WAN YU CHIH
Eng. Naval, Escola Politécnica da USP, 1979
Economista, Faculdade de Economia e Administração da USP, 1983

# INFLUENCIA DOS CUSTOS FIXOS E VARIAVEIS NA ROTEIRIZAÇÃO DE FROTAS DE VEICULOS DE CAPACIDADES VARIADAS

Dissertação apresentada à Escola Politécnica da USP para a obtenção do Título de Mestre em Engenharia.

São Paulo, 1987

Orientador: Prof. Dr. Marco Antonio Brinati
Professor Livre-Docente do Departamento de Engenharia Naval da EPUSP.

# AGRADECIMENTOS

Ao Prof.Dr. MARCO ANTONIO BRINATI desejo prestar meus sinceros agradecimentos pela sua dedicada orientação e apoio durante todo o programa de mestrado, bem como pelas suas valiosas idéias e sugestões. Sem a sua participação não teria sido possível a realização deste trabalho.

Aos meus pais, pelo incentivo e estímulo dado aos estudos desde os meus primeiros passos.

Ao Hamilton, pelo gosto e motivação que me inspirou pelos modelos matemáticos e pela informática.

À minha querida esposa por todo o apoio dado ao longo do trabalho.

# RESUMO

Neste estudo examina-se a influência dos custos fixos e variáveis na determinação de roteiros de veículos de diferentes capacidades para entregas em zona urbana.

A primeira parte do trabalho ocupa-se em traçar, com base em ampla pesquisa bibliográfica, um painel abrangente do problema, mostrando suas principais características, as diversas formulações matemáticas, os métodos exatos, com suas limitações, e os algorítmos heurísticos para sua solução.

Na segunda parte, apresenta-se um procedimento desenvolvido com o objetivo de minimização do custo total sujeito às restrições de capacidade dos veículos, tempo máximo de jornada e demanda dos pontos de entrega. A heurística proposta é uma modificação do algorítmo do roteiro gigante SGT de Golden et alii..

Os resultados obtidos mostram que o modelo proposto aproxima melhor as soluções em relação aos roteiros que seriam gerados na prática, pois consideram custos específicos por tipo de veículo, ao contrário da maioria dos modelos existentes. Além disso, o modelo permite obter soluções que utilizam veículos menores em múltiplas viagens por dia, ao invés de veículos maiores alocados a uma única rota.

# ABSTRACT

This study analizes the effect of fixed and variable costs on the routing of a variable size vehicle fleet for delivery in a urban zone.

The first part of the study is intended to draft, through a vast bibliography research, an wide scenery of the vehicle routing problem, showing its main features, the different mathematical formulations, the exact methods, with their limitations, and the heuristic algorithms for its solution.

In the second part one presents a heuristic method developed with the aim of minimizing the total vehicle routing cost subject to vehicle capacity, maximum travel time, and delivery point demand constraints. The heuristic is a modification of the single giant tour - SGT algorithm proposed by Golden et al..

The results obtained mean that the proposed model produces better aproximatations of the routes that would be observed in pratice, due to the consideration of the specific vehicle costs by type, than the majority of the existing models. Besides this the model permits the utilization of smaller vehicles in shorter routes instead of greater vehicles in an unique route.

SUMARIO

INDICE DAS TABELAS

# CAPITULO 1

# INTRODUÇAO

Apresenta-se neste Capítulo uma breve descrição da situação atual da utilização de processos sistematizados de roteirização de veículos no Brasil, mostrando que pouco se fez nesta área ainda, mas que pode se esperar um grande avanço em função da popularização da informática. E apresentada também a questão da influência dos custos veiculares na determinação da frota e dos roteiros. Em seguida, em função do panorama exposto, são apresentados os objetivos do presente estudo e a estrutura de apresentação do trabalho, indicando-se o conteúdo dos seus capítulos.

## 1.1 Contexto Atual

Diariamente, milhares de frotas de veículos são roteirizadas com o objetivo de atender às entregas urbanas, distribuindo diversos tipos de produtos e obedecendo a critérios de economicidade, disponibilidade de veículos e horários de atendimento.

Dadas as diferentes características de densidade das zonas de entregas, torna-se geralmente conveniente a utilização de veículos com diversas capacidades, cada qual com diferentes custos fixos e variáveis. Periodicamente, as frotas devem ser revistas, adequando-as em número e composição por tipos.

Como nas indústrias o faturamento é realizado diariamente, existe uma limitação de tempo para a execução da roteirização, que deve ser feita em tempo hábil para que o carregamento possa ser realizado nas primeiras horas da manhã.

A maioria das empresas no Brasil emprega roteiristas para a execução manual desta tarefa, separando e agrupando pedidos baseados na sua experiência. Pouco se pode saber a respeito do grau de eficiência dos roteiros elaborados dada a inexistência de parâmetros que possibilitem a comparação. Por outro lado, raramente se tem notícia de empresas que estejam utilizando algum sistema automatizado de roteirização baseado em algorítmo. Apesar do interêsse que muitos profissionais da área têm no assunto, pouco se tem feito no Brasil.

Um levantamento realizado em 1986 nos EUA pela Arthur Andersen [25] sobre softwares aplicativos existentes para a área de distribuição física mostrou que existiam 91 softwares sobre roteirização e programação de tráfego. Deste número, 44 eram voltados para computadores de grande porte, 22 para minicomputadores, e 25 para microcomputadores. É contrastante observar que no Brasil existe um único software (para computador de grande porte), que é o VSPX da IBM [28], desenvolvido nos anos 60. Pode-se especular que a utilização de microcomputadores nesta atividade é um fator que possivelmente acelerará a sua disseminação no Brasil, tão logo haja softwares disponíveis no mercado.

Um aspecto que chama a atenção é que se pode constatar, através da literatura, que praticamente inexistem algorítmos que considerem os custos veiculares na roteirização e no dimensionamento de frotas de veículos com diversas capacidades. Frequentemente ocorre na prática a realização de várias viagens diárias por um mesmo veículo, que após a entrega de toda carga do primeiro embarque do dia, retornam para um novo carregamento e posterior distribuição no mesmo dia. É comum encontrar situações em que veículos pequenos (furgões de 1 t de capacidade) realizam duas ou três viagens diárias e veículos médios (caminhões de 3 t) e grandes (caminhões de 12 t) realizam até duas viagens por dia.

Isto traz à tona a seguinte questão: qual é a diferença de custo total anual entre frotas dimensionadas e roteirizadas por:

1. um modelo que desconsidera o custo fixo e o custo variável dos veículos.
2. um modelo que considera somente o custo fixo dos veículos.
3. um modelo que considera somente o custo variável.
4. um modelo que considera tanto o custo fixo como o variável individualizados por tipo de veículo.

E possível se ilustrar a importância das questões acima. Dois veículos muito utilizados na distribuição física urbana de carga sêca são a Kombi Standard de 0.8 t de capacidade e o caminhão Mercedes Benz L608 de 3.5 t. Os respectivos custos fixos e variáveis envolvidos são mostrados na tabela 1.

TABELA 1.
CUSTOS DE VEICULOS NACIONAIS DE CARGA

| | Custo Fixo | Custo Variável |
|---|---|---|
| Kombi | Cz$ 4749/mês | Cz$ 0,6665/km |
| MB L608D | Cz$ 7818/mês | Cz$ 1,1454/km |

(Valores de maio de 1986, Revista Transporte Moderno)

Admitindo que um veículo médio (MB L608D) atenda 4 clientes, cada qual com 0.8 t de carga, em cerca de 3 h e percorrendo 33 km no total, o custo mensal de operação (supondo 22 dias de operção por mês) seria:

CT veículo médio = 7818 + 1,1454*33*22 = Cz$ 8650/mês

A mesma operação poderia ser feita por um veículo pequeno (Kombi) em 4 viagens de 30 km no total, em cerca de 6 horas. O custo mensal de operação seria:

CT veículo pequeno = 4749 + 0,6665*4*30*22 = Cz$ 6509/mês

Os algorítmos que não consideram os diferentes custos veiculares dão preferência para a adoção do veículo médio para essa operação, pois não permitem avaliar alternativas de troca de tipos de veiculos em funçào do seu custo relativo. Uma forma de se contornar esse problema seria através de um pós-processamento dos resultados, trocando-se veículos nas rotas quando possível. Todavia os roteiros gerados podem ter qualidade inferior em relação a um algorítmo que tivesse o custo fixo e variável de diferentes tipos de veículos incorporado ao critério de roteirização.

Dentro do panorama apresentado, os objetivos propostos para este estudo são os abaixo destacados.

## 1.2 Objetivos do Estudo

(1) Realizar uma pesquisa bibliográfica analisando a possibilidade de considerar a participação dos custos de veículos de diversas capacidades nos diversos modelos encontrados na literatura.

(2) Formular um modelo de roteirização de veículos de diversas capacidades e custos diferentes, alocados a um depósito único, que pudesse ser implementado em microcomputador.

(3) Analisar a influência dos custos fixos e variáveis na roteirização e dimensionamento da frota para entregas em zona urbana.

(4) Estudar as importância de se considerar a possibilidade de realização de múltiplas viagens por dia, por um mesmo veículo, no custo total.

## 1.3 Estrutura do Trabalho

No Capítulo 1 apresentou-se uma introdução do trabalho, destacando a importância do assunto abordado e os objetivos do estudo.

No Capítulo 2, faz-se uma análise dos diversos tipos de problemas existentes de roteirização, os fatores e restrições envolvidos, procurando caracterizar o problema de roteirização de veículos.

Com a finalidade de servir de apoio à descrição dos diversos métodos exatos e algorítmos heurísticos a serem apresentados nos Capítulos 4 e 5, são apresentadas no Capítulo 3 as diversas formulações do problema, começando pelo clássico Problema do Caixeiro Viajante, a partir do qual são desenvolvidas as demais formulações do Problema de Roteirização de Veículos.

No Capítulo 6 são apresentados os algorítmos de Determinação do Tamanho e Composição da Frota de Golden et alii. que serviram de base para o modelo que foi desenvolvido e implementado em microcomputador para a avaliação da influência dos custos fixos e variáveis na roteirização, cujos resultados são apresentados no Capítulo 7.

Finalmente, as conclusões do estudo são apresentadas no Capítulo 8, juntamente com as recomendações para continuidade posterior.

# CAPITULO 2

# CARACTERIZAÇÃO DO PROBLEMA

Neste Capítulo apresenta-se uma classificação básica dos . Problemas de Roteirização de Veículos com as características que os descrevem e em seguida é feita a descrição de uma série de aspectos adicionais, de natureza prática, que elevam o grau de complexidade do problema.

## 2.1 Classificação dos Problemas

Um roteiro é uma sequência de pontos de entrega (ou coleta) que um veículo precisa percorrer, com início e término num depósito. Uma programação de viagem (ou sequenciamento) de um veículo é também uma sequência de pontos que um veículo precisa percorrer, com a condição adicional de ter horários pré-estabelecidos de chegada e partida (conhecido como janelas de tempo ou "time windows"), ou então, relações de precedência entre pontos a serem cumpridos. Deste modo, conforme Magnanti [33], define-se o que seriam as duas formas mais frequentes de planejamento de frotas: o Problema de Roteirização e o Problema de Programação de Veículos.

Uma outra classificação é sugerida por Golden et alii.[21], que consideram três categorias básicas de problemas descritas abaixo:

(1) Problema de Roteirização de Veículos

Parte-se de uma determinada frota pré-estabelecida com todos os veículos de uma mesma capacidade. Adota-se a premissa de que todos os veículos já foram adquiridos e que, portanto, todo o custo fixo já foi realizado. Assim, somente os custos variáveis devem ser considerados na roteirização. O objetivo do problema é minimizar a distância total percorrida pela frota.

(2) Problema de Dimensionamento da Frota

A decisão de roteirizar é geralmente precedida e condicionada pela questão do tamanho da frota, ou seja, o número de veículos a adquirir a fim de atender à demanda. Neste caso, tanto o custo fixo como o variável devem ser considerados. Supõe-se que os veículos sejam todos iguais. O objetivo é determinar o tipo de veículo e quantidade mais econômica para a distribuição.

(3) Problema do Tamanho da Frota e Composição

Supõe-se, neste caso, que os veículos a serem utilizados sejam de diversos tipos, com diferentes características de custo, capacidade e desempenho. O objetivo buscado é a determinação do número de veículos por tipo.

A rigor o problema (2) é um caso particular do problema (3), tendo sido destacado por Golden et alii. possivelmente devido aos diferentes graus de complexidades envolvidos.

Na prática, o Problema de Roterização de Veículos (ou PRV) pode ser encontrado com diversas características, de modo que diferentes abordagens se tornam necessárias para a sua resolução. Segundo uma classificação elaborada por Bodin et alii.[5], essas características são as seguintes:

A. Horário de atendimento do cliente
   1. pré-estabelecido
   2. janelas de tempo
   3. não especificado

B. Número de depósitos
   1. um depósito

2. múltiplos depósitos

C. Número de veículos

1. um veículo

2. múltiplos veículos

D. Composição da frota

1. um só tipo de veículo

2. vários tipos de veículos

E. Natureza da demanda

1. determinística

2. estocástica

F. Localização da demanda

1. em nós

2. nos arcos

3. mista

G. Rede viária

1. não direcionada

2. direcionada

3. mista

H. Capacidade dos veículos

1. imposta e todas iguais

2. imposta e diversas

3. não imposta

I. Tempo máximo de permanência na rota por veículo

1. imposto e todos iguais

2. imposto e diversos

3. não imposto

J. Custos

1. variáveis

2. fixos

K. Operação

1. coleta

2. entrega

3. mista

L. Objetivo

1. minimizar custo variável

2. minimizar custo fixo mais variável
3. minimizar o número de veículos necessário (custo fixo)

M. Outros

Os roteiros baseados em nós são os mais frequentes: distribuição de jornais e revistas em bancas, produtos de consumo para supermercados e varejo, produtos industriais para fábricas, produtos farmacêuticos para as drogarias e hospitais, etc. Os roteiros baseados em arcos ocorrem em menor número, porém, com significativa frequência: serviço de entrega postal, medição de consumo de água e luz, coleta e entrega de bujões de gás, coleta de lixo, limpeza urbana, conservação de rede elétrica, etc..

Neste estudo irá se tratar somente dos problemas relacionados com demandas localizadas em pontos. Como o problema de coleta é basicamente o mesmo que o de entrega, mudando somente o sentido do fluxo de carga, passa a se referir a partir deste momento, para efeito de simplificação da apresentação, somente ao problema de entregas.

Algumas características (além das apresentadas por Bodin et alii.) podem elevar o grau de complexidade do PRV e são apresentadas a seguir.

## 2.2 Restrições Adicionais

Schrage [42] aponta as seguintes restrições que podem caracterizar o PRV na prática:

(1) Em determinados problemas de atendimento por frequência de visitas, alguns nós ou arcos devem ser visitados um determinado número de vezes, dentro de um certo período de tempo (uma semana, por exemplo). Problemas típicos deste tipo são a coleta de lixo em zona urbana, a entrega de

combustível para postos de gasolina e visitas periódicas de vendedores. (Russell e Igo [39] desenvolveram um modelo dentro desta linha, para o problema de roteirização de entregas ao longo de uma semana).

