híbrido do Problema do Carteiro Chinês Misto (Pccm) Destinado a Otimização de Rotas para Coleta do Lixo Doméstico** – SBMAC, Porto Alegre/RS, 2004.

PAPADIMITRIOU, C. H., **On the Complexity of the Edge Traversing**, J. Assoc. Comput. Mach. 23, 1976.

PASCOAL Jr., A.; OLIVEIRA FILHO, P. C.; **Análise de Rotas de Coleta de Resíduos Sólidos Domiciliares com o Uso de Geoprocessamento**. Revista Acadêmica, Ciências Agrárias e Ambientais, Curitiba, V. 8, nº 2, Abril/Junho, 2010.

PEARN, W. L.; LI, M. L., **Algorithms for the Windy Postman Problem**, Computers Ops Res. Vol. 21, N°6, 1994.

PEARN, W. L.; LIU, C. M. **Algorithms for the Chinese Postman Problem**, On Mixed Networks, Computers Ops Res. Vol. 22, N°5, 1995.

Pedro Henrique de Almeida Konzen; João Carlos Furtado; Claudia Wollmann Carvalho; Marco Flôres Ferrão; Rolf Fredi Molz; Irionson Antonio Bassani; Silvio Luiz Hüning – **Otimização de métodos de controle de qualidade de fármacos usando algoritmo genético e busca tabu** - Pesquisa Operacional – Artigo Scielo, vol.23, nº.1 Rio de Janeiro, Jan./Apr. 2003. Acessado em 07/04/2011 às 14:03h –

http://www.scielo.br/scielo.php?pid=S0101-74382003000100014&script=sci_arttext

PEDROSO, k.; CERUTI, C., **PROGRAMA DE GERENCIAMENTO DE RESÍDUOS SÓLIDOS URBANOS: UMA PROPOSTA PARA PEQUENOS MUNICÍPIOS BRASILEIROS**. Artigo apresentado na VII Semana de Engenharia Ambiental – Keylla Pedroso; Fabiane Cristina Ceruti; Proposta de Gerenciamento de Resíduos Sólidos Urbanos: Uma Proposta para Pequenos Municípios Brasileiros. Artigo publicado na VII Semana de Engenharia Ambiental, UNICENTRO – Campus de Irati, de 01 a 04 de junho 2009. Disponível em http://www.unicentro.br/graduacao/deamb/semana_estudos/pdf_09/PROGRAMA%20DE%20GERENCIAMENTO%20DE%20RES%20CDDUOS%20S%20D3LIDOS%20URBANOS%20UMA%20PROPOSTA%20PARA%20PEQUENOS%20MUNIC%20CDPIOS%20BRASI.pdf - Acessado em 02/02/2011 às 18:08 hrs.

PNRS. **Panorama dos Resíduos Sólidos no Brasil**. Disponível em: <http://www.abrelpe.org.br/> Acesso em: 02/02/2011.

RICHEY, M. B.; PARKER, R. G.; **A cubic algorithm for the directed Eulerian sub graph problem**. European Journal of Operational Research 50, 1991.

ROSENTHAL, R. E. **GAMS – A User's Guide**. GAMS Development Corporation, Washington, DC, USA, 2011.

SAMPAIO, R. M.; YANASSE, H. H., **Estudo e Implementação de Algoritmos de Roteamento sobre Grafos em um Sistema de Informações Geográficas**, INPE – Instituto Nacional de Pesquisas Espaciais, LAC – Laboratório Associado de Computação e Matemática Aplicada, 2007 – <http://www.dcc.ufla.br/infocomp/artigos/v3.1/art12.pdf>- acessado em 25/02/2011 às 21:10hrs.

SEMA – Desperdício Zero. Acessado em 29/04/2009, às 14:20hrs. <http://www.meioambiente.pr.gov.br/>

SHERAFAT, Hassan. **Algoritmos Heurísticos de Cobertura de Arcos**. Tese de Doutorado, Universidade Federal de Santa Catarina, Florianópolis-SC, 2004.

SILVA, A. F.; MARINS, F. A. S.; SILVA, G. M.; LOPES, P. R. M. A.; **Pesquisa Operacional: Desenvolvimento e Otimização de Modelos Matemáticos por meio de Modelos Matemáticos da Linguagem GAMS**. Notas de Aula. UNESP, (2009)

SILVA, C.A.; SOUZA, S.R.; **Uma Aplicação da Meta-heurística Híbrida Simulated Annealing-Iterated Local Search ao Problema de Fluxo Multiproduto sob o Espaço Capacitado** - Programa de Pós-Graduação em Modelagem Matemática e Computacional, CEFET-MG, 10/09/2010. Acessado em 07/04/2011, às 15:36h; http://www.sbmec.org.br/tema/seletas/docs/v9_1/17Sil_Sou.pdf

SILVA, A. F. **Modelagem do planejamento agregado da produção de uma usina sucroalcooleira**. Programa de Pós-graduação em Engenharia de Produção, Universidade Federal de Itajubá, Dissertação de Mestrado, Itajubá/MG, 2009.

SOUZA, A., **JORNAL HOJE CENTRO SUL** - <http://www.hojecentrosul.com.br/geral/ibge-divulga-resultados-extra-oficiais-do-censo-2010/> - acessado em 25/02/2011 às 19:46hrs.

XAVIER, R.; LISBOA, A.; VIEIRA, D.; SALDANHA, R.; **Heurísticas para modelagem e minimização do consumo de combustível para rotas de coleta de lixo**. XLII SBPO – Bento Gonçalves/RS, 2010.

ZIONTS, S. **Linear and Integer Programming**. Prentice-Hall, Inc, Englewood Cliffs, New Jersey, United States of America, 1974.

<http://www.irati.pr.gov.br/municipio/localizacao geografica.asp>

- Página Prefeitura Municipal de Irati, acessada em 05/12/2011 às 13:24hrs.

https://docs.google.com/viewer?a=v&q=cache:r5dgCM6te6QJ:www.din.uem.br/~ademir/grafos/arquivos/E1-GrafoEuler_Hamilton.pdf+uma+solu%C3%A7%C3%A3o+ao+problema+%C3%A9+a+seguinte.+Transforme+o+grafo+em+grafo+euleriano,+duplicando+as+arestas+que+formam+o+caminho+mais+curto+entre+esses+dois+v%C3%A9rtices+de+grau+%C3%ADmpar&hl=pt-BR&gl=br&pid=bl&srcid=ADGEESg-

[px4APf85LtpWM8B8ZvJruUfg8i_vB_EGNq7KexluuKVNYNi_XnDcsLvKppNT14o71et
hm9SKGT8j_KLsbz2XwpWbFC110YeNct_EOVFh4_ZKxD9CCGmOW8kD_KXfNEs
RIEul&sig=AHIEtbSpfzOaMnPTMvivL0HwBYBbUwsmgg](#) - acessado em
08/12/2011, às 14:39hrs.

ANEXO I – EXEMPLO DO PROBLEMA DO CARTEIRO CHINÊS

De acordo com Goldberg (2005), o PCC é um exemplo de roteamento que admite solução em tempo polinomial, apresentando um ciclo Euleriano no grafo representado, onde a solução para o caso é o caminho fechado {1-2-6-7-2-3-4-5-1-6-8-1}.

