

Dedico este trabalho ao meu querido irmão  
Geraldo Vidal de Souza (in memoriam), de quem  
sentirei saudades eternamente.

## AGRADECIMENTOS

Ao meu marido Ademir Zamboni, e aos meus filhos Gustavo e Caroline, pela sua preciosa colaboração e compreensão por todas as horas de trabalho e estudo em que tive que me ausentar do seu convívio

À professora Maria Teresinha Arns Steiner, pela colaboração, orientação e grande presteza, durante o curso e também pela sua dedicação e amor ao trabalho que conduziram à escolha do tema e elaboração desta dissertação.

À minha colega Deise Maria Bertholdi Costa, com quem dividi várias horas de estudo, seminários e muitos trabalhos em comum.

Aos professores Celso Carnieri, Yuan, Raimundo, Rubens e todos os professores que contribuíram com seus ensinamentos durante todo o curso e pelo seu pronto atendimento quando solicitados.

À Escola Bom Jesus, que gentilmente forneceu todos os dados para a realização deste trabalho.

À Maxi-Data, que contribui com o trabalho, colocando seu mapa à disposição e propondo-se também a esclarecer quaisquer dúvidas sobre o manuseio do trabalho.

À todas as pessoas que direta ou indiretamente colaboraram para a conclusão e êxito neste curso.

# SUMÁRIO

	página
<b>CAPÍTULO I</b>	
1. INTRODUÇÃO.....	1
1.1 Objetivos do Trabalho.....	1
1.2 Importância do Trabalho.....	2
1.3 Limitações do Trabalho.....	3
1.4 Estrutura do Trabalho.....	3
1.5 Os problemas de roteirização.....	4
<b>CAPÍTULO II</b>	
2. DESCRIÇÃO DO PROBLEMA REAL	
2.1 Introdução ao problema.....	6
2.2 Como o problema é resolvido atualmente.....	6
2.3 Descrição Geral do problema.....	8
2.4 Algumas particularidades sobre o transporte escolar.....	10
<b>CAPÍTULO III</b>	
3. ALGUMAS NOÇÕES BÁSICAS SOBRE GRAFOS	
3.1 Algumas definições sobre grafos.....	14
3.1.1 Grafo ou rede.....	14
3.1.2 Caminho.....	15
3.1.3 Caminho Simples.....	16
3.1.4 Caminho Elementar.....	16
3.1.5 Cadeia.....	16
3.1.6 Grafo Conexo.....	16
3.1.7 Custo de um caminho.....	16
3.1.8 Circuito.....	16
3.1.9 Circuito Elementar.....	17
3.1.10 Circuito Hamiltoniano.....	17
3.1.11 Ciclo.....	17

3.1.12 Representação Matricial de um Grafo.....	18
3.1.13 Grau de um vértice.....	18

## CAPÍTULO IV

4. MÉTODOS UTILIZADOS PARA A RESOLUÇÃO DO PROBLEMA REAL	
4.1. Introdução.....	19
4.2. O problema do caixeiro viajante.....	24
4.2.1. Origem do problema do caixeiro viajante.....	24
4.2.2. O problema do caixeiro viajante.....	24
4.2.3. Descrição dos métodos para solucionar o problema do caixeiro viajante.....	26
a) Métodos Exatos.....	26
a.1) Problema de Designação para resolução do pcv.....	26
a.2) Primeira Formulação, via PLI, para solucionar o PCV.....	33
a.3) Segunda Formulação, via PLI, para solucionar o PCV.....	35
b) Métodos heurísticos para solucionar o PCV.....	36
b.1) Procedimentos de Construção de rotas.....	36
b.1.1) Algoritmo Heurístico que utiliza os <i>savings</i> de Clarke and Wright.....	37
b.1.2) Inserção mais econômica.....	38
b.1.3) Algoritmo Heurístico para obtenção das sementes.....	39
c) Procedimentos de melhorias de rotas.....	40
c.1) Melhoria 2-opt e 3-opt.....	40
c.2) Refinamento do k-opt (Or-opt) .....	42
c.3) Troca de nós entre duas rotas.....	44
d) Procedimentos de Composição de Algoritmos.....	45
e) Variações que podem ser efetuadas.....	46
4.3. O problema dos m-caixeiros viajantes.....	48
4.3.1. Transformação do m-caixeiro viajante em um	

único caixeiro viajante.....	50
4.4. Problema Clássico de Roteirização.....	51
4.5. Algoritmo de Floyd.....	54
4.6. Algoritmos Heurísticos para a solução do problema do caixeiro viajante com múltiplos depósitos.....	56
4.6.1 Algoritmo das razões para problemas com para problemas com Múltiplos Depósitos.....	56
4.6.2 Algoritmo dos <i>savings</i> para problemas com múltiplos depósitos.....	57
4.6.3 Algoritmo dos <i>savings</i> para problemas com múltiplos depósitos para problemas grandes.....	60

## CAPÍTULO V

5. IMPLEMENTAÇÃO COMPUTACIONAL.....	62
5.1. Introdução.....	62
5.2. Estudo da situação atual da escola.....	63
5.2.1. Divisão em grupos .....	65
5.2.2. Sequência dos pontos de parada.....	66
5.2.3. Apresentação da situação atual.....	67
5.3. Implementação Computacional.....	68
5.3.1. Cadastramento de dados.....	68
5.3.2. Processo Inicial - Separação dos pontos em dois grupos.....	70
5.3.3. Utilização da Programação Inteira.....	72
5.3.4. Determinação das Sementes.....	76
5.3.5. Determinação dos Clusters ótimos.....	78
5.3.6. Determinação da Sequência ótima em que os pontos serão atendidos.....	81
5.3.7. Ajustes nas respostas.....	84

5.4.Obtenção dos resultados.....	85
5.5.Ajustes finais.....	88
5.6.Resumo dos resultados.....	90
5.7.Análise dos resultados.....	91
5.8.Apresentação da solução.....	92
CAPÍTULO VI	
6. CONCLUSÕES E SUGESTÕES PARA FUTUROS TRABALHOS.....	96
6.1 Conclusões.....	96
6.2 Sugestões para futuros trabalhos.....	100
REFERÊNCIAS BIBLIOGRÁFICAS.....	101
ANEXO I.....	104

## LISTA DE FIGURAS

Figura 3a - Ilustração dos tipos de gráficos.....	15
Figura 3b - Ilustração de um ciclo com seis nós.....	17
Figura 4a - Exemplificação do PCV.....	25
Figura 4b - Exemplificação da Solução do Problema de Atribuição.....	28
Figura 4c - Exemplo de Ramificação do PCV.....	30
Figura 4d - Ilustração da melhoria 2-opt.....	41
Figura 4d1 - Ilustração da melhoria 3-opt.....	42
Figura 4d2 - Ilustração do algoritmo Or-opt.....	44
Figura 4e - Ilustração da troca de dois nós.....	45
Figura 4f - Ilustração do m-pcv.....	48
Figura 4g - Ilustração de 2 caixeiros para duas sub-rotas.....	51
Figura 4h - Ilustração de 5 caixeiros com 2 depósitos.....	59
Figura 5a - Distribuição dos pontos de parada na cidade de Curitiba.....	64
Figura 5b - Divisão em 25 clusters .....	65
Figura 5c - Sequência dos pontos de parada adotada pela escola.....	66
Figura 5d - Rotas realizadas pela escola.....	66a
Figura 5e - Separação dos pontos atendidos pela escola e pela E.T.....	71
Figura 5f - Agrupamentos centrais.....	76
Figura 5g - Localização das sementes.....	77
Figura 5h - Formação dos 9 clusters.....	79
Figura 5i - Roteiro de um ônibus a nível de bairro.....	82
Figura 5j - Roteiro de um ônibus a nível de quadras.....	83
Figura 5l - Formação de algumas rotas com a sequência dos pontos de parada na região central da cidade.....	87
Figura 5m - Formação dos clusters ótimos.....	94
Figura 5n - Formação das rotas pelo processo de otimização.....	95



## LISTA DE QUADROS

Quadro 5.1 - Situação atual dos ônibus da escola.....	67
Quadro 5.2 - Quantidades e capacidades dos ônibus da escola.....	72
Quadro 5.3 - Quantidades e capacidades dos ônibus da escola.....	72
Quadro 5.4 - Resultados obtidos da resolução do PLI.....	74
Quadro 5.5 - Resultados obtidos da resolução do PLI.....	75
Quadro 5.6 - Resultados obtidos através da utilização dos algoritmos.....	86
Quadro 5.7 - Resultados obtidos para os ônibus através da otimização.....	90
Quadro 5.8 - Resultados obtidos para as distâncias das rotas.....	91
Quadro 6.1 -Comparação das distâncias percorridas pela frota.....	97

## RESUMO

O objetivo deste trabalho é apresentar uma solução para o problema de roteirização do Transporte Escolar.

O problema foi estudado para um caso real de uma escola com sede na região metropolitana de Curitiba, porém a roteirização foi feita na cidade, já que a grande maioria dos alunos residem nesta.

Considerou-se neste problema, as distâncias a serem percorridas por  $m$  veículos, a disponibilidade e capacidade destes, e além disso, as demandas de cada um dos  $n$  pontos.

Para a resolução deste problema foram estudados vários algoritmos heurísticos e exatos, e testes foram realizados com alguns destes algoritmos, dos quais foram selecionados aqueles que forneceram os melhores resultados.

Preliminarmente à utilização destes algoritmos, foi utilizado um modelo matemático de Programação Linear Inteira, para a obtenção das quantidades ótimas de veículos e suas respectivas capacidades; em seguida foi utilizado o algoritmo heurístico para obtenção das sementes (depósitos artificiais), procedimentos para obtenção dos clusters ótimos [Bodin et al., 1983] que numa primeira fase é utilizado sem considerar as capacidades dos veículos e na segunda fase considera as capacidades [Clarke and Wright, 1964] e [Bodin et al., 1983] e o algoritmo da Inserção mais Econômica [Bodin et al., 1983], para a construção das rotas.

O problema do Transporte Escolar requer vários veículos partindo de um depósito central, visitando vários pontos de parada para a coleta de alunos e

levando-os até a escola, onde permanecem até o final das aulas devendo então ser entregues às suas residências.

O objetivo principal do trabalho é minimizar a distância total percorrida pela frota inteira, fazendo reduzir assim, o tempo de permanência dos alunos nos ônibus escolares.

Pode-se avaliar a boa performance dos métodos utilizados, através dos resultados que mostraram uma redução de custos de transporte em torno de 20%. Isto sem considerar a economia que a empresa obteve, em realizar o trabalho com um ônibus a menos do que vinha utilizando, acarretando com isto uma redução nos custos com pessoal e também com a manutenção e locação de um veículo. com pessoal e também com a manutenção e locação de um veículo.

## **ABSTRACT**

The objective of this work is to present a solution for the problem of school bus transportation problem.

The problem was studied taking an account a real situation of a school located in a metropolitan region of Curitiba, but the routing was done in the city, once most of the students live in this area.

To solve this problem, many exact and heuristic algorithms were studied, tests were done with some of these algorithms, from wich were selected those that provided the best results.

Previously to the utilization of these algorithms, it was used a mathematical model of Linear and Integer Programing, in order to get the optimal quantities of vehicles and their respective capacities. After that, it was used the heuristic algorithms in order to get the seeds (dummy depots) and then procedures to get the optimal clusters (Bodin et al., 1983) that in an initial phase they were used without considering the vehicles capacities, and in a second phase they considered the capacities (Clarke and Wright, 1964) and (Bodin, 1983). Finally, the algorithm of More Cheapest Insertion (Bodin, 1983) was used to get the routes.

The problem of school bus transportation requires many vehicles leaving from a central depot, visiting many demand points to pick up the students and take them to the school, where they stay until the end of the classes, and afterwards delivere them to their homes.

The main objective of the work is to minimize the total distance covered by the total fleet, in order to reduce the time that the students stay in the school buses.

It is possible to analyze the performance of the methods used, through the results that showed a saving transportation costs in about 20% to do the same work with one less bus than that they had been using, generating a reduction in costs with the staff, as well as with the maintenance and rent of one vehicle.

# CAPÍTULO I

## 1.INTRODUÇÃO

### 1.1. Objetivos do trabalho

O objetivo principal do trabalho de roteirização de veículos aplicadas ao Transporte Escolar, é minimizar o percurso efetuado por todos os ônibus que prestam serviços à escola, fazendo com isto reduzir o tempo de permanência dos estudantes dentro dos ônibus e possibilitando aos pais maior conforto em termos de horários.

O trabalho visa também efetuar a organização de todo o processo do transporte escolar, desde a base. As fases do trabalho são estruturadas da seguinte forma:

- Cadastrar todos os alunos que utilizam o transporte escolar, tendo todas as informações a respeito de cada um, como: endereço, responsável, ônibus que utiliza, horário em que o ônibus apanhará e entregará o estudante em casa, e outros.
- Após a realização do cadastramento, poder-se-á ter uma visão geral do grupo de alunos que deverá ser atendido. Esta visualização poderá ser efetuada no computador em um mapa digitalizado da cidade.

- O próximo passo será efetuar a divisão deste grupo de pontos em grupos menores, os quais serão destinados a algum dos ônibus disponíveis na empresa.
- Em seguida é apresentada a sequência em que os pontos de parada deverão ser atendidos.

## **1.2. Importância do trabalho**

O serviço de transporte escolar vem crescendo a cada dia, já que a população vem aumentando consideravelmente, fazendo com isto crescer a demanda nas escolas. Outro fator que influencia no aumento do número de estudantes, é que as mães cada vez mais estão saindo para trabalhar fora, não podendo cuidar de seus filhos em tempo integral, assim as crianças estão sendo colocadas na escola cada vez mais cedo, e muitas vezes os pais não têm o tempo disponível para levar e buscar seus filhos nas escolas. Por este motivo cresce a cada dia a procura pelos serviços de transporte escolar.

Com um grande número de ônibus e com estudantes espalhados em todas as regiões da cidade, torna-se necessário um trabalho de otimização nos serviços do transporte escolar, visando a melhoria das rotas e principalmente a redução do tempo gasto para percorrer cada rota.

## **1.3. Limitações do trabalho**

Os problemas que podem limitar a implementação do trabalho é a falta de sentidos nas ruas do mapa digitalizado, o que influencia consideravelmente na sequência em que os pontos deverão ser atendidos, pois pode não ser possível efetuar a coleta nos pontos de parada na ordem oferecida pelo programa. Porém esta sequência poderá ser modificada em função do melhor caminho real para se chegar de um ponto a outro.

#### **1.4. Estrutura do trabalho**

O presente trabalho está dividido em seis capítulos, incluindo esta introdução.

No capítulo II é feita a descrição do problema real, onde é definida a forma com que as empresas que efetuam o transporte escolar trabalham, as particularidades e as dificuldades encontradas na realização deste serviço e finalmente a abordagem real do problema.

O capítulo III cita algumas definições necessárias para o perfeito entendimento da linguagem utilizada no decorrer do trabalho.

O capítulo IV tem a finalidade definir o problema como um todo e descrever os algoritmos existentes e os que foram utilizados na resolução do problema do transporte escolar.

No capítulo V mostra-se a aplicação dos algoritmos vistos no capítulo IV para o problema descrito no capítulo II, bem como os resultados obtidos e comparações com o trabalho que vinha sendo realizado pela escola, apresenta-se também as conclusões sobre o trabalho.



Finalmente no capítulo VI são apresentadas as sugestões para futuros trabalhos.

### **1.5. Os problemas de roteirização**

O problema de roteirização de veículos é um dos mais importantes problemas de otimização e possui muitas aplicações. Em algumas empresas o custo de transportes chega a representar 80% do custo total da empresa, um percentual altíssimo, motivo pelo qual as empresas estão cada vez mais interessadas em aumentar sua produtividade, evitando ao máximo a sua ociosidade.

Como exemplo de aplicação podemos citar as empresas de distribuição de bebidas, coleta de lixo, entregas postais, coleta de alunos por ônibus escolares, distribuição de cigarros e de muitos outros produtos.

Já existem muitos estudos sobre o problema de roteirização de veículos, mas implementados na prática são poucos. Existem alguns pacotes prontos, porém não atendem as particularidades de cada problema.

Justamente pelo grande número de particularidades que cada empresa possui é que os problemas de roteirização são muito difíceis de serem resolvidos; cada caso deve ser estudado separadamente.

Atualmente, existem empresas que estão fazendo mapas digitalizados das cidades, facilitando com isto a localização dos pontos e também viabilizando o processo de roteirização informatizada.

No problema de roteirização de veículos, o conjunto de pontos deve ser dividido em áreas ou regiões de atendimento, formando agrupamentos (clusters) de pontos para então decidir quantos veículos serão necessários para atender cada região e posteriormente como serão formadas as rotas.

O objetivo geral dos problemas de roteirização é minimizar a distância total percorrida assim como a frota de veículos utilizados para atender a demanda, minimizando com isto o custo total de transportes, maximizando a produtividade e satisfação dos usuários.

Os veículos possuem restrições de capacidade e tempo, e podem iniciar e terminar suas rotas em um depósito central, caracterizando um problema de múltiplos caixeiros viajantes, ou ainda ter o ponto inicial distinto do ponto final.

Existem muitos algoritmos para a construção de rotas, dos quais alguns são descritos e utilizados na resolução deste problema.