(2) O tempo de percurso nas zonas urbanas pode variar drásticamente durante os horários de pico de tráfego, implicando em tempos de percurso variáveis para cada faixa horária. Quando o tempo de viagem de uma rota completa atravessa diversas faixas horárias em que o tempo de percurso varia, tem-se um problema de maior complexidade ainda.

(3) Um veículo pode ser caracterizado pelo seu custo operacional (fixo e variável) e se opera em rotas abertas ou fechadas. As rotas abertas são operadas por terceiros, que não retornam ao depósito após encerrado o roteiro; as rotas fechadas são operadas por veículos da frota própria, que retornam ao local de origem após o roteiro.

(4) Via de regra, a maioria das empresas têm, por política de atendimento, não dividir a carga destinada a um cliente em várias entregas, preferindo fazê-la em uma única vez. Entretanto, dependendo das cargas envolvidas e da geometria do problema, pode-se tornar interessante dividir as cargas, particularmente nos casos em que grandes volumes localizam-se mais próximas ao depósito. Com a divisão em vários veículos torna-se possível que estes façam outras entregas menores adicionais, gerando roteiros mais balanceados em termos de tempo de duração da jornada.

(5) Geralmente, a função objetivo a minimizar é a distância total dos roteiros. Na prática, pode ser preferível, por exemplo, a utilização de rotas mais longas, sobre as quais incide o custo de horas-extras, do que o emprego de um

número maior de veículos. Neste caso, a função objetivo deve considerar também o custo da hora-extra.

(6) Os tempos de percurso bem como os de parada (atendimento) são geralmente aleatórios. Se os roteiros estão sendo feitos para serem utilizados por um determinado período de tempo, então as demandas de cada cliente podem também ser tratadas como variáveis aleatórias, porque elas deverão variar de viagem a viagem. O objetivo poderá ser o de se escolher frotas que sejam soluções de compromisso entre o custo mínimo de operação e a perda de vendas por falta de veículos.

(7) Problemas de roteirização podem surgir como integrantes de um problema mais abrangente como o de localização de depósitos e centros de distribuição. A questão a ser analisada nestes casos é o nível de detalhe e precisão que deve ter o modelo.

Outros pontos relativos ao problema devem ser cuidadosamente considerados na implementação do modelo, conforme indicado no manual do VSPX da IBM [28]:

(1) Janelas de tempo para determinados pontos. Pode-se evitar através disto a realização de entregas em determinados horários onde o tráfego é bastante adverso, o estacionamento difícil, ou quando existe proibição pela Prefeitura de acesso de veículos de carga em determinados períodos, ou ainda devido a requisitos do próprio cliente.

(2) Tempo de parada fixo para todos os pontos. Serve para se considerar um tempo médio de espera, manobra e documentação igual para todos os clientes.

(3) Tempo adicional específico para determinados pontos. Serve

para se considerar um tempo de parada maior para determinados clientes, por exemplo.

(4) Tempo médio de carregamento e descarregamento por unidade de carga. Pode ser utilizado para se determinar um tempo de duração da entrega mais preciso.

(5) Horário limite inferior de início e horário limite superior de término da operação da frota. Nenhum veículo deve começar a operação antes ou terminar após os horários limites estabelecidos.

(6) Máxima duraçào da jornada. Pode ser utilizado para se evitar que se ultrapasse a jornada máxima diária de trabalho dos operadores, ou ainda quando existir ociosidade, para se balancear as cargas de trabalho entre as equipes. Quando especificada para alguns determinados veículos, pode-se fazer com que veículos mais velhos sejam alocados às rotas mais curtas e veículos mais novos às rotas mais longas, aumentando a confiabilidade nas entregas.

(7) Número máximo de pontos por rota. E uma forma adicional de se controlar a duração das jornadas. Por outro lado, serve para certos tipos de distribuição, como a de produtos refrigerados, para os quais há um número máximo permissível de abertura das portas, a fim de se evitar a deterioração dos produtos por perda de calor.

(8) Múltiplas jornadas em um dia. Alguns veículos são designados às rotas mais longas ou mais demoradas. Outros podem ser designados a um número múltiplo de rotas curtas e rápidas. Quando desejável, isto deve ser especificado adequadamente.

(9) Limitações de tamanho do veículo para determinados pontos. Podem ocorrer devido a dificuldades de acesso físico dentro

das instalações do cliente, ou em determinadas áreas centrais da cidade.

(10) Especificações de tipo de veículo para determinados pontos. Servem, por exemplo, para se vincular motoristas às zonas, ou para alocar veículos especialmente adaptados para determinadas operações, como a entrega de carga paletizada.

(11) Velocidade por tipo de veículo. Apesar de variar conforme o tipo do veículo, frequentemente utiliza-se a suposição de que seja uniforme. Todavia, este pode ser, conforme o caso, um requisito a ser considerado.

(12) Veículos multicompartimentados. Pode ser eventualmente necessário se indicar a existência de veículos com compartimentos de carga isolados (para cargas incompatíveis como carga seca e refrigerada, por exemplo). Mesmo quando isto não ocorre, pode ser interessante se utilizar esta especificação para adequar o carregamento de carga volumétrica com carga densa (como colchões com baterias para automóvel).

(13) Especificação dual da unidade de carga. As unidades de carga podem ser especificadas em termos de peso e volume. Com a especificação dual da carga, pode-se impor a restrição de capacidade máxima de peso e de volume dos veículos, a ser verificada na montagem de carga.

(14) Pedidos com baixa prioridade. Estes casos podem ser indicados de modo a serem abandonados na roterização do dia, no caso de se ter excedido a capacidade de carga. Estes pedidos poderão ser deixados para o dia seguinte ou para serem transportados por terceiros.

A qualidade dos resultados obtidos com um modelo depende, em grande parte, da qualidade dos dados de entrada utilizados. A matriz de distâncias empregada deve se aproximar ao máximo das distâncias reais percorridas sobre o sistema viário, considerando inclusive as mãos de direção. O levantamento de uma matriz desta natureza apresenta, geralmente, dificuldades devido à constante dinâmica de crescimento e mudança que as principais cidades brasileiras sofrem, que, aliadas ao seu porte, tornam o custo dessa atividade algo muito elevado. Os sucessivos avanços da informática devem possibilitar um barateamento dessa operação a medida que permitirão uma maior produtividade na montagem dos dados da rede viária, como por exemplo, através do uso de leitoras de mesa e softwares gráficos para o desenho de mapas na tela e em "plotter".

A IBM, no manual do programa VSPX [28], apresenta duas abordagens para a montagem e análise da rede:

(1) Método das Distâncias reais

Como o nome indica, baseia-se nas distâncias reais entre diversos pontos da rede viária. Isto requer que sejam especificados todos os pontos de interesse sobre o mapa (zonas, clientes, esquinas), como também os diversos caminhos viáveis entre eles. Dados os pontos e as distâncias, o computador pode determinar os melhores caminhos entre todos os pares de pontos da rede. A precisão deste método pode ser aumentada definindo-se velocidades de percurso para cada ligação entre nós.

(2) Método das Coordenadas

Somente os clientes (ou zonas) são representados como pontos localizados sobre o mapa. O método pressupõe a existência de uma grade traçada sobre o mapa definindo um sistema de coordenadas. Essas coordenadas são também utilizadas para descrever a posição das barreiras, tais como rios ou montanhas onde a passagem inexiste, ou áreas de congestionamento, onde a velocidade média de percurso é inferior à velocidade média do sistema. A matriz de distâncias é montada através do cálculo das distâncias em linha reta entre cada par de pontos, multiplicadas por um fator de sinuosidade médio para considerar os desvios realizados em relação às distâncias reais.

O primeiro método permite uma aproximação melhor das distâncias e tempos de percurso reais. Contudo, envolve, via de regra, custo de preparação e tempo de processamento mais elevados. (Mais detalhes são fornecidos no Apêndice A).

Finalmente, deve-se apontar a questão da amostragem a ser utilizada para o planejamento. Algumas empresas adotam por política, o dimensionamento da frota para atender ao pico de demanda. Outras dimensionam pela média deixando o pico para ser atendido por terceiros. Todavia, algumas situações não permitem que isso seja feito. E o caso da cobrança ou venda feita pelo próprio motorista, ou quando se trata de veículo que demande um investimento específico para o tipo de transporte a ser realizado (veículos refrigerados etc.). Após a obtenção das diversas composições de frota para os diversos dias da amostra, deve-se estabelecer uma frota que seja a mais adequada ao longo do período projetado. Do mesmo modo, para se considerar a sazonalidade, é conveniente se obter composições ideais para cada período específico do ano, e posteriormente se definir a composição mais adequada para todos os períodos. O mesmo tipo de análise pode ser necessário em alguns casos para variações ao longo dos dias da semana. O critério de decisão final pode ser a a utilização plena da frota ou o pleno atendimento da demanda.

Apesar de reconhecer a validade dos diversos aspectos apontados para a caracterização do PRV, a modelagem matemática, os métodos exatos e os algorítmos heurísticos mostrados nos capítulos seguintes conseguem incorporar, via de regra, apenas um pequeno conjunto desses aspectos.

$$\sum_{i=1}^{n} d_i \left( \sum_{j=1}^{n} x_{ijv} \right) \leqslant K_v \qquad v = 1,\ldots,NV \qquad (3.25)$$

$$x_{ijv} = 0 \text{ ou } 1 \qquad \text{para todos } i,j,v \qquad (3.26)$$

$$X \in S \qquad (3.27)$$

onde: $c_{ij}$ = custo do percurso do ponto i ao ponto j,

$x_{ijv}$ = variáveis de decisão binárias que indicam se o veículo v viaja do ponto i ao ponto j ($x_{ijv} = 1$), ou não ($x_{ijv} = 0$),

$K_v$ = capacidade do veículo v,

S = conjunto de restrições de formação de sub-rotas.

As restrições (3.21) e (3.22) garantem que cada ponto seja atendido por um único veículo. A restrição (3.23) assegura que um mesmo veículo entra e sai pelo mesmo ponto. A inequação (3.24) faz com que cada veículo saia uma única vez do depósito. A inequação (3.25) faz a limitação da capacidade $K_v$ de cada veículo. Uma última condição que deve ser imposta é a da proibição de ocorrência de sub-rotas que não contenham o depósito. Existem diversas formas para isto. Por exemplo, S poderia ser composta por restrições de formação de sub-rotas impostas sobre cada tipo de veículo. S seria então a união dos conjuntos $S_v$ definidos por:

$$S_v = \{ x_{ijv} : \sum_{i \in Q} \sum_{j \in Q} x_{ijv} \leqslant |Q| - 1 \text{ para todo} \qquad (3.28)$$
$$\text{subconjunto não vazio } Q \text{ de pontos } \{1,2,\ldots,n\}\}$$

onde: $|Q|$ = número de elementos do subconjunto Q.

Observa-se que quando NV = 1, a restrição (3.23) torna-se redundante, assim como a restrição (3.25). Consequentemente, para aplicações de um único veículo, esta formulação se reduz ao modelo usual de designação do PCV.

### 3.2.3 Formulação do Fluxo de Bens

Esta formulação combina restrições de designação do PCV com restrições de fluxo de movimentação de bens e é a seguinte:

$$\text{minimizar} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad (3.29)$$

$$\text{sujeito a} \quad \sum_{i=1}^{n} x_{ij} = 1 \qquad j = 2,\dots,n \qquad (3.30)$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad i = 2,\dots,n \qquad (3.31)$$

$$\sum_{j=1}^{n} x_{1j} \leq NV \qquad (3.32)$$

$$\sum_{i=1}^{n} y_{ijk} - \sum_{l=1}^{n} y_{jlk} = \begin{cases} d_k & \text{se } j=k \\ 0 & \text{se } j \neq 1 \text{ ou } j \neq k \end{cases} \quad \text{para todo } j,k \qquad (3.33)$$

$$\sum_{k=1}^{n} y_{ijk} \leq K x_{ij} \qquad i \neq j = 1,\dots,n \qquad (3.34)$$

$$y_{ijk} \geq 0 \qquad \text{para todo } i,j,k \qquad (3.35)$$

$$x_{ij} = 0 \text{ ou } 1 \qquad \text{para todo } i,j \qquad (3.36)$$

onde: $c_{ij}$ = custo do percurso do ponto i ao ponto j,

$x_{ij}$ = variáveis de decisão binárias e indicam se um veículo se move do ponto i ao ponto j ($x_{ij}=1$), ou não ($x_{ij}=0$),

$y_{ijk}$ = variáveis de fluxo e especificam o quanto de demanda destinada ao ponto k é transportado do ponto i ao ponto j.

K = capacidade do veículo.

As duas primeiras restrições (3.30) e (3.31) asseguram que cada ponto é atendido por um único veículo. A inequação (3.32) limita o número de veículos e (3.33) é a restrição correspondente ao balanço de massa para o fluxo de bens. A restrição (3.34) assegura que o movimento de bens de i para j não excede a capacidade do veículo e se nenhum veículo viaja de i para j ($x_{ij}=0$) então nenhum bem é transportado naquela ligação.

Segundo Magnanti, para estender esta formulação ao caso da frota não homogênea, pode-se substituir (3.30) a (3.32) por (3.21) a (3.24) e repetir a restrição (3.34) NV vezes, colocando $K_v\ x_{ijv}$ no lugar de $K\ x_{ij}$.

As formulações apresentadas até o momento consideram somente as restrições de demanda dos pontos e da frota. Para tornar a modelagem mais completa são necessárias, na prática, a consideração de algumas restrições adicionais, que são apresentadas a seguir.

### 3.2.4 Restrições Adicionais

Bodin et alii. [5] apontam algumas restrições do problema que podem ocorrer na prática e que elevam o grau de complexidade da modelagem:

(1) Para se considerar a limitação do tempo de duração da jornada de trabalho, ou tempo de permanência na rota, pode-se incluir a seguinte restrição nas formulações anteriores:

$$\sum_{i=1}^{n} t_{iv} \sum_{j=1}^{n} x_{ijv} + \sum_{i=1}^{n} \sum_{j=1}^{n} t_{ijv}\ x_{ijv} \leq T_v \quad (3.37)$$

$$v = 1,\ldots,NV$$

onde: $T_v$ = tempo máximo de permanência na rota para o veículo v,

$t_{iv}$ = tempo de parada do veículo v no nó i ($t_{1v} = 0$),

$t_{ijv}$ = tempo de viagem de um veículo v do nó i ao nó j.