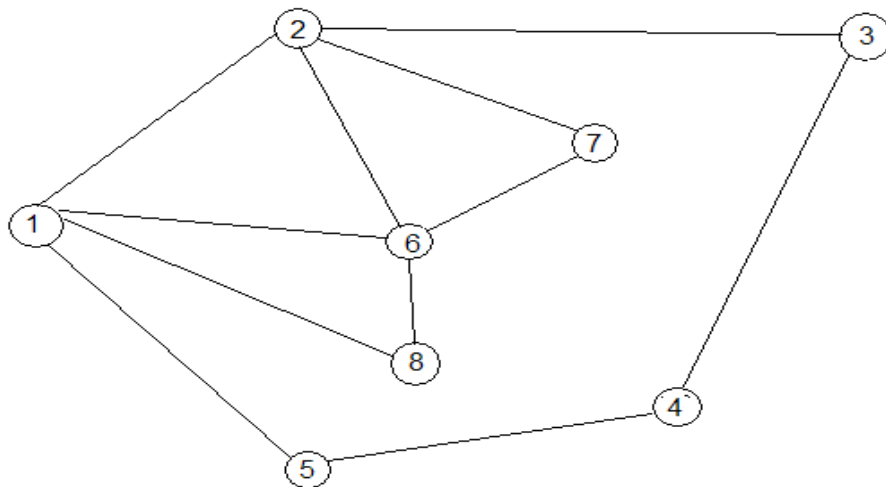


FIGURA 12: GRAFO G
FONTE: GOLDBARG (2005)

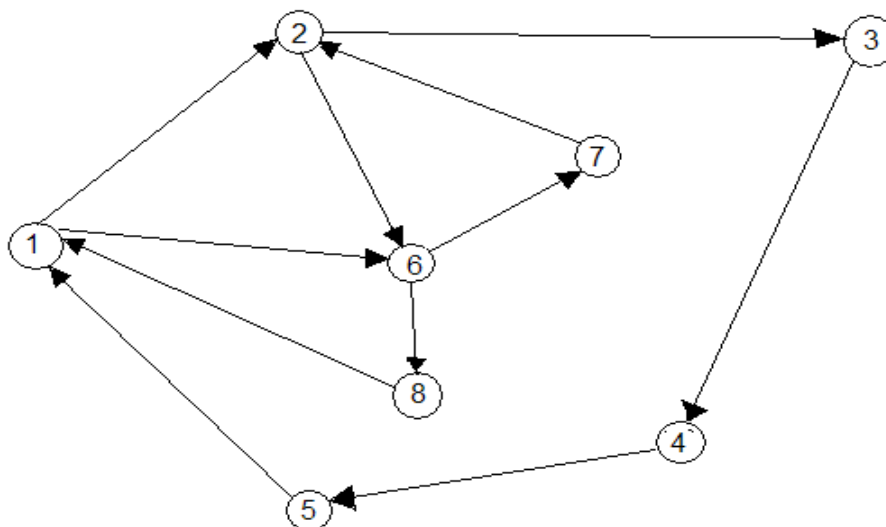


FIGURA 13: GRAFO EULERIANO
FONTE: GOLDBARG (2005)

ANEXO II – RELATÓRIOS DE VISITAÇÃO DO CAMINHÃO DE COLETA DE RESÍDUOS

Secretaria Municipal de Meio Ambiente - Irati - Paraná								
Data: 07/02/2011		Roteiro:						
Caminhão/Placa: ANK 3580		Segunda feira Motorista Marcelo						
Horário	Percurso			Conver-sões	Reves-timento	Decli-vidade	Tipo de Tráfego	Observações
	Rua	Rua A	Rua B					
	CENTRO							
7:21	Maria Bittar do Prado (Patio)	Pátio	João C. Ferreira	VE	AS	PL	MD	
:	João C. Ferreira	Maria Bittar do Prado (Patio)	Paulo Xisto	SF	PR	S+	MD	
:	João C. Ferreira	Paulo Xisto	Pacifico Borges	VE	PR	PL	MD	
:	Pacifico Borges	João C. Ferreira	Paulo Xisto	VD	AS	PL	MD	
:	Paulo Xisto	Pacifico Borges	João Wasilevski	VD	PR	S-	MD	
:	João Wasilevski	Paulo Xisto	Moises de Oliveira	VD	PR	PL	MD	
:	Moises de Oliveira	João Wasilevski	Lino Esculápio	VE	AS	D-	MD	
:	Lino Esculápio	Moises de Oliveira	Caetano Zarpelon	VE	AS	PL	MD	
:	Caetano Zarpelon	Lino Esculápio	Padre Pedro	VE	RP	S-	MD	
:	Padre Pedro	Caetano Zarpelon	João Wasilevski	VE	RP	S-	MD	
:	João Wasilevski	Padre Pedro	Moises de Oliveira	VE	AS	PL	MD	
:	Moises de Oliveira	João Wasilevski	Lino Esculápio	SF	AS	D-	MD	
:	Moises de Oliveira	Lino Esculápio	João C. Ferreira	VE	AS	D-	MD	
:	João C. Ferreira	Moises de Oliveira	Caetano Zarpelon	VE	PR	PL	MD	
:	Caetano Zarpelon	João C. Ferreira	Lino Esculápio	RR	AS	PL	MD	
:	Caetano Zarpelon	Lino Esculápio	João C. Ferreira	SF	AS	PL	MD	
:	Caetano Zarpelon	João C. Ferreira	Exp. José de Lima	VE	AS	PL	MD	
:	Exp. José de Lima	Caetano Zarpelon	Basilio Szpak	VD	AS	PL	MD	
:	Basilio Szpak	Exp. José de Lima	Exp. José B. Susko	SF	AS	PL	MD	
:	Basilio Szpak	Exp. José B. Susko	Antonio Valença	MR/VD	AS	PL	MD	Apenas até metade da quadra e retornou
:	Basilio Szpak	Antonio Valença	Exp. José B. Susko	VD	AS	PL	MD	
:	Exp. José B. Susko	Basilio Szpak	Arcélio B. Teixeira	VD	AS	PL	MD	
:	Arcélio B. Teixeira	Exp. José B. Susko	Antonio Valença	SF	AS	PL	MD	GG Condomínio
:	Arcélio B. Teixeira	Antonio Valença	Av. Dr. Vicente Machado	RR	PR	D-	MD	GG Posto
:	Arcélio B. Teixeira	Av. Dr. Vicente Machado	Antonio Valença	VE	PR	S-	MD	
:	Antonio Valença	Arcélio B. Teixeira	Basilio Szpak	SF	AS	PL	MD	
:	Antonio Valença	Basilio Szpak	Casemiro Sinderski	VD	AS	PL	MD	
:	Casemiro Sinderski	Antonio Valença	Exp. José de Lima	VD	AS	PL	MD	
08:00	Exp. José de Lima	Casemiro Sinderski	Caetano Zarpelon	MR/RR	AS	PL	MD	SC / Voltado de ré até o meio da quadra
:	Exp. José de Lima	Casemiro Sinderski	Moises de Oliveira	VD	AS	PL	MD	
:	Moises de Oliveira	Exp. José de Lima	João C. Ferreira	VE	AS	S-	MD	
:	João C. Ferreira	Moises de Oliveira	Pacifico Borges	VE	PR	PL	MD	
:	Pacifico Borges	João C. Ferreira	Exp. José de Lima	VE	PR	D-	MD	
:	Exp. José de Lima	Pacifico Borges	Moises de Oliveira	VD	PR	D-	MD	
:	Moises de Oliveira	Exp. José de Lima	Av. Dr. Vicente Machado	VD	AS	D-	MD	
:	Av. Dr. Vicente Machado	Moises de Oliveira	Pacifico Borges	VD	AS	S-	MD	
:	Pacifico Borges	Av. Dr. Vicente Machado	Exp. José de Lima	VE	PR	S-	MD	
:	Exp. José de Lima	Pacifico Borges	Paulo Xisto	VE	AS	PL	MD	
:	Paulo Xisto	Exp. José de Lima	Av. Dr. Vicente Machado	VE	AS	D-	MD	Travessa Paulo Xisto é peço a pé trecho curto
:	Av. Dr. Vicente Machado	Paulo Xisto	Pacifico Borges	VE	AS	S-	MD	
:	Pacifico Borges	Av. Dr. Vicente Machado	Exp. José de Lima	VE	PR	S-	MD	SC
:	Exp. José de Lima	Pacifico Borges	Paulo Xisto	VD	AS	PL	MD	SC
:	Paulo Xisto	Exp. José de Lima	João C. Ferreira	VE	AS	PL	MD	
:	João C. Ferreira	Paulo Xisto	Maria Bittar do Prado (Patio)	SF	AS	D+	MD	SC
:	João C. Ferreira	Maria Bittar do Prado (Patio)	TV João Candido Ferreira	VD	AS	D-	MD	
:	TV João Candido Ferreira	João C. Ferreira	Sem Saída	RR	RP	PL	MD	
:	TV João Candido Ferreira	Sem Saída	João C. Ferreira	SF	RP	PL	MD	SC
:	TV João Candido Ferreira	João C. Ferreira	Sem Saída	RR	RP	PL	MD	
:	TV João Candido Ferreira	Sem Saída	João C. Ferreira	VE	RP	PL	MD	
:	João C. Ferreira	TV João Candido Ferreira	AV. Dr. Armando Van Der Laars	SF	AS	S-	MD	
:	João C. Ferreira	AV. Dr. Armando Van Der Laars	Rua Benedito de Moraes	SF	AS	MD	S-	
:	João C. Ferreira	Rua Benedito de Moraes	Nossa Senhora de Fátima	SF	AS	M.U	S+	
:	João C. Ferreira	Nossa Senhora de Fátima	Prof. Vitor do Amaral	SF	AS	M.U	S+	
:	João C. Ferreira	Prof. Vitor do Amaral	Antonio Candido Cavalin	VD	PR	M.U	PL	
:	Antonio Candido Cavalin	João C. Ferreira	Lino Esculápio	SF	AS	M.U	D-	
:	Antonio Candido Cavalin	Lino Esculápio	Zeferino Bitencourt	VD	AS	M.U	D-	
:	Zeferino Bitencourt	Antonio Candido Cavalin	Prof. Vitor do Amaral	SF	AS	MD	D-	SC
:	Zeferino Bitencourt	Prof. Vitor do Amaral	Nossa Senhora de Fátima	VE	AS	MD	S-	SC
:	Nossa Senhora de Fátima	Zeferino Bitencourt	Poto de Coleta do Hospital Mun.	RR	AS	MD	S-	Coleta no Hospital Municipal/ Coletado pela man
:	Nossa Senhora de Fátima	Poto de Coleta do Hospital Mun.	Zeferino Bitencourt	VD	AS	MD	D-	SC
:	Zeferino Bitencourt	Nossa Senhora de Fátima	Prof. Vitor do Amaral	SF	D-	AS	MD	SC
:	Zeferino Bitencourt	Prof. Vitor do Amaral	Antonio Candido Cavalin	VD	S-	AS	MD	SC
:	Antonio Candido Cavalin	Zeferino Bitencourt	Abilio Carvalho Bastos	SF	D-	AS	M.U	
:	Antonio Candido Cavalin	Abilio Carvalho Bastos	Nereu Ramos	SF	S+	AS	M.U	
:	Antonio Candido Cavalin	Nereu Ramos	Julio Vieira Lisboa	SF	S+	AS	UM	
:	Antonio Candido Cavalin	Julio Vieira Lisboa	Manoel Alves do Amaral	VE	D-	AS	UM	
08:54	Manoel Alves do Amaral	Antonio Candido Cavalin	Coleta no Cavalim Bora	RR	PL	AS	MD	GG CAVALIM BORA 8 Min.
:	Manoel Alves do Amaral	Coleta no Cavalim Bora	Antonio Candido Cavalin	VE	PL	AS	MD	
:	Antonio Candido Cavalin	Manoel Alves do Amaral	José Augusto da Silva	SF	D-	AS	UM	
:	Antonio Candido Cavalin	José Augusto da Silva	Francisco Stroparo	SF	S+	AS	UM	
:	Antonio Candido Cavalin	Francisco Stroparo	Traiano Grácia	VD	D-	AS	UM	
:	Traiano Grácia	Antonio Candido Cavalin	Joaquim Ferreira Guimarães	SF	D-	AS	MD	
:	Traiano Grácia	Joaquim Ferreira Guimarães	Daniel Moreira	VD	D-	AS	MD	
:	Daniel Moreira	Traiano Grácia	Alfredo Nunes do Nascimento	VE	PL	AS	MD	SC
:	Alfredo Nunes do Nascimento	Daniel Moreira	Argentina	VD	S-	RP	MD	SC
:	Argentina	Alfredo Nunes do Nascimento	AV Presidente G. Vargas	SF	PL	PR	MD	Coleta especial casa de um portadorde necessi
:	Argentina	AV Presidente G. Vargas	João Zarpelon	VD	PL	PR	MD	SC
:	João Zarpelon	Argentina	Daniel Moreira	VD	PL	AS	MD	SC
:	Daniel Moreira	João Zarpelon	AV Presidente G. Vargas	SF	PL	AS	MD	SC
:	Daniel Moreira	AV Presidente G. Vargas	Traiano Grácia	VD	D-	AS	MD	SC