## **CAPÍTULO II**

### **2. DESCRIÇÃO DO PROBLEMA REAL**

#### **2.1. Introdução ao Problema**

Em geral, as escolas que possuem frota própria de ônibus escolar, ou empresas que fazem o serviço, realizam a roteirização de maneira intuitiva. Tomou-se como exemplo uma escola situada na região de Curitiba para coletar dados reais sobre o problema de roteirização no transporte escolar, onde a roteirização é feita manualmente por uma pessoa com bastante experiência neste serviço e que para tanto, conheça muito bem a cidade.

#### **2.2. Como o problema é resolvido atualmente:**

O trabalho todo é realizado manualmente. Existem muitas dificuldades em resolver este problema. A maior delas é localizar os pontos num mapa impresso em tamanho A0. Este processo é lento e difícil, pois os nomes das ruas são pequenos, as quadras curtas, endereços não constantes no mapa, numeração das casas fora de sequência e outros problemas que dificultam muito a localização dos pontos. Em seguida é feita uma divisão da cidade em regiões, levando em consideração aspectos geográficos, como rodovias movimentadas, Br's, e outros, para evitar que as rotas fiquem cruzando ruas de difícil travessia. Após este passo

é feita uma estimativa do número de alunos em cada região para se poder calcular a quantidade de ônibus que será utilizada em cada uma destas regiões. Levando em conta as considerações anteriores, o responsável pelo trabalho começa o serviço de roteirização, ou seja, a determinação da sequência de coleta dos alunos, formando rotas uma a uma até que a lotação de cada ônibus esteja completa.

A rota de cada ônibus é feita intuitivamente, utilizando o conhecimento dos motoristas e de algum acompanhante. Durante mais ou menos uma semana antes do início das atividades, os motoristas saem com os carros para conhecerem o trajeto que vai sendo aperfeiçoado, conforme a experiência, e além disso, podem calcular o tempo de cada rota e assim avisar os pais que o ônibus passará num determinado horário.

Além das dificuldades de se realizar o trabalho manualmente, da demora e do desgaste das pessoas envolvidas, ainda muitos problemas podem ocorrer tais como: cruzamento de rotas, passagem duas ou até três vezes pelo mesmo local, trajetos longos e demorados e outros. Isto acontece porque o trabalho não está otimizado, fazendo com que a empresa ou escola, possam estar desperdiçando tempo e dinheiro.

O objetivo no trabalho de otimização do transporte escolar é minimizar o número de ônibus necessários, realizando o melhor trajeto possível, ou seja, minimizando as distâncias totais percorridas acarretando a redução do tempo de

permanência do aluno dentro do veículo e proporcionar maior satisfação à seus usuários.

### **2.3. Descrição Geral do problema**

Para o desenvolvimento e implementação das técnicas abordadas neste trabalho, foi estudado o caso de uma escola da cidade de Curitiba/Pr.

Atualmente a escola atende 997 alunos distribuídos em 717 pontos de parada, os roteiros são feitos manualmente, por um encarregado deste trabalho há muitos anos.

No período da manhã os ônibus iniciam suas rotas às 6:30hs e apanham as crianças até às 8:00hs. No período da tarde o serviço começa às 11:30hs e vai até às 13:00hs e na hora de entregar os alunos, o serviço começa às 17:00hs e vai até às 18:30.

A escola pesquisada possui ônibus com capacidades variadas: 32, 43, 44, 45, 46, 49, 50, 51, 56, 58 ou 60 lugares. Os ônibus menores são utilizados para as regiões centrais da cidade, onde o fluxo de veículos é grande e existe dificuldade de tráfego.

Ao todo a escola possui 27 ônibus distribuídos da seguinte maneira: 18 são alocados para a região de Curitiba, 1 deles é alocado para rodar na região metropolitana e o outro para os alunos que praticam esportes, como natação ou

ballet, o restante dos ônibus ficam à disposição, como reservas, para qualquer eventualidade. Os outros ônibus necessários para o transporte dos alunos são terceirizados de uma empresa que realiza o Transporte Escolar (E.T.).

Os estudantes permanecem na escola em tempo integral três dias por semana, nos outros dois dias eles têm atividades apenas no período da tarde. Os ônibus que atendem esta sede da escola são de propriedade da escola, e os que atendem os alunos que estudam nas outras sedes são da E.T.

Os ônibus atendem várias regiões da cidade, atualmente distribuídos da seguinte maneira:

Para o bairro Champagnat são utilizados três ônibus, na Água Verde e Batel são utilizados dois, um para Santa Felicidade e mais uma para uma parte de Santa Felicidade e São Braz, dois para a Vila Isabel, um para a Vila Guaiúra, um para Santa Quitéria, um para o Seminário, um para Campo Comprido, um para a região Central, um para o Cristo Rei, um para o Capanema, dois para o Cabral, um para Santa Cândida, um para Boa Vista, um para Mercês e Pilarzinho, um para Boqueirão e Vila Hauer, um para o Jardim das Américas, um para Santa Bárbara e Almirante da Serra, e um para o Jardim Social.

No decorrer do ano ocorrem mudanças nos pontos de parada, tais como, alunos que devem ser apanhados ou devolvidos em locais distintos dos originais, ou que não precisam ser apanhados, e outras situações mais.

## 2.4. Algumas particularidades sobre o transporte escolar

- Os estudantes preenchem fichas onde consta o seu endereço, porém nem sempre ele é apanhado neste local. Muitas vezes o motorista combina com os pais ou responsáveis pelo aluno, um local próximo de sua casa para que ele seja apanhado, tais como uma esquina, um posto de gasolina e outros;
- São vinte e cinco os ônibus que fazem os roteiros, atualmente. Um deles roda apenas na Região Metropolitana da cidade, isto é, coleta alunos na cidade de Campo Largo.
- As rotas de segunda, quarta e sexta feiras são iguais, pois todos os alunos possuem aula no período integral. Nas terças e quintas as crianças só têm aula no período da tarde, mas têm a opção de fazer aula de natação pela manhã, então as rotas destes dois dias são diferentes. As crianças que fazem natação devem ser entregues pelos pais no Colégio Central, onde o ônibus as apanha e as leva para o local das aulas onde elas ficam direto para as aulas do período da tarde. Os motoristas devem saber quais alunos fazem estas aulas extras, pois então podem modificar a sua rota para não precisar passar pelas casas dos estudantes que já foram para a escola mais cedo. Porém, às vezes, os pais avisam que seu filho não vai para a natação, por exemplo, então o motorista deverá incluí-la no roteiro da tarde. Esta flexibilidade dada aos alunos é um fator de complicação para a escola.

- Ocorre também dos responsáveis ligarem para a garagem dizendo que o filho deve ser apanhado junto com algum amigo da mesma rota ou de outra rota (problema do chefe de transportes). Então o motorista pode melhorar a rota para não passar na casa da criança que mudou e ficar atento para apanhá-la em outro endereço com as demais crianças;
- A responsabilidade dos motoristas, chefe de transportes e de outros envolvidos é muito grande. A partir do momento em que o aluno entra no ônibus, durante o percurso, até a entrega na escola ou para os pais ou responsáveis, conforme conste na ficha, o pessoal envolvido é responsável absoluto pelo aluno, devendo portanto lhe proporcionar conforto (ônibus em boas condições), segurança (condições de segurança e entrega para a pessoa indicada na ficha), e satisfação (passar o menor tempo possível dentro do ônibus);
- Há também, casos de alunos que são filhos de pais separados e que a cada mês devem ser apanhados em locais distintos. Existe um edital que contém avisos sobre as crianças que possuem este tipo de problema, sendo que os cuidados devem ser redobrados.
- Todos os motoristas possuem acompanhantes que os ajudam nos trajetos, a conhecerem as crianças, etc. A acompanhante permanece na porta do ônibus para auxiliar a criança no embarque e desembarque. Às vezes no início do ano a acompanhante precisa tocar o interfone ou campainha para avisar que o ônibus chegou;



- É necessário que se conheça muito bem o percurso que cada cada ônibus faz em cada dia da semana, na ida e na volta, pois caso ocorra algum problema é mais fácil mandar outro ônibus para substituição.
- Pela manhã existe um ônibus que fica em Campo Largo para apanhar os motoristas (50% deles moram lá, e vêm para Curitiba para iniciarem suas atividades);
- Na pasta das fichas dos alunos consta a sequência na qual eles devem ser apanhados, mas o motorista é quem decide o trajeto que fará para alcançar cada ponto e pode, inclusive, ir melhorando o trajeto a cada dia se achar necessário e se for possível, motivo pelo qual, as atividades especiais só iniciam um mês após o início das aulas.
- Muitas atualizações são feitas após o início das aulas, e vão sendo adaptadas ao roteiro mais próximo, porém, às vezes, não é possível incluir um novo ponto de parada em uma rota já pronta, por causa da lotação. O ponto é então incluído numa outra rota. O que ocorre é que com todas estas alterações (inclusões, exclusões e mudanças de endereço) ao final do ano letivo, as rotas estão muito ruins, com trajetos longos e muitos cruzamentos.
- Os ônibus possuem porta dos dois lados para que a crianças possam embarcar e desembarcar com segurança. Pode ocorrer que os ônibus tenham que fazer uma volta em U para poder apanhar a criança do lado certo e isto acaba atrasando bastante o percurso;

- Os ônibus são equipados com rádios, todos interligados e ligados à garagem. Qualquer problema que ocorra durante o trajeto deve ser comunicado à garagem, para que o chefe de transportes tome as decisões cabíveis a cada caso;
- Quando o pai ou responsável pela criança não se encontra no local e horário combinados para receber a criança, o motorista a conduz para a garagem, onde serão apanhadas pelos pais ou responsáveis;

Por todas estas particularidades é que o problema de roteirização não é de fácil resolução, cada caso deve ser estudado separadamente para que possam ser atendidas as peculiaridades de cada escola.

## CAPÍTULO III

### 3. ALGUMAS NOÇÕES BÁSICAS SOBRE GRAFOS

[Christofides, 1975]

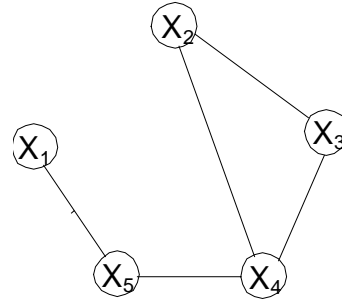
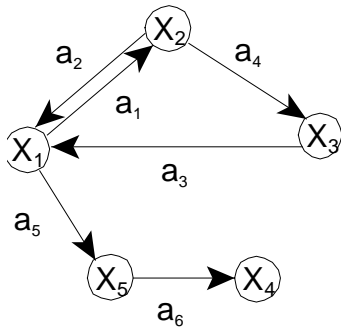
O objetivo deste capítulo é esclarecer alguns termos e definições que serão utilizados no decorrer deste trabalho.

#### 3.1. Algumas Definições

**3.1.1. Grafo ou Rede:** é uma coleção de vértices (pontos, ou nós)  $x_1, x_2, \dots, x_n$  e uma coleção de linhas (arcos ou arestas)  $a_1, a_2, \dots, a_m$  conectando todos ou alguns destes vértices.

O grafo  $G$  é denotado e descrito por  $G(X, A)$ , onde  $X$  é o conjunto de vértices e  $A$  é o conjunto de linhas.

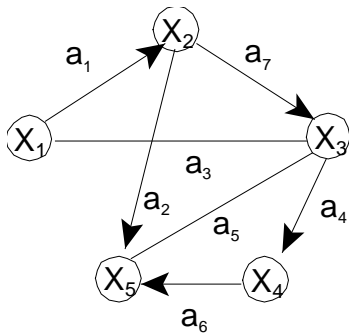
Quanto às linhas, o grafo poderá ser direcionado, isto é, as linhas possuem uma direção; não-direcionado, quando suas linhas não possuem direção ou misto com linhas direcionadas e não direcionadas.



As figuras a seguir ilustram os três tipos de grafos:

3a - Grafo direcionado

3b - Grafo não direcionado



3c - Grafo misto

Figura 3a - Ilustração dos tipos de grafos

### 3.1.2. Caminho (direcionado)

Um caminho em um grafo direcionado é qualquer sequência de arcos onde o vértice final de um é o vértice inicial do seguinte.

### **3.1.3. Caminho Simples**

É um caminho no qual um arco não figura mais que uma vez.

### **3.1.4. Caminho Elementar**

É um caminho no qual um vértice não figura mais que uma vez.

### **3.1.5. Cadeia (não direcionado)**

Uma cadeia em um grafo é qualquer sequência de arestas na qual cada aresta é conectada com suas arestas adjacentes através dos seus dois vértices terminais.

### **3.1.6. Grafo Conexo**

Um grafo é dito conexo se existir entre quaisquer pares de nós do grafo  $G$  pelo menos uma cadeia. Caso contrário o grafo é dito não conexo.

### **3.1.7. Custo (ou peso ou comprimento) de um caminho:**

Se a cada arco  $(x_i, x_j)$  de um grafo for associado um custo  $c_{ij}$ , o custo de um caminho  $\mu$  é igual a soma dos custos dos arcos que formam este caminho.

### **3.1.8. Circuito**

É um caminho  $a_1, a_2, \dots, a_q$ , no qual o vértice inicial de  $a_1$  coincide com o vértice final de  $a_q$ .

### 3.1.9. Circuito Elementar

É um circuito no qual um vértice só aparece uma vez (exceto o inicial e o final que são os mesmos).

### 3.1.10. Circuito Hamiltoniano

É um circuito elementar que contém todos os  $n$  vértices do grafo  $G$ .

### 3.1.11. Ciclo

É uma parte não direcionada de um circuito. Assim, um ciclo é uma sequência de arcos onde o vértice inicial e final são os mesmos.

Um exemplo de um ciclo é apresentado na figura 3b a seguir:

$a_1, a_3, a_4, a_7, a_1$

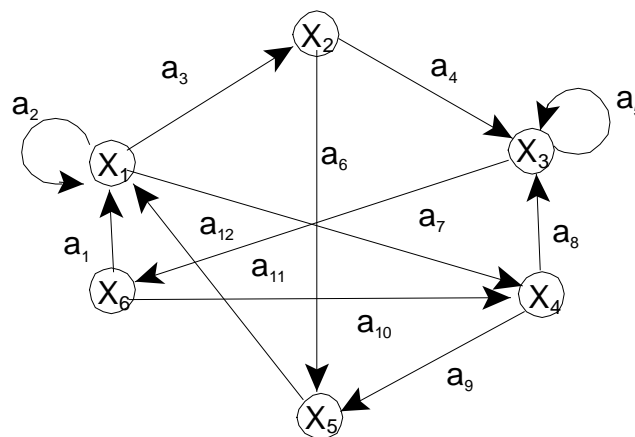


Figura 3b - Ilustração de um ciclo com 6 nós

### 3.1.12. Representação Matricial de um grafo

**a) Matriz de Adjacência:** Para um grafo  $G(X, A)$ , sua matriz de adjacência, denotada por  $A = [a_{ij}]$ , é definida como:

$$a_{ij} = \begin{cases} 1, & \text{se o arco } (x_i, x_j) \exists \text{ em } G; \\ 0, & \text{se o arco } (x_i, x_j) \nexists \text{ em } G. \end{cases}$$

**b) Matriz de Incidência:** Para um grafo  $G(X, A)$ , sua matriz de incidência, denotada por  $B = [b_{ij}]$ , é uma matriz  $n \times m$ , onde  $m$  é o número de arcos, definida como:

$$b_{ij} = \begin{cases} 1, & \text{se } x_i \text{ é o vértice inicial de } a_j \\ -1, & \text{se } x_i \text{ é o vértice final de } a_j \\ 0, & \text{se } x_i \text{ não é um vértice terminal de } a_j, \text{ ou se } a \text{ é um laço} \end{cases}$$

Se  $G(X, A)$  for um grafo não direcionado, a matriz de incidência é definida como acima, exceto trocando os “-1” por “1”.

### 3.1.13. Grau de um vértice

O número de arcos que têm  $x_i$  como vértice inicial é o grau de saída do vértice  $x_i$ , e é denotado por  $d_s(x_i)$  e similarmente o número de arcos que tem  $x_i$  como seu vértice final é chamado o grau de entrada do vértice  $x_i$  e é denotado por  $d_e(x_i)$ . Assim, na figura 3b, o grau de saída exterior do vértice  $x_6$ , por exemplo, é 2, e o grau de entrada do vértice  $x_6$  é 1.

A soma dos graus de entrada e de saída de todos os vértices deve ser igual ao número de arcos,  $m$ , de  $G$ , isto é:

$$\sum_{i=1}^n d_s(x_i) = \sum_{i=1}^n d_e(x_i) = m$$



## CAPÍTULO IV

### 4.MÉTODOS UTILIZADOS PARA A RESOLUÇÃO DO PROBLEMA REAL

#### 4.1. Introdução

O objetivo deste capítulo é mostrar os procedimentos pesquisados para a resolução do problema de roteirização no transporte escolar.

O problema a ser resolvido pode ser formulado da seguinte maneira:

**Problema Geral:** Um certo número de clientes de uma região, situados em locais diferentes, devem ser atendidos por vários veículos que iniciam suas rotas em um depósito comum. Um único veículo não pode ser designado a todos os locais por causa de sua capacidade e horário de trabalho.