(2) Pode-se incorporar restrições de janela de tempo no modelo

de roteirização. Define-se $a_j$ como o instante de chegada no nó j. As restrições de horário limite inferior ($a_j min$) e horário limite superior ($a_j max$) podem ser representadas pelas seguintes equações não lineares:

$$a_j = \sum_{v=1}^{NV} \sum_{i=1}^{n} ( a_i + t_{iv} + t_{ijv} ) \; x_{ijv} \quad j = 1,\ldots,n \qquad (3.38)$$

$$a_1 = 0 \qquad (3.39)$$

$$a_j min \leqslant a_j \leqslant a_j max \qquad j = 2,\ldots,n \qquad (3.40)$$

As restrições não lineares acima podem ser substituídas pelas seguintes restrições lineares:

$$a_j \geqslant ( a_i + t_{iv} + t_{ijv} ) - ( 1 - x_{ijv} ) \; T \qquad (3.41)$$

$$a_j \leqslant ( a_i + t_{iv} + t_{ijv} ) + ( 1 - x_{ijv} ) \; T \qquad (3.42)$$

para todo i,j,v

onde: $T = max_v \{ T_v \}$.

(3) As formulações do PRV apresentadas acima consideram uma disponibilidade limitada de veículos. Para se formular um problema com disponibilidade ilimitada deve-se substituir, na formulação do fluxo de veículos, NV por NT = número de tipos de veículos nas expressões (3.23) a (3.25), e v por k como indicador do tipo de veículo utilizado nas expressões (3.20) a (3.26).

(4) A formulação de programação inteira do PRV pode ser facilmente alterada para incorporar múltiplos depósitos, segundo Golden et alii.[23]. Denominando de nó 1,2,...,M, aos M depósitos, obtém-se a formulação desejada substituindo-se os índices das restrições (3.21) e (3.22) para j = M+1,...,n (no lugar de j = 2,3,...,n), e a

restrição (3.24) por

$$\sum_{i=1}^{M} \sum_{j=M+1}^{n} x_{ijv} \leqslant 1 \qquad v = 1,\ldots,NV \qquad (3.43)$$

$$\sum_{p=1}^{M} \sum_{i=M+1}^{n} x_{ipv} \leqslant 1 \qquad v = 1,\ldots,NV \qquad (3.44)$$

Cabe para finalizar tecer alguns comentários sobre as alterações que seriam necessárias para se considerar o custo fixo e variável de veículos de capacidades diferentes nas formulações do PRV apresentadas. Para se considerar o custo fixo de cada tipo de veículo seria necessário se incluir na função objetivo uma parcela de custo relativa ao custo fixo dada $\sum_{v=1}^{NV} F_v$ onde $F_v$ seria o custo fixo (diário, se for o caso) do veículo do tipo v. Para se incluir o custo variável distinto por tipo de veículo, basta se fazer $c_{ijk}$ toda vez que aparece o parâmetro $c_{ij}$ nas formulações apresentadas do PRV. O parâmetro $c_{ijk}$ seria o custo de percurso do veículo de tipo k entre os pontos i e j, possível de se obter simplesmente multiplicando a distância do percurso pelo fator de custo quilométrico. (No caso da formulação da partição de conjunto se faria $c_{jk}$ que seria o custo variável do roteiro j, no lugar de $c_j$).

# CAPITULO 4

# METODOS EXATOS

Os métodos exatos para a resolução do Problema de Roteirização de Veículos (PRV) são derivados, via de regra, dos mesmos métodos empregados para o Problema do Caixeiro Viajante (PCV). O PCV é um problema clássico que possui uma formulação bastante simples, mas que, entretanto, é de difícil resolução e sempre desafiou os estudiosos do assunto. Por este motivo apresenta-se, em primeiro lugar, uma síntese em ordem cronológica do desenvolvimento dos métodos exatos para o PCV, para em seguida tratar dos métodos para o PRV.

## 4.1 Problema do Caixeiro Viajante

Dantzig et alii. [13] em 1954, propuseram um método para resolução do PCV por meio de programação linear, relaxando algumas restrições de formação de sub-rotas. O método foi proposto para ser utilizado manualmente, através de representações gráficas, sendo aplicado a um problema de 49 pontos.

Flood [18], em 1955, foi o primeiro a tentar abordar o problema como um problema de designação, sem no entanto chegar a formular um algorítmo.

Little et alii. [32] formularam, em 1963, um procedimento para a resolução exata do PCV através do método de "branch-and-bound". Inicialmente, relaxam-se as restrições de valores 0-1 das variáveis do PCV. Parte-se de um resultado obtido por programação linear e, em cada estágio, uma variável (com valor não inteiro) é selecionada e seus dois valores inteiros mais próximos (superior e inferior) são determinados, bifurcando assim o conjunto corrente de soluções viáveis em dois outros subconjuntos. Através da solução de um problema de designação avalia-se o limite inferior em cada um dos subconjuntos. O problema de designação não é resolvido enquanto o limite inferior não for obtido (cujo valor é o mesmo para cada um dos dois subconjuntos). O processo de bifurcação continua até se encontrar uma solução viável, caso contrário o subconjunto do estágio corrente é reprocessado. Lawler e Wood [29] também estudaram extensivamente a aplicação de métodos de "branch-and-bound" para o PCV.

Christofides [9], em 1966, procurou obter limites inferiores da função objetivo do PCV para a sua utilização como ponto de partida nos algorítmos de "branch-and-bound". Aplicando o método em 14 problemas simétricos e assimétricos, Christofides concluiu que os valores encontrados do limite eram somente, em média, 4% menores que a solução ótima do PCV. Christofides et alii. [8] utilizaram essa mesma abordagem para a resolução do PCV por programação dinâmica.

Bellmore e Nemhauser [4], em 1966, apresentaram um levantamento dos métodos exatos para o PCV, tendo concluído que os métodos baseados em programação dinâmica eram os mais indicados para problemas pequenos de até 13 nós. Para problemas maiores de até 70 nós, indicavam um método baseado no procedimento de "branch-and-bound". Para problemas que não pudessem ser resolvidos por este último, recomendava-se um algorítmo heurístico (3-opt de Lin).

Até os anos 70, a abordagem exata era considerada inadequada para a resolução de problemas encontrados na prática, seja por limitações de tempo ou de capacidade de armazenamento de dados. Todavia a partir dessa época começaram a surgir resultados bastante animadores para o PCV.

Parker e Rardin [38] observaram que novas abordagens para o problema foram surgindo e atualmente problemas de tamanho bem superiores aos manuseados anteriormente podem ser hoje resolvidos rotineiramente. Contudo, observam que parte deste progresso também é devido aos avanços dos equipamentos de computação. Como o mérito dos métodos de PCV é geralmente avaliado pelo tamanho do problema que podem resolver, Parker e Rardin alertam sobre a questão de que os resultados estão sempre sujeitos a desvios devidos à configuração dos problemas e às variações das eficiências dos equipamentos e programas utilizados.

Uma abordagem desenvolvida por Held e Karp [26],[27], em 1969, baseou-se na formulação do "minimum spannig tree" para derivar um limite próximo para as variáveis do problema. Parker e Rardin observam que esta abordagem é superior às demais para problemas simétricos, enquanto que o procedimento de eliminação de sub-rotas é claramente melhor para problemas assimétricos.

Balas e Christofides [1] formularam, em 1979, um método incorporando algumas caracteristicas do tratamento de eliminação de sub-rotas e das técnicas de dualidade lagrangeana. Balas e Christofides realizaram testes computacionais para problemas assimétricos de 200 a 325 nós em tempos bastante pequenos. Parker e Rardin observaram que o método de Balas e Christofides é o melhor conhecido para problemas assimétricos.

Miliotis [34],[35] adotou, em 1975, uma abordagem de programação inteira partindo do princípio de que existem rotinas de programação linear que permitem a incorporação de novas restrições para um novo reprocessamento logo em seguida. Os valores não inteiros são eliminados por rotinas de "branch-and-bound" e planos de corte e, em seguida, a solução é testada quanto à viabilidade da solução. Caso a solução seja inviável, aplica-se uma rotina para geração das restrições violadas e, em seguida, submete-se novamente o problema à rotina de programação linear até que uma solução viável seja obtida. Ao contrário dos métodos de Little et alii. [32] e de Held e Karp [27],[28], que não geram soluções não inteiras, o algorítmo de Miliotis permite a ocorrência de valores não inteiros. Quando isto ocorre, o problema é bifurcado em dois novos sub-problemas, utilizando aproximações inteiras das variáveis. Segundo Miliotis [35], os tempos de processamento demandados por este método são significativamente menores quando se usa o algorítmo do plano de corte ao invés do "branch-and-bound" e que por sua vez são bem menores que os obtidos por Held e Karp [28].

Em 1980, Crowder e Padberg [11] relataram excelentes resultados com a aplicação de planos de corte para gerar restrições que impeçam a formação de sub-rotas, através de uma abordagem semelhante à de Miliotis, tendo resolvido um problema simétrico de 318 pontos. O método proposto opera em dois estágios. No primeiro, um grande número de planos de corte são gerados e anexados à formulação de designação. No segundo estágio, é aplicado o método de "branch-and-bound". Se o resultado desse segundo estágio for um roteiro, o processo termina. Caso contrário conclui-se que se trata de um sistema de sub-rotas. Um novo corte é realizado e o estágio dois é repetido. O método foi implementado através dos softwares MPSX de programação linear e MIP de programação inteira por "branch-and-bound", ambos da IBM. Parker e Rardin observam que o sucesso do método é garantido pelos cortes gerados no primeiro estágio, resultando numa aproximação muito boa do problema original pelos resultados obtidos por programação linear. Assim, os tempos de processamento para o primeiro estágio são significativamente maiores do que para o do segundo estágio. Este foi o maior PCV simétrico resolvido por um método exato, até aquela data.

Parker e Rardin em [38] finalizam concluindo que o PCV dificilmente será algum dia considerado um problema de resoluçào fácil, como passaram a ser alguns outros problemas de natureza combinatória. Neste sentido, é de se esperar que os métodos heurísticos continuarão a ser muito estudados e utilizados.

## 4.2 O Problema de Roteirização de Veículos

Ao contrário do PCV que pode ser resolvido, conforme visto acima, com relativo sucesso através dos métodos exatos, o PRV apresenta dificuldades bem maiores devido à maior complexidade da sua formulação, com um maior número de equações e variáveis, tornando o trabalho de resolução relativamente bem maior em termos computacionais.

### 4.2.1 Restrições Computacionais

Nos casos práticos, o tamanho de cada um dos modelos formulados no Capítulo 3 excede, segundo Magnanti, às capacidades computacionais de qualquer software de programação inteira genérica. Por exemplo, um problema de 100 pontos e 5 veículos gera 100 restrições e $2^{100}$ variáveis no modelo de partição de conjunto, 710 restrições e 50000 variáveis no modelo de fluxo de veículos (mesmo sem considerar as restrições para evitar sub-rotas), 20301 restrições 10100 variáveis discretas e cerca de 1 milhão de variáveis contínuas no modelo de fluxo de bens.

Torna-se compreensível o fato de pouco ter sido feito em termos de resolução por algoritmo exato, principalmente na fase embrionária do desenvolvimento das técnicas de programação inteira. Golden et alii. [23] fazem referência a uma citação de Christofides (1974) de que o maior problema de roterização de veículos já resolvido através de método exato envolvia somente 23 pontos.

Christofides e Eilon [7], em 1969, foram um dos poucos a tentarem solucionar o PRV através de método exato. Para isto, basearam-se no algorítmo de "branch-and-bound" de Little et alii. [32] para o PCV e utilizaram a forma clássica de conversão do PCV múltiplo num PCV simples. M depósitos artificiais foram criados e localizados na mesma posição do original, tantos quanto eram os veículos necessários. Como não se conhecia o número de veículos de antemão, o problema foi resolvido para diversos valores de M, e a melhor solução foi adotada. M possui um limite inferior que foi o ponto de partida do processo de busca:

$$M \geqslant \sum_{i=1}^{n} d_i/K$$

onde: $d_i$ = demanda do ponto i,
K = capacidade do veículo utilizado.

O algorítmo de "branch-and-bound" de Christofides e Eilon foi utilizado somente em dois problemas de 6 e 13 pontos por restrições de memória e tempo de processamento.

Segundo Magnanti [33], o PRV possui uma natureza combinatória que sugere, ao menos em princípio, que poderia ser formulado e resolvido como um problema de programação inteira. Porém, raros são os trabalhos publicados que se baseiam em técnicas de resolução exata do problema. Ao contrário, observa-se, históricamente, que as soluções desenvolvidas têm-se baseado quase totalmente em algorítmos de natureza heurística.

Magnanti ao analisar a evolução dos métodos de resolução desenvolvidos, discute vários aspectos da questão tais como: o papel desempenhado pela otimização no contexto da roteirização e programação de veículos; as circunstâncias limitadas nas quais a otimização provou ter sucesso; se a noção de solução ótima seria atualmente um ideal ultrapassado que deve ser abandonado em favor de uma solução menos rigorosa, porém mais fácil de ser obtida; e se a roteirização de veículos difere de outros problemas de otimização combinatória.

O PCV é o integrante básico (e ao mesmo tempo a versão mais estudada) do PRV. A roteirização de um único veículo através de uma série de pontos de demanda designados a ele é um componente essencial à maioria das aplicações de roteirização de veículos. Poderia se esperar assim que os algoritmos de roteirização de veículos fossem evoluir e progredir significativamente da mesma forma como ocorreu com os algoritmos do PCV. Porém, isto não ocorreu. E notável a frequência com que novas técnicas em otimização combinatória foram surgindo para a resolução do PCV, ao passo que raras foram as vezes em que puderam ser aplicadas à roteirização de veículos.

O sucesso dos procedimentos baseados em otimização, ainda segundo Magnanti , tem sido limitado em parte devido ao fato de que, até recentemente, os algoritmos se baseavam somente em versões especializadas de métodos de "branch-and-bound" e técnicas de planos de corte. Pouco se havia feito, utilizando-se os recentes avanços da programação inteira (como as técnicas de relaxação lagrangeana).

Apesar das aparentes dificuldades em se resolver o PRV através de um método exato, Magnanti abre algumas perpectivas para a questão, conforme se descreve a seguir.

### 4.2.2 Perspectivas Futuras

Magnanti [33] afirma que alguns tipos de planejamento de frotas são bem adequados à modelagem exata (geralmente aqueles formulados como um fluxo em rede bem estruturado), ao passo que outros, particularmente os que envolvem a roteirização de veículos, são consideralvelmente mais difíceis de se resolver por método exato. Para esta categoria, os algoritmos heurísticos têm provado ser uma alternativa atraente. Geralmente são mais fáceis de se entender e consequentemente mais rapidamente aceitos pelo pessoal envolvido na sua utilização. São menos sofisticados e, portanto, mais fáceis de se programar e manter em computador. Além disso, são geralmente considerados eficazes para a resolução de uma gama larga de problemas práticos, fornecendo soluções usualmente aceitas como boas ou razoáveis. Estes argumentos sugerem que os algoritmos heurísticos continuarão sendo um instrumental bastante utilizado para análise de problemas de planejamento de frotas. Todavia, ainda segundo Magnanti, existem algumas razões para se preferir soluções exatas e se esperar uma crescente utilização dos métodos exatos:

(1) Custo-benefício

Em algumas situações, como na roteirização de grandes navios petroleiros, o custo de implementação e utilização de métodos exatos com a garantia de otimalidade é mais do que justificado pelos grandes valores envolvidos. Considerando que os custos de processamento eletrônico de dados devem continuar a cair e que os custos de transportes devem continuar a se elevar, os métodos de otimização devem se tornar crescentemente mais viáveis econômicamente. Com a popularização cada vez maior da informática é de se esperar que se utilize, cada vez mais, análises se sensibilidade para tomada de decisão, fomentando o uso de métodos exatos.