FIGURA 14 – ROTA ATUAL 1
FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

Secretaria Municipal de Meio Ambiente - Irati - Paraná

Data: 01/02/2011		Roteiro: Terça feira Motorista Marcelo		Conver-sões	Reves-timento	Decli-vidade	Tipo de Tráfego	Observações
Caminhão/Placa: ANK 3580		Percurso						
Horário	Rua	Rua A	Rua B					
								CENTRO
7:12	Rua do Pátio	patio	João Candido Ferreira	VE	AS	PL	MD	PO
	João Candido Ferreira	Travessa João Candido Ferreira	Paulo Xisto	VE	AS	S-	MD	SC/PP
	Paulo Xisto	João Candido Ferreira	Pacifico Borges	SF	AS	PL	MD	SC/PP
	Paulo Xisto	Pacifico Borges	João Wasilewski	VD	PR	PL	MD	SP/coletado apenas da creche
	João Wasilewski	Paulo Xisto	Moises de Oliveira	VD	PR	PL	MD	SC/PP
	Moises de Oliveira	João Wasilewski	Lino Esculapio	SF	AS	D-	MD	SC
	Moises de Oliveira	Lino Esculapio	João Candido Ferreira	VD	AS	D-	MD	SC
	João Candido Ferreira	Moises de Oliveira	Pacifico Borges	SF	AS	D-	MD	SC
	João Candido Ferreira	Pacifico Borges	Paulo Xisto	SF	AS	D-	MD	SC
	João Candido Ferreira	Paulo Xisto	Rua do Pátio	SF	AS	PL	MD	SC
	João Candido Ferreira	Rua do Pátio	Trav. João Candido Ferreira	SF	AS	PL	MD	SC
	João Candido Ferreira	Trav. João Candido Ferreira	Dr. Armando Van Der Laars	SF	AS	PL	MD	SC
	João Candido Ferreira	Dr. Armando Van Der Laars	Benedito de Moraes	SF	AS	PL	MD	SC
	João Candido Ferreira	Benedito de Moraes	N. S. de Fátima	SF	AS	PL	MD	SC
	João Candido Ferreira	N. S. de Fátima	Prof. Vitor do Amaral	SF	AS	PL	MD	SC
	João Candido Ferreira	Prof. Vitor do Amaral	Antonio C. Cavalin	VD	AS	PL	MD	SC
	Antonio C. Cavalin	João Candido Ferreira	Lino Esculapio	SF	AS	PL	M.U.	PP
	Antonio C. Cavalin	Lino Esculapio	Zeferino Bittencout	SF	AS	PL	M.U.	
	Antonio C. Cavalin	Zeferino Bittencout	Abilio C. Bastos	SF	AS	PL	M.U.	
	Antonio C. Cavalin	Abilio C. Bastos	Nereu Ramos	SF	AS	PL	M.U.	
	Antonio C. Cavalin	Nereu Ramos	Julio Vieira Lisboa	SF	AS	PL	M.U.	
	Antonio C. Cavalin	Julio Vieira Lisboa	Manoel A. do Amaral	VE	AS	PL	M.U.	
	Manoel A. do Amaral	Antonio C. Cavalin	Traiano Grácia	RR	AS	PL	MD	GG Cavalin Bora
	Manoel A. do Amaral	Traiano Grácia	Antonio C. Cavalin	VE	AS	PL	MD	
	Antonio C. Cavalin	Manoel A. do Amaral	Dr. José Augusto da Silva	SF	AS	PL	M.U.	
	Antonio C. Cavalin	Dr. José Augusto da Silva	Francisco Stroparo	SF	AS	S+	M.U.	
	Antonio C. Cavalin	Francisco Stroparo	Traiano Grácia	VD	AS	PL	M.U.	
	Traiano Grácia	Bifurcação em Y	Des. Joaquim Ferreira Guimarães	SF	AS	D-	MD	
	Traiano Grácia	Des. Joaquim Ferreira Guimarães	Daniel Moreira	SF	AS	D-	MD	
	Traiano Grácia	Daniel Moreira	Olavo Bilac	SF	AS	PL	MD	
	Traiano Grácia	Olavo Bilac	José Filus	SF	AS	PL	MD	
	Traiano Grácia	José Filus	Prof. Tico Lopes	SF	AS	PL	MD	Rotatória
	Traiano Grácia	Prof. Tico Lopes	Snr. Lopes	SF	AS	PL	MD	
	Traiano Grácia	Snr. Lopes	Decuciano de Quadros	VD	AS	PL	MD/QM	