**Ojetivo Geral:** O objetivo geral é determinar inicialmente os agrupamentos de pontos que cada veículo atenderá, e posteriormente a sequência em que estes pontos serão atendidos, de forma a minimizar a distância total viajada por todos os veículos, atendendo à todas as demandas, sujeito a restrições de capacidade de cada veículo e distância. A restrição de distância pode ser trocada por uma restrição de tempo.

Existem métodos exatos e heurísticos para a resolução de problemas de roteirização de veículos, porém os métodos exatos se tornam limitados em termos de implementação computacional, ou seja, são aplicáveis apenas, para problemas pequenos com poucas restrições e variáveis.

Os problemas exatos pertencem à uma classe de problemas chamada NP-completos ou NP-hard que são problemas para os quais não existem algoritmos de ordem polinomial que os resolvam. Os problemas pertencentes à classe P são aqueles para os quais existe um algoritmo de ordem polinomial para resolvê-los, ou seja, o número de operações elementares necessárias para solucioná-los cresce de forma polinomial, enquanto que os NP-hard são de difícil tratamento e o número de operações cresce exponencialmente.

Os algoritmos exatos existentes para a solucionar o problema do caixeiro viajante tornam-se inviáveis quando o número de cidades a serem visitadas aumenta, pois enquanto  $n$  cresce, o número de roteiros cresce mais rapidamente. O número de roteiros possível é  $n!$ , se  $c_{ij}$  é diferente de  $c_{ji}$ . Assim, por exemplo, se  $n=4$ , existem 24 roteiros possíveis e, se  $n=8$  o número de roteiros possíveis será 40.320, o que torna inviável a solução exata. No entanto, se a matriz de custos for simétrica então o número de roteiros é  $n!/2$ , ainda assim se  $n$  é grande o número de roteiros que se pode obter é razoavelmente grande para serem resolvidos através de métodos exatos.

Basicamente existem dois procedimentos para resolver os problemas de roteirização, que são os métodos exatos e heurísticos. Os exatos dificilmente são

utilizados na prática, pois “explodem” em termos computacionais, e quanto aos heurísticos, existem muitos, dos quais alguns serão estudados e aplicados na solução do problema abordado neste trabalho.

Na tentativa de encontrar a solução exata pode-se utilizar o algoritmo Branch and Bound ou Programação Inteira Mista [Bodin, 1983] que só são viáveis para resolver problemas relativamente pequenos. Os procedimentos heurísticos são mais utilizados, pois além de fornecerem boas soluções são bem mais rápidos e não possuem restrições quanto ao tamanho do problema.

As soluções obtidas através dos métodos heurísticos são soluções aproximadas, assim sendo, o que se pode fazer para verificar a eficiência do método, é aplicar a técnica a um conjunto de testes e comparar os resultados com a solução ótima, se este resultado for conhecido. Caso contrário, deve-se comparar estes resultados com os obtidos por outras heurísticas para se verificar a performance do método. Outra maneira de se analisar os resultados é escolher o pior desvio possível do ótimo que a heurística pode produzir e comparar os resultados. Esta escolha do pior caso é pessimista, porém jamais violará o problema.

Serão citados a seguir alguns métodos heurísticos que podem ser utilizados na busca da solução para os problemas de roteirização.

a) O algoritmo Sweep [Billy E. Gillett e Leland R. Miller, 1971] ou técnica da varredura, é um procedimento eficiente para construção de rotas. A localização dos pontos de parada, que formarão cada rota, são determinadas de acordo com o ângulo de sua coordenada polar para cada local. Um procedimento iterativo é então utilizado para melhorar a distância total viajada em todas as rotas.

O algoritmo possui a característica de que a quantidade de cálculos aumenta linearmente com o número médio de locais para cada rota.

b) O critério do Vizinho mais Próximo [Bodin et al., 1983], inicia o procedimento por qualquer nó da rota, encontra o mais próximo do último e adiciona-o ao roteiro, repetindo estes passos até que um circuito Hamiltoniano seja formado.

c) Método da Inserção [Rosenkrantz, Sterns and Lewis, 1977], seleciona sub-rotas iniciais e procura um nó a ser inserido nesta rota determinando posteriormente qual é a melhor posição para a inserção deste nó. As primeiras sub-rotas podem ser os próprios nós.

Existem várias maneiras de se fazer a inserção de pontos numa rota, entre elas pode-se citar os seguintes:

c.1) *Inserção mais próxima* [Bodin, 1983], inicializa uma sub-rotas com um nó somente; encontra o nó mais próximo deste último e o insere na rota, repetindo este passo até que todos os nós estejam roteirizados.

O algoritmo da Inserção mais próxima necessita de um número de cálculos da ordem de  $n^2$ .

c.2) *Inserção Arbitrária* [Bodin, 1983], inicia com uma sub-rotas contendo um nó  $i$  somente, encontra o nó  $k$  cujo custo,  $c_{ik}$ , seja mínimo e o insere na rota. Seleciona-se então um nó arbitrário para entrar na rota e o circuito vai sendo formado, o procedimento é repetido até que se tenha um circuito Hamiltoniano.

Existem, vários outros procedimentos de construção de rotas que podem ser ainda citados, tais como: Inserção do mais distante, Inserção mais rápida, Cobertura Convexa, Inserção do maior ângulo, e outros descritos com maiores detalhes em Bodin et al., 1983.

## **4.2. O Problema do Caixeiro Viajante (PCV)**

### **4.2.1. A origem do Caixeiro Viajante** [A. J. Hoffman, P. Wolfe, 1985]

A primeira vez que se utilizou o termo “Caixeiro Viajante” foi no início da década de 1930 nos meios matemáticos. O precursor do termo é desconhecido, porém o principal responsável pela sua divulgação foi Merrill Flood, motivado em propor um desafio intelectual por modelos Matemáticos como o da Teoria dos Jogos. A outra razão da popularização do PCV está ligada aos problemas de designação, mais geralmente com os problemas de transporte e chamou atenção também pelo seu título.

O primeiro artigo que propôs uma “Solução de Larga Escala do Problema do Caixeiro Viajante” foi publicado por Dantzig, Fulkerson e Johnson em 1954 no Jornal da Sociedade de Pesquisa Operacional da América, que foi o principal evento da história da Otimização Combinatória.

#### **4.2.2. Problema do Caixeiro Viajante (PCV)**

O problema do Caixeiro Viajante envolve um veículo que deve partir de um depósito central (nó1) e visitar (n-1) localidades, uma única vez, para em seguida retornar ao depósito central. O problema pode ser simétrico, ou seja, não depende da direção ou sentido das ruas, ou assimétrico que considera o sentido das ruas.

Este problema considera um único depósito e um único veículo que deverá atender várias localidades, passando uma única vez por cada ponto, de modo a minimizar a distância total viajada. A demanda em cada nó é determinística e o veículo tem a capacidade conhecida. A figura 4a mostra um exemplo do problema do caixeiro caixeiro viajante com cinco nós.

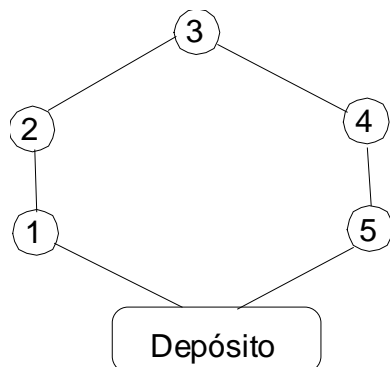


Figura 4a - Exemplificação do PCV

O problema do caixeiro viajante pode ser definido da seguinte forma:

Seja uma rede  $G(N,A,C)$  onde  $N$  = número de nós,  $A$  = número de arcos e  $C = [c_{ij}]$  a matriz dos custos. O Problema do Caixeiro Viajante necessita de um circuito Hamiltoniano em  $G$  de mínimo custo total.

#### **4.2.3. Descrição dos métodos para solucionar o Problema do Caixeiro Viajante**

A seguir serão descritos detalhadamente alguns algoritmos que podem ser utilizados na resolução do problema do caixeiro viajante.

**a) Métodos exatos para a solução do PCV [Bodin et al., 1983]**

As soluções ótimas para o Problema do Caixeiro Viajante estão baseadas na Programação Matemática.

**a.1) Problema de designação para a resolução do PCV**

Os problemas de Designação envolvem uma alocação biunívoca de, por exemplo, operários à tarefas, esta condição pode ser assegurada pela criação de operários ou tarefas fictícias sendo que os custos (tempos),  $c_{ij}$ , para o  $i$ -ésimo operário completar a  $j$ -ésima tarefa são conhecidos.

O objetivo é designar cada um dos operários à uma determinada tarefa, de modo que estas sejam concluídas em um tempo mínimo total.

Os problemas de designação podem ser abordados como problemas de transporte, considerando-se os operários como origens e tarefas como destinos, em que todas as ofertas e demandas são iguais a 1.

Seja um problema de designação definido da seguinte forma:

$$\text{Min } z = \sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij}$$

s.a

$$\sum_{j=1}^m x_{ij} = 1, \quad i = 1, \dots, n$$



$$\sum_{i=1}^m x_{ij} = 1, j = 1, \dots, n$$

$x_{ij} \geq 0$  e inteiros

Onde, para simplificar, considera-se que  $c_{ij} = c_{ji}$ , ou seja, a matriz das distâncias é simétrica e  $c_{ii} = +\infty$ , para  $i = 1, 2, \dots, n$ . Considera-se  $m$  origens com  $a_i = 1, 2, \dots, m$  unidades disponíveis e  $n$  destinos com  $b_j = 1, 2, \dots, n$  unidades necessárias e  $c_{ij}$  é o custo do arco  $i - j$ .

$$x_{ij} = \begin{cases} 1, & \text{se } i - j \text{ está no trajeto} \\ 0, & \text{caso contrário} \end{cases}$$

Os problemas de designação selecionam de uma matriz  $X=(x_{ij})$  de variáveis de decisão aquelas para os quais somente um arco  $(i,j)$  saia do nó  $i$  e somente um arco,  $(i,j)$ , chegue ao nó  $j$ , com a mínima distância. Isto porém, não assegura que a formação de uma rota, mas ao invés disso, a designação pode resultar em sub-rotas.

Um exemplo de sub-rota é mostrado na figura 4b.

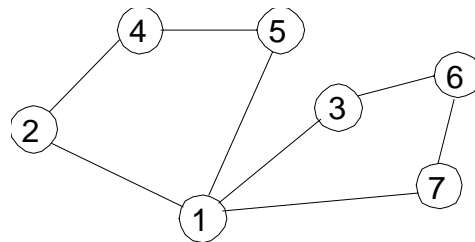


Figura 4b - Exemplificação da solução de problema de Atribuição  
com a formação de sub-rotas: 1-2-4-5-1 e 1-3-6-7-1

Associando-se o problema do caixeiro viajante a um problema de atribuição pode-se obter um limite inferior para o custo da solução do problema.

Considere então, um problema no qual cada nó  $x_j$  deve ser atribuído a um nó  $x_i$ , que será seu único predecessor na solução do problema. Cada atribuição tem um custo  $c_{ij}$  associado dado pela matriz de custos resultante do algoritmo de Floyd, que será descrito posteriormente.

Se um único circuito for formado e se for Hamiltoniano, a solução obtida é ótima para o problema do caixeiro viajante. Porém, nem sempre esta solução é viável, pelo fato de serem formados diversos circuitos disjuntos. Neste caso, o problema de atribuição em questão poderá ser modificado, a fim de inviabilizar a formação de diversos circuitos, acrescentando-se a ele restrições adicionais, como os procedimentos de ramificações sugerido por Christofides, 1975.

Considere-se um caso genérico no qual a solução para o problema de atribuição contém um circuito não Hamiltoniano, ou seja, a solução contém dois ou mais circuitos. Denotaremos este problema por  $P_0$ . Dentre os circuitos formados, seja

$$S\{x_1, x_2, \dots, x_k\}$$

o circuito de menor cardinalidade. O objetivo é eliminá-lo, forçando a formação do circuito Hamiltoniano. Fazendo  $c(x_1, x_j) = \infty$ ,  $c(x_2, x_j) = \infty$ , ...,  $c(x_k, x_j) = \infty \quad \forall x_j \in S$  e

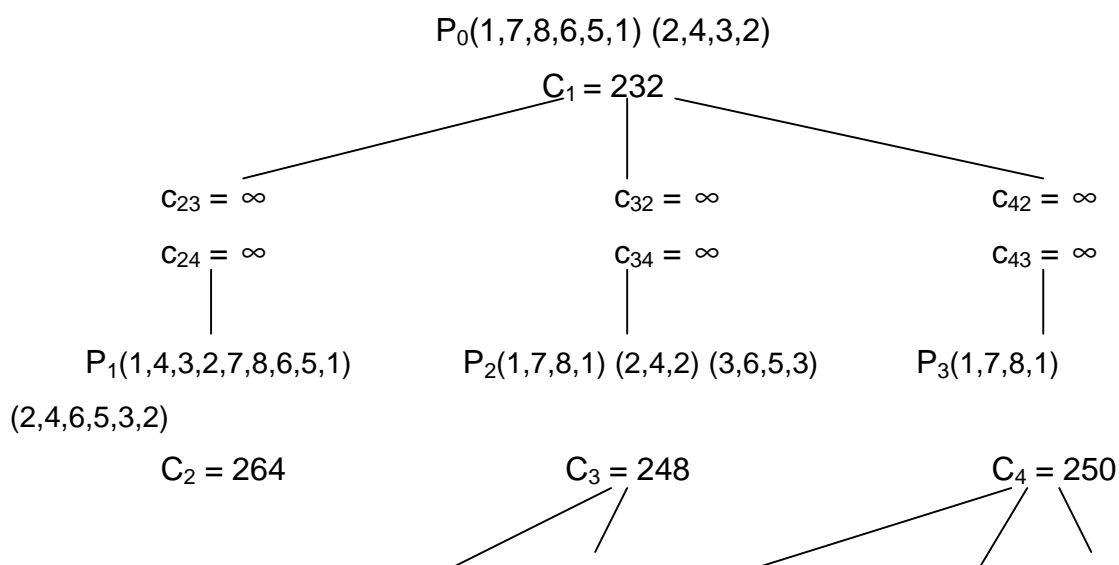
mantendo-se todos os demais custos inalterados, obtém-se a formação de  $k$  sub-problemas,  $P_1, P_2, \dots, P_k$ .

Um exemplo da ramificação de um problema é apresentado na figura a seguir.

Seja a seguinte matriz de custos, de uma rede com 8 nós:

$\infty$	76	43	38	51	42	19	80
42	$\infty$	49	26	78	52	39	87
48	28	$\infty$	36	53	44	68	61
72	31	29	$\infty$	42	49	50	38
30	52	38	47	$\infty$	64	75	82
66	51	83	51	22	$\infty$	37	71
77	62	93	54	69	38	$\infty$	26
42	58	66	76	41	52	83	$\infty$

Considere que  $P_1$  é uma solução inicial para o problema  $P_0$ , com dois circuitos  $(2,4,3,2)$  e  $(1,7,8,6,5,1)$  e com custo inicial  $C_1=232$ :



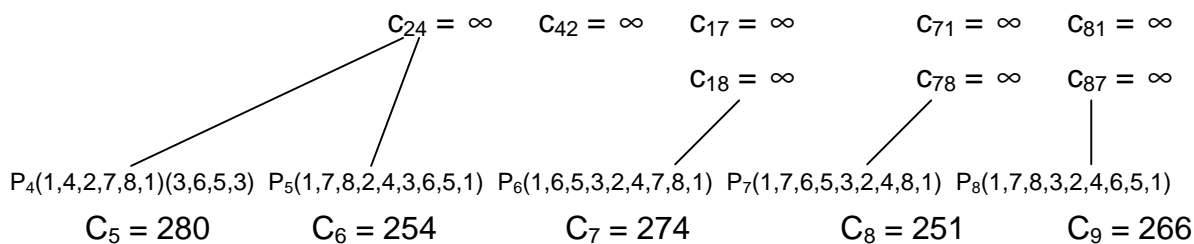


Figura 4i - Exemplo do procedimento de Ramificação na procura da solução ótima para o problema do caixeiro viajante

Toma-se em  $P_0$  o circuito de menor cardinalidade,  $(2,3,4,2)$  o qual deseja-se eliminar. Através do procedimento de ramificação este circuito dá origem a 3 sub-problemas  $P_1$ ,  $P_2$  e  $P_3$ . Os custos dos sub-problemas e as soluções para seus problemas de atribuição estão indicados na figura 4i.

A solução para o sub-problema  $P_2$  é um circuito Hamiltoniano de custo 264. Entretanto, os limites inferiores para os sub-problemas  $P_2$  e  $P_3$  são menores que este valor, sabe-se portanto, que ao continuar a busca, efetuando as ramificações a partir destes sub-problemas, poder-se-á obter uma solução melhor. O menor destes limites é:  $\min \{248, 250\} = 248$  que corresponde ao subproblema  $P_2$ . Continua-se a ramificação a partir deste subproblema eliminando o circuito  $(2,4)$ . E assim prossegue-se até que um circuito Hamiltoniano seja encontrado e que possua custo menor que qualquer outro circuito Hamiltoniano ou não Hamiltoniano.

A solução para o exemplo acima está no subproblema  $P_7$  que apresenta o circuito Hamiltoniano  $(1,7,6,5,3,2,4,8,1)$  de custo 251.

O método Húngaro, descrito a seguir, é muito utilizado para resolver problemas de designação [McLaughlin, 1979].