(2) Análise de sensibilidade

Frequentemente, a validação dos modelos, o estudo de políticas e demais análises demandam conhecimento sobre a sensibilidade do modelo aos parâmetros do problema. Os métodos heurísticos, apesar de amparados por análises de margem de erro, são bem menos úteis para este propósito. Torna-se difícil discernir se qualquer mudança na função objetivo é devido à mudança do parâmetro ou de alguma peculiaridade da heurística empregada.

(3) Desenvolvimento de novas técnicas.

Apesar de que algumas categorias de problemas sempre continuarão a desafiar as capacidades computacionais e as limitações de custo, novos avanços tecnológicos associados aos avanços na área de computação permitirão um maior desenvolvimento de novas técnicas de resolução por método exato.

(4) Heurísticas baseadas em otimização.

Espera-se que cada vez mais os métodos exatos venham a ser utilizados como componentes de processos heurísticos (ver exemplos no ítem 5.2.5 do capítulo seguinte).

Para encerrar este capítulo cabe apresentar uma colocação muito interessante de Magnanti [33] sobre a importância dos métodos exatos. Ele aponta que um argumento frequentemente utilizado contra os métodos exatos, é o de que o esforço dispendido para se determinar uma solução ótima é geralmente perdido uma vez que o próprio modelo, bem como os dados utilizados, são apenas aproximações da realidade, justificando-se assim o emprego dos métodos heurísticos ao invés dos exatos. Magnanti observa que é importante compreender que um modelo exato aproxima a representação de um problema real enquanto que um algorítmo heurístico aproxima a resolução do modelo, compondo erros bem maiores desta forma. Apesar dos modelos exatos serem mais difíceis de se resolver, eles ainda fornecem a melhor aproximação do problema real.

# CAPITULO 5

# METODOS HEURISTICOS

Os métodos heurísticos geralmente possibilitam a resolução de problemas de roteirização em tempos de processamento relativamente bem menores que os métodos exatos, fornecendo soluções com precisão aceitável.

Por motivos didáticos, bem como para seguir a evolução cronológica do desenvolvimento dos diversos algorítmos heurísticos, irá se focalizar inicialmente os algorítmos do Problema do Caixeiro Viajante (PCV) para depois apresentar os do Problema de Roteirização de Veículos (PRV).

## 5.1 O Problema do Caixeiro Viajante

Considerando que os métodos exatos de resolução de PCV não são, em geral, aplicáveis a problemas de grande porte, por razões computacionais, uma variedade de procedimentos heurísticos foi sendo desenvolvida. Existe uma multiplicidade de algorítmos heurísticos propostos para a resolução de grandes PCV. Porém, segundo Golden et alii.[22], são três as principais classes de abordagens existentes:

(1) procedimentos de Construção, que geram uma rota proxima da ótima, a partir da matriz de distâncias.

(2) procedimentos de Melhoria, que buscam encontrar úma rota melhor a partir de uma dada rota inicial.

(3) procedimentos Compostos, que geram uma rota inicial a partir de um procedimento de construção e, depois, tentam aperfeiçoar a rota, utilizando um ou mais procedimentos de melhoria.

# CAPITULO 3

# FORMULAÇOES MATEMATICAS

As representações matemáticas do Problema de Roteirização de Veículos (PRV) são em sua maioria derivadas da formulação original do Problema do Caixeiro Viajante (ou PCV). Por isso, as formulações do PCV são apresentadas inicialmente, para em seguida se apresentar as diversas formulações do PRV.

Por convenção, adota-se em todas as formulações apresentadas que o depósito é o ponto de índice 1 e que existam n-1 pontos de demanda (clientes) a serem roteirizados.

## 3.1 O Problema do Caixeiro Viajante

O PCV consiste no clássico problema de determinação do melhor caminho entre uma série de pontos a serem visitados, partindo e retornando a um mesmo local de origem. O objetivo é o de minimizar a distância percorrida.

Apresenta-se inicialmente a formulação original do problema para um único viajante: o PCV Simples. Em seguida, apresenta-se a formulação derivada para vários viajantes: o PCV Múltiplo.

### 3.1.1 O Problema do Caixeiro Viajante Simples

A formulação do PCV Simples (Golden et alii. [23]) é a seguinte:

minimizar $\quad \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \, x_{ij} \qquad (3.1)$

sujeito a $\quad x_{ij} = 1 \qquad j = 1,\ldots,n \qquad (3.2)$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad i = 1,\ldots,n \qquad (3.3)$$

$$X \in S \qquad (3.4)$$

$$x_{ij} = 0 \text{ ou } 1 \qquad i,j = 1,\ldots,n \qquad (3.5)$$

onde: $c_{ij}$ = custo (ou distância) do percurso de i a j.
$x_{ij}$ = variável inteira que indica se a ligação i a j existe ($x_{ij}$ = 1) ou não ($x_{ij}$ = 0).

S pode ser qualquer conjunto de restrições que proíbam a formação de sub-rotas e que satisfaçam às restrições (3.2),(3.3) e (3.5). Existem diversas formas de se definir esse conjunto S. Uma formulação bastante prática baseia-se na introdução de variáveis de fluxo $y_{ij}$ entre os pontos i e j, admitindo-se que n-1 unidades de um bem são embarcadas na origem e que todo nó a ser visitado demanda uma dessas unidades. Esta formulação é apresentada por Bodin et alii. [5] e é devida a Gavish e Graves:

$$\sum_{j=1}^{n} y_{ij} - \sum_{K=1}^{n} y_{ki} = -1 \qquad i = 2,\ldots,n \qquad (3.6)$$

$$y_{ij} \leqslant U\, x_{ij} \qquad i,j = 1,\ldots,n \qquad (3.7)$$

$$y_{ij} \geqslant 0 \qquad i,j = 1,\ldots,n \qquad (3.8)$$

onde: $U \geqslant n-1$

### 3.1.2 O Problema do Caixeiro Viajante Múltiplo

O PCV múltiplo é uma generalização do PCV, que vem acomodar a necessidade de se utilizar mais que um único vendedor. Neste problema, M vendedores têm que visitar n nós, de tal forma que a distância total percorrida seja a mínima. Cada vendedor precisa viajar por uma sub-rota de nós, incluindo o depósito, e todo nó exceto o depósito precisa ser visitado somente uma vez e por um único vendedor. A formulação matemática apresentada a seguir (Golden et alii. [23]) é uma extensão natural da formulação de designação do PCV:

minimizar $$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{3.9}$$

sujeito a $$\sum_{i=1}^{n} x_{ij} = \begin{cases} M \text{ se } j = 1 \\ 1 \text{ se } j = 2,\ldots,n \end{cases} \tag{3.10}$$

$$\sum_{j=1}^{n} x_{ij} = \begin{cases} M \text{ se } i = 1 \\ 1 \text{ se } i = 2,\ldots,n \end{cases} \tag{3.11}$$

$$X \in S \tag{3.12}$$

$$x_{ij} = 0 \text{ ou } 1 \qquad i,j = 1,\ldots,n \tag{3.13}$$

onde: $c_{ij}$, $x_{ij}$ e $S$ são os mesmos definidos anteriormente.

Este problema pode ser convertido em um problema equivalente ao PCV simples. Basta se criar M cópias do depósito, chamando-os de $D_1, D_2, \ldots, D_M$, cada qual conectado aos outros nós exatamente como no problema original. Os M depósitos por sua vez não são conectados entre si, ou então são ligados por arcos de comprimentos superior a

$$\sum_{i=1}^{n} \sum_{j=1}^{n} |c_{ij}| \tag{3.14}$$

Deste modo, garante-se que uma rota nunca usará um arco conectando duas cópias do depósito.

## 3.2 O Problema de Roteirização de Veículos

A primeira formulação do problema da roterização de veículos foi feita por Dantzig e Ramser [14] em 1958. Eles reconheceram que as variáveis de decisão do problema, $x_{ij}$, só poderiam assumir valores inteiros 0 ou 1. Como na época não existia ainda nenhuma técnica, segundo eles, que pudesse resolver este tipo de problema, Dantzig e Ramser procuraram a solução relaxando as variáveis $x_{ij}$, fazendo

$$0 \leqslant x_{ij} \leqslant 1$$

e aplicando em seguida o método de programação linear.

A ocorrência de quebras dos $x_{ij} \neq 0$ ou 1 foram resolvidas por tentativa e erro, tornando a solução inteira. Porém, eles reconheceram que o método não garantia a solução ótima.

Atualmente existe uma variedade razoável de modelos de programação inteira para o PRV. Contudo, segundo Magnanti [33], a maioria desses modelos são elaborados a partir de três abordagens básicas: partição de conjunto, fluxo de bens e fluxo de veículos. Magnanti denominou a primeira abordagem - partição de conjunto - de "set covering". Entretanto como as restrições são de igualdade a formulação pode ser também designada como "set partitioning".

Para a apresentação dos modelos, irá se admitir que uma frota de NV veículos, alocados a um depósito comum, precisa ser roteirizada para entregar bens a n-1 pontos de demanda com requisitos de entrega especificos, $d_2,...,d_n$ e, em seguida, retornar ao depósito. Cada modelo apresentado pode ser implementado para incorporar características de múltiplos depósitos, restrições de máximo tempo em rota, janelas de tempo etc..

As formulações do PRV são dadas por Magnanti [33] e são apresentadas de acordo com a classificação proposta por ele. Admite-se em todas as formulações, que a demanda de qualquer ponto a ser atendido é sempre menor que a capacidade de um veículo existente. Quando isto não ocorrer, pode-se, na prática, quebrá-la em várias partes menores.

### 3.2.1 Formulação da Partição de Conjunto

Esta formulação representa a abordagem de "agrupar primeiro e roterizar depois", onde cada ponto de demanda é designado aos veículos e, em seguida, cada veículo é roteirizado através dos pontos a ele designados para se determinar a sequência de entregas. Em princípio, a formulação requer a enumeração de toda possível designação dos pontos de demanda às partições associadas aos veículos.

As variáveis de decisão do problema $z_j$ são binárias 0-1 e especificam se a partição j é usada ($z_j = 1$), ou não ($z_j = 0$). Faz-se $a_{ij} = 1$ se o ponto i de demanda pertence à partição j e $a_{ij} = 0$, em caso contrário. Os coeficientes $a_{ij}$ são fixados e definidos para cada partição j. A formulação do problema é a seguinte:

$$\text{minimizar} \quad \sum_{j=1}^{J} c_j \, z_j \qquad (3.14)$$

$$\text{sujeito a} \quad \sum_{j=1}^{J} a_{ij} \, z_j = 1 \qquad i = 2,\dots,n \qquad (3.15)$$

$$z_j = 0 \text{ ou } 1 \qquad j = 1,\dots,J \qquad (3.16)$$

onde: $c_j$ = custo mínimo do roteiro de um veículo passando pelos pontos de demanda da partição j a ele designados. Este custo de roteirização é a solução de um PCV.

As restrições estabelecem que cada ponto i precisa ser designado a somente uma das partições possíveis $j = 1,\dots,J$.

Os coeficientes $a_{ij}$ devem satisfazer a toda restriçào imposta para a partição. Por exemplo, se K é a capacidade de qualquer veículo da frota (que se assume homogênea para efeito de simplificaçào), então os coeficientes $a_{ij}$ de qualquer partição devem satisfazer à restrição

$$\sum_{i=1}^{n} a_{ij}\, d_i \leqslant K \qquad (3.17)$$

Para acomodar veículos não homogêneos na frota basta realizar uma pequena alteração na formulação (3.14) a (3.16). Seja $C_v$, para $v = 1,\ldots,NV$, o conjunto de partições candidatas a serem designadas ao veículo v. Adiciona-se à formulação (3.14) a (3.16) as restrições

$$\sum_{j \in C_v} z_j = 1 \qquad v = 1,\ldots,NV \qquad (3.18)$$

que estabelecem que cada veículo é designado a somente uma única partição de clientes. As restrições (3.17) tornam-se dependentes do tipo de veículo. Assim, para qualquer partição j do conjunto de candidatos $C_v$, a capacidade $K_v$ do veículo v, substitui a constante K em (3.17), que se altera para:

$$\sum_{i=1}^{n} a_{ij}\, d_i \leqslant K_v \qquad \text{para todo } j \in C_v \qquad (3.19)$$

3.2.2 <u>Formulação do Fluxo de Veículos</u>

Esta formulação é uma extensão direta da formulação de designação do PCV. A formulação é a seguinte:

$$\text{minimizar} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{v=1}^{NV} c_{ij}\, x_{ijv} \qquad (3.20)$$

$$\text{sujeito a} \quad \sum_{v=1}^{NV} \sum_{i=1}^{n} x_{ijv} = 1 \qquad j = 1,\ldots,n \qquad (3.21)$$

$$\sum_{v=1}^{NV} \sum_{j=1}^{n} x_{ijv} = 1 \qquad i = 1,\ldots,n \qquad (3.22)$$

$$\sum_{i=1}^{n} x_{ipv} - \sum_{j=1}^{n} x_{pjv} = 0 \qquad p = 1,\ldots,n; \quad v = 1,\ldots,NV \qquad (3.23)$$

$$\sum_{j=2}^{n} x_{1jv} \leqslant 1 \qquad v = 1,\ldots,NV \qquad (3.24)$$

A seguir são apresentadas as principais heurísticas de cada uma dessas classes. Para efeito de simplificação, admite-se que a matriz de distâncias ou custos seja simétrica.

### 5.1.1 Procedimentos de Construção

a. Vizinho mais próximo (Adjacente mais próximo)

1. Iniciar a rota por qualquer nó.
2. Encontrar o nó mais próximo ao último adicionado à rota. Acrescentá-lo à rota.
3. Repetir o passo 2 até que todos os nós estejam contidos no roteiro. Então, ligar o primeiro ao último nó.

O procedimento pode ser repetido várias vezes, cada vez iniciando-se o roteiro por um nó diferente. A melhor solução encontrada será aquela a ser adotada.

b. Clarke e Wright (Método das economias)

1. Escolher um nó como ponto de partida do roteiro.
2. Criar subrotas para todos os nós existentes i e j, a partir do nó 1 de início, do tipo 1-2-1, 1-3-1, 1-4-1, etc.
3. Calcular para cada subrota formada a economia ($s_{ij}$) que seria gerada caso fosse fundida a uma outra subrota através da fórmula:

$$s_{ij} = c_{i1} + c_{1j} - c_{ij} \qquad (5.1)$$

onde: $c_{ij}$ = custo do percurso entre os pontos i è j.