FIGURA 15 – ROTA ATUAL 2
FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

Secretaria Municipal de Meio Ambiente - Irati - Paraná

Data: 03/02/2011		Roteiro: Centro, Fósforo, Jardim Virginia		Conver-sões	Reves-timento	Decli-vidade	Tipo de Tráfego	Observações
Caminhão/Placa: ANK 3580		Percurso						
Horário	Rua	Rua A	Rua B					
								CENTRO
7:13	Maria Bittar do Prado (Pátio)	Pátio	João Candido Ferreira	VE	AS	PL	MD	
	João Candido Ferreira	Maria Bittar do Prado (Pátio)	Paulo Xisto	VE	AS	S+	MD	
	Paulo Xisto	João Candido Ferreira	Pacifico Borges	SF	AS	PL	MD	
	Paulo Xisto	Pacifico Borges	Moises de Oliveira	VD	PR	S-	MD	
	Moises de Oliveira	Paulo Xisto	João Candido Ferreira	SF	AS	D-	MD	QM
	Moises de Oliveira	João Candido Ferreira	Expedicionário José de lima	VD	AS	D-	MD	
	Expedicionário José de lima	Moises de Oliveira	Pacifico Borges	SF	AS	D-	MD	
	Expedicionário José de lima	Pacifico Borges	Paulo Xisto	SF	AS	D-	MD	
	Expedicionário José de lima	Paulo Xisto	Sam Saida	RR	AS	S-	MD	
	Expedicionário José de lima	Sam Saida	Paulo Xisto	VE	AS	PL	MD	
	Paulo Xisto	Expedicionário José de lima	João Candido Ferreira	VE	AS	PL	MD	
	João Candido Ferreira	Paulo Xisto	Armando van der laars	VD	AS	D-	MD	
	Armando van der laars	João Candido Ferreira	Fundos da Secretária de Saúde	RR	RP	PL	MD	SP
	Armando van der laars	Fundos da Secretária de Saúde	João Candido Ferreira	VD	RP	PL	MD	
	João Candido Ferreira	Armando van der laars	Benedito de Moraes	SF	AS	S-	MD	
	João Candido Ferreira	Benedito de Moraes	Nossa Senhora de Fátima	SF	AS	S-	MD	SC
	João Candido Ferreira	Nossa Senhora de Fátima	Professor Vitor do Amaral	SF	PR	S+	MD	SC
	João Candido Ferreira	Professor Vitor do Amaral	Antonio Candido Cavalin	VD	PR	S+	MD	SC
	Antonio Candido Cavalin	João Candido Ferreira	Lino Esculapio	VE	PR	S+	MD	SC
	Lino Esculapio	Antonio Candido Cavalin	Dezenove de Dezembro	RR	PR	PL	MD	Até meio da quadra e retorno
	Lino Esculapio	Dezenove de Dezembro	Antonio Candido Cavalin	VE	AS	PL	M.U.	
	Antonio Candido Cavalin	Lino Esculapio	Zeferino Bittencourt	SF	AS	PL	M.U.	
	Antonio Candido Cavalin	Zeferino Bittencourt	Abilio Carvalho Bastos	SF	AS	PL	M.U.	
	Antonio Candido Cavalin	Abilio Carvalho Bastos	Nereu Ramos	SF	AS	S+	M.U.	
	Antonio Candido Cavalin	Nereu Ramos	Julio Vieira Lisboa	SF	AS	S+	M.U.	
	Antonio Candido Cavalin	Julio Vieira Lisboa	Manoel Alves do Amaral	VE	AS	PL	M.U.	
	Manoel Alves do Amaral	Antonio Candido Cavalin	Dezenove de Dezembro	RR	AS	PL	MD	GG Cavalin Bora 7min. (7:52)
	Manoel Alves do Amaral	Dezenove de Dezembro	Antonio Candido Cavalin	VE	AS	PL	M.U.	
	Antonio Candido Cavalin	Manoel Alves do Amaral	Dr. José Augusto da Silva	SF	AS	PL	M.U.	
	Antonio Candido Cavalin	Dr. José Augusto da Silva	Francisco Stroparo	SF	AS	S+	M.U.	
	Antonio Candido Cavalin	Francisco Stroparo	Traiano Grácia	VD	AS	PL	M.U.	
	Traiano Grácia	Antonio Candido Cavalin	Des. Joaquim F. Guimarães	SF	AS	D-	MD	
	Traiano Grácia	Des. Joaquim F. Guimarães	Daniel Moreira	RR	AS	D-	MD	Coleta Posto Nadir
	Traiano Grácia	Daniel Moreira	Des. Joaquim F. Guimarães	SF	AS	S-	MD	
	Traiano Grácia	Des. Joaquim F. Guimarães	Francisco Stroparo	SF	AS	S-	MD	SC
	Traiano Grácia	Francisco Stroparo	José Augusto da Silva	SF	AS	PL	M.U.	SC
	Traiano Grácia	José Augusto da Silva	Manoel Alves do Amaral	SF	AS	PL	M.U.	SC
	Traiano Grácia	Manoel Alves do Amaral	Julio Vieira Lisboa	VE	AS	PL	M.U.	SC
	Julio Vieira Lisboa	Traiano Grácia	Antonio Candido Cavalin	SF	AS	D-	MD	SC
	Julio Vieira Lisboa	Antonio Candido Cavalin	Professor Vitor do Amaral	SF	AS	D-	MD	SC
	Julio Vieira Lisboa	Professor Vitor do Amaral	Nossa Senhora de Fátima	VD	AS	D-	MD	SC
	Nossa Senhora de Fátima	Julio Vieira Lisboa	Nereu Ramos	VE	AS	PL	MD	SC
	Nereu Ramos	Nossa Senhora de Fátima	Benedito de Moraes	SF	AS	D+	MD	
	Nereu Ramos	Benedito de Moraes	Pres. Getulio Vargas	VE	AS	D-	MD	