**ALGORITMO:** Método Húngaro

**Passo 1:** Obter a matriz de custos pelo algoritmo de Floyd;

**Passo 2:** Subtraia de cada linha e ou coluna o seu menor elemento, tal que exista no mínimo um zero em cada linha e em cada coluna; e nenhum valor negativo;

**Passo 3:** Faça o máximo número de atribuições às células zero, da seguinte maneira: (Faça um para atribuir e um X para retirar)

**3.1.** Em cada linha com apenas um zero, faça e X em todos os zeros daquela coluna;

**3.2.** Em cada coluna com apenas um zero, faça e X em todos os zeros restantes daquela coluna e daquela linha;

**3.3.** Repita 3.1 e 3.2. até que todos os zeros da matriz estejam marcados. (com ou X).

**3.4.** Certifique-se de que não existe mais do que um em cada linha e coluna;

**Passo 4:** Se existe um em cada linha e coluna, então estas são atribuições mínimas. Marque as correspondentes células do problema original e obtenha o custo total. Pare. A solução ótima foi obtida.

**Passo 5:** Se não existe um em cada linha e coluna, obtenha mais zeros da seguinte maneira:

**5.1.** Marque todas as linhas com nenhuma atribuição;

**5.2.** Marque todas as colunas com X's nas linhas marcadas;

**5.3.** Marque todas as linhas com      nas colunas marcadas;

**5.4.** Repita 5.2 e 5.3 até a exaustão.

**Passo 6:** Alinhe todas as linhas não marcadas e colunas marcadas;

**Cheque:** O número de linhas deverá ser igual ao número de      ;

Todo zero deverá estar “alinhado”.

**Passo 7:** Nesta matriz proceda da seguinte forma: (para produzir no mínimo mais um zero)

**7.1.** Ache o mínimo valor, não “alinhado”;

**7.2.** Subtraia este valor de todas as linhas marcadas;

**7.3.** Adicione este valor a todas as colunas marcadas.

**Passo 8:** Faça o máximo número de atribuições, como no passo 3 e repita os passos 5, 6 e 7 até que exista um      em cada linha e em cada coluna, como no passo 4.

Obs.: No passo 3, se todas as linhas e colunas com um único zero já terminaram e tem-se somente linhas e colunas com dois ou mais zeros, isto significa que existe atribuições alternativas de minimização para estas linhas e colunas. Quaisquer destes zeros podem ser atribuídos que irá produzir o mínimo custo total. Estas soluções alternativas devem ser sempre registradas.

**a.2) Primeira formulação, via PLI, para solucionar o PCV**

Para evitar que sub-rotas sejam formadas, algumas restrições adicionais são impostas na seleção de arcos na matriz  $X$ , como é mostrado no modelo de Programação Linear Inteira, a seguir:

$$\min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

s.a

$$\sum_{i=1}^n x_{ij} = 1 \quad (j=1, \dots, n)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad (i=1, \dots, n)$$

$$X=(x_{ij}) \in S$$

$$x_{ij} = 0 \text{ ou } 1 \quad (i, j=1, \dots, n)$$

O conjunto  $S$  pode ter quaisquer restrições que proibam as soluções em sub-rotas. Estas restrições são chamadas de quebra de sub-rotas. As possíveis escolhas para  $S$  são:

$$S=\{(x_{ij}) : \sum_{i \in Q} \sum_{j \notin Q} x_{ij} \geq 1 \text{ para todo subconjunto próprio não vazio } Q \text{ de } N\}; \quad (1)$$

$$S=\{(x_{ij}) : \sum_{i \in R} \sum_{j \in R} x_{ij} \leq |R| - 1 \text{ para todo subconjunto não vazio } R \text{ de } \{2, 3, \dots, n\}\}; \quad (2)$$

$S$  contém  $2^n$  restrições de quebra de sub-rotas em (1) e (2), mas somente  $n^2 - 3n + 2$  restrições no conjunto (3).

O primeiro conjunto de restrições afirma que todo subconjunto próprio  $Q$  de nós deve estar conectado aos outros nós da rede na solução. O segundo conjunto implica que os arcos seleccionados em  $X$  não contém ciclo, pois um ciclo sobre  $R$  deve conter  $|R|$  arcos. Isto exclui a possibilidade mostrada na figura (4b).

### a.3) Segunda Formulação, via PLI, para solucionar o PCV

[Held - Karp e Christofides]

Nesta formulação, substitui-se dois conjuntos de restrições dados anteriormente.

$$\min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

s.a

$$\sum_j x_{1j} = 2 \quad (j=1, \dots, n)$$

$$\sum_j x_{ij} + \sum_k x_{ki} = 2 \quad (i=1, \dots, n) \quad (i=1, 2, \dots, n) \quad (\text{estas duas restrições}$$

afirmam que cada nó tem grau 2)



$$\sum_i \sum_j x_{ij} = n \quad (\text{garante que um total de } n \text{ arcos está na solução})$$

$$\sum_{i=1}^n x_{ij} = 1 \quad (j=1, \dots, n)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad (i=1, \dots, n)$$

$$X=(x_{ij}) \in S$$

$$x_{ij}=0 \text{ ou } 1 \quad (i,j=1, \dots, n)$$

Estas duas formulações, a.1 e a.2, porém tornam-se inviáveis quando se trata de resolver problemas grandes, pois o número de variáveis e restrições tornam-se muito grandes, inviabilizando a solução através da Programação Linear Inteira.

#### **b) Métodos Heurísticos para a resolução do PCV:**

Existem três grandes classes para resolver o problema do caixeiro viajante, são elas:

- Procedimentos de Construção de rotas, que constróem rotas ótimas ou quase ótimas considerando a matriz de distâncias (custos);
- Procedimentos de melhorias de rotas, que efetuam melhorias em rotas já existentes;

- Procedimentos de composição de algoritmos, que constroem uma rota inicial com o auxílio de um dos procedimentos de construção de rotas e utilizam a melhoria para obter um resultado mais eficiente, evitando cruzamentos entre rotas e outros problemas.

A seguir serão descritos alguns dos métodos heurísticos utilizados na resolução do Problema do Caixeiro Viajante.

### **b.1) Procedimentos de Construção de Rotas**

Serão descritos detalhadamente, neste item, os procedimentos de construção de rotas utilizados para resolver o problema abordado neste trabalho.

#### **b.1.1) Algoritmo heurístico que utiliza os *savings* de CLARKE and WRIGHT**

[Clarke, Wright, 1963], [Ballou, 1992]

O algoritmo dos *savings* de Clarke and Wright é um procedimento que constrói rotas simultaneamente, e a cada iteração vai efetuando trocas nestas rotas por um conjunto melhor de rotas.

Inicia-se o processo com tantas rotas quanto são o número de nós, ou seja, supõe-se que existem  $n$  veículos para atender as demandas de cada um dos nós individualmente. Verifica-se então, a possibilidade de trocar dois veículos que atendem os nós  $i$  e  $j$  por apenas um.

No próximo passo, as rotas vão sendo construídas de acordo com o maior *savings*  $s(i,j)$ , que é calculado da seguinte forma:

$$s(i,j) = [2c(0,i) + 2c(0,j)] - [c(0,i) + 2c(0,j) + c(i,j)] = c(0,i) + c(0,j) - c(i,j)$$

**ALGORITMO:**

**Passo 1:** Selecione qualquer nó como depósito central (nó 0), inicialize as n rotas, ligando cada cidade, exclusivamente ao depósito obtendo-se as rotas (0 - i - 0), para  $i = 1, \dots, n$

**Passo 2:** Calcule os *savings* entre os pontos de demanda  $P_i$  e  $P_j$ , da seguinte forma:

$$s(i,j) = c(0,i) + c(0,j) - c(i,j) \quad \forall i, j$$

**Passo 3:** Ordene os savings do maior para o menor;

**Passo 4:** Começando do topo da lista dos *savings* e movendo para baixo, forme sub-rotas considerando que:

- $P_i$  e  $P_j$  sejam nós "de ponta" de suas respectivas sub-rotas.

Por exemplo, dadas as sub-rotas (0 - 3 - 5 - 4 - 2 - 0) e (0 - 6 - 1 - 0), as pontas são as cidades 3, 2, 6 e 1. Os savings  $s(i,j)$  para  $i = 5, 4$  não precisam ser considerados.

- $P_i$  e  $P_j$  não devem estar alocados à mesma sub-rotas, ou seja, se  $i$  pertence à mesma sub-rotas de  $j$ , os savings de  $i$  para  $j$  não precisam ser considerados.
- $d(R_i) + d(R_j) < K_v$ , ou seja, que a demanda do veículo da sub-rotas  $i$  unida à demanda do veículo da sub-rotas  $j$  não deve exceder à capacidade de cada veículo;

**Passo 5:** Se as três condições do passo anterior forem satisfeitas, unir as sub-rotas dos pontos  $P_i$  e  $P_j$ ;

**Passo 6:** Se não existir  $s(i,j)$  que satisfaça as condições do passo 4, pare.  
Caso contrário volte ao passo 4.

### **b.1.2) Inserção mais Econômica**

**Passo 1:** Comece com uma sub-rota consistindo de um nó somente;

**Passo 2:** Encontre o nó  $k$  tal que  $c_{ik}$  é mínimo e forme a sub-rota  $i-k-i$ ;

**Passo 3:** (Passo da Inserção). Encontre o arco  $(i,j)$  na sub-rota que minimize  $c_{ik} + c_{kj} - c_{ij}$ , então insira  $k$  entre  $i$  e  $j$ .

**Passo 4:** Vá para o passo 3 até que se tenha um circuito Hamiltoniano.

### **b.1.3) Algoritmo heurístico para obtenção das sementes (depósitos artificiais)** *[Paraíba et al., 1990]*

Este algoritmo seleciona as melhores posições para inserir as sementes (depósitos artificiais), para as quais serão agrupados pontos com designação temporária, para posteriormente serem alocados definitivamente à cada veículo.

#### **ALGORITMO:**

A escolha das  $m$  cidades sementes se dá da seguinte forma:

**Passo 1:** Escolher as  $m$  cidades-sementes que inicializarão as  $m$ -rotas  $R_i$ :

**Passo 2:** Faça  $k = 1$ . Encontrar  $i_k \in U$  tal que  $c(0, i_k) = \max\{c(0, i_k)\}$ , para todo  $i$ , onde  $0$  é o depósito central, ou seja, encontre o ponto de demanda mais distante do depósito central, este ponto será a primeira semente;

**Passo 3:** Encontrar as demais sementes, faça  $k = k + 1$ . Encontre  $i^*$ , ainda não roteirizado, tal que:

$$c(i, i_{k-1}) + c(i, i_{k-2}) + \dots + c(i, i_2) + c(i, i_1), \text{ para } k = 2, \dots, m, \text{ seja máximo.}$$

Esta seleção evita que as  $m$  rotas sejam inicializadas por sementes próximas entre si, ou seja, evita que as rotas sejam inicialmente concorrentes em um mesmo agrupamento de pontos.

### c) Procedimentos de melhorias de rotas

Os procedimentos de melhorias de rotas têm como objetivo melhorar ainda mais a distância da rota, efetuando trocas entre os nós da mesma rota e entre rotas, evitando cruzamentos dentro da rota ou entre rotas, respectivamente. Alguns destes procedimentos, utilizados para a solução do problema, são citados a seguir:

#### c.1) Melhoria 2-Opt e 3-Opt

Os procedimentos para melhoria de rotas mais utilizados são o 2-opt e 3-opt que foram introduzidos por Lin em 1965 e o procedimento  $k$ -opt, para  $k \geq 3$ , foi apresentado por Lin e Kernighan mais recentemente em 1973.



**ALGORITMO:**

**Passo1:** Encontre uma solução inicial por qualquer um dos métodos de construção de rotas, descritos neste capítulo;

**Passo2:** Faça a melhoria da rota utilizando uma das heurísticas de trocas;

**Passo3:** Continue com o passo 2 até que não haja mais trocas a serem efetuadas.

O procedimento de trocas é o seguinte: Para um dado  $k$ , definimos uma  $k$ -troca de uma rota consistindo da deleção de  $k$  arcos numa rota que serão trocados por  $k$  outros arcos de modo a formar uma nova rota. Uma rota é  $k$ -ótima se não for mais possível efetuar trocas para melhorar a distância total viajada. Geralmente o 3-opt chega bem próximo da solução e o 4-opt gera uma melhoria que não justifica o custo computacional.

A figura 4d ilustra a melhoria 2-opt. Decidiu-se por desligar os arcos (A,B) e (E,F) introduzindo os arcos (A,E) e (B,F) de forma a proporcionar um custo menor, ou seja, uma rota de menor custo que a original. Neste caso é necessário que a matriz de distâncias seja simétrica, pois os arcos (B,C), (C,D) e (D,E) foram invertidos.

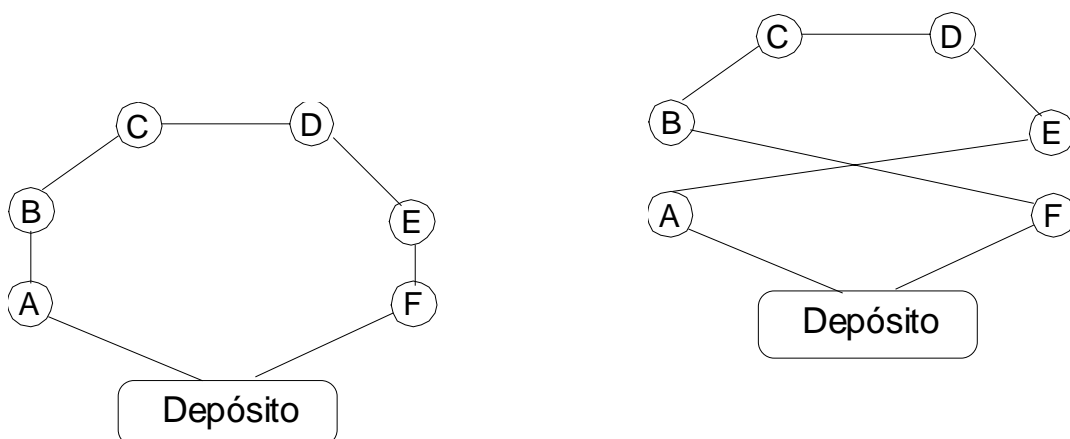


Figura 4d - Ilustração da melhoria 2-opt.

A figura 4e ilustra o caso 3-opt. Desligando-se 3 arcos de cada vez e a cada passo verificando-se a melhoria da nova rota em relação à anterior:

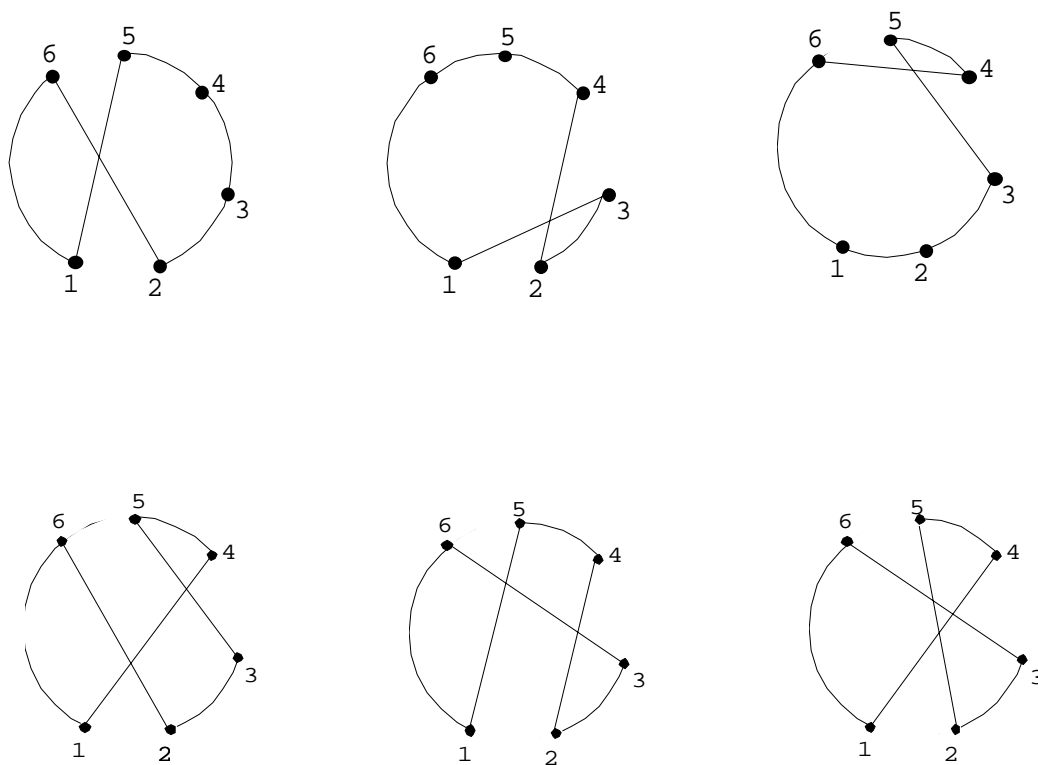


Figura 4e - Ilustração da melhoria 3-opt.