4. Listar as economias em ordem decrescente.
5. Tomar o primeiro par de nós indicado na lista de economias criando a subrota 1-i-j-1.
6. Descer a lista e tomar o primeiro par de nós indicado que

contiver o ponto i ou j (o par de pontos j-m por exemplo) e aumentar a subrota para 1-i-j-m-1.

7. Seja $1-i_1-i_2-i_3-...-i_k-1$ a subrota existente em determinado estágio do procedimento. Percorrer a lista do $s_{ij}$ até encontrar a maior economia envolvendo um dos nós externos $i_1$ ou $i_k$ (nós ligados 1) e nós não pertencentes à subrota existente. Se a maior economia for $s_{j\,i1}$, aumentar a subrota para $1-j-i_1-i_2-i_3-...-i_k-1$. Se a maior economia for $s_{ik\,j}$, aumentar a subrota para $1-i_1-i_2-i_3-...-i_k-j-1$.
8. Repetir o passo 7 até esgotar todos os nós.

Do mesmo modo que para a heurística anterior, esse procedimento poderá ser repetido várias vezes inciando-se o roteiro por nós diferentes. A melhor solução será aquela que apresentar a menor distância total. (Este algorítmo foi criado originalmente para o PRV que se encontra apresentado no ítem 5.2.3 deste Capítulo).

c. Inserção

Um procedimento de construção por inserção forma a rota determinando, primeiramente, o nó ainda não pertencente à sub-rota que deve ser inserido e, em seguida, determinando o par de nós aonde a inserção deve ser realizada. Do mesmo modo que nos casos anteriores, os procedimentos de inserção descritos abaixo podem ser repetidos diversas vezes, iniciando-se o roteiro por nós diferentes (com exceção do procedimento c5).

c1. Inserção mais próxima

1. Iniciar a rota por um nó i qualquer.
2. Encontrar o nó k tal que $c_{ik}$ seja mínimo, e formar a sub-rota i-k-i.
3. Dada uma sub-rota, encontrar o nó k, ainda não pertencente à ela, mais próximo a qualquer nó da sub-rota.

4. Achar o arco (i,j) da sub-rota que minimiza a expressão

$$c_{ik} + c_{kj} - c_{ij} \qquad (5.2)$$

onde: $c_{ij}$ = custo do percurso entre os pontos i e j.
Inserir o nó k entre i e j.

5. Repetir os passos 3 e 4 até envolver todos os nós.

c2. Inserção mais econômica

Substituir os passos 3 e 4 do algorítmo anterior pelo seguinte: encontrar (i,j) na sub-rota, e um nó k não pertencente a ela, tal que

$$c_{ik} + c_{kj} - c_{ij}$$

seja mínimo. Inserir k entre i e j.

c3. Inserção arbitrária

Substituir o passo 3 do algorítmo c1 pelo seguinte: escolher arbitrariamente um nó k não pertencente à sub-rota para inserção.

c4. Inserção pelo ponto mais distante

Substituir o passo 2 do algorítmo c1 pelo seguinte: encontrar o nó k tal que $c_{ik}$ seja máximo, e formar a sub-rota i-k-i. Substituir o passo 3 pelo seguinte: Dada uma sub-rota, encontrar o nó k, ainda não pertencente à ela, mais distante a qualquer nó da sub-rota.

c5. Polígono Convexo ("Convex Hull")

1. Formar um polígono convexo envolvendo todos os pontos. O perímetro do polígono fornece a sub-rota inicial.

2. Para cada nó k ainda não pertencente à sub-rota, determinar o par de nós i e j onde k deverá ser inserido tal que

$$c_{ik} + c_{kj} - c_{ij}$$

seja mínimo.

3. De todos os conjuntos (i,k,j) encontrados no passo 2, determinar (i*,k*,j*) tal que minimize

$$(c_{i^*k^*} + c_{k^*j^*})/(c_{i^*j^*}) \qquad (5.3)$$

4. Inserir o nó k* na sub-rota entre os nós i* e j*.
5. Repetir os passos 2 a 4 até envolver todos os nós.

Bodin et alii.[5] acrescentam à relação apresentada acima os seguintes algorítmos de inserçào:

c6. Inserção pelo maior ângulo

E uma variação do algorítmo do Polígono Convexo. Consiste em se substituir os passos 2 e 3 pelo seguinte: determinar o nó k não pertencente a sub-rota, e o arco (i,k) da sub-rota tal que o ângulo formado pelos arcos (i,k) e (k,j) seja o maior possível.

c7. Inserção pela diferença multiplicada pela taxa

O procedimento é o mesmo da inserção pelo maior ângulo exceto pelo passo 2 que é substituido pelo seguinte: escolher o nó k* não pertencente à sub-rota e o arco (i*,j*) da sub-rota, tais que o produto seguinte seja o menor possível:

$$( c_{i^*k^*} + c_{k^*j^*} - c_{i^*j^*} )\times( (c_{i^*k^*} + c_{k^*j^*})/c_{i^*j^*} ) \qquad (5.4)$$

### 5.1.2 Procedimentos de Melhoria

Christofides e Eilon [7] mencionam que Flood [18] foi o primeiro a observar que o roteiro ótimo do PCV nunca se intercepta a si mesmo, por motivos geométricos facilmente compreensíveis. Segundo eles, Croes (1958) descreveu como rotas sem intersecção poderiam ser obtidas a partir de uma rota arbitrária inicial, abrindo-se 2 arcos, e reconectando os 4 pontos abertos por 2 novos arcos (chamado de "inversão" por Croes).

Christofides e Eilon observam que Lin (1965) chamou de 2-opt (2-ótimo) à rota sem intersecção que não pode ser reduzida em comprimento através da troca de pares de arcos. O termo 2-opt implica que nenhuma melhoria é mais possível de ser obtida pela troca de 2 arcos. O roteiro 2-opt não é necessariamente igual ao roteiro ótimo, sendo possível se ter diversos roteiros 2-opt para um mesmo problema. Lin mostrou que o mesmo raciocínio era válido para o caso de se trocar 3 arcos, chamando-o a solução obtida assim de 3-opt.

Prosseguindo, Lin e Kernighan chegaram ao método de determinação de rotas k-opt [36]. Para uma dada rota, existem $\binom{n}{k}$ combinações possíveis de trocas de k arcos. A aplicação do k-opt para o PRV pode ser encontrada em Russell [40] à semelhança do que foi feito com o 3-opt por Christofides e Eilon.

O procedimento de trocas de arcos gera soluções de ótimo local. Uma troca de k=2 arcos gera resultados inferiores aos gerados por k=3 ou mais. Entretanto um procedimento 3-opt requer cerca de n vezes o número de opreções que um 2-opt. Para se obter melhores aproximações da rota ótima se poderia repetir o procedimento 3-opt para diversas rotas iniciais. Computacionalmente isso pode se tornar impraticável rapidamente, dependendo do tamanho do problema. Por isso, Christofides e Eilon propuseram um método, classificado por Golden et alii. [22] como um procedimento composto, que será apresentado a seguir.

### 5.1.3 Procedimentos Compostos

O procedimento composto de Christofides e Eilon [7] para o PCV é o seguinte:

(1) começar com uma rota aleatória
(2) encontrar a rota 2-opt
(3) encontrar partir da rota gerada no passo 2 uma rota 3-opt.
(4) repetir o procedimento a partir do passo 1 diversas vezes e selecionar a melhor solução.

Bodin et alii. [5] sugerem algumas variantes para o procedimento composto apresentado acima, visando reduzir o tempo de computação sem prejudicar a qualidade dos resultados:

(1) Executar o procedimento sem o passo 2.
(2) Executar o procedimento sem o passo 3. Esta variante fornece um processamento mais rápido e resultados muito bons, não melhores, porém, que os gerados pelo procedimento composto de três etapas.
(3) Executar o procedimento composto algumas vezes usando diferentes algoritmos de construção de rotas no passo 1. Esta variante apesar de mais lenta que o procedimento de três etapas, gera melhores resultados.
(4) No passo 1 do procedimento composto, executar somente o procedimento de construção a partir de um pequeno número de nós iniciais. Executar os passos 2 e 3 a partir da melhor destas soluções.

Os algorítmos apresentados por Golden et alii. foram utilizados para resolver uma série de 8 problemas de 25 a 100 pontos. Obteve-se assim, além do comprimento da rota, o comportamento do pior caso, ou seja, a relação máxima observada entre o comprimento do roteiro obtido e o comprimento ótimo, bem como o número de cálculos computacionais necessários. Golden et alii. concluiram o seguinte:

(1) Utilizando-se procedimentos de construção foram encontradas rotas distantes 5 a 7% da optimalidade.

(2) Dentre os procedimentos de construção destacaram-se os algorítmos de Clarke e Wright, Inserção mais distante, Inserção arbitrária e Polígono Convexo. Os algorítmos do Vizinho mais próximo, Inserção mais próxima e Inserção mais econômica tiveram um desempenho muito ruim na maioria dos casos.

(3) Os procedimentos de melhoria, em particular o 2-opt e 3-opt, partindo de soluções obtidas aleatoriamente, operaram aproximadamente com a mesma eficiência e eficácia dos melhores procedimentos de construção.

(4) Os procedimentos compostos produziram soluções 2 a 3% distantes da optimalidade com alta regularidade, sendo entretanto relativamente mais demorados que os outros.

(5) Foram obtidos resultados 1 a 2% distantes da optimalidade aplicando-se o procedimento composto repetidamente.

(6) O algorítmo do Polígono Convexo acoplado ao 2-opt teve muito bom desempenho em termos de precisão e eficiência.

(7) Os algorítmos da Inserção arbitrária e Inserção mais distante foram utilizados como procedimentos de construção dentro de diversas variantes de procedimentos compostos, com excelentes resultados. Nenhuma conclusão pôde ser obtida quanto à melhor combinação. A consistência dos resultados para os diversas combinações testadas foi muito grande.

Além dos procedimentos acima apresentados, existe um tratamento bastante interessante baseado em simulação de Monte Carlo, desenvolvido por Shimizu [43], para a resolução do PCV. São empregadas diversas técnicas de redução da variância por simulação. Foi resolvido um problema de 10 nós e os resultados foram comparados com a solução por método gráfico de Barachet [2]. A técnica de "variáveis antitéticas" apresentou muito bom resultado, sendo o custo mínimo estimado apenas 3% superior que o de Barachet. Esta técnica associa faixas de números aleatórios de distribuição uniforme entre 0 e 1, às ligações entre pares de pontos de modo inversamente proporcional a sua distância. Shimizu conclui que as técnicas de redução de variância são válidas para o tipo de problema. Seria interessante se essa abordagem fosse comparada com outras técnicas heurísticas para a avaliação do seu desempenho.

Os algorítmos heurísticos do PCV servem de base para a maioria dos métodos heurísticos para o PRV, conforme se constata a seguir.

## 5.2 O Problema de Roteirização de Veículos

Segundo Bodin et alii.[5], as estratégias de resolução heurística dos problemas de roteirização podem ser classificadas em uma das seguintes abordagens:

(1) agrupar primeiro e roteirizar depois
(2) roteirizar primeiro e agrupar depois
(3) economia e inserção
(4) melhoria e troca
(5) baseada em programação matemática
(6) otimização interativa

As quatro primeiras abordagens têm sido extensivamente utilizadas no passado, enquanto que as duas últimas são mais recentes.

### 5.2.1 Agrupar Primeiro e Roteirizar Depois

Estes procedimentos formam, inicialmente, grupos de pontos a serem vinculados a cada veículo, para numa segunda etapa serem roteirizados. O algorítmo de Varredura ("Sweep") de Gillet e

Miller é baseado neste tipo de enfoque e é apresentado abaixo.

a. Algorítmo de Varredura ("Sweep") de Gillet e Miller [20]

Este algorítmo utiliza a abordagem de agrupar primeiro e roteirizar depois. Consiste em se varrer o mapa com uma semi-reta centrada no depósito, movimentando-a sempre num mesmo sentido. Toda vez que os nós varridos violarem alguma das restrições do problema retorna-se ao ponto imediatamente anterior constituindo um roteiro. A varredura tem sua contagem zerada, iniciando a partir daí um novo grupamento até se esgotar todos os nós. Numa segunda etapa, os nós pertencentes a cada grupamento são roteirizados por algum algorítmo de resolução do PCV.

Gillet e Miller fizeram uma comparação, utilizando o algorítmo heurístico de trocas 2-opt de Lin (em Golden et alii. [22]), com o algorítmo exato de Held e Karp [26] (baseado em uma técnica de "branch-and-bound"), elegendo o primeiro pelo seu melhor desempenho global. O algorítmo proposto foi comparado com o algorítmo de Gaskell [19] (abordagem de economia múltipla) e de Christofides e Eilon [7] (abordagem 3-opt), (considerados segundo Gillet e Miller como os melhores algorítmos conhecidos na época,1971), através de 7 problemas. O algorítmo de Varredura mostrou ser superior aos outros nos problemas maiores (50 a 100 pontos). Nos problemas menores (21 a 32 pontos) os resultados obtidos foram equivalentes. Além da precisão dos resultados a grande vantagem apontada é que o tempo de processamento cresce linearmente com o número de pontos (se o número de pontos por rota permanecer constante).

5.2.2 Roteirizar Primeiro e Agrupar Depois

Estes procedimentos funcionam na ordem inversa ao procedimento anterior de agrupar primeiro e roteirizar depois. Primeiro, uma rota gigante (geralmente inviável, violando algumas restrições do problema) é construída passando por todos os pontos de atendimento. Em seguida, esta rota gigante é particionada em uma série de rotas menores, porém viáveis. Christofides e Eilon [7] citam o método de Tyagi (1968), que utilizou o algorítmo do Vizinho mais Próximo para a formação da rota gigante. Golden et alii. [22] desenvolveram diversos algorítmos para o PRV a partir desta abordagem, criando a rota gigante através do procedimento 2-opt de Lin e que serão apresentados no Capítulo 6.

### 5.2.3 Economias e Inserção

Estes procedimentos constroem a solução repetindo o seguinte processo: são avaliadas, para cada ponto ainda não pertencente a nenhuma rota, as economias que seriam geradas, caso fossem inseridos na rota em construção. O ponto que apresentar a maior economia é inserido. O procedimento continua até que todos os pontos tenham sido inseridos nas rotas formadas. O algorítmo clássico deste tipo é o de Clarke e Wright [10] que encontra-se apresentado abaixo. Métodos baseados neste algorítmo para o PRV podem ser encontrados em Christofides e Eilon [7], Gaskell [19], Golden et alii. [21],[22],[23],[24], Norback e Love [36] e Yellow [45]. Russell e Igo [39] utilizaram esta abordagem com uma restrição adicional para o problema de frequência semanal de visitas.

a. Algorítmo de Clarke e Wright

Clarke e Wright desenvolveram em 1962 um algorítmo heurístico para o PRV com veículos de capacidades variadas, que passou a ser conhecido também como método de economias e inserção. O algorítmo consiste basicamente no seguinte:

1. Designar o depósito como nó 1.
2. Criar rotas para todos os nós existentes, a partir do nó 1 do tipo 1-2-1, 1-3-1, 1-4-1, etc.
3. Calcular para cada nó remanescente a economia que seria

gerada caso fosse incorporada à rota construída até o momento, através da fórmula:

$$S_{ij} = C_{i1} + C_{1j} - C_{ij} \qquad (5.5)$$

4. Tomar o nó que apresenta a maior economia e inseri-lo no roteiro, após o último nó inserido anteriormente.
5. Repetir o passo 4 até esgotar todos os nós.