FIGURA 16: ROTA ATUAL 3
FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

Secretaria Municipal de Meio Ambiente - Irati - Paraná

Data: 04/02/2011		Roteiro:									
Caminhão/Placa: ANK 3580		Rota de Sexta motorista Marcelo									
Horário	Rua	Rua A	Rua B	Conver-	Reves-	Decli-	Tipo de	Observações			
				sões	tamento	vidade	Tráfego				
								CENTRO			
7:12	Maria Bittar do Prado (Pátio)	Pátio	João Candido Ferreira	VE	AS	S+	MD				
:	João Candido Ferreira	Maria Bittar do Prado (Pátio)	Paulo Xisto	SF	AS	PL	MD				
:	João Candido Ferreira	Paulo Xisto	Pacifico Borges	VE	PR	PL	MD				
:	Pacifico Borges	João Candido Ferreira	Paulo Xisto	VD	AS	PL	MD				
:	Paulo Xisto	Pacifico Borges	Moisés de Oliveira	VD	PR	PL	MD				
:	Moisés de Oliveira	Paulo Xisto	Lino Esculápio	VE	AS	D-	MD				
:	Lino Esculápio	Moisés de Oliveira	Caetano Zarpelon	VE	AS	PL	MD				
:	Caetano Zarpelon	Lino Esculápio	João Wasilevski	VE	RP	S-	MD				
:	João Wasilevski	Caetano Zarpelon	Moisés de Oliveira	VE	AS	PL	MD				
:	Moisés de Oliveira	João Wasilevski	Lino Esculápio	SF	AS	D-	MD	SC			
:	Moisés de Oliveira	Lino Esculápio	João Candido Ferreira	VE	AS	D-	MD				
:	João Candido Ferreira	Moisés de Oliveira	Caetano Zarpelon	VE	PR	PL	MD				
:	Caetano Zarpelon	João Candido Ferreira	Lino Esculápio	RR	AS	PL	MD				
:	Caetano Zarpelon	Lino Esculápio	João Candido Ferreira	SF	AS	PL	MD				
:	Caetano Zarpelon	João Candido Ferreira	Exp. José de Lima	VE	AS	PL	MD				
:	Exp. José de Lima	Caetano Zarpelon	Basilio Szpak	VD	AS	PL	MD				
:	Basilio Szpak	Exp. José de Lima	Sem Nome	VE	AS	PL	MD				
:	Sem Nome	Basilio Szpak	Arcelio Batista Teixeira	VE	AS	PL	MD				
:	Arcelio Batista Teixeira	Sem Nome	Exp. José de Lima	VD	AS	PL	MD				
:	Exp. José de Lima	Arcelio Batista Teixeira	Coleta na Policia Militar	RR	AS	PL	MD				
:	Exp. José de Lima	Coleta na Policia Militar	Arcelio Batista Teixeira	VE	AS	PL	MD	SC			
:	Arcelio Batista Teixeira	Exp. José de Lima	Sem Nome	SF	AS	PL	MD	SC			
:	Arcelio Batista Teixeira	Sem Nome	Antonio Valença	SF	AS	PL	MD	GG Condomínio			
:	Arcelio Batista Teixeira	Antonio Valença	Dr. Vicente Machado	RR	PR	PL	MD				
:	Arcelio Batista Teixeira	Dr. Vicente Machado	Antonio Valença	VE	PR	PL	MD	SC			
:	Antonio Valença	Arcelio Batista Teixeira	Basilio Szpak	SF	AS	PL	MD				
:	Antonio Valença	Basilio Szpak	Casemiro Sinderski	VD	AS	PL	MD				
:	Casemiro Sinderski	Antonio Valença	Exp. José de Lima	VE	AS	PL	MD				
:	Exp. José de Lima	Casemiro Sinderski	Moisés de Oliveira	VD	AS	PL	MD				
:	Moisés de Oliveira	Exp. José de Lima	João Candido Ferreira	VE	AS	PL	MD				
:	João Candido Ferreira	Moisés de Oliveira	Pacifico Borges	VE	PR	PL	MD				
:	Pacifico Borges	João Candido Ferreira	Exp. José de Lima	VE	PR	PL	MD				
:	Exp. José de Lima	Pacifico Borges	Moisés de Oliveira	VD	PR	D-	MD				
08:03	Moisés de Oliveira	Exp. José de Lima	AV. Dr. Vicente Machado	VD	AS	PL	MD				
:	AV. Dr. Vicente Machado	Moisés de Oliveira	Pacifico Borges	VD	AS	PL	MD				
:	Pacifico Borges	AV. Dr. Vicente Machado	Exp. José de Lima	VE	PR	PL	MD				
:	Exp. José de Lima	Pacifico Borges	Paulo Xisto	VE	PR	PL	MD				
:	Paulo Xisto	Exp. José de Lima	TV. Paulo Xisto	SF	AS	D-	MD	Na travessa os Catadores pegam a pé			
:	Paulo Xisto	TV. Paulo Xisto	AV. Dr. Vicente Machado	VE	AS	D-	MD				
:	AV. Dr. Vicente Machado	Paulo Xisto	Pacifico Borges	VE	AS	S-	MD	QM			
:	Pacifico Borges	AV. Dr. Vicente Machado	Exp. José de Lima	VE	PR	PL	MD	SC			
:	Exp. José de Lima	Pacifico Borges	Paulo Xisto	VD	PR	PL	MD	SC			
:	Paulo Xisto	Exp. José de Lima	João Candido Ferreira	VE	AS	S-	MD				
:	João Candido Ferreira	Paulo Xisto	Maria Bittar do Prado (Pátio)	SF	AS	D-	MD	SC			
:	João Candido Ferreira	Maria Bittar do Prado (Pátio)	TV. João Candido Ferreira	VD	AS	D-	MD				
:	TV. João Candido Ferreira	João Candido Ferreira	Sem Saida	RR	RP	PL	MD				
:	TV. João Candido Ferreira	Sem Saida	João Candido Ferreira	VD	RP	PL	MD				
:	João Candido Ferreira	TV. João Candido Ferreira	Dr. Armando Van Der Laars	SF	AS	S-	MD				
:	João Candido Ferreira	Dr. Armando Van Der Laars	N. Senhora de Fátima	SF	AS	S+	MD				
:	João Candido Ferreira	N. Senhora de Fátima	Prof. Vitor do Amaral	SF	PR	S+	MD				
:	João Candido Ferreira	Prof. Vitor do Amaral	Antonio Candido Cavalim	VD	PR	PL	MD				
:	Antonio Candido Cavalim	João Candido Ferreira	Lino Esculápio	SF	AS	D-	M.U				
:	Antonio Candido Cavalim	Lino Esculápio	Zeferino Bittencourt	VD	AS	D-	M.U				
:	Zeferino Bittencourt	Antonio Candido Cavalim	Prof. Vitor do Amaral	SF	AS	D+	MD				
:	Zeferino Bittencourt	Prof. Vitor do Amaral	N. Senhora de Fátima	VE	AS	S+	MD				
:	N. Senhora de Fátima	Zeferino Bittencourt	Abilio Carvalho Bastos	RR	AS	S+	MD	SP Hospital de Irati			
08:30	N. Senhora de Fátima	Abilio Carvalho Bastos	Zeferino Bittencourt	VD	AS	D+	MD	SC			
:	Zeferino Bittencourt	N. Senhora de Fátima	Prof. Vitor do Amaral	SF	AS	D-	MD	SC			
:	Zeferino Bittencourt	Prof. Vitor do Amaral	Antonio Candido Cavalim	VD	AS	PL	MD	SC			
:	Antonio Candido Cavalim	Zeferino Bittencourt	Abilio Carvalho Bastos	SF	AS	PL	M.U				
:	Antonio Candido Cavalim	Abilio Carvalho Bastos	Nereu Ramos	VE	AS	S-	M.U				
:	Nereu Ramos	Antonio Candido Cavalim	Dezenove de Dezembro	RR	AS	PL	MD	GG Italiano 5 min			
:	Nereu Ramos	Dezenove de Dezembro	Antonio Candido Cavalim	VE	AS	PL	MD	SC			
:	Antonio Candido Cavalim	Nereu Ramos	Julio Vieira Lisboa	SF	AS	S+	M.U				
:	Antonio Candido Cavalim	Julio Vieira Lisboa	Manoel Alves do Amaral	VE	AS	PL	M.U				
:	Manoel Alves do Amaral	Antonio Candido Cavalim	Coleta no CAVALIM BORA	RR	AS	PL	MD	GG CAVALIM BORA 10 Min.			
:	Manoel Alves do Amaral	Coleta no CAVALIM BORA	Antonio Candido Cavalim	VE	AS	PL	MD	SC			
:	Antonio Candido Cavalim	Manoel Alves do Amaral	Dr. José Augusto da Silva	SF	AS	PL	M.U				
:	Antonio Candido Cavalim	Dr. José Augusto da Silva	Francisco Stroparo	SF	AS	S+	M.U				
:	Antonio Candido Cavalim	Francisco Stroparo	Trajano Grácia	VD	AS	PL	M.U				
:	Trajano Grácia	Antonio Candido Cavalim	Des. Joaquim Ferreira Guimarães	SF	AS	D-	MD				
:	Trajano Grácia	Des. Joaquim Ferreira Guimarães	Daniel Moreira	SF	AS	D-	MD				
:	Trajano Grácia	Daniel Moreira	Olavo Bilak	SF	AS	PL	MD				

FIGURA 17: ROTA ATUAL 4

FONTE: PREFEITURA MUNICIPAL DE IRATI (2011)

APÊNDICE 1 – PARTE DA MATRIZ DE ADJACÊNCIAS (e_{ij})

e(i,j)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	0	0	0	0	0
14	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

FIGURA 18: MATRIZ DE ADJACÊNCIAS
FONTE: O autor (2011)

APÊNDICE 2 – CUSTOS DOS ARCOS REPRESENTADOS POR NÓ_INICIAL,
NÓ_FINAL, CUSTO – MATRIZ (a_{ij})