### c.2) Refinamento do k-opt (Or-opt)

Um algoritmo mais poderoso foi proposto por Lin e Kernighan, [17], que decide qual será o valor de k, ou seja, a cada iteração o algoritmo decide quantos arcos serão trocados. Porém este procedimento requer muito mais cálculos que os procedimentos 2-opt eo 3-opt, entretanto produz uma solução muito mais próxima do ótimo.

Tendo-se decidido quantos arcos, s, serão desligados, uma série de testes são realizados para determinar se s+1 arcos deverão ser trocados. Este procedimento é repetido até que uma das condições de parada seja atingida. As trocas são escolhidas de modo a assegurar que as novas rotas formadas sejam viáveis.

#### ALGORITMO:

Seja  $G_p^*$  a melhoria ou segurança que pode ser realizada se os p arcos atuais são trocados de forma a gerar uma nova rota, melhor que a anterior.

**Passo 1:** Escolha um roteiro inicial.

**Passo 2:** Faça  $G_0^* = 0$ . Selecione qualquer nó como nó inicial (por ex. l) e considere um dos arcos adjacentes a este nó, da solução atual (ex.:{l,k}) para remover. Faça p=1.

**Passo 3:** Dos arcos restantes, escolha um arco que não esteja na solução atual (ex.:{k,i}) tal que  $g_1 = c_{lk} - c_{ki} > 0$ . O arco é escolhido de modo a maximizar  $g_1$ .

**Passo 4:** Tendo escolhido {l,k} para sair e {k,l} para entrar na solução da iteração anterior, o arco que deixará o circuito na p-ésima iteração é retirado. Adicionando o arco {j,l} reconstrói-se a rota.  $G_1^* = g_1 + c_{ij} - c_{jl}$ . Se  $g_1 > G_1^* > 0$ , faça p = p+1.



**Passo 5:** O arco  $\{j,l\}$  não é necessariamente o arco escolhido para entrar na solução desta iteração. Procura-se o nó  $q$  que maximize  $g_p = c_{ij} - c_{iq}$ . Se  $q = m$  o

arco escolhido para entrar é  $\{j, m\}$ . Calcule  $G_p = \sum_{s=1}^p g_s$  e então,  $G_p^* = G_p + c_{mn} - c_{nl}$ .

Seja

$G^* = \max \{G_0^*, G_1^*, \dots, G_p^*\}$ . Faça  $p = p+1$  e repita o passo 5 até que:

- Não existam mais trocas possíveis que melhorem a rota;
- A solução corrente já é uma rota ótima;
- $G_p \leq 0$  ou
- $G_p \leq G^*$ .

Se uma destas condições estiver satisfeita, constrói-se a rota associada com a melhor solução de  $\{G_0^*, G_1^*, \dots, G_p^*\}$ .

O procedimento é repetido até que todos os nós tenham sido utilizados como nó inicial e não existam mais melhorias a serem efetuadas na rota.

A Figura 4f ilustra a melhoria efetuada pelo algoritmo.

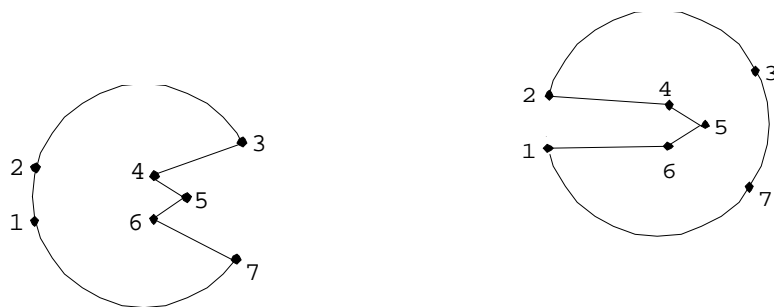


Figura 4f - Ilustração do algoritmo *Or-opt* para  $s = 3$  trocas

### c.3) Troca de nós entre duas rotas [Paraíba, 1990]

Seja  $N_k$  um nó situado na rota  $R_N$  entre os nós  $N_{k-1}$  e  $N_{k+1}$  e outro nó  $M_j$  situado na rota  $R_M$  entre os nós  $M_{j-1}$  e  $M_{j+1}$ . Se o custo

$$c(N_{k-1}, M_j) + c(M_j, N_{k+1}) + c(M_{j-1}, N_k) + c(N_k, M_{j+1}) < c(N_{k-1}, N_k) + c(N_k, N_{k+1}) + c(M_{j-1}, M_j) + c(M_j, M_{j+1})$$

Então troque  $N_k$  por  $M_j$  na rota  $R_N$  e  $M_j$  por  $N_k$  na rota  $R_M$

Uma ilustração da troca entre duas cidades é mostrado na figura 4f.

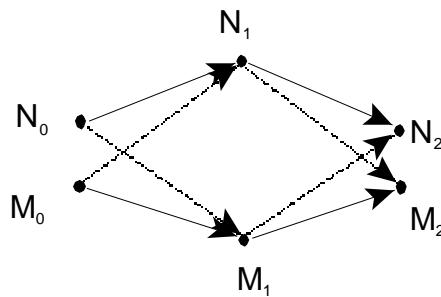


Figura 4f - Ilustração da troca de dois nós entre duas rotas

#### d) Procedimento de Composição de Algoritmos

A composição de algoritmos consiste do seguinte:

**Passo 1:** Obter uma solução inicial usando algum dos procedimentos de construção de rotas;

**Passo 2:** Aplicar o 2-OPT para a rota encontrada no passo 1;

**Passo 3:** Aplicar o 3-OPT para a rota encontrada no passo 1.

O procedimento de composição de algoritmos é relativamente rápido computacionalmente e fornece excelentes resultados.

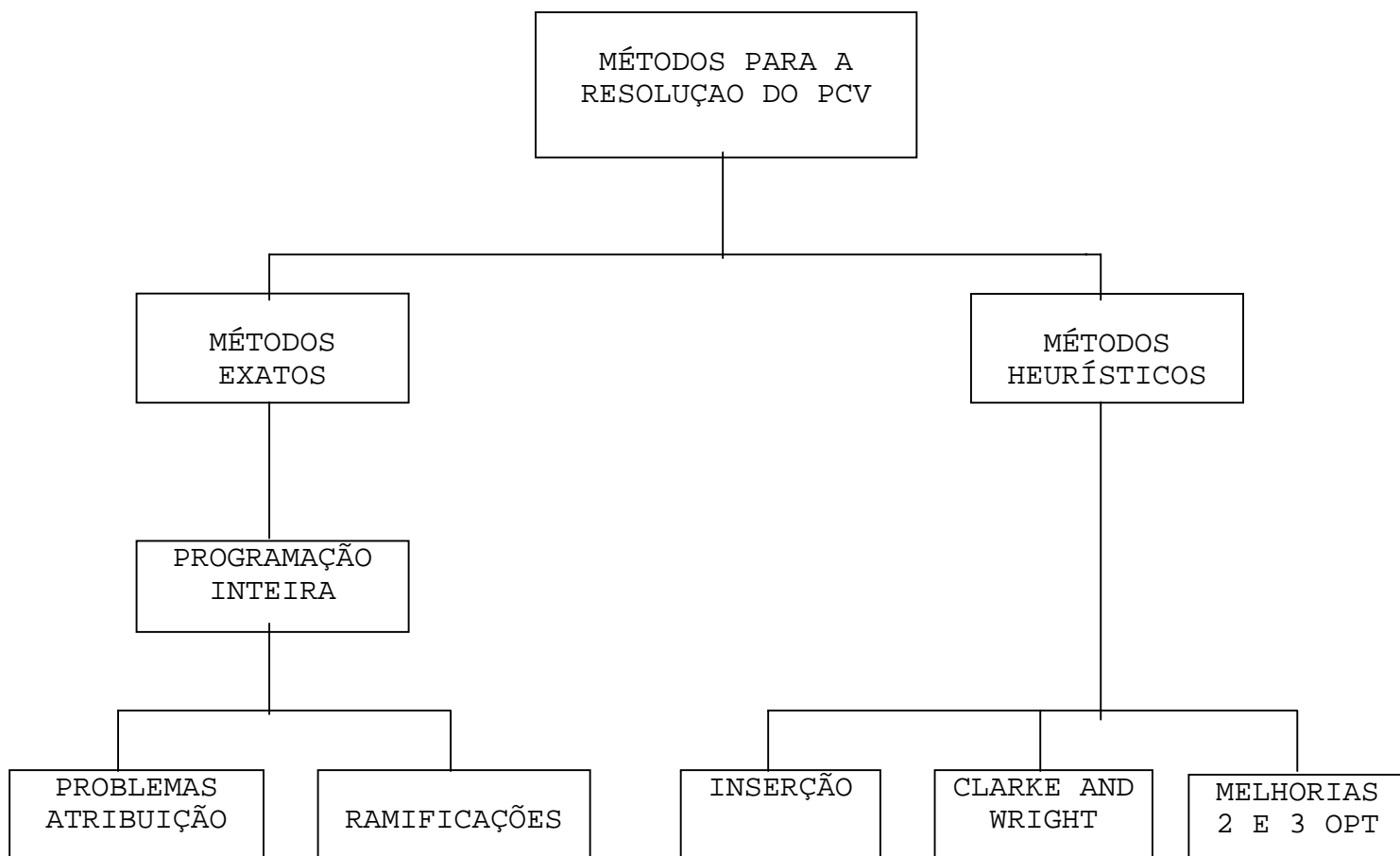
A principal função da Composição de Procedimentos é melhorar rapidamente a solução inicial e esperar que o 2-OPT e 3-OPT forneçam uma solução quase-ótima. O algoritmo 3-OPT pode ser utilizado somente uma vez, ou até que não haja mais trocas a serem efetuadas de forma a melhorar a solução obtida.

#### **e) Variações que podem ser efetuadas**

- Rodar o procedimento somente com o passo 2;
- Rodar o procedimento somente com o passo 3. Esta variação deixa o procedimento muito mais rápido, porém não tão eficiente;
- Utilizar diferentes procedimentos de construção de rotas para comparar os resultados e escolher o melhor deles.

Todas as heurísticas utilizadas aqui necessitam que matrizes de distâncias sejam simétricas.

Apresenta-se aqui um fluxograma que esquematiza os métodos que foram utilizados para solucionar o problema do caixeiro viajante, aplicado ao problema.



## Fluxograma de alternativas para a resolução do PCV

### **4.3. O Problema dos M-Caixeiros Viajantes (MPCV)**

O problema dos m-caixeiros viajantes é uma generalização do problema do caixeiro viajante com um único caixeiro e possui mais aplicações práticas.

Os m-veículos, de uma determinada frota, devem sair de um depósito central visitar os  $n$  nós de uma rede, uma única vez e retornar ao depósito, de forma a minimizar a distância total percorrida pelos m-caixeiros

Cada veículo efetua uma viagem, visita cada nó somente uma vez e retorna ao depósito central.

A figura 4f ilustra o problema dos m-caixeiros viajantes para  $m = 3$ .

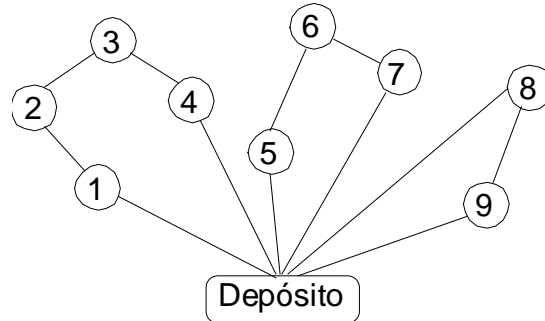


Figura 4f - Ilustração do M-Caixeiro viajante para o caso de 3 caixeiros

A formulação exata, através de um modelo de PLI, para o problema dos m-caixeiro viajantes é apresentada a seguir: [Bodin, 1983]

$$\min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

s.a.

$$\sum_{i=1}^n x_{ij} = b_j = \begin{cases} M, \text{ se } j = 1 \\ 1, \text{ se } j = 2, 3, \dots, n \end{cases} \quad (1)$$

$$\sum_{j=1}^n x_{ij} = a_i = \begin{cases} M, \text{ se } i = 1 \\ 1, \text{ se } i = 2, 3, \dots, n \end{cases} \quad (2)$$

$$X = (x_{ij}) \in S \quad (3)$$

$$x_{ij} = 0 \text{ ou } 1 \quad (i, j = 1, \dots, n) \quad (4)$$

A restrição (1) impõe a condição de que todos os m-caixeiros sejam utilizados.

A formulação exata do problema acima exposto, muitas vezes não é possível de ser resolvida por Programação Inteira, pois em problemas de porte médio, o número de restrições e variáveis torna-se muito grande.

Pode-se então utilizar um método que transforma o problema dos m-caixeiros viajantes em um problema de um único problema do caixeiro viajante e depois aplica-se o algoritmo de atribuição e em seguida, se necessário, o procedimento de ramificação sugerido por Christofides. Estes procedimentos serão descritos a seguir.

#### 4.3.1. Transformação do M-PCV em um único PCV [Golden e Bodin, 1983]

Será apresentado a seguir, um método para a transformação do problema dos múltiplos caixeiros viajantes em um único caixeiro viajante.

##### ALGORITMO:

**Passo 1:** Cria-se m cópias do depósito denotadas por D1, D2, ..., Dm, cada um conectado aos outros nós da mesma forma que eram com o depósito original, isto é, com arcos com os mesmos comprimentos, formando uma rede expandida.

**Passo 2:** As m-cópias não são conectadas, ou são conectadas com arcos com comprimento excedendo a  $\sum_{i=1}^n \sum_{j=1}^n |c_{ij}|$ . Desta maneira uma rota ótima para um único

caixeiro na rede expandida, sobre o conjunto de nós  $\{D_1, D_2, \dots, D_m, 2, 3, \dots, n\}$ , nunca utilizará um arco conectando duas cópias do depósito.

**Passo 3:** Reunindo as cópias em um único nó novamente, a rota para um único caixeiro viajante se decompõe em  $m$  sub-rotas como exigido pelo problema dos  $m$ -caixeiros.

As figuras 4g1 e 4g2, mostram um exemplo de um problema com 4 nós e 2 caixeiros. Originalmente temos os nós 1, 2, 3 e 4 considerando o nó 1 como depósito central. A solução do problema na rede expandida contém os nós  $D_1$ ,  $D_2$ , 2, 3 e 4, onde os nós  $D_1$  e  $D_2$  representam os 2 caixeiros. Fazendo-se a contração da rede, tem-se a solução conforme mostrado na figura 4g2. Pode-se ver que a rota  $D_1-3-D_2-4-2-D_1$  na rede expandida é equivalente a interpretação de dois caixeiros para as sub-rotas  $1-3-1$  e  $1-4-2-1$ , mostradas na figura 4g2.

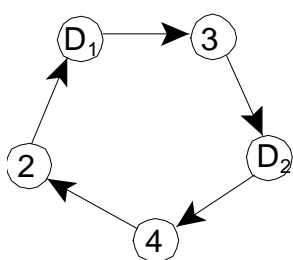


Figura 4g1

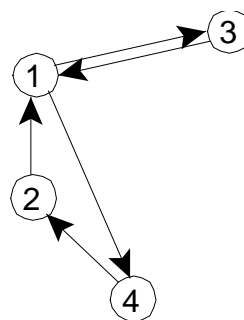


Figura 4g2

Figura 4g - Ilustração de dois caixeiros para duas sub-rotas

Figura 4g1 - Solução na rede expandida

Figura 4g2 - Solução na rede contraída



Para a resolução do MPCV, conforme apresentado na figura (4g), pode-se aplicar qualquer procedimento exato, já que a rede expandida do MPCV pode ser tratada como tal.

#### 4.4. Problema Clássico de Roteirização [Bodin, 1983]

O problema de roteirização de veículos necessita de um conjunto de sub-rotas de entregas a partir de um depósito central para vários pontos de demanda, cada um tendo necessidades de serviços, com o objetivo de minimizar a distância total a ser percorrida pela frota inteira. Os veículos têm restrição de capacidade e tempo. Todos os veículos iniciam e terminam sua sub-rotas em um depósito central.

Este problema, chamado de problema genérico de roteirização de veículos, tem a formulação exposta a seguir, devido a Golden et al., 1977.

$$(4.4.1) \quad \text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{v=1}^V c_{ij} x_{ijv}$$

s.a.

$$(4.4.2) \quad \sum_{i=1}^n \sum_{v=1}^V x_{ijv} = 1 \quad (j=2, \dots, n)$$

$$(4.4.3) \quad \sum_{j=1}^n \sum_{v=1}^V x_{ijv} = 1 \quad (i=2, \dots, n)$$

$$(4.4.4) \quad \sum_{i=1}^n x_{ipv} - \sum_{j=1}^n x_{pjh} = 0 \quad (v=1, \dots, V; p=1, \dots, n)$$

$$(4.4.5) \quad \sum_{i=1}^n d_i \left( \sum_{j=1}^n x_{ijv} \right) \leq K_v \quad (v=1, \dots, V)$$

$$(4.4.6) \quad \sum_{i=1}^n t_{iv} \sum_{j=1}^n x_{ijv} + \sum_{i=1}^n \sum_{j=1}^n t_{ijv} x_{ijv} \leq T_v \quad (v=1, \dots, V)$$

$$(4.4.7) \quad \sum_{j=2}^n x_{1jv} \leq 1 \quad (v=1, \dots, V)$$

$$(4.4.8) \quad \sum_{i=2}^n x_{i1v} \leq 1 \quad (v=1, \dots, V)$$

$$(4.4.9) \quad X \in S$$

$$(4.4.10) \quad x_{ijv} = 0 \text{ ou } 1 \text{ para todos } i, j, v$$

onde:

$n$  - número de nós;

$V$  - número de veículos;

$K_v$  - capacidade do veículo  $v$ ;

$T_v$  - tempo máximo permitido para a rota do veículo  $v$

$d_i$  - demanda do nó  $i$  ( $d_i = 0$ , nó 1 = depósito);

$t_{iv}$  - tempo necessário do veículo  $v$  para entrega ou coleta no nó  $i$

$t_{ijv}$  - tempo de viagem do veículo  $v$  a partir do nó  $i$  até o nó  $j$  ( $t_{ijv} = \infty$ );

$c_{ij}$  - custo da viagem do nó  $i$  para o nó  $j$ ;

$x_{ijv} = 1$  se o arco  $i$ - $j$  é percorrido pelo veículo  $v$ ;

$x_{ijv} = 0$  caso contrário.