Esta formulação foi chamada por Gaskell de método sequencial de economias uma vez que uma nova rota só é iniciada após encerrada a anterior.

b. Algorítmos de Economias múltiplas de Gaskell [19]

Gaskell, partindo da formulação feita por Clarke e Wright para o cálculo da economia e inserção, elaborou novos critérios, chamando-os de:

(1) Método da Economia múltipla

Parte-se da lista de economias em ordem decrescente, calculadas conforme Clarke e Wright. O arco (A-B) no topo da lista de economias é o primeiro a iniciar a rota (formando a sequência 1-A-B-1). Toma-se em seguida o segundo arco (B-C, por exemplo) da lista de economia. Se existir um ponto em comum ao primeiro arco (o ponto B no caso) e que não viole as restrições do problema, o arco é incorporado à rota existente (formando a sequência 1-A-B-C-1) e todos os demais arcos envolvendo o ponto comum (B) são eliminados da lista. Caso contrário, o arco é utilizado para se iniciar uma nova rota. Deste modo uma multiplicidade de rotas vão sendo construídas ou criadas a cada estágio. O processo continua até que todos os arcos listados tenham sido utilizados ou cancelados.

(2) Método do $\lambda$ múltiplo

O procedimento de busca de novos pontos é o mesmo do método (1), com exceção do cálculo das economias que é feito por:

$$\lambda_{ij} = S_{ij}(dm + d_{1i} - d_{1j} - d_{ij}) \qquad (5.6)$$

onde: $S_{ij}$ = economia calculada por Clarke e Wright,
$d_{ij}$ = distância do ponto i ao ponto j,
dm = média dos $d_{1i}$

(3) Método do $\pi$ múltiplo

O procedimento de busca de novos pontos é o mesmo do método (1), com excessão do cálculo das economias que é feito por:

$$\begin{aligned} \pi_{ij} &= 2d_{1j} - (d_{1j} + d_{ij} - d_{1i}) - d_{ij} \\ &= d_{1i} + d_{1j} - 2d_{ij} \\ &= S_{ij} - d_{ij} \end{aligned} \qquad (5.7)$$

Os métodos (2) e (3) são casos especiais do critério

$$M_{ij} = S_{ij} - \theta d_{ij} \qquad (5.8)$$

sendo $\theta$ um parâmetro dependente da forma do problema, que coloca maior ou menor ênfase na posição relativa de dois pontos em relação ao depósito. E interessante observar que Yellow [45] desenvolveu, praticamente ao mesmo tempo, um trabalho utilizando o mesmo enfoque de Gaskell. Golden et alii. [23] exploraram este artifício para reduzir o tempo de processamento de PRV de 1000 pontos.

Gaskell conclui que a abordagem múltipla é usualmente superior à sequencial de Clarke e Wright (ver Apêndice B).

### 5.2.5 Melhoria e Troca

O 2-opt e 3-opt de Lin (em Golden et alii. [22]) e o k-opt de Lin e Kernighan [31] são os integrantes desta classe de algorítmos para o PCV. Posteriormente, foram aplicados no PRV por Christofides e Eilon [7] (cujo trabalho é apresentado abaixo) e por Russel [39],[40]. Baseiam-se na tentativa sucessiva de trocas de arcos entre rotas diferentes, sempre observando as restrições do problema, de modo a se obter melhorias no custo total da roteirização. O procedimento continua até que mais nenhuma redução de custo seja possível. Dror e Levy [15] utilizaram um procedimento composto 2-opt e 3-opt para realizar trocas de arcos entre rotas diferentes, geradas inicialmente pelo procedimento de Clarke e Wright, para o problema da roteirização de entregas ao longo de uma semana.

a. Algorítmo de abordagem "3-opt" de Christofides e Eilon [7]

Utilizando o procedimento de roteirizar primeiro e zonear depois, Christofides e Eilon geraram uma rota gigante através do procedimento composto descrito anteriormente (ítem 5.1.3) para posteriormente particioná-lo em veículos, em função das suas capacidades. Em seguida, cada par de arcos existente (seja da mesma partição ou não) é testado quanto a melhoria que seria gerada no custo total da roteirização caso fossem trocados entre si, respeitadas as restrições do problema.

Christofides e Eilon compararam os resultados obtidos com o algorítmo de Clarke e Wright na resolução de 10 problemas de 6 a 100 pontos e concluíram que a abordagem "3-opt" é significativamente superior.

### 5.2.5 Programação Matemática

Esta abordagem abrange os algoritmos que são diretamente derivados da formulação de programação matemática do problema de roteirização. Um exemplo é o modelo de Locação e Alocação de Cooper apresentado abaixo. Um outro excelente exemplo é o modelo de Fisher e Jaikumar que também é apresentado abaixo. Neste último algorítmo, a formulação matemática do problema é primeiramente elaborada na forma exata. Dois componentes interrelacionados são identificados. Um é o problema da designação generalizada e o outro é o PCV. Os procedimentos desenvolvidos por Fisher e Jaikumar aproveitam-se do fato de que estes dois problemas têm sido estudados extensivamente e potentes abordagens de programação matemática encontram-se disponíveis. Deve-se observar que tanto o método de Fisher e Jaikumar como o de Cooper baseiam-se também na abordagem de agrupar primeiro e roteirizar depois. A etapa de agrupamento é feita por programação matemática.

a. Algorítmo de Locação-Alocação ("Location-Allocation") de Cooper

Este algorítmo é citado por Cullen et alii. [12] e foi desenvolvido por Cooper(1969). A solução é buscada iterativamente através de duas etapas. Primeiramente são localizados alguns pontos base, que atuarão como depósitos regionais espalhados pela cidade. Em seguida, os pontos de entrega são alocados a cada ponto base. Supõe-se no modelo que os veículos saem do depósito central, vão diretamente a um ponto base e, a partir daí, realizam suas entregas sempre indo do ponto base ao ponto de entrega e voltando ao ponto base sucessivamente, até atender a todos os pontos alocados àquele ponto base, retornando após isso ao depósito central. A distância total a ser minimizada nesta fase é a somatória das distâncias que ligam cada ponto ao seu respectivo ponto base.

A localizaçào dos pontos base é feita inicialmente de modo arbitrário. A alocação dos pontos de entrega aos pontos base é feita através de um algorítmo de designação 0-1 que resolva o seguinte problema de programação inteira:

$$\text{minimizar} \quad \sum_{v=1}^{NV} \sum_{i=2}^{n} \left[ (x_i-a_v)^2 + (y_i-b_v)^2 \right]^{1/2} z_{iv} \qquad (5.9)$$

$$\text{sujeito a} \quad \sum_{i=2}^{NV} z_{iv} \leq K \qquad v = 1,\ldots,NV \qquad (5.10)$$

$$\sum_{v=1}^{NV} z_{iv} = 1 \qquad i = 2,\ldots,n \qquad (5.11)$$

$$z_{iv} = 0 \text{ ou } 1 \qquad \text{para todo } i,v \qquad (5.12)$$

$$z_{iv} = 1 \qquad v = 1,\ldots,NV \qquad (5.13)$$

onde: $n$ = número de pontos de entrega mais 1,
$NV$ = número de veículos,
$K$ = capacidade do veículo (supostamente única para efeito de simplificação),
$x_i$ e $y_i$ = coordenadas do ponto de entrega $i$,
$a_v$ e $b_v$ = coordenadas do ponto base $v$,
$z_{iv}$ = variável binária que indica se o ponto de entrega $i$ está alocado ao ponto base $v$ ($z_{iv}=1$), ou não ($z_{iv}=0$).

Uma vez feita a alocação dos pontos de entrega aos pontos base, é realizada uma relocalização dos pontos base para cada grupamento de pontos de entrega. O procedimento de alocação e locação continua até que mais nenhuma melhoria na função objetivo seja alcançada. (Apesar de não estar claro no artigo, o algorítmo para a resolução do problema de localização deve ser possivelmente o do baricentro, também conhecido como método dos pesos).

Cullen et alii. não fornecem maiores indicações sobre o desempenho e eficiência do método. Contudo, pode se inferir pela sua formulação que o algorítmo deve ser bastante limitado computacionalmente, devido à resolução do problema de designação. Por outro lado, pode-se inferir, ainda, que o modelo seja muito sujeito a casos patológicos, particularmente na fase de relocalização dos pontos base através do algorítmo do baricentro. Pontos de atendimento não convenientes poderão ser ligados ao ponto base, se forem consideradas ligações em linha reta ao invés da distância real.

b. Algorítmo de Designação Generalizada de Fisher e Jaikumar [17]

Esta heurística encara o PRV como consistindo de dois componentes interrelacionados. Um é o problema da designação generalizada (que faz o agrupamento dos pontos) e o outro é o PCV (que faz a roteirização dos pontos de cada agupamento). Para que se possa entender melhor o mecanismo do algorítmo, apresenta-se abaixo a formulação do PRV reescrita de uma outra forma em relação ao que foi apresentado anteriormente no ítem 3.2.2 . (formulação do fluxo de veículos) de modo a permitir o reconhecimento das duas partes envolvidas:

minimizar $$\sum_{ijv} c_{ij}\, x_{ijv} \qquad (5.14)$$

sujeito a $$\sum_{i} d_i\, z_{iv} \leqslant K_v \qquad v = 1,\ldots,NV \qquad (5.15)$$

$$\sum z_{iv} = \begin{cases} NV & \text{se } i=1 \\ 1 & \text{se } i = 2,\ldots,n \end{cases} \qquad (5.16)$$

$$z_{iv} = 0 \text{ ou } 1 \qquad \text{para todo } i,v \qquad (5.17)$$

$$\sum_{i} x_{ijv} = z_{jv} \qquad \text{para todo } j,v \qquad (5.18)$$

$$\sum_{j} x_{ijv} = z_{iv} \qquad \text{para todo } i,v \qquad (5.19)$$

$$\sum_{i,j \in S} x_{ijv} \leqslant |S| - 1 \qquad (5.20)$$

$$x_{ijv} = 0 \text{ ou } 1 \qquad i,j = 1,\ldots,n \qquad (5.21)$$

onde: NV = número de veículos disponível,

$K_v$ = capacidade de veículo v,

$d_i$ = demanda do ponto i,

$c_{ij}$ = custo do percurso do ponto i ao ponto j,

$x_{ijv}$ = variável binária que indica se o veículo v trafega no arco de ligação entre os pontos i e j ($x_{ijv}$ = 1), ou não ($x_{ijv}$ = 0),

$z_{iv}$ = variável binária que indica se o ponto i é servido pelo veículo v ($z_{iv}$ = 1), ou não ($z_{iv}$ = 0).

As restrições (5.15) a (5.17) são as restrições do problema de designação generalizada que asseguram que cada rota começa e termina no depósito (ponto 1), que todo ponto é atendido por algum veículo e que a carga alocada a um veículo está dentro de sua capacidade. Se os $z_{iv}$ são fixados para satisfazer (5.15) a (5.17), então para um dado v, as restrições (5.18) a (5.21) definem um PCV para os pontos designados ao veículo v.

Fisher e Jaikumar observam que as restrições de capacidade dos veículos não exercem o devido papel nas regras de decisão da maioria dos modelos heurísticos desenvolvidos até o momento, devido à relativa miopia dos processos no ato da escolha da capacidade dos veículos a serem utilizados. (Christofides e Eilon [7] fizeram a mesma observação particularmente sobre o algorítmo de Clarke e Wright). Assim, problemas que possuem restrições severas de capacidade geralmente terminam com um resultado muito pobre.

O algorítmo desenvolvido por Fisher e Jaikumar resolve o problema da designação dos pontos aos veículos (5.14) a (5.17) através de uma aproximação linear da função objetivo. Em seguida, obtém-se a solução final através da roteirização dos pontos alocados a cada veículo, por meio de um algorítmo exato para o PCV.

O algorítmo proposto superou, segundo os autores, os melhores conhecidos até o momento. Além disso, segundo os mesmos, ele tem a vantagem considerar, através do problema da designação, o impacto da alocação de um ponto a um veículo em relação a todas as demais alternativas possíveis, à luz das restrições de capacidade dos veículos. Evita-se assim o problema inerente às heurísticas de designação sequenciais (heurísticas de construção), que fazem alocações iniciais "boas" levando eventualmente a alocações muito caras no final, a fim de manter a viabilidade da solução, conforme mencionado acima.

A idéia básica da abordagem apresentada é descrita pelas reformulação das expressões (5.14) a (5.21) anteriores como um problema de designação não-linear:

$$\text{minimizar} \quad \sum_{v} f(Z_v) \qquad (5.22)$$

$$\text{sujeito a} \quad \sum_{i} d_i \, z_{iv} \leqslant K_v \qquad v = 1,\ldots,NV \qquad (5.23)$$

$$\sum_{v} z_{iv} = \begin{cases} NV & \text{se } i = 1 \\ 1 & \text{se } i = 2,\ldots,n \end{cases} \qquad (5.24)$$

$$z_{iv} = 0 \text{ ou } 1 \qquad \text{para todo } i,v \qquad (5.25)$$

onde: $f(Z_v)$ = custo do roteiro ótimo do PCV para os pontos alocados ao veículo v.

A função $f(Z_v)$ é definida matematicamente por

$$f(Z_v) = \min Z_{ijv} \, c_{ij} \, x_{ijv} \qquad (5.26)$$

$$\text{tal que} \quad \sum_{i} x_{ijv} = z_{jv} \qquad j = 1,\ldots,n \qquad (5.27)$$

$$\sum_{j} x_{ijv} = z_{iv} \qquad i = 1,\ldots,n \qquad (5.28)$$

$$\sum_{i,j \in S} x_{ijv} \leqslant |S| - 1 \qquad (5.29)$$

$$x_{ijv} = 0 \text{ ou } 1 \qquad \text{para } i,j = 1,\ldots,n \qquad (5.30)$$

A função $f(Z_v)$ é extremamente complexa e não serve para o propósito em vista. Assim, a heurística baseia-se numa aproximação linear da função objetivo (5.22) dada por

$$\sum_{v=1}^{NV} \sum_{i=2}^{n} d_{iv} \, z_{iv} \qquad (5.31)$$

As distâncias $d_{iv}$ podem ser obtidas de diversas maneiras. Fisher e Jaikumar utilizaram o seguinte procedimento:

(1) Alocar NV pontos base ("seed") associados aos NV veículos. A alocação pode ser feita arbitrariamente por uma pessoa, o que torna o método bastante interessante em termos de interação do homem com a máquina, ou automaticamente conforme descrito adiante.