Arco, Custo	Arco, Custo	Arco, Custo	Arco, Custo	Arco, Custo	Arco, Custo	Arco, Custo	Arco, Custo	Arco, Custo
1,2,40.5528	16,14,144.1783	30,33,64.3282	48,47,74.9788	69,70,82.4561	89,88,95.2033	107,117,91.3415	124,127,228.4035	140,141,86.3909
1,5,105.3991	16,15,33.0805	30,41,163.4214	48,44,110.0220	70,71,72.9895	89,83,112.6946	108,107,74.6469	124,125,66.9715	140,144,29.2243
2,1,40.5528	16,19,69.2395	31,42,70.6558	48,49,80.4639	70,75,79.5907	89,92,100.2704	108,109,108.9647	125,124,66.9715	141,142,44.2436
2,3,76.7837	16,17,50.7297	31,29,74.9503	49,48,80.4639	71,72,124.2220	90,88,51.3077	108,116,69.3251	125,126,119.5537	141,137,61.6197
2,5,138.9777	17,28,75.6312	31,32,117.9163	49,50,85.8532	71,66,59.0994	90,148,118.6494	109,115,68.1864	126,125,119.5537	141,140,86.3909
3,2,76.7837	17,13,197.0517	32,31,117.9163	49,54,100.8483	72,156,87.2865	91,90,51.0063	109,108,108.9647	126,128,127.7992	142,141,44.2436
3,4,182.4105	17,16,50.7297	32,41,66.2390	49,151,131.8203	72,79,131.2145	91,92,94.9183	109,110,194.1261	126,129,108.3776	142,143,73.6732
3,30,96.0191	17,20,110.9107	32,33,160.6740	49,152,56.8639	73,146,123.5321	92,89,100.2704	109,97,60.5368	127,114,66.2055	142,78,117.8654
3,29,250.5930	18,15,67.2366	32,44,71.8526	50,49,85.8532	73,87,46.0934	92,91,94.9183	110,114,49.3497	127,124,228.4035	143,142,73.6732
4,5,47.7238	18,19,38.4595	33,32,160.6740	50,51,53.0979	73,62,51.1790	92,93,73.6997	110,109,194.1261	127,128,41.8303	143,113,118.7116
4,3,182.4105	18,21,136.4319	33,30,64.3282	50,158,36.3038	74,75,81.4004	93,92,73.6997	110,111,24.1047	128,135,55.4404	143,136,103.8765
4,30,120.7644	19,18,38.4595	33,34,229.6280	51,50,53.0979	74,71,67.5042	93,86,98.6752	111,110,24.1047	128,127,41.8303	144,145,54.2024
4,35,431.4246	19,16,69.2395	33,53,76.9375	52,158,42.0656	75,74,81.4004	93,94,69.8031	111,96,32.7237	128,126,127.7992	144,140,29.2243
5,4,47.7238	19,20,33.0829	34,33,229.6280	53,158,86.6517	75,77,82.8713	94,93,69.8031	112,113,58.1451	128,129,55.3657	145,132,150.4636
5,6,89.4395	19,24,131.9378	34,39,257.5036	53,33,76.9375	76,28,95.2390	94,85,105.9628	112,111,43.1429	129,130,84.1717	145,144,54.2024
5,2,138.9777	20,17,110.9107	34,40,68.9575	53,54,81.3513	76,69,186.7039	95,28,50.2834	113,112,58.1451	129,126,108.3776	146,147,49.4301
5,1,105.3991	20,19,33.0829	35,4,431.4246	54,53,81.3513	76,77,113.0893	95,96,47.1593	113,114,121.3858	129,128,55.3657	146,73,123.5321
5,7,427.2748	20,27,33.1062	35,8,55.2299	54,49,100.8483	77,76,113.0893	95,112,32.6267	113,143,118.7116	129,134,202.4249	146,151,121.7298
6,5,89.4395	20,25,133.5262	35,36,81.6174	54,56,87.7936	77,78,85.2583	96,95,47.1593	113,76,104.0170	130,131,77.4259	147,148,51.2355
7,9,430.3686	21,18,136.4319	36,39,86.7167	54,55,50.0602	77,143,173.4949	96,97,200.7435	114,113,121.3858	130,129,84.1717	147,150,122.2807
7,8,47.4397	21,24,31.5475	36,35,81.6174	55,54,50.0602	78,77,85.2583	97,96,200.7435	114,110,49.3497	131,130,77.4259	148,90,118.6494
7,5,427.2748	21,22,179.9214	36,67,111.8400	56,54,87.7936	78,74,80.1688	97,98,51.1191	114,127,66.2055	131,133,81.0453	148,149,122.0155
8,7,47.4397	22,21,179.9214	37,38,71.1624	56,57,29.4478	78,79,122.6183	97,109,60.5368	114,115,209.3899	131,132,124.5417	149,148,122.0155
8,35,55.2299	22,23,90.0113	38,36,66.2522	57,56,29.4478	79,144,173.4949	98,97,51.1191	115,124,65.2946	132,145,150.4636	149,150,52.3099
8,11,265.2699	23,22,90.0113	38,37,71.1624	57,40,78.5278	79,78,122.6183	98,99,122.4748	115,116,98.9028	132,133,55.1325	149,153,118.9264
9,7,430.3686	23,24,240.2590	39,36,86.7167	57,61,177.8466	79,80,76.2409	98,26,39.7832	115,109,68.1864	132,131,124.5417	150,149,52.3099
9,14,80.4853	23,99,165.5435	39,40,200.3544	58,65,144.0158	80,79,76.2409	98,100,102.7404	115,114,209.3899	133,132,55.1325	150,147,122.2807
9,10,74.4077	24,19,131.9378	39,34,257.5036	58,64,58.6338	80,81,54.2390	99,98,122.4748	116,117,114.5850	133,131,81.0453	150,151,48.8674
10,9,74.4077	24,21,31.5475	40,39,200.3544	58,61,35.0406	80,82,32.5531	100,98,102.7404	116,108,69.3251	133,139,68.0328	150,152,120.0681
10,11,92.1457	24,23,240.2590	40,57,78.5278	59,60,63.7599	81,80,54.2390	100,107,60.0607	116,115,98.9028	133,134,178.9312	151,150,48.8674
10,13,63.7713	24,25,32.4363	40,59,95.1756	59,58,51.6865	82,156,100.5083	100,101,98.3369	117,116,114.5850	134,133,178.9312	151,49,131.8203
11,8,265.2699	25,24,32.4363	41,30,163.4214	60,59,63.7599	82,80,32.5531	101,100,98.3369	117,107,91.3415	134,129,202.4249	151,146,121.7298
11,10,92.1457	25,20,133.5262	41,32,66.2390	61,57,177.8466	82,87,147.7876	101,106,45.3100	117,118,123.2196	134,138,117.9312	152,49,56.8639
11,12,61.0402	25,26,32.6605	41,29,117.5199	61,58,35.0406	82,83,59.4086	101,102,117.8620	118,117,123.2196	134,135,105.8965	152,150,120.0681
11,68,74.2720	26,25,32.6605	42,31,70.6558	61,63,52.4173	83,84,46.3120	102,101,117.8620	118,119,142.4674	135,128,55.4404	152,154,131.4986
12,11,61.0402	26,27,131.4834	42,43,45.7384	61,62,38.3739	83,82,59.4086	102,103,136.3928	118,120,37.8732	135,134,105.8965	152,153,50.6444
12,13,105.2369	26,98,39.7832	42,45,84.2839	62,63,40.8588	83,85,42.8905	103,102,136.3928	119,118,142.4674	135,136,104.8741	153,149,118.9264
12,69,57.1534	27,26,131.4834	43,42,45.7384	63,61,52.4173	83,89,112.6946	103,104,56.5544	119,104,57.5046	136,135,104.8741	153,152,50.6444
12,28,221.9254	27,20,33.1062	43,157,22.6527	63,64,38.0815	84,83,46.3120	104,103,56.5544	119,120,161.0747	136,137,72.0604	153,155,114.7375
13,10,63.7713	27,28,194.2454	43,44,75.4175	64,66,136.2934	85,83,42.8905	104,105,193.4312	120,121,25.2366	136,143,103.8765	154,47,61.7271
13,14,94.8803	28,17,75.6312	44,43,75.4175	64,72,66.3970	85,86,55.2392	104,119,57.5046	120,123,66.5408	137,138,30.0129	154,152,131.4986
13,12,105.2369	28,95,50.2834	44,32,71.8526	65,39,85.2806	85,94,105.9628	105,106,44.7732	120,118,37.8732	137,136,72.0604	154,155,53.2739
13,17,197.0517	28,27,194.2454	44,48,110.0220	65,58,144.0158	86,85,55.2392	105,104,193.4312	120,119,161.0747	137,141,61.6197	155,154,53.2739
14,9,80.4853	28,76,95.2390	45,42,84.2839	66,65,52.4660	86,93,98.6752	105,121,34.9825	121,105,34.9825	138,139,84.0328	155,153,114.7375
14,13,94.8803	28,12,221.9254	45,46,48.0033	66,67,72.8827	87,73,46.0934	106,101,45.3100	121,122,63.6831	138,134,117.9312	156,73,55.9397
14,15,115.5841	29,3,250.5930	45,47,53.2659	67,68,70.5208	87,82,147.7876	106,107,81.2174	121,120,25.2366	138,137,30.0129	156,82,100.5083
15,14,115.5841	29,41,117.5199	46,45,48.0033	67,70,59.4673	87,88,22.6568	106,105,44.7732	122,121,63.6831	139,133,68.0328	157,43,22.6527
15,16,33.0805	29,31,74.9503	47,45,53.2659	68,38,103.5656	88,89,95.2033	107,100,60.0607	122,106,32.1767	139,140,93.7566	158,50,36.3038
15,18,67.2366	30,3,96.0191	47,48,74.9788	68,11,74.2720	88,87,22.6568	107,106,81.2174	123,122,19.2016	139,138,84.0328	158,52,42.0656
	30,4,120.7644	47,154,61.7271	69,68,66.8244	88,147,117.2547	107,108,74.6469	124,115,65.2946	140,139,93.7566	158,53,86.6517