- As restrições (4.4.1) indicam que a distância total deve ser minimizada;
- As restrições (4.4.2) e (4.4.3) asseguram que cada nó deve ser atendido por exatamente um veículo.
- As restrições (4.4.4) assegura a continuidade da rota;
- As restrições (4.4.5) são referentes às capacidades dos veículos;
- As restrições (4.4.6) são restrições de tempo da rota;
- As restrições (4.4.7) e (4.4.8) garantem que a disponibilidade dos veículos não são excedidas e
- As restrições (4.4.9) são restrições de quebra de sub-rotas.

Como as restrições (4.4.2) e (4.4.4) implicam em (4.4.3), e (4.4.4) e (4.4.7) implicam em (4.4.8), pode-se excluir do modelo as equações (4.4.3) e (4.4.7) que são redundantes. Assume-se que  $\max_{1 \leq i \leq n} d_i < \min_{1 \leq v \leq V} K_v$ , isto é, a demanda em cada nó não excede à capacidade total de qualquer veículo.

Para resolver este problema deve-se utilizar alguma das heurísticas de construção de rotas que geram rotas quase-ótimas a partir da matriz de custos e em seguida aplicar algoritmos para a melhoria de rotas já citados neste capítulo.

#### 4.5 Algoritmo de Floyd

Para a aplicação de qualquer método exato ou heurístico do PCV ou do MPCV, deve-se conhecer as distâncias (custos) entre todos os pares de nós. Estas distâncias, euclidianas ou reais, podem ser obtidas através do algoritmo de Floyd [*Christofides, 1975*]. Este algoritmo fornece as distâncias mínimas entre os pares de nós de uma rede.

Seja  $C = [c_{ij}]$  a matriz de custos associada a um grafo  $G(X,A)$ , onde

$$c_{ij} = \begin{cases} 0, & \text{para todo par } (x_i, x_j) \in A \\ \infty, & \text{para todo par } (x_i, x_j) \notin A \\ \text{custo associado ao arco } (x_i, x_j) \in A \end{cases}$$

**ALGORITMO:****Passo 1:** Faça  $k = 0$ **Passo 2:**  $k = k+1$ **Passo 3:** Para todo  $i \neq k$  tal que  $c_{ik} \neq \infty$  e todo  $j \neq k$  tal que  $c_{kj} \neq \infty$  calcule:

$$c = \min[c_{ij}, (c_{ik} + c_{kj})].$$

**Passo 4.a.** Se  $k = n$  pare. A solução ótima foi alcançada e  $[c_{ij}]$  fornece os custos mínimos para cada par de nós.**Passo 4.b.** Se  $k < n$ , volte ao passo 2.

O algoritmo de Floyd pode ser complementado de modo a apresentar, além da matriz dos mínimos custos, os caminhos nos quais estes custos foram obtidos. Para isto T. C. Hu, 1969 sugere o armazenamento de uma segunda matriz  $\theta = [\theta_{ij}]$ , onde  $\theta_{ij}$  é o nó predecessor do vértice  $x_j$  no caminho mínimo entre os vértices  $x_i$  e  $x_j$ . Inicialmente, considera-se  $\theta_{ij} = x_i \forall j$ .

A atualização desta matriz é feita da seguinte forma:

$$\theta_{ij} = \begin{cases} \theta_{kj}, & \text{se } (c_{ik} + c_{kj}) < c_{ij} \\ \theta_{ij}, & \text{se } (c_{ik} + c_{kj}) \geq c_{ij} \end{cases}$$

#### **4.6. Algoritmos heurísticos para a solução do PCV com múltiplos depósitos (PCRIV).** [Gillet e Johnson, 1976]

Serão citados aqui, alguns dos algoritmos heurísticos que foram utilizados na resolução do problema abordado neste trabalho.

##### **4.6.1. Algoritmo das razões para problemas com Múltiplos Depósitos** [Bodin, 1983] [Clarke and Wright, 1964]

Inicialmente cada um dos pontos de demanda são designados aos depósitos, por meio de algoritmos que serão citados a seguir, e em seguida vários PCRIV's com um único depósito são resolvidos usando alguma heurística de construção de rotas.

Considera-se, inicialmente, que todos os pontos de parada estão sem designação. Para cada nó  $i$ , faça  $t'(i)$  ser o depósito mais próximo para  $i$  e  $t''(i)$  o segundo mais próximo depósito para  $i$ .

Para cada nó  $i$ , calcule a razão:

$$r(i) = c_{it}(i) / c_{it''(i)}$$

e ordene em ordem crescente pelos valores de  $r(i)$ , assim fica determinada a ordem na qual os nós são designados a um depósito<sup>1</sup>. Após um certo número de nós ter sido designado da lista de  $r(i)$ , um pequeno grupo é formado ao redor de cada depósito.

Os pontos  $i$  tais que a razão  $r(i)$  esteja próximo de 1, são designados como se segue:

Se dois nós adjacentes  $j$  e  $k$  já estão designados a um depósito, inserindo  $i$  entre  $j$  e  $k$  numa rota ligada a  $t$  cria um custo adicional igual a  $c_{ji} + c_{ik} - c_{jk}$  o qual representa uma parte de um custo total (ou distância). Ou seja, o algoritmo designa o ponto  $i$  para o depósito  $t$  inserindo  $i$  entre dois nós já designados ao depósito  $t$ , que fornece um custo mínimo.

Utiliza-se a seguir algum algoritmo para cada grupo separadamente de modo a construir e sequenciar rotas, de modo a obter o menor custo possível. Um número de refinamentos são então feitos para melhorar a solução obtida.

#### 4.6.2. Algoritmo dos *savings* para problemas com múltiplos Depósitos

[Tillman e Cain, 1972], [Bodin, 1983]

Este algoritmo começa com uma solução inicial consistindo de servir cada nó exclusivamente por uma rota do depósito mais próximo. Seja  $c_{ij}$  = custo ou distância entre os nós  $i$  e  $j$ , e  $c_i^k$  = custo ou distância entre o nó  $i$  e o depósito  $k$ . O custo inicial total de todas as rotas é:

$$D = \sum_{i=1}^n 2 \cdot \min\{c_i^k\}$$

onde  $n$  é o número de nós de parada. O método liga pares de nós de maneira a reduzir o custo total. Uma regra básica é utilizada no algoritmo: a designação inicial dos nós para o depósito mais próximo é temporária, mas uma vez que dois ou mais nós tenham sido designados para uma mesma rota de um depósito, os nós não são redesignados para outro depósito. Além disso, como no algoritmo original dos *savings*, os nós  $i$  e  $j$  podem ser ligados somente se os nós  $i$  e  $j$  não forem interiores a uma rota existente.

Para cada passo, a escolha de unir um par de nós  $i$  e  $j$  numa mesma rota do depósito  $k$  é feita através do cálculo dos *savings* quando se liga  $i$  e  $j$  a  $k$ . Os nós  $i$  e  $j$  podem ser unidos somente se nenhuma restrição for violada.

O cálculo dos *savings* não é tão direto como no caso de um único depósito. Os *savings*  $s_{ij}^k$  são associados com cada combinação de nós de parada  $i$ ,  $j$  e o depósito  $k$ , e representam a redução no custo total ou na distância total viajada quando se une  $i$  e  $j$  a  $k$ . A fórmula é dada por:

$$s_{ij}^k = c_i^k + c_j^k - c_{ij}$$

onde:

$$c_i^k = \begin{cases} 2\min\{c_i^t\} - c_i^k, & \text{se } i \text{ ainda não recebeu designação} \\ c_i^k, & \text{caso contrário} \end{cases}$$

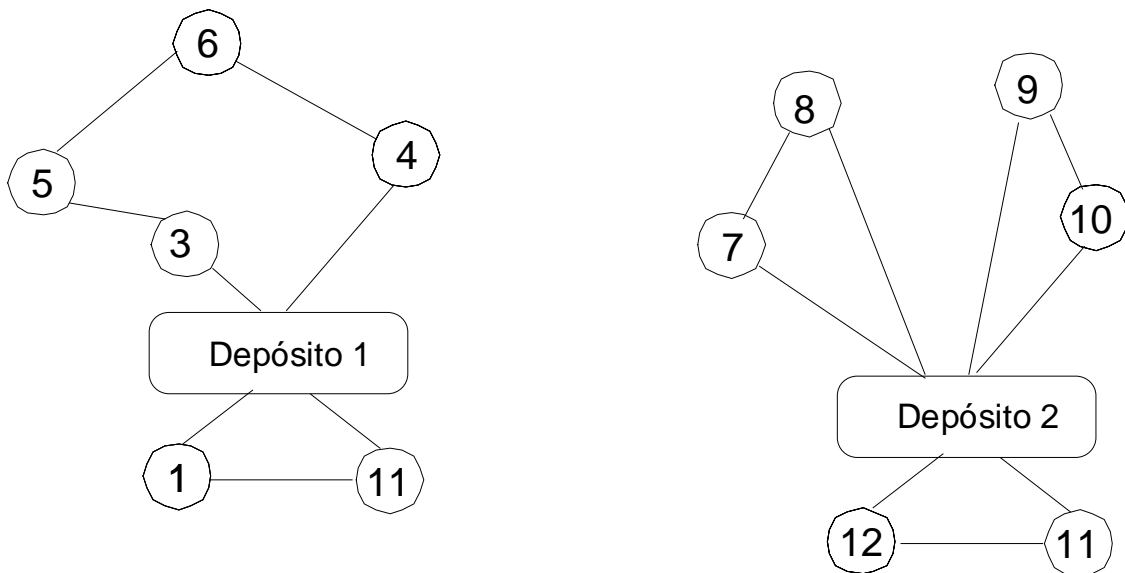
---

<sup>1</sup> Depósito: aqueles que estão relativamente mais próximos são considerados primeiro

No primeiro caso, o nó  $i$  está sendo redesignado do seu depósito mais próximo e a designação previamente feita de custo  $2 \min \{c_i^t\}$  é economizada; no segundo caso, o nó  $j$  está sendo inserido entre o depósito  $k$  e o nó  $i$ , e a ligação de  $i$  a  $k$  é retirada da solução atual.

Os *savings*  $s_{i,j}^k$  são calculados para  $i, j = 1, \dots, n$  para cada passo. Eles podem ser armazenados em  $m$  matrizes  $n \times n$ . A cada passo a ligação  $i - j$  em  $k$  é escolhida tal que maximize  $s_{i,j}^k$  e que forneça uma solução viável (respeitando a capacidade e outras restrições).

A figura 4h, a seguir, mostra um exemplo do  $m$ -caixeiro com múltiplos depósitos.



depósitos.

Figura 4h - Ilustração de 5 caixeiros viajantes com 2 depósitos



#### **4.6.3. Algoritmo dos savings para problemas com múltiplos depósitos, para problemas grandes [Golden et al., 1978]**

O algoritmo resolve o problema em dois estágios: primeiramente os pontos devem ser alocados aos depósitos, e então, as rotas devem ser construídas unindo nós designados ao mesmo depósito. O ideal seria realizar simultaneamente os dois passos, como nos *savings* com múltiplos depósitos já discutido anteriormente.

Inicialmente o algoritmo utiliza as razões como no algoritmo descrito em (4.6.1).

Ignora-se os nós que não sejam fronteira e o algoritmo dos *savings* é aplicado ao conjunto dos nós da fronteira.

Para todo nó  $i$  determina-se o depósito mais próximo  $k_1$  e o segundo mais próximo  $k_2$

$$\text{Se } r(i) = \begin{cases} c_i^{k_1} / c_i^{k_2} < \delta, & \text{designar } i \text{ para } k_1 \\ c_i^{k_1} / c_i^{k_2} \geq \delta, & \text{diz-se que } i \text{ é um nó fronteira} \end{cases}$$

para  $0 \leq \delta \leq 1$ .

$$\text{Se } \delta = \begin{cases} 0, & \text{todos os nós são fronteira} \\ 1, & \text{todos os nós são designados ao depósito mais próximo} \end{cases}$$

Pode-se fixar o número de nós fronteira como se queira, variando  $\delta$ .

#### **ALGORITMO:**

**Passo 1:** Ignora-se os nós que não sejam fronteira e o algoritmo dos *savings* é aplicado ao conjunto dos nós fronteira.

**Passo 2:** Aloca-se os nós fronteira aos depósitos e simultaneamente constrói-se segmentos de rotas conectando estes nós.

Ao final do segundo passo, todos os nós do problema em questão estão designados a algum depósito e todos os nós fronteira estão ligados a alguma rota.

**Passo 3:** Utiliza-se alguma das técnicas de construção de rotas encontrando a solução para cada um dos depósitos.

**Passo 4:** Os segmentos de rotas que estão contruídos com os nós fronteira são extendidos para os nós remanescentes.

## CAPÍTULO V

### 5. IMPLEMENTAÇÃO COMPUTACIONAL DAS TÉCNICAS ABORDADAS E OBTENÇÃO DOS RESULTADOS

#### 5.1. Introdução

A resolução deste problema, envolve muitas fases e procedimentos que serão descritos neste capítulo. Por ser um processo dinâmico o problema se torna de difícil tratamento, pois requer:

- A localização manual dos pontos de parada no mapa, que são em grande quantidade;
- Frequentes mudanças de endereço dos alunos, e portanto, mudanças nos pontos de parada;
- Grande variedade de tipos de ônibus, ou seja, vários ônibus de diferentes capacidades.

O objetivo é dividir o grande grupo de pontos de parada em vários pequenos grupos e determinar qual tipo de ônibus deverá atender a qual agrupamento de pontos de parada, de forma a minimizar a distância total percorrida, obtendo-se como consequências:

- Minimização do tempo de permanência das crianças nos ônibus;
- Economia de combustível, manutenção e outros;
- Maior satisfação por parte dos pais e alunos.

E posteriormente, determinar a sequência ótima em que estes pontos de parada serão atendidos, evitando cruzamentos nos roteiros dos ônibus.

## 5.2. Estudo da situação atual da escola

Atualmente, o serviço de roteirização da escola, é feito da seguinte forma: primeiramente as fichas dos alunos contendo seus endereços, são separadas por regiões e então vão sendo alocadas aos ônibus até que a lotação de cada um esteja completa. É claro que não está sendo feito um trabalho de otimização, ou seja, existem problemas tais como: cruzamento de rotas fazendo com que haja reclamações dos pais quanto aos horários que os ônibus atendem às crianças, maior demora no trajeto dos ônibus e taxa de ocupação inadequada.

Além destes problemas, o trabalho de localização dos pontos no mapa é um trabalho que requer muito tempo e concentração, o trabalho de inserir os pontos de parada à mão num mapa impresso demora em torno de três meses para ficar pronto. O trabalho de localização manual dos pontos no mapa é feito depois que as rotas já estão formadas, assim sendo, estas rotas estão sendo formadas sem a visão geral do grupo, o que faz com que ocorram muitos cruzamentos de rotas e constantes reclamações dos pais.

Com o trabalho de otimização estes problemas serão evitados e a localização dos pontos fica muito mais rápida e eficaz, pois é feita diretamente num software que localiza e insere os pontos de parada no mapa, e o trabalho de roteirização não demora mais do que um dia para ficar pronto.

A figura 5a mostra a disposição dos pontos de parada distribuídos na cidade de Curitiba.

Figura 5a - Distribuição dos pontos de parada na cidade de Curitiba

### **5.2.1. Divisão em grupos**

A figura 5b a seguir, mostra a divisão dos pontos de parada em vinte e cinco grupos (clusters), um para cada ônibus. Esta divisão foi realizada pela escola, para o período letivo de 1997.

Figura 5b - Divisão em 25 clusters realizada pela escola

### **5.2.2. Sequência dos pontos de parada**

A sequência em que os pontos de parada estão sendo atendidos está representada no mapa mostrado na figura 5c a seguir:

Figura 5c - Sequência dos pontos de parada adotada pela escola

### **5.2.3. Apresentação da situação atual**

Atualmente, a situação da escola com relação a quantidade, capacidade e distribuição das demandas e distâncias, é apresentada no quadro 5.1 a seguir:



Tipos de ônibus	Quantidade (disponível)	Capacidade (disponível)	Demanda (utilizada)	Distâncias Percorridas
1 (fictício)*	999	01	--	--
02	02	32	32/19	21.826/7.933
03	05	43	36/41/40/36	19.236/12.266/ 17.089/12.376
04	01	45	44	24.369
05	08	49	48/47/49/40/43 /51/45/47	15.469/20.374/ 20.213/11.117/ 25.984/31.680/ 13.519/15.819/
06	02	50	--	--
07	01	56	57	24.826
08	01	58	50	5.839
09	01	60	55	10.983
10	02	44	29/20	4.066/12.343
11	03	46	41/31/37	7.351/13.422/ 12.053
12	01	49	28	33.482
13	01	51	31	31.379
Total	27	1210	997	425.014**

Quadro 5.1 - Situação atual dos ônibus da escola

Pode-se observar, que existem ônibus com capacidades excedidas e também alguns com sua utilização bem menor do que sua capacidade, esta estratégia é utilizada visando reduzir a quilometragem feita por cada um dos veículos, tentando com isso reduzir o tempo de permanência dos alunos nos ônibus, mesmo assim ocorrem rotas com distâncias muito grandes.