(2) As distâncias $d_{iv}$ são obtidas pela expressão

$$d_{iv} = \min \{(c_{1\,i} + c_{i\,ik} + c_{i\,k1}), (c_{1\,ik} + c_{ik\,i} + c_{i\,1})\} - (c_{1\,ik} + c_{ik\,1})$$

(5.32)

onde: $c_{i\,j}$ = custo do percurso entre os pontos i e j.

De acordo com Fisher e Jaikumar, existem muitas vantagens em se empregar a seleção manual dos pontos base. Segundo eles, geralmente o usuário tem um sentimento do problema e pode colocar a sua experiência a serviço do sistema computadorizado de roteirização. Isto cria uma sensação de envolvimento e controle do processo, que é fundamental para o sucesso de uma boa implementação.

Para a seleção automática dos pontos base, Fisher e Jaikumar desenvolveram um algorítmo muito semelhante à varredura de Gillet e Miller [20]. O mapa é dividido em setores proporcionais à demanda dos pontos envolvidos, centrados no depósito. Os pontos base são posicionados sobre os raios que dividem esses setores ao meio, a uma distância do centro tal que envolva 75% da demanda do setor.

O algorítmo utilizado para a solução do problema de designação,segundo Fisher, é baseado na técnica de relaxação lagrangeana. O PCV para cada grupamento de pontos resultante foi resolvido pelo algorítmo exato de Miliotis que consiste numa combinação de "branch-and-bound" com planos de corte.

Os testes computacionais foram realizados resolvendo-se 12 problemas de 50 a 100 pontos. O algorítmo proposto foi comparado com o algorítmo de Clarke e Wright, de Varredura de Gillet e Miller, de pesquisa em árvore de Christofides et alii., e de 2 fases de Christofides et alii.. Em termos de qualidade da solução, o algorítmo proposto claramente teve melhor desempenho que os demais. O algorítmo encontrou a melhor solução em 9 dos 12 problemas e teve a menor média de custo para todos os casos. O tempo de processamento médio também foi dos menores, somente perdendo para o algorítmo de Clarke e Wright por uma diferença muito pequena. O concorrente mais próximo em precisão foi o algorítmo de pesquisa em árvore de Christofides et alii.. Todavia este algorítmo tem o inconveniente de sofrer um progressivo aumento do tempo de processamento com o número de pontos.

### 5.2.6 Otimização Interativa

E uma abordagem onde um alto grau de participação humana é incorporada ao processo de resolução do problema. A idéia básica dessa estratégia é a de que uma pessoa experiente tem a capacidade de definir e revisar parâmetros e introduzir premissas dentro do modelo de resolução, baseada na sua intuição e conhecimento do problema. Tem-se verificado que, desta forma, aumenta na prática a tendência de que o modelo venha a ser implementado e utilizado. Um exemplo desse procedimento interativo é o algorítmo de Fisher e Jaikumar apresentado anteriormente, que permite a seleção dos pontos base a partir dos quais será feita a designação dos pontos. Ainda dentro dessa categoria, Bodin et alii. [5] (pag.99) fazem referência aos trabalhos de Krolak et alii..

### 5.2.7 Estatística

Uma forma alternativa de abordagem não apontada na classificação acima e que foi encontrada em alguns artigos é a estatística, cabendo, portanto, tecer alguns comentários sobre o assunto.

Novaes [37] elaborou um modelo de zoneamento de entregas para uma frota, a partir de uma regressão estatística que fornecia a distância a ser percorrida por um veículo numa zona, em função do número de pontos de entrega e da área da zona de entrega. Este procedimento permitiu a definição de zonas de entrega fornecendo assim o dimensionamento da frota. A regressão foi obtida através do ajuste de uma curva Erlang com dados obtidos pela roteirização manual de diversos problemas. Novaes não comparou os resultados com algum outro procedimento conhecido para que se pudesse ter uma idéia do desempenho da abordagem proposta.

Um outro trabalho que utiliza a abordagem estatística é o de Golden [24], que elaborou uma regressão para estimativa da distância total correspondente à solução ótima do PCV através da distribuição de Weibul e comparou os resultados com as soluções obtidas pelo procedimento de economias. Segundo Golden, a heurística proposta gera soluções 5 a 10% distantes da solução ótima e é um instrumento muito útil para avaliações rápidas.

Uma desvantagem dos procedimentos apresentados acima é a limitação do seu uso nos casos práticos onde há restrições múltiplas a obedecer, tais como horários limites e capacidades variadas, que seriam muito dificeis de ser incorporados ao modelo.

Como se pôde ver, os métodos heurísticos permitem uma modelagem mais simples, trazendo consigo um forte apelo a sua utilização. O Apêndice B fornece outros detalhes interessantes relativos à modelagem com vistas a sua implementação computacional. Terminada a exposição da pesquisa bibliográfica realizada, irá se apresentar no capítulo seguinte o modelo desenvolvido para a análise da influência dos custos veiculares sobre a roteirização.

# CAPITULO 6

## MODELO DESENVOLVIDO PARA ANALISE

Apesar do método de Fisher e Jaikumar ser bastante eficiente (conforme visto no Capítulo anterior) permitindo se obter soluções muito próximas da ótima em tempos de computação relativamente pequenos, ele apresenta alguns inconvenientes que impedem a sua utilização para o objetivo proposto neste trabalho:

(1) Não se dispunha de programa de computador para a resolução do problema de designação generalizada pelo método de relaxação lagrangeana e a utilização de um pacote genérico de programação inteira (como o TEMPO da Burroughs ou o MSPX da IBM) provavelmente não seria viável para os problemas de maior porte.

(2) A modelagem do problema de designação generalizada, que faz a alocação dos nós aos veículos, não contempla as restrições de jornada de trabalho, que serão incluídas nos problemas a serem resolvidos no Capítulo 7. Por se basear na formulação exata do PRV, a introdução dessas restrições adicionais de tempo, se tornaria rapidamente impraticável computacionalmente ou, pelo menos, tornaria o método menos interessante quando comparado com um método de abordagem heurística, em termos de rapidez e simplicidade.

(4) Finalmente, para atender ao objetivo deste estudo seria necessário que o método de Fisher e Jaikumar permitisse a introdução de parâmetros de capacidades variadas bem como de custo fixo e custo variável para cada tipo de veículo, o que tornaria a sua formulação e resolução relativamente bastante

complexa e difícil.

Por estes motivos procurou-se encontrar uma outra abordagem de resolução heurística que servisse de ponto de partida para o desenvolvimento do modelo de análise. Neste sentido, o trabalho que mais se destacou na pesquisa bibliográfica realizada e que mais atende aos objetivos deste estudo é o apresentado no recente artigo de Golden et alii. [21]. Os algorítimos propostos nesse artigo foram utilizados como base do modelo utilizado para o estudo e são descritos a seguir.

## 6.1 Algorítmos de Determinação do Tamanho e Composição da Frota de Golden et alii.

O trabalho de Golden et alii. [21] procurou avaliar o desempenho de uma série de algorítmos desenvolvidos por eles, os quais consideram os custos fixos de veículos de diferentes capacidades para a da frota e dos roteiros. O custo variável foi admitido como independente do tipo do veículo e igual a um. O objetivo buscado foi o de se minimizar o custo total da operação.

Para efeito de simplificação admitiu-se que os custos variáveis fossem simétricos, isto é $c_{ij} = c_{ji}$ (custo variável para ir de i a j igual ao custo variável para ir de j a i).

Golden et alii. formularam e testaram 12 algoritmos heurísticos que foram os seguintes:

1. CW (Clarke e Wright)
2. CS (Economias combinadas, "Combined savings")
3. OOS (Oportunidades otimistas de economias, "Optimistic opportunity savings")
4. ROS (Oportunidades realísticas de economias, "Realistic opportunity savings")
5. ROS $\gamma$
6. SGT (Rota gigante simples, "Single giant tour")
7. MGT (Rota gigante múltipla , "Multiple giant tour")

8. MGT$^{m}$

9. (MGT+2-opt)

10. (MGT+OROPT)

11. (MGT+2-opt)$^{m}$

12. (MGT+OROPT)$^{m}$

O primeiro algorítmo é o de Clarke e Wright. Os algorítmos 2 a 5 são derivações feitas do algorítmo de Clarke e Wright, de economias e inserção. Os algorítmos 6 a 12 são baseados no procedimento do tipo roteirizar primeiro e agrupar depois. Todos os algorítmos foram desenvolvidos para considerar o custo fixo, com excessão do algorítmo 1, que foi mantido conforme a formulação original.

O cálculo da economia de inserção de rota para o algorítmo de Clarke e Wright (CW) é dado por

$$S_{ij} = c_{1i} + c_{1j} - c_{ij} \qquad (6.1)$$

Para considerar os diversos custos fixos no cálculo da economia, Golden et alii. substituiram essa formulação por

$$SCS_{ij} = S_{ij} + F(z_i) + F(z_j) - F(z_i + z_j) \qquad (6.2)$$

e criaram o algorítmo CS, onde F(z) é o custo fixo do menor veículo que pode atender à demanda z, sendo uma função discreta do tipo escada.

Para superar o problema que o algorítmo CS apresenta de ignorar as possíveis economias permitidas pela utilização de um veículo maior, criou-se o OOS, que é caracterizado pela forma de cálculo da economia por

$$\begin{aligned} SOOS_{ij} &= SCW + F(z_i) + F(z_j) - F(z_i + z_j) \\ &\quad + F(P(z_i + z_j) - z_i - z_j)) \\ &= SCS + F(P(z_i + z_j) - z_i - z_j)) \end{aligned} \qquad (6.3)$$

onde $P(z)$ é a capacidade do menor veículo que pode atender à demanda z.

Através dos testes realizados com diversos problemas, verificou-se que o algorítmo CS supera o OOS. Na verdade, ambos os algorítmos apresentam pontos fracos básicos. O CS subestima as economias, o OOS superestima. A solução que veio à tona foi o método ROS onde a economia é calculada por

$$SROS_{ij} = SCS_{ij} + F'(P((z_i+z_j) - z_i - z_j) \qquad (6.4)$$

onde $F'(z)$ é o custo fixo do maior veículo cuja capacidade é menor ou igual a z (se $z<K_1$, $F'(z)=0$). Os testes computacionais confirmaram a superioridade deste algorítmo.

Para aprimorar esta abordagem foi introduzido o parâmetro g de formato do roteiro que modifica a preferência de inserções dos pontos em função da distância entre eles e da sua distância ao depósito (descrito em detalhes no Anexo B), gerando o algorítmo ROS$\gamma$ , cujo cálculo de economia é dado por

$$SROS\gamma_{ij} = SROS_{ij} + (1 - \gamma)c_{ij} \qquad (6.5)$$

Dos 20 testes realizados em problemas de 12 a 100 pontos, o ROS$\gamma$ mostrou ser o melhor de todos da categoria de economia e inserção acima apresentada.

Um segundo tipo de abordagem foi utilizada para a definição dos algorítmos 6 a 12. A partir de uma rota gigante criada pelo algorítmo 2-opt de Lin, as rotas de cada veículo foram formadas através da resolução de um problema de determinação da distância mínima.

Golden et alii. denominaram de sequência 1-s(1)-s(2)-...-s(n)-1 ao roteiro gigante resultante do procedimento 2-opt, onde 1 é o índice do depósito, s(1) o primeiro ponto, s(2) o segundo e assim sucessivamente.

O SGT forma os custos de cada tramo da rede para o problema da distância mínima através da seguinte formulação

$$\begin{aligned} CUSTO(k,m) = c(1,s(k)) &+ \sum_{r=k}^{m-2} c(s(r),s(r+1)) \\ &+ c(s(m-1),1) \\ &+ F\left(\sum_{r=k}^{m-1} d(s(r))\right) \qquad (6.6) \end{aligned}$$

onde: $s(k)$ = posição sequencial do ponto k no roteiro gigante,

CUSTO(k,m) = custo fixo mais variável do roteiro que sai do depósito, atende os pontos $s(k)$, $s(k+1)$, $s(k+2)$,...., $s(m-1)$, e retorna ao depósito,

$c(i,j)$ = custo do percurso do ponto i ao ponto j, associado à distância,

$d(i)$ = demanda a ser entregue ao ponto i.

O MGT parte da construção de uma rota inicial igual à do SGT porém sem incluir o depósito. São criadas cópias de M nós que são localizadas exatamente na posição dos nós originais. São calculados os custos totais de ligação entre os conjuntos de pontos através da expressão (6.6) e a partição dos roteiros é feita pela determinação da menor distância da rede. O número M de nós a serem copiados é calculado pela seguinte expressão:

$$\sum_{r=1}^{M-1} d(s(r)) < CMV \leq \sum_{r=1}^{M} d(s(r)) \qquad (6.7)$$

onde: CMV = capacidade do maior veículo

O MGT* utiliza a mesma sistemática do MGT com a diferença que o processo é repetido 5 vezes inciando-se a partir de 5 rotas gigantes diferentes.

O (MGT+2-opt) faz um pós aperfeiçoamento da solução, partindo-se do resultado do MGT. E aplicado o procedimento de melhoria e trocas 2-opt de Lin para arcos de roteiros diferentes, sempre se observando as restrições do problema.

O (MGT+OROPT) é uma adaptação, segundo Golden et alii., do procedimento 3-opt proposto por Or (do qual Golden et alii. não dão maiores detalhes), que basicamente segue o mesmo raciocínio do algorítmo anterior.

Finalmente, repetindo-se o procedimento realizado no MGT$^{m}$, definiu-se os algorítmos (MGT+2-opt)$^{m}$ e (MGT+OROPT)$^{m}$ que são os mesmos definidos anteriormente repetidos 5 vezes de modo a se escolher o de melhor resultado.

O (MGT+OROPT)$^{m}$ foi o algorítmo que apresentou o melhor resultado de custo, porém apenas marginalmente melhor que o (MGT+2-opt)$^{m}$. Conforme mencionado anteriormente, o ROS-g foi o melhor dos algorítmos de economias e inserção. Pelos resultados de Golden et alii. verifica-se que seu desempenho é comparável aos métodos do segundo tipo (SGT e MGT) sem o pós aperfeiçoamento da solução (MGT+2-opt ou MGT+OROPT).

## 6.2 Modelo Utilizado

Os modelos de Golden et alii. integrantes da estratégia de construção inicial de um roteiro gigante são facilmente implementáveis para a consideração de custos variáveis específicos por tipo de veículo. Basta-se tomar a função CUSTO(k,m) dada pela expressão (6.6) e substitui-la por:

$$\begin{aligned} CUSTOA(k,m) = & \ dist(1,s(k))*cvv(v) \\ & + \sum_{r=k}^{m-2} dist(s(r)),s(r+1))*cvv(v) \\ & + dist(s(m-1),1)*cvv(v) \\ & + F( \sum_{r=k}^{m-1} d(s(r))) \end{aligned} \tag{6.8}$$

onde: dist(i,j) = distância do ponto i ao j,

cvv(v) = custo variável do veículo do tipo v,

v = menor veículo que pode atender a soma das demandas da rota.