FIGURA 19: CUSTO DOS ARCOS
FONTE: O autor (2011)

APÊNDICE 3 – MODELO UTILIZADO NA LINGUAGEM DE PROGRAMAÇÃO
GAMS

Set NosNós/1*158/;

alias (Nos,i)

alias (Nos,j)

alias (Nos,k)

Tablee(i,j) distância entre jornadas

Obs.: aqui é inserida a matriz com os valores dos dados e(i,j)

TableCustos(i,j) custo dos arcos

Obs.: aqui é inserida a matriz com os valores dos custos e(i,j)

Variables

z soma das distâncias

integervariable a;

Equations

DistanciaTotal minimizar distância

restr1(k)

restr2(i,j)

restr3(i,j)

restr4(i,j);

DistanciaTotal.. z =e= sum((i,j), Custos(i,j)*a(i,j));

restr1(k).. sum(i,e(i,k)*a(i,k))-sum(j,e(k,j)*a(k,j)) =e= 0;

restr2(i,j).. e(i,j)*a(i,j) + e(j,i)*a(j,i) =g= 1\$(e(i,j) = 1);

restr3(i,j).. a(i,j) =g= (1-e(i,j))*(-1);

restr4(i,j).. a(i,j) =l= e(i,j)*100000;

Model CarteiroChines /ALL/;

solveCarteiroChines using MIP minimizing z;

Display a.l;

APÊNDICE 4 – RESULTADOS DO MODELO DE PROGRAMAÇÃO LINEAR
INTEIRA

TABELA 6: RESULTADO NEOS

	1	2	3	4	5	6
1		1.000				
2			1.000		1.000	
3		1.000		1.000		
4					1.000	
5	1.000					1.000
6					1.000	
29			1.000			
30			1.000			
35				1.000		
+	7	8	9	10	11	12
5	1.000					
7		1.000	1.000			
8	1.000				1.000	
9				1.000		
10			1.000			
11				1.000		1.000
28						1.000
35		1.000				
68					1.000	
+	13	14	15	16	17	18
9		1.000				
10	1.000					
12	1.000					
13		1.000			1.000	
14			1.000	1.000		
15				1.000		1.000
16			1.000		1.000	
17				1.000		
21						1.000
28					1.000	

continua

continuação							
	0	19	20	21	22	23	24
	16	1.000					
	17		1.000				
	18	1.000		1.000			
	19		1.000				1.000
	21				1.000		
	22					1.000	
	23						1.000
	24			1.000			
	99					1.000	
+		25	26	27	28	29	30
	3						1.000
	4						1.000
	17				1.000		
	20	1.000		1.000			
	24	1.000					
	25		2.000				
	26			1.000			
	27				2.000		
	31					1.000	
	41					1.000	1.000
	76				1.000		
+		31	32	33	34	35	36
	29	1.000					
	30			1.000			
	32	1.000					
	33		1.000				
	34			1.000			
	36					2.000	
	39						2.000
	40				2.000		
	44		1.000				
	67						1.000

continua

continuação						
	37	38	39	40	41	42
+						
					1.000	
						1.000
					1.000	
			1.000			
		1.000				
		1.000				
	1.000					
			1.000			
						1.000
				2.000		
				1.000		
			1.000			
+	43	44	45	46	47	48
	1.000		1.000			
		1.000				
	1.000					
				1.000	1.000	
			1.000			
		1.000				1.000
						1.000
					1.000	
	1.000					
+	49	50	51	52	53	54
					1.000	
	1.000					
						1.000
	1.000		1.000			
		1.000				
						1.000
					1.000	
						1.000

continua

continuação							
	152	1.000					
	158		1.000		1.000		
+		55	56	57	58	59	60
	54	1.000	1.000				
	56			1.000			
	58					1.000	
	59						1.000
	60					1.000	
	61			1.000	1.000		
	64				1.000		
+		61	62	63	64	65	66
	39					1.000	
	58					1.000	
	62	1.000		1.000			
	63	1.000	1.000				
	64			1.000			
	65						1.000
	66				1.000		
	71						1.000
	72				1.000		
	73		1.000				
+		67	68	69	70	71	72
	12			1.000			
	38		1.000				
	66	1.000					
	67		1.000				
	68			1.000			
	69				1.000		
	70	1.000				1.000	
	71						1.000
	74					1.000	
	75				1.000		
	156						1.000

continua

continuação						
+	73	74	75	76	77	78
	69			1.000		
	74		1.000			
	75	1.000				
	76				1.000	
	77		1.000			1.000
	78	1.000				
	87	1.000				
	113			1.000		
	142					1.000
	143				1.000	
	146	1.000				
	156	1.000				
+	79	80	81	82	83	84
	72	1.000				
	78	1.000				
	79		2.000			
	80		1.000	2.000		
	81	1.000				
	83			1.000		1.000
	84				1.000	
	85				1.000	
	89				1.000	
	144	1.000				
+	85	86	87	88	89	90
	82		2.000			
	83				1.000	
	85	1.000				
	86	1.000				
	87			1.000		
	88				1.000	1.000
	89			1.000		
	90			1.000		
	92				1.000	
	93	1.000				
	94	1.000				
	148					1.000

coninua

continuação						
+	91	92	93	94	95	96
					2.000	
			1.000			
		1.000				
	1.000					
		1.000				
			1.000			
				1.000		
						2.000
					1.000	
+	97	98	99	100	101	102
			1.000			
		1.000				
	1.000					
	1.000		1.000	1.000		
		1.000				
		1.000			2.000	
						1.000
				2.000		
+	103	104	105	106	107	108
				1.000		
	1.000					
		1.000				
			1.000			
				1.000		
					2.000	
						1.000
						1.000
					1.000	
		1.000				
			1.000			
				1.000		

continua

continuação						
+	109	110	111	112	113	114
	95			1.000		
	96		1.000			
	97	2.000				
	108	1.000				
	109		1.000			
	110			1.000		1.000
	111		1.000		1.000	
	112				1.000	
	113					1.000
	143				1.000	
+	115	116	117	118	119	120
	104				1.000	
	108	1.000				
	109	2.000				
	114	1.000				
	115		1.000			
	116			1.000		
	117				1.000	
	118		1.000			1.000
	119					1.000
	120			1.000		
	123					1.000
+	121	122	123	124	125	126
	105	1.000				
	106		1.000			
	115			2.000		
	120	1.000				
	121		1.000			
	122			1.000		
	124				1.000	
	125					1.000
	128					1.000

continua

continuação							
+		127	128	129	130	131	132
	114	1.000					
	124	1.000					
	126		1.000	1.000			
	127		2.000				
	128			1.000			
	129				1.000		
	130					1.000	
	131						1.000
	132					1.000	
	145						1.000
+		133	134	135	136	137	138
	128			1.000			
	129		1.000				
	131	1.000					
	132	1.000					
	133		1.000				
	134			1.000			1.000
	135				2.000		
	136					1.000	
	138					1.000	
	139						1.000
+		139	140	141	142	143	144
	79						1.000
	133	1.000					
	136					1.000	
	137			2.000			
	138	1.000					
	139		1.000				
	140			1.000			1.000
	141		1.000		2.000		
	142					1.000	

continua

continuação							
+		145	146	147	148	149	150
	73		1.000				
	88			1.000			
	144	1.000					
	146			1.000			
	147				1.000		1.000
	148					1.000	
	149				1.000		1.000
	151		1.000				1.000
	153					1.000	
+		151	152	153	154	155	156
	47				1.000		
	49	1.000					
	73						1.000
	82						1.000
	150	1.000	2.000				
	152			1.000	1.000		
	153		1.000				
	154					1.000	
	155			1.000			
+		157	158				
	43	1.000					
	52		1.000				
	53		1.000				