Com este trabalho tem-se a intenção de procurar fazer com que os ônibus com maiores trajetos possuam menos pontos de parada, reduzindo assim, o tempo de percurso e que, em compensação, os ônibus com trajetos menores atendam mais pontos de parada.

### 5.3. Implementação computacional

As heurísticas e procedimentos de melhoria utilizados na implementação computacional, foram descritas no capítulo IV.

Os procedimentos foram implementados em linguagem Visual Basic 4.0 e as particularidades necessárias para a obtenção dos resultados são comentadas.

### **5.3.1. Cadastramento dos dados**

O cadastramento dos alunos que utilizam o transporte escolar, foi feito da seguinte maneira:

- Cada aluno deve preencher uma ficha, onde estarão todos os seus dados pessoais como: endereço contendo nome da rua, número da casa e as transversais. Um modelo desta ficha é apresentado no Anexo I.
  
- As fichas são organizadas em ordem alfabética pelos nomes das ruas, para facilitar a digitalização dos endereços, e também para cadastrar a demanda de cada ponto, pois pode ocorrer que em cada ponto exista mais do que um aluno utilizando o transporte escolar, que é o caso de irmãos ou alunos que residem no mesmo prédio.

---

\* Onibus fictício utilizado para ajustes feitos nos programas computacionais

\*\* As distâncias estão apresentadas em metros

- Os pontos de parada são cadastrados um a um. Este cadastramento é realizado com a ajuda de um Software que possui o mapa de Curitiba digitalizado, e também um localizador de endereços, no qual é digitado o nome da rua e número da residência. O endereço é então localizado no mapa, neste ponto é inserido um símbolo. O ponto possui coordenadas geográficas que serão utilizadas mais tarde pelo programa.
- Um banco de dados é então formado, contendo as coordenadas geográficas dos pontos, e a cada ponto existe uma chave associada. Esta chave foi construída de modo que contivesse informações sobre o ponto de parada a que ela está associada. Cada quadra possui um eixo com um determinado número. A chave de acesso de cada ponto de parada é constituída por este número seguido do número da casa ou edifício.

### **5.3.2. Processo Inicial - Separação dos pontos em dois grupos**

Primeiramente elaborou-se um programa para separar os pontos de parada que a Escola atenderá e os pontos que serão atendidos pela empresa de terceirização (E.T.). São selecionados neste passo, os pontos que estão situados

mais próximos de cada uma das garagens. O resultado desta divisão está apresentado na figura 5d.

Pelo resultado desta divisão a E.T. atenderá 249 alunos e a escola 780 alunos. Neste passo já se pode notar uma melhoria, pois a escola designou para a empresa 217 crianças e terceirizou 7 ônibus, enquanto que através do processo de otimização, foram designados 6 ônibus para as 249 crianças e mais 18 ônibus da escola para atender as outras 748 crianças.

Figura 5d - Separação dos pontos atendidos pela escola e pela E.T.

### **5.3.3. Utilização da Programação Inteira**

- O primeiro passo para a resolução do Problema do Transporte Escolar foi a busca de dados reais em relação aos ônibus. Foi fornecido pela escola uma listagem contendo os tipos de ônibus, a quantidade de cada tipo e a capacidade de cada um. Como a escola não possui uma quantidade suficiente de ônibus para atender à demanda, o serviço é terceirizado pela

E.T., que aloca alguns ônibus para a escola. Os dados sobre os ônibus estão disponíveis nos Quadros 5.2 e 5.3 a seguir:

#### TIPOS E QUANTIDADES DE ÔNIBUS UTILIZADOS PELO COLÉGIO

TIPOS DE ÔNIBUS	CAPACIDADE	QUANTIDADE
1	32	2
2	43	5
3	45	1
4	49	8
5	50	2
6	56	1
7	58	1
8	60	1

Quadro 5.2 - Quantidades e capacidades dos ônibus da escola

#### TIPOS E QUANTIDADES DE ÔNIBUS UTILIZADOS PELA E.T.

TIPOS DE ÔNIBUS	CAPACIDADE	QUANTIDADE
1	44	2
2	46	3
3	19	1
4	51	1

Quadro 5.3 - Quantidades e capacidades dos ônibus disponíveis da E.T.

#### a) Formulação de um modelo de Programação Inteira para a Escola

Para obter as quantidades e capacidades dos veículos a serem utilizados, foi construído um modelo de Programação Linear Inteira, considerando a quantidade de veículos disponível e a capacidade de cada um.

A formulação é apresentada a seguir:

$$(1) \quad \text{Min}_{x_i} z = \sum_{i=1}^8 x_i$$

s.a

$$(2) \quad x_1 \leq 2$$

$$(3) \quad x_2 \leq 5$$

$$(4) \quad x_3 \leq 1$$

$$(5) \quad x_4 \leq 8$$

$$(6) \quad x_5 \leq 0$$

$$(7) \quad x_6 \leq 1$$

$$(8) \quad x_7 \leq 1$$

$$(9) \quad x_8 \leq 1$$

$$(10) \quad x_1 \geq 2$$

$$(11) \quad (32 - 2)x_1 + (43 - 3^*)x_2 + (45 - 3)x_3 + (49 - 3)x_4 + (50 - 3)x_5 + \\ + (56 - 3)x_6 + (58 - 3)x_7 + (60 - 3)x_8 \geq 748^{**}$$

$$(12) \quad x \geq 0, \quad i = 1, \dots, 8 \text{ e inteiros}$$

onde,  $x_i$  representa a quantidade de ônibus de cada tipo a serem utilizados pela escola.

Observação: o ônibus fictício não entra na formulação.

O que se deseja na função objetivo é minimizar a quantidade total de ônibus utilizada.

As restrições de (2) a (9) indicam quantidade disponível de cada tipo de ônibus;

Combinando as restrições (2) e (10) tem-se  $x_1 = 2$ , o que indica que devem ser usados os 2 ônibus do tipo 1, pois é uma exigência da escola que os dois micro-ônibus sejam utilizados para realizarem o trajeto no centro da cidade;

---

\* Folga de 2 lugares exigida pela escola e mais 1 para os ajustes do programa computacional

\*\* Número de alunos a serem atendidos pela escola

A restrição (6) indica que o ônibus do tipo 5 deve ficar à disposição, ou seja  $x_5 = 0$  (ônibus-reserva);

A restrição (11) indica que a capacidade total dos ônibus deve ser maior ou igual ao número de alunos que utilizam o transporte escolar.

Os resultados obtidos estão disponíveis no Quadro 5.4

TIPO DE ÔNIBUS	QUANT./TIPO
1	2
2	5
3	1
4	7
5	0
6	1
7	1
8	1
TOTAL	18

Quadro 5.4 - Resultados obtidos da resolução do modelo de PLI

### b) Formulação do Problema de Programação Inteira para a E.T.

As mesmas considerações são feitas para se resolver o PLI para a E.T.

$$(1) \quad \text{Min } z = x_1 + x_2 + x_3$$

s.a

$$(2) \quad x_1 \leq 3$$



- (3)  $x_2 \leq 2$
- (4)  $x_3 \leq 1$
- (5)  $(44 - 3)x_1 + (46 - 3)x_2 + (49 - 3)x_3 \geq 249^*$
- (6)  $x \geq 0$ , e inteiros

As restrições e variáveis têm o mesmo significado que no caso anterior.

Os resultados obtidos estão disponíveis no Quadro 5.5

TIPO DE ÔNIBUS	QUANT./TIPO
1	3
2	2
3	1
TOTAL	6

Quadro 5.5 - Resultados obtidos da resolução do PLI

O resultado mostrou que o problema pode ser resolvido utilizando 18 ônibus da escola e 6 da E.T. Concluí-se então, que o problema pode ser solucionado com 24 ônibus.

#### 5.3.4. Determinação das Sementes - Depósitos Fictícios

Conhecendo-se o número de veículos a serem utilizados, pode-se definir então, as sementes. Inicialmente foram definidas as sementes<sup>1</sup> para a região central da cidade, denominadas sementes críticas, pois foram designadas manualmente e se referem a regiões onde só podem circular micro-ônibus, pela dificuldade de tráfego. No caso são dois micro-ônibus disponíveis pela escola, ou seja, duas sementes foram escolhidas manualmente sendo que cada uma destas

\* Número de alunos a serem atendidos pela E.T.

sementes agrupou no máximo 30 alunos, tendo sido definido para estas duas sementes uma folga de dois lugares, diferente das folgas definidas para as demais regiões.

A figura a seguir mostra os agrupamentos de pontos formados na região central.

#### Figura 5e - Agrupamentos Centrais

A seguir foram determinadas as demais sementes (figura 5f) utilizando o algoritmo que seleciona a melhor posição para as sementes [Paraíba, 1990]. Ficaram definidas 22 sementes, no total uma para cada veículo, porém a partir da 10ª. semente as suas localizações começaram a ficar muito próximas; considerou-se, então, apenas 9 sementes (contando com as sementes críticas), que são os pontos mais distantes entre si e mais distantes do depósito, onde

$$C_{i i_k-1} + C_{i i_k-2} + \dots + C_{i i_2} + C_{i i_1}$$

é máximo e  $i_k$  são os pontos não roteirizados.

---

<sup>1</sup> Sementes - Depósitos fictícios

Figura 5f - Localização das sementes

### 5.3.5. Determinação dos Clusters Ótimos

Este procedimento é dividido em duas fases:

1<sup>a</sup>. Fase:

Nesta 1<sup>a</sup>. Fase é efetuada a pré-designação dos pontos de parada à estas sementes, utilizando o algoritmo com múltiplos depósitos descrito no capítulo IV, item 4.6. Este algoritmo agrupa os pontos mais próximos aos depósitos através do cálculo das razões:

$$r(i) = c_{it'}(i) / c_{it''}(i)$$

As capacidades dos veículos não são consideradas nesta etapa, apenas o cálculo da razão  $r(i)$ . Para pontos duvidosos, isto é, que apresentaram a razão  $r(i) > 0.7$ , fez-se a pré-designação de um ponto  $i$  de acordo com a expressão  $c(j, i) + c(i, k) - c(j, k)$ . Assim foram formados os 9 agrupamentos ao redor das sementes. Estes agrupamentos estão apresentados na figura 5g.

Figura 5g - Formação dos 9 clusters (sem considerar as capacidades dos ônibus)

2<sup>a</sup>. Fase:

Na 2<sup>a</sup>. Fase, considerando-se as capacidades dos veículos, e aplicando-se o procedimento devido a Tillman e Cain, 1972, que começa com uma solução inicial consistindo por servir cada ponto de parada exclusivamente por uma rota a partir do depósito mais próximo. O método liga sucessivamente os pares de pontos de parada, visando sempre, um custo total mínimo. A cada passo, a escolha de unir dois pares de pontos de parada é feita em termos dos *savings* de Clarke and Wright, 1964, este procedimento foi descrito com detalhes no capítulo IV, item 4.6.1.

Neste passo foram formadas 31 rotas para a escola, sendo que 13 destas rotas são fictícias, pois o resultado da otimização mostrou que são necessários apenas 18 ônibus da Escola. Para a E.T. foram formadas sete rotas, sendo que

uma delas é fictícia, pois utilizou-se apenas 6 ônibus terceirizados, ao invés de 7 como era feito pela escola

As rotas fictícias são aquelas que são designadas à ônibus fictícios, ao se inicializar o procedimento. Estas rotas devem ser incluídas em outros roteiros reais. Para isto são feitos alguns ajustes nas respostas tentando sempre inserir as rotas fictícias nas rotas formadas e procurando sempre incluir primeiramente os pontos com maiores demandas, pois se estes forem deixados para o final poderão ter como resultado a inclusão em uma rota saturada, fazendo com isto que a rota tenha uma solução pior, ou seja uma distância maior.

Justamente por causa da inclusão destes pontos é que são deixados folgas de alguns lugares nos ônibus. Vários testes foram realizados com diferentes folgas, o que apresentou melhor resultado foi o teste com a folga de cinco lugares, duas delas são deixadas por exigência da escola, que às vezes precisa para levar professores ou para mudanças de endereços ou mesmo para alunos novos, e os outros três lugares são necessários para a inclusão dos pontos que foram alocados à rotas fictícias.

Os agrupamentos ótimos considerando-se as capacidades dos veículos estão apresentadas na figura 51, na página 94.

### **5.3.6. Determinação da sequência ótima em que os pontos devem ser atendidos**

Para encontrar a sequência ótima em que os pontos de parada devem ser atendidos, aplicou-se o algoritmo da Inserção mais econômica citado no cap.IV, item b.1.2, onde considerou-se como primeiro ponto da rota, o ponto mais próximo

da garagem e como último ponto, o mais próximo da sede da escola. A inserção foi feita separadamente para cada um dos 24 grupos formados na etapa anterior. A partir do primeiro ponto, procurou-se qual o ponto que forneceria maior economia, em termos de distância, ou seja, tal que  $c_{ik}+c_{kj}-c_{ij}$  fosse mínimo. Este ponto foi inserido na rota entre o primeiro e o último, o procedimento foi sendo repetido até que todos os pontos da rota estivessem ligados.

O veículo inicia seu roteiro pelo primeiro ponto, passa coletando os alunos e vai até o último ponto, que fica localizado mais próximo da estrada que leva à escola. Porém o trajeto que será utilizado para alcançar cada ponto é de responsabilidade do motorista.

Observa-se que a distância utilizada foi a euclidiana.

A figura 5h mostra o roteiro de um dos ônibus a nível de bairros, e a figura 5i mostra o roteiro a nível de quadras.

Figura 5h - Roteiro de um ônibus a nível de bairro



Figura 5i - Roteiro de um ônibus a nível de quadras

### **5.3.7. Ajustes nas respostas**

a) O primeiro ajuste trata de incluir as rotas fictícias nas rotas formadas, como já mencionado no item 5.3.5. A inclusão dos nós fictícios nas rotas se dá da seguinte maneira:

- Primeiro seleciona-se os nós com maior demanda, a fim de evitar que mais tarde estes pontos tenham que entrar em rotas de forma a tornar muito grande a distância da mesma. Procura-se então, a rota mais próxima da rota fictícia para que o nó seja inserido e a melhor posição nesta rota, sempre verificando a capacidade de ônibus, por causa das trocas efetuadas.

Com este ajuste, ficaram agora, um total de dezoito rotas formadas para o colégio e seis rotas para a E.T., perfazendo o total de vinte e quatro rotas;

Aplica-se, então, o algoritmo de melhorias nas rotas, 2-Opt e 3-Opt, descritos no capítulo IV, item c.1. Neste passo procura-se trocar a sequência em que os

pontos de parada são atendidos, desligando-se respectivamente, dois e três arcos de cada vez e efetuando-se novas ligações entre os pontos de forma a reduzir a distância total percorrida pelo veículo. Este algoritmo é aplicado até que não haja mais trocas possíveis na rota, que possam minimizar a distância total.

b) O segundo ajuste, efetua um remanejamento dos ônibus verificando dentre os ônibus que estão sendo utilizados, se existe possibilidade de troca entre ônibus, de forma a alocar os de menor capacidade às menores rotas;

c) O terceiro ajuste verifica entre todos os ônibus disponíveis, incluindo os de reserva, se existe algum que possa ser trocado com algum daqueles alocados às rotas, de forma a melhorar a taxa de ocupação dos veículos.

#### **5.4. Obtenção dos resultados**

Aplicando os métodos acima descritos, obteve-se a resposta para o problema de roteirização do transporte escolar. Para medir o grau de eficiência das técnicas abordadas, os resultados obtidos foram comparados com a solução realizada pela escola. Estes resultados encontram-se no Quadro 5.6.

Para poder realizar esta comparação, foram cadastradas as rotas efetuadas pela escola, com a divisão feita pelos responsáveis do setor de transportes. Em cada uma destas rotas foi calculada a distância euclidiana, para se poder comparar com os resultados obtidos pelo processo de otimização.

O quadro a seguir mostra a utilização dos ônibus após a implementação das heurísticas.

TIPOS DE ÔNIBUS	QUANTIDADE	CAPACIDADE	DEMANDAS UTILIZADAS	1°. AJUSTE	2°. AJUSTE	3°. AJUSTE
1	999	6	4/2/2/6/5/3/2/ 1/2/1/1/1	--	--	--
2	2	32	30/30	30/30	30/30	30/30
3	5	43	37/25/37/37/ 37	41/33/37/ 40	33/37/40/ 36	33/37/34/ 40/36
4	1	45	--	--	--	40
5	8	49	44/32/44/44/ 44/44/37/36	44/34/44/ 44/44/44/ 40/36	41/44/34/ 44/44/44/ 40/36	41/44/44/ 44/44
6	2	50	--	48	48	48
7	1	56	51	51	51	51
8	1	58	53	53	53	53
9	1	60	55	55	55	55
10	3	44	32/41/42	34/41/42	42/30/42	42/30/42
11	2	46	43/43	43/43	44/44	44/44

\* Ônibus fictício utilizado para as rotas fictícias

12	1	49	46	46	47	47
TOTAL	27	1263	997	24	24	24

Quadro 5.6 -Resultados obtidos através da utilização dos algoritmos e ajustes

Um exemplo de algumas rotas da região central da cidade é mostrada na figura 5j a seguir:

Figura 5j - Formação de algumas rotas com a sequência dos pontos de parada, da região central da cidade

### **5.5. Ajustes Finais**

Este trabalho é um processo dinâmico, pois a todo momento ocorrem mudanças de endereço e novas matrículas na escola, assim são necessários ajustes constantes para a utilização das soluções. Foi feito então, um programa que realiza estas de atualizações, que consta do seguinte:

Mudança de Endereço:

- Verificar se no endereço de onde o aluno está saindo, existe mais de um aluno. Se existir, aquele ponto de parada não poderá ser retirado do Banco de Dados, pois existem outros alunos que deverão ser apanhados naquele endereço.
- Verificar se no endereço para o qual o aluno está indo, já existe algum aluno, se já existir deve-se utilizar o mesmo ponto de parada atualizando apenas, a demanda daquele ponto. Se não existir, deve-se criar um e novo ponto de parada para aquele aluno e incluí-lo em

algum dos roteiros existentes, utilizando as folgas de lugares deixadas nos ônibus, também para este fim. Verifica-se qual é a rota mais próxima para a inserção daquele ponto e qual é a melhor posição para a inserção na rota. A capacidade também deve ser verificada, pois pode ocorrer que não existir mais lugares sobrando no ônibus, para acolher esta demanda. O programa prevê também a possibilidade de trocar algum ponto da rota para a inclusão deste novo ponto. Neste caso, poderão ser efetuadas várias trocas, sempre procurando obter a menor distância possível nas rotas.

#### Endereço Novo

- Verificar se o endereço da criança já existe. Se existir apenas atualiza-se a demanda daquele ponto. Caso contrário insere-se o ponto no mapa e no banco de dados, e procura-se qual é o roteiro mais próximo daquele ponto de parada. Se a rota estiver saturada deve-se verificar a possibilidade de troca de algum dos pontos da rota com outro ponto de outra rota, de modo a deixar lugar suficiente para agrupar o novo ponto.

Com isto pode-se analisar qualquer situação e efetuar as mudanças necessárias, tendo os dados e a demanda de cada ponto atualizados. Estas alterações devem ser efetuadas sempre que necessário.

## 5.6. Resumo dos resultados

Serão apresentados nesta seção o resumos dos resultados obtidos através da otimização das rotas para o problema do transporte escolar. Estes resultados podem ser encontrados no Quadro 5.7.

QUANTIDADES E CAPACIDADES			PPI		SOLUÇÃO OTIMIZADA		
TIPOS DE ÔNIBUS	QUANTIDADE	CAPACIDADE	QUANTIDADE	CAPACIDADE	QUANTIDADE	DEMANDAS	SOBRAS
1	999	6	--	--	--	----	
2	2	32	2	64	2	30/30	4
3	5	43	5	215	5	33/37/34/ 40/36	35
4	1	45	0	0	1	40	5
5	8	49	8	392	6	41/44/44/ 44/44/44/	131
6	2	50	0	0	1	48	2
7	1	56	1	56	1	51	5
8	1	58	1	58	1	53	5
9	1	60	1	60	1	55	5

---

\* Ônibus fictício

10	3	44	3	132	3	42/30/42	18
11	2	46	2	92	2	44/44	4
12	1	49	1	49	1	47	2
TOTAL	27	1263	24	1118	24	997	216

Quadro 5.7 - Resultados obtidos para os ônibus, através da otimização

Pela experiência de anos anteriores, o pessoal que cuida da parte de transportes do escola, sabe que até o final do primeiro semestre, a situação já terá se alterado muito, por causa das frequentes mudanças de endereço ocorridas no período e alunos que entram e saem da escola. Isto acarreta muitas mudanças nos roteiros, fazendo com que apresente resultados piores do que os iniciais, ou seja, uma distância total a ser percorrida maior do que a inicial. O que se deve fazer então, é reavaliar a situação podendo até mesmo refazer todo o processo desde o início.

### 5.7. Análise dos resultados

Nesta seção é feita uma análise dos resultados e as comparações dos resultados obtidos pelo processo de otimização e pela solução fornecida pela escola.

DISPONÍVEL			RESULTADOS OBTIDOS	
Tipos de ônibus	Quantidade disponível	Capacidade	Demanda	Distâncias Percorridas
1 (fictício)	999	6	--	--
2	2	32	30/30	3.351/6.537
3	5	43	34/37/34/40/36	13.912/8.669/ 23.735/13.476/ 14.895
4	1	45	40	18.799
5	8	49	41/44/44/44/44 /44	13.880/10.396/ 6.748/10.353/ 10.423/9.200
6	2	50	48	24.560
7	1	56	51	6.235
8	1	58	53	5.452



9	1	60	55	5.383
10	3	44	42/30/42	17.167/24.961/ 37.831
11	2	46	44/44	24.108/15.000
12	1	49	47	13.187
Total	27	1263	997	338.258

Quadro 5.8 - Resultados obtidos para as distâncias das rotas  
Comparando as distâncias do Quadro 5.8 com as distâncias do Quadro 5.1, pode-se perceber uma redução de 111.125m no percurso total de uma viagem. Considerando que este percurso é feito duas vezes ao dia, a redução é de 222.250m diários, isto representa economia em termos de desgaste dos veículos, combustível, e principalmente do tempo de permanência da criança no ônibus, o que constitui a principal reclamação dos pais. Além de reduzir também, os custos com um motorista, uma acompanhante e um ônibus, já que o processo mostrou que o problema pode ser solucionado com apenas 24 ônibus e mesmo assim reduzir a quilometragem total rodada duas vezes ao dia.

É válido lembrar que em todas as situações citadas neste trabalho, nos testes realizados e também nas rotas executadas pela escola, foram consideradas as distâncias Euclidianas, para se poder comparar os resultados na mesma grandeza.

### 5.8. Apresentação da solução

O fornecimento dos resultados é feito da seguinte maneira:

- 1) É implantado na escola um sistema contendo o mapa digitalizado que permite:

- A visualização dos pontos de parada um a um no mapa digitalizado;
  - A visualização de qual é o aluno que está num determinado ponto de parada;
  - Visualização do mapa global com todos os pontos de parada inseridos;
  
  - Visualização de todas as rotas simultaneamente ou de cada uma separadamente;
  
  - A verificação rápida do cadastro de qualquer um dos alunos, seu endereço e sua localização no mapa, qual ônibus está utilizando, nome do responsável e qualquer outra informação necessária.
  
  - A localização rápida de novos pontos de parada.
- 1) Cada motorista recebe um relatório sobre os pontos de parada por ele atendidos, onde consta a sequência dos pontos de parada, os dados sobre os alunos, tais como endereço, apartamento, telefone, responsável, e outros.
  
  - 2) É também fornecido para cada motorista, um mapa com todos os pontos de parada e com a sequência de atendimento à estes pontos.
  
  - 4) A escola recebe um mapa global, contendo todos os pontos de parada e todas as rotas.

Na figura 5l a seguir, pode-se observar os clusters formados pelo processo de otimização, está mais concentrado do que na figura 5b pág 65.

Figura 5l - Formação dos clusters ótimos  
(considerando as capacidades dos veículos)

Nesta figura estão apresentadas todas as rotas formadas pelo processo de otimização, pode-se perceber bem menos cruzamentos do que na figura 5c da página 66.

Figura 5m - Formação das rotas pelo processo de otimização

## CAPÍTULO VI

### 6. CONCLUSÕES E SUGESTÕES PARA FUTUROS TRABALHOS

#### 6.1 Conclusões

A fase de maior importância do trabalho é, sem dúvida, a divisão da cidade em setores ótimos. Procurando concentrar ao máximo os pontos de parada, pois é olhando para o todo que se percebe a grande melhoria no trabalho. Verificando-se separadamente o percurso de cada rota, observa-se que já está sendo feito o melhor trajeto possível, isto pela própria experiência dos motoristas que vão a cada dia melhorando seus roteiros.

Fica estabelecida uma solução inicial para o problema, mas como existem constantes mudanças de endereços, e novos alunos utilizando o transporte escolar, os ajustes são realizados procurando efetuar as menores alterações possíveis nos roteiros dos outros ônibus.

Adotando-se os procedimentos de otimização percebe-se que o percentual de melhoria é bastante significativo, e que reduz o tempo de permanência dos alunos nos ônibus. Estes resultados comparativos estão mostrados no Quadro 5.9.

	ESCOLA	OTIMIZAÇÃO	MELHORIA OBTIDA POR VIAGEM
Distância Inicial	65.962	67.236	--
Distância para percorrer os pontos	425.014	338.258	86.756 (21%)
Distância Final	145.483	137.213	--
Distância Total	636.459	542.707	93.752 (14,7%)

Quadro 6.1 - Comparação das distâncias percorridas pela frota

A distância inicial é definida como sendo a distância da garagem até o primeiro ponto;

A distância final é definida como sendo a distância do último ponto de parada até a saída para o Colégio

A distância total é a distância percorrida pelos ônibus por viagem;

Uma viagem é um percurso de ida ou de volta;

O tempo médio de permanência dos alunos no ônibus é dado pelo tempo gasto para percorrer a distância entre os pontos mais o tempo gasto para percorrer a distância final.

No processo realizado pela escola, os alunos permanecem o tempo necessário para percorrer 570.497m diários, enquanto que no processo de otimização, os estudantes permanecem nos ônibus, o tempo necessário para percorrer 475.471m diários, havendo portanto, uma redução 16,6% do tempo de permanência dos alunos nos ônibus.

Muitas dificuldades foram encontradas ao se tentar solucionar o problema do transporte escolar, tais como:

- A localização exata de cada ponto de demanda no mapa;
- Escolher dentre os inúmeros procedimentos heurísticos existentes, aqueles que seriam utilizados, e escolher a melhor combinação de métodos de forma a encontrar uma boa solução para o problema;
- A solução obtida, embora matematicamente seja uma solução quase ótima, pode não ser a melhor solução para a escola, pois este problema possui muitas particularidades, como por exemplo motoristas que não estão acostumados aos roteiros fornecidos pelo programa e que não desejam adaptar-se a novas regiões, e os sentidos das ruas que muitas vezes obriga que a sequência de coleta ou entrega nos pontos de parada seja alterada;
- As dificuldades em se comparar os resultados, pois na escola se efetua ajustes impossíveis de serem feitos através de um programa, como por exemplo, alunos que são apanhados em locais distintos de suas residências por combinarem com o motorista; ônibus com lotação excedida, e outros ajustes manuais que reduzem consideravelmente as distâncias percorridas pelos ônibus.
- Porém, a maior das dificuldades foi, sem dúvida, o fato do processo ser muito dinâmico e ter que sofrer muitos ajustes durante o ano. As atualizações são muito frequentes, de 2 a 3 por semana. Estas



atualizações se referem à alunos que mudam de endereço, alunos novos e que saem da escola ou deixam de utilizar o transporte escolar.

As alterações vão sendo realizadas, procurando sempre, minimizar as alterações nos roteiros dos ônibus.

Apesar de todas estas dificuldades, observou-se uma melhoria que não pode ser desprezada, pois reflete diretamente nos custos da empresa e na satisfação dos usuários.

## **6.2 Sugestões para futuros trabalhos**

Como sugestão para futuros trabalhos ou para as próximas aplicações pode-se citar os seguintes tópicos:

- Considerar primeiramente a estrutura da cidade com relação à avenidas movimentadas como Br's, vias rápidas e outras, para evitar que um veículo fique atravessando estas rodovias várias vezes durante um trajeto.
- Considerar em cada quadra somente um ponto de parada, ou seja aglutinando todos os pontos de parada da quadra num único ponto de forma a evitar que dois ou mais ônibus apanhem crianças de uma mesma quadra.
- Considerar os sentidos das ruas, mãos e contra-mãos para que o roteiro fornecido aos motoristas seja mais preciso.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BALLOU, R. H., *Logística Empresarial*. São Paulo, Editora Atlas, (1992).
- [2] BODIN, L., GOLDEN, B., ASSAD, A., BALL, M., *Routing and Scheduling of Vehicles and Crew - The State of The Art*, Journal of Computer and Operations Research, Vol.10, N°2, (1983),
- [3] BODIN, L., GOLDEN, B., *Classification in Vehicle Routing and Scheduling*. College of Business and Management, University of Maryland at College Park. Networks, Vol.11 97-108, (1981).
- [4] CHRISTOFIDES, N., EILON, S., *Na Algorithm for the Vehicle-Dispatching Problem*, Imperial College of Science and Technology, Operational Research Quarterly, Vol.20 N°3, (1969).
- [5] CHRISTOFIDES, N., *Graph theory - An Algorithm Approach*. New York, Academic Press Inc., (1975).
- [6] CHRISTOFIDES, N., MINGOZZI, A., TOTH, P., *Exact Algorithms for the Vehicle Routing Problem, Based on Spanning Tree and Shortest Path Relaxations*. Mathematical Programming. (1981).
- [7] CHRISTOFIDES, N., MINGOZZI, A., TOTH, P., *State-Space Relaxation Procedures for the Computation of Bounds to Routing Problem*. Networks, John Wiley & Sons. Vol.11. (1981).
- [8] CLARKE, G., WRIGHT, J. W., *Scheduling of vehicles from a Central Depot to a Number of Delivery Points*, Operations Research, Vol.12, N°4, (1964).
- [9] DESROCHERS, M., DESROSIERS, J., SOLOMON, M., (1991), *A New Optimization Algorithm for the vehicle Routing Problem with Time Windows*, Gérard and École Polytechnique, Montreal, Canadá, (1991).
- [10] FISCHER, M. L., JAIKUMAR, R., *A generalized Assignment Heuristic for Vehicle Routing*, Networks, Vol. 11 109-124, John Wiley and Sons, Inc.,(1981).
- [11] GASKELL, T. J., *Bases for Vehicle Fleet Scheduling*. Royal College of Advanced Technology, Salfordt. Operational Research. Quarterly Vol. 18 N° 3. (1967).
- [12] GILLETT, B. E., MILLER, L. R., *A Heuristic Algorithm for the Vehicle-Dispatch Problem*, Bowling Green State University, Bowling Green, Ohio, (1971)

- [13] GOLDEN, B., MAGNANTI, T., NGUYEN, *Implementing Vehicle Routing Algorithms*. Networks Vol.7, 113-148, (1977).
- [14] GOLDEN, B., BODIN, L., DOYLE, T., STEWART, W., JR., *Approximate Traveling Salesman algorithms*, University of Maryland, College Park, Maryland, (1979).
- [15] GRACIOLLI, O. D., *Planejamento de Roteiros de Veículos Coletores de Resíduos Sólidos de Serviço de Saúde*. Dissertação submetida à Universidade Federal de Santa Catarina para obtenção do Grau de Mestre em Engenharia. Florianópolis. (1994).
- [16] HOFFMAN, A. J., WOLFE, P., *The Traveling Salesman Problem*. IBM Thomas j. Watson Research Center, yorktown Heights. (1985).
- [17] LAWLER, E. L., LENSTRA, J. K., RINNOOY KAN, A. H. G., SHMOYS, D. B., *The Traveling Salesman Problem*. Jhon Wiley & Sons. New York, (1984).
- [18] LIN, S., KERNIGHAN, B. W., *An Effective Heuristic Algorithm for the Traveling Salesman Problem*, Bell Telephone Laboratories, Incorporated, Murray Hill, N. J., (1971).
- [19] MCLAUGHLIN, F. S., Pickhardt R. C., *Quantitative techniques for Manegment Decisions*. Boston, Houghton Mifflin Company, (1979).
- [20] NOVAES, A. G., *Sistemas Logísticos - Transporte, Armazenagem e Distribuição Física de Produtos*. Edgard Bucler Ltda. (1989).
- [21] PARAÍBA, L. C., FERNANDES, J. F. R., ANDO, A. S., *Um Algoritmo Heurístico de Construção Paralela para o problema do M-Caixeiro viajante*. Campinas. UNICAMP, (1990).
- [22] RONEN, D., *Perspectives on Pratical Aspects of Truck Routing and Scheduling*. European Journal of Operational Research 35, 137-145. (1988).
- [23] RUSSEL, R. A., *An Efective Heuristic for the M-tour Traveling Salesman Problema with Some Side Conditions*. University of Tulsa - Tulsa/Oklahoma. Operations Research, Vol.25, N°3. (1977).
- [24] WATERS, C. D. J., *A Solution Procedure for the Vehicle-Scheduling Problem Based on Iterative Route Improvement*. University of Strathclyde, Glasgow. Op. Res. Soc. Vol.38, N° 9, pp 833-839,(1987).
- [25] WURMANN, K., *Desarrollo de Modelos Matemáticos Aplicados Al Transporte en el Sector Forestal.*, Revista Ingenieria de Sistemas, Vol. 10 N°2, (1993).