FONTE: O autor (2012)

APÊNDICE 5:PARTE DO PROGRAMA PARA PCCNO DESENVOLVIDO NO MATLAB

```

% Este programa tem por objetivo resolver um problema do carteiro chinês
% não direcionado

clear % limpa toda área de trabalho.

dados % carrega as coordenadas cartesianas de cada um dos 158 pontos do
grafo associado à cidade de Irati

distancias % Esta rotina tem por objetivo, calcular a matriz de distâncias
% W sem as orientações do grafo de Irati, isto é, considerando um
problema não direcionado.
W(1:n,1:n)=zeros;
a=1/0;
for i=1:n
    for j=2:6
        if ad(i,j)~=0
            W(i,ad(i,j))=sqrt((coord(i,2)-coord(ad(i,j),2)).^2+(coord(i,3)-
coord(ad(i,j),3)).^2);
            W(ad(i,j),i)=W(i,ad(i,j)); % para tornar o grafo simétrico
        end
    end
end
for i=1:n
    for j=1:n
        if (W(i,j)==0)&(i~=j)
            W(i,j)=a;
        end
    end
end

floyd % gera as matrizes W das distâncias mínimas entre dois nós quaisquer
% e P das trajetórias).
for i=1:n
    for j=1:n
        P(i,j)=i;
    end
end

for k=1:n
    for i=1:n
        for j=1:n
            if (W(i,k)+W(k,j))<W(i,j)
                P(i,j)=P(k,j);
            end
            W(i,j)=min(W(i,j),W(i,k)+W(k,j));
        end
    end
end

emparel % Este programa tem por objetivo realizar um emparelhamento
% (problema de matching, através de uma heurística de melhorias semelhante
aos procedimentos 2-Opt.
for i=1:n
    if paridade2(i,2)==0
        matrizimp(i,1:n)=inf;
        matrizimp(1:n,i)=inf;
    end
end

```



```

end

% Agora atribuiremos também infinito para os termos da diagonal principal.
for i=1:n
    matrizimp(i,i)=inf;
end

matrizimp2=matrizimp;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Agora iremos gerar um emparelhamento inicial, unindo primeiramente os
% nós ímpares que possuem menor distância entre si.
% Lista dos nós ímpares
m=0;
for i=1:158
    if paridade2(i,2)==1
        m=m+1;
        nosimpares(m)=paridade2(i,1);
    end
end
%nosimpares

% x e y são os nós ímpares que tem a menor distância e este valor é mínimo
for i=1:46

    [ab,x]=min(matrizimp);
    [bb,y]=min(ab);
    x=x(y);
    minimo=matrizimp(x,y);
    pares(i,1)=x;
    pares(i,2)=y;
    pares(i,3)=minimo;
    matrizimp(x,1:n)=inf;
    matrizimp(1:n,x)=inf;
    matrizimp(y,1:n)=inf;
    matrizimp(1:n,y)=inf;

end
%nosimpares
pares
valorinicial=sum(pares(1:46,3))
% Agora que já temos o emparelhamento inicial, vamos começar com as
% melhorias:
mudou=1;
while mudou~=0
    aux10=0;
    for i=1:46
        for j=1:46
            if i~=j

newdelta(j,1)=matrizimp2(pares(i,1),pares(j,2))+matrizimp2(pares(i,2),pares
(j,1));
newdelta(j,2)=matrizimp2(pares(i,1),pares(j,1))+matrizimp2(pares(i,2),pares
(j,2));

                else
                    newdelta(j,1)=inf;
                    newdelta(j,2)=inf;
                end
            end
        % Agora vamos achar a melhor opção de troca
        [ab,x]=min(newdelta);
        [bb,y]=min(ab);

```

```

x=x(y);
menorvalor=newdelta(x,y);
delta=pares(i,3)+pares(x,3);
if (menorvalor<delta)&(y==1)
    aux1=pares(i,2);
    aux2=pares(x,2);
    pares(i,2)=aux2;
    pares(x,2)=aux1;
    pares(i,3)=matrizimp2(pares(i,1),pares(i,2));
    pares(x,3)=matrizimp2(pares(x,1),pares(x,2));
    aux10=aux10+1;
elseif (menorvalor<delta)&(y==2)
    aux1=pares(i,2);
    aux2=pares(x,1);
    pares(i,2)=aux2;
    pares(x,1)=aux1;
    pares(i,3)=matrizimp2(pares(i,1),pares(i,2));
    pares(x,3)=matrizimp2(pares(x,1),pares(x,2));
    aux10=aux10+1;
else
end
novovalor=sum(pares(1:46,3))
clear newdelta
end
if aux10~=0
    mudou=1;
else
    mudou=0;
end
end
pares
novovalor

roteiro % esta rotina calculará o roteiro que passará por todos os arcos
% Primeiramente será construída a matriz MatrizArcos de zeros, uns ou dois.
% indicando quantas vezes o carteiro passará por cada arco.
for i=1:158
    for j=1:158
        MatrizArcos(i,j)=0;
    end
end

MatrizArcos;

for i=1:158
    for j=2:11
        if adinc2(i,j)~=0
            MatrizArcos(i,adinc2(i,j))=1;
        end;
    end
end

% Agora acrescentaremos uns aos arcos duplicados:
for i=1:46
    ii=pares(i,1);
    jj=pares(i,2);
    % Antes encontraremos o vetor t da trajetória de ii até jj:
    clear t
    if W(ii,jj)~=a
        p=1;
    end
end

```

```
t(p)=jj;
while t(p)~=ii
    p=p+1;
    t(p)=P(ii,t(p-1));
end
t=fliplr(t);
end
[t,I]=sort(t);
%t=fliplr(t)
v=max(I);
for kk=1:v-1
    MatrizArcos(t(kk),t(kk+1))=MatrizArcos(t(kk),t(kk+1))+1;
    MatrizArcos(t(kk+1),t(kk))=MatrizArcos(t(kk+1),t(kk))+1;
end
end
MatrizArcos2=MatrizArcos;
```

APÊNDICE 6: CÓDIGO DO PROGRAMA DESENVOLVIDO EM MATLAB PARA CRIAR ROTA

```

Sub CriarRota()
'Carregar Matriz com as adjacências
Max = 1
For i = 1 To NumNos
    If NArcosDisp(i) > Max Then
        Max = NArcosDisp(i)
    End If
Next
ReDim MatrizAdjacências(NumNos, Max)
ReDim MatrizRotas(TotalArcos, TotalArcos + 2)
ReDim NumNosRota(TotalArcos)
Dim PosiçãoRota As Long
For i = 1 To NumNos
    For j = 1 To Max
        MatrizAdjacências(i, j) = ThisWorkbook.Worksheets("Dados").Cells(i + 1, j + 1)
    Next
Next
NumRota = 1
NóAtual = 1
MatrizRotas(NumRota, 1) = NóAtual
PosiçãoRota = 2
Do While TotalArcos > 0
    If NArcosDisp(NóAtual) > 0 Then
        MatrizRotas(NumRota, PosiçãoRota) = MatrizAdjacências(NóAtual, 1)
        PosiçãoRota = PosiçãoRota + 1
        Nó1 = MatrizAdjacências(NóAtual, 1)
        AtualizarDadosAdjacências NóAtual
        NóAtual = Nó1
        TotalArcos = TotalArcos - 1
    ElseIf NóAtual = MatrizRotas(NumRota, 1) Then
        NumNosRota(NumRota) = PosiçãoRota - 1
        ImprimirRota
        NumRota = NumRota + 1
        EncontrarArco
        MatrizRotas(NumRota, 1) = NóAtual
        PosiçãoRota = 2
    End If
Loop
NumNosRota(NumRota) = PosiçãoRota - 1
ImprimirRota
caminho = ThisWorkbook & "/Rota.txt"
Open caminho For Output As #1
Print #1, rotastring
Close #1
EncaixarRotas
MsgBox "Fim!"
End Sub

```