Através disto, foram implementados os modelos SGT, MGT e

MGT$^{m}$ num mesmo programa computacional que se denominou de WGT, com opção de geração de M còpias de nós para o MGT e opção de repetição de geração de n processamentos para n roteiros aleatórios gigantes.

Uma outra característica implementada no WGT foi a restrição de duração máxima da jornada de trabalho na rota. Primeiramente, todas as viagens têm o seu tempo de duração estimados através da seguinte expressão:

$$
\begin{aligned}
Trot = Tfixo + Ttaxa * & \sum_{r=k}^{m-1} d(s(r)) \\
& + (\ dist(1, s(k)) \\
& + \sum_{r=k}^{m-2} dist(s(r)), s(r+1)) \\
& + dist(s(m-1), 1)\ )/Vel
\end{aligned}
\qquad (6.9)
$$

onde: Trot = tempo de duração da rota em formação,

Tfixo = tempo constante de atendimento por ponto,

Ttaxa = taxa de descarga por unidade de demanda,

Vel = velocidade média de percurso.

Em seguida, o tempo de duração da rota é comparado à jornada máxima de trabalho. Se o tempo de rota for maior, a função CUSTOA(k,m) adquire o valor infinito. Caso contrário, assume o valor calculado pela expressão (6.8). Deste modo impede-se a formação de rotas com duração maior que a permissível.

Para permitir a análise de incentivo à formação de rotas mais curtas de modo a se utilizar veículos de menor capacidade e portanto menor custo fixo, na realização de viagens múltiplas, conforme ilustrado na introdução do trabalho, foi implementado uma política de prêmio às viagens mais curtas. Esta política funciona da seguinte maneira: todas as viagens (veículos) com duração menor que a metade da jornada máxima diária têm seu custo fixo reduzido à metade de modo a levar o algorítmo de determinação do menor caminho à adotá-la. Este fator é incorporado ao modelo através da função CUSTOB(k,m) conforme o que se segue:

$$CUSTOB(k,m) = CUSTOA(k,m) - P \qquad (6.10)$$

onde: P = prêmio igual à metade do custo fixo do veículo alocado à ligação.

Para realizar a consolidação das rotas mais curtas obtidas com a política incorporada acima, deve-se alocar os veículos às viagens múltiplas mediante um pós-processamento. Nos casos mais simples esta tarefa pode ser feita manualmente. Entretanto, nos casos que apresentam um maior número de combinações de rotas possíveis, torna-se necessário um método mais eficiente. As alternativas existentes neste caso são duas. A primeira delas é por programação matemática baseada na seguinte formulação inteira:

$$\text{minimizar} \sum_{j=1}^{q} y_j \qquad (6.11)$$

sujeito a:

$$\sum_{i=1}^{q} Trot_i \, x_{ij} \leq Tmax_j \, y_j \qquad j=1,\ldots,q \qquad (6.12)$$

$$\sum_{j=1}^{q} x_{ij} = 1 \qquad i=1,\ldots,q \qquad (6.13)$$

$$y_j, x_{ij} = 0 \text{ ou } 1 \qquad i,j=1,\ldots,q \qquad (6.14)$$

onde: $Trot_i$ = tempo de duração da rota i, para i= 1,...q,
q = número de rotas formadas,
$y_j$=1 se o veículo j é utilizado, $y_j$=0 caso contrário,
$x_{ij}$=1 se o roteiro i é alocado ao veículo j, $x_{ij}$=0 caso contrário.

A segunda alternativa disponível para a execução do pós-processamento de fusão das rotas é um algorítmo heurístico. Neste trabalho, para efeito de simplificaçào dos casos analisados, considera-se que um veículo possa realizar no máximo duas viagens por dia. Isto representa o que ocorre na maioria dos casos na prática. Assim, o algorítmo heurístico proposto faz a determinação do número de veículos necessário, combinando as rotas através de um critério bastante simples, sempre se respeitando a duração máxima da jornada de trabalho. Este algorítmo consiste no seguinte:

1. Listar, para um determinado tipo de veículo, as rotas formadas pelo modelo WGT proposto e os respectivos tempos de duração.
2. Determinar o menor tempo de jornada da lista.
3. Subtrair este número da duração máxima possível.
4. Eliminar todas as rotas com tempo superior ao valor encontrado em 3, alocando um veículo para cada uma delas.
5. Fazer a fusão da rota remanecescente de maior duração com a rota de menor duração encontrada em 2 e eliminar as duas da lista, alocando um veículo único para as duas viagens.
6. Repetir os passos 2 a 5 até eliminar todas as rotas da lista.
7. Contar o número total de veículos necessários somando o número obtido no passo 4 (número de veículos que fazem uma viagem), com o número obtido no passo 5 (número de veículos que fazem duas viagens diárias).

Os procedimentos descritos acima são válidos para fusão de rotas alocadas a veículos do mesmo tipo. E possível se alocar veículos de tipos diferentes, contanto que não haja incompatibilidade de capacidade, devendo para isto se implementar os modelos de forma adequada.

## 6.3 Implementação Computacional do Modelo

A partir da formulação do modelo apresentada acima, foi desenvolvido um programa de microcomputador em linguagem BASIC estruturado cuja listagem encontra-se no Apêndice E. O programa tem a seguinte estrutura de funcionamento:

1. Rotina XYP
   Permite a entrada de dados da geometria do problema e da demanda por digitação das coordenadas (x,y) e demanda de cada ponto. Permite também a gravação e leitura de arquivos gravados em diskette.

2. Rotina DISTANCIAS
   Executa o cálculo da matriz de distâncias. As distâncias entre os pontos são lineares e simétricas. Esta rotina permite também entrar com a matriz de distâncias por digitação, quando necessário. Como o cálculo é relativamente demorado, principalmente para os problemas maiores, a matriz de distâncias pode ser gravada em diskette, para ser lida posteriormente.

3. Rotina ROTALEATORIA
   Cria um roteiro gigante aleatório através do seguinte procedimento: a cada passo é gerado um número aleatório entre 1 e n+1 (n=número de pontos do problema com o depósito). Se o ponto correspondente a este número já tiver sido alocado à rota, o número é desprezado e gera-se um outro número aleatório. Caso contrário, este ponto é acrescido à rota em formação, após o último ponto anteriormente incorporado. O processo continua até se esgotar todos os pontos a serem roteirizados.

4. Rotina OPT2
   Executa o procedimento 2-opt de Lin de trocas de arcos,

partindo do roteiro gigante aleatório gerado pela rotina anterior.

5. Rotina FROTA

Permite a entrada, leitura e gravação em diskette, de dados descritivos da frota (tipos, capacidades, custos fixos e variáveis).

6. Rotina TEMPO

Permite a entrada de dados de tempo máximo de jornada, tempo de corte (tempo máximo de duração da primeira viagem), tempo de atendimento constante em cada ponto, taxa de descarga e velocidade média de percurso.

7. Rotina ZONAS

A partir do roteiro gigante formado pela Rotina OPT2, e pelos dados da FROTA e TEMPO, é feito o particionamento do roteiro através das seguintes sub-rotinas:

a. Sub-rotina COSTKM

Calcula a matriz CUSTOB(k,m) para cada alternativa de particionamento.

b. Sub-rotina CRITICO

Faz a determinação do menor caminho baseado no algorítmo clássico de Dijkstra (apresentado no Apêndice C), utilizando como distâncias dos diversos arcos existentes na rede os valores da matriz COSTB(k,m) calculada pela sub-rotina anterior.

c. Sub-rotina RESUMO

Uma vez determinado o particionamento do roteiro gigante pela sub-rotina CRITICO, a sub-rotina RESUMO elabora o detalhamento de cada roteiro formado, permitindo a visualização dos resultados da roteirização no monitor e a impressão de listagens.

O programa implementado foi utilizado para permitir a realização de diversas análises de influência dos parâmetros de custo e tempos sobre a roteirização e que serão assunto do próximo capítulo.

# CAPITULO 7

# RESULTADOS

A partir do modelo estabelecido no Capítulo anterior, implementou-se um programa de computação com a finalidade de resolver uma série de problemas, para estabelecer relações entre custos fixos e variáveis, tempos de jornada e tipos de veículos. Irá se apresentar neste Capítulo uma descrição dos problemas resolvidos, do trabalho computacional e finalmente das análises obtidas.

## 7.1 Problemas Resolvidos

Para efeito de simplificação, cada problema é referenciado pelo código dos dados e da frota (o problema 1 da tabela 7.1, por exemplo, é dado pelo problema de 13 pontos definido por Clarke e Wright CW13 e pela frota #1 definida por Golden et alii., respectivamente, conforme discriminado no Apêndice D).

TABELA 7.1

RELAÇÃO DE PROBLEMAS RESOLVIDOS

| PROBLEMA | | FROTA | PONTOS(1) | DEMANDA MIN | MAX | MED |
|---|---|---|---|---|---|---|
| 1 | CW13 | #1 | 13 | 11 | 19 | 15 |
| 2 | CE21-51A | #3 | 21 | 3 | 41 | 17,7 |
| 3 | CW31 | #7 | 31 | 0 | 19 | 8 |
| 4 | CE51-76 | #13 | 51 | 5 | 37 | 19,5 |
| 5 | CE51 | #15 | 51 | 3 | 41 | 15,5 |
| 6 | CE76 | #17 | 76 | 1 | 37 | 18,2 |
| 7 | CE101 | #19 | 101 | 1 | 41 | 14,6 |

Obs.: (1) Pontos de entrega mais o depósito.

Os dados de frota utilizados são de Golden et alii. [21] e guardam uma boa proporcionalidade em relação aos problemas. Adotou-se somente uma frota para cada problema, conforme relação acima. Como o objetivo buscado não era o de se testar exaustivamente o desempenho do modelo comparativamente a um outro, não se teve a preocupação de se processar um número maior de casos. Os problemas escolhidos foram suficientes para se analisar os pontos básicos pretendidos.

Golden et alii. consideraram diferentes custos fixos para cada veículo e custo variável igual a 1, para todos, ou seja, uma unidade de custo para cada unidade de distância percorrida. Nas diversas análises que serão apresentadas neste Capítulo foram utilizadas diversas composições de custo variável (e eventualmente de custos fixos diferentes ao de Golden et alii.). Quando isto ocorre, os dados utilizados são devidamente assinalados. E importante ressaltar que estes dados foram adotados arbitrariamente sem a pretensão de que fossem representativos, mas que servissem aos propósitos da análise.

## 7.2 Aferição do Programa Computacional

O programa desenvolvido (WGT) foi aferido com os resultados do SGT de Golden et alii. [21]. Foram comparados os resultados de 13 dos 20 problemas resolvidos por Golden et alii., tendo-se constatado uma diferença média de apenas 1,9% , permitindo assim constatar sua validade. A tabela 7.2 mostra os valores de custo total encontrados. A pequena diferença apresentada se deve ao fato de o WGT ter sido processado somente uma única vez para cada problema, por motivos de tempo de processamento, enquanto que o SGT de Golden et alii. representa o melhor resultado de 25 processamentos.

TABELA 7.2

AFERIÇÃO DO PROGRAMA WGT

| PROBLEMA | | WGT | SGT | DIFERENÇA (WGT/SGT) |
|---|---|---|---|---|
| 1 | CW13 #1 | 642 | 620 | 3,5% |
| 2 | CW13 #2 | 736 | 722 | 1,9% |
| 3 | CE21-51A #3 | 983 | 965 | 1,9% |
| 4 | CE21-51A #4 | 7363 | 6918 | 6,4% |
| 5 | CE21-51B #5 | 1050 | 1027 | 2,2% |
| 6 | CE21-51B #6 | 7379 | 7391 | 0,0% |
| 7 | CE51-76 #13 | 2477 | 2449 | 1,1% |
| 8 | CE51-76 #14 | 9670 | 9637 | 0,3% |
| 9 | CE51 #15 | 2756 | 2722 | 1,2% |
| 10 | CE51 #16 | 2881 | 2855 | 0,9% |
| 11 | CE76 #17 | 1851 | 1815 | 3,9% |
| 12 | CE76 #18 | 2511 | 2479 | 1,3% |
| 13 | CE101 #19 | 9349 | 9283 | 0,7% |
| MEDIA | | – | – | 1,9% |

O programa computacional foi implementado em linguagem BASIC estruturado e foi processado num microcomputador de 16bits-compatível IBM. Os tempos de processamento obtidos apresentados na tabela 7.3, referem-se ao equipamento sem co-processador 8087 e não consideram os tempos de leitura dos dados.

O WGT é equivalente em tempo de processamento ao SGT de Golden et alii. aplicado uma única vez, isto é sem repetir 25 vezes o processo para 25 diferentes roteiros gigantes aleatórios. Os tempos apresentados para o SGT são estimativos e foram obtidos pela multiplicação dos dados obtidos com o WGT por 25. O MGT de Golden et alii. também foi implementado e a sua utilização gerou os tempos apresentados na coluna MGT, tendo-se estimado a partir daí os tempos para o MGT$^{\underline{s}}$ pela simples multiplicação dos tempos por 5.

TABELA 7.3

TEMPOS DE PROCESSAMENTO

| PROBLEMA | | PONTOS | WGT | SGT | $MGT^1$ | $MGT^5$ |
|---|---|---|---|---|---|---|
| 1 | #1 | 12 | 36" | 15' | 4' | 20' |
| 2 | #2 | 12 | 47" | 20' | 5' | 25' |
| 3 | #3 | 20 | 1'20" | 33' | 8' | 40' |
| 4 | #4 | 20 | 1'20" | 33' | 8' | 40' |
| 5 | #5 | 20 | 1'20" | 33' | 8' | 40' |
| 6 | #6 | 20 | 1'20" | 25' | 8' | 40' |
| 7 | #13 | 50 | 10' | 4h | 1h | 5h |
| 8 | #14 | 50 | 9' | 3h30' | 1h | 5h |
| 9 | #15 | 50 | 9' | 4h | 1h | 5h30' |
| 10 | #16 | 50 | 6' | 2h30' | 1h | 3h30' |
| 11 | #17 | 75 | 21' | 8h30' | 3h | 16h |
| 12 | #18 | 75 | 23' | 9h30' | 5h30' | 28h |
| 13 | #19 | 100 | 40' | 17h | - | - |

Os resultados de Golden et alii. indicam que o SGT apresenta desempenho superior ao $MGT^1$, só sendo superado pelo $MGT^5$. Como os tempos de processamento encontrados para o SGT e para o $MGT^5$ foram excessivamente grandes, principalmente para os problemas maiores, decidiu-se pela adoção do modelo WGT, sem perda significativa da qualidade dos resultados, conforme foi aferido inicialmente (a perda é inferior a 2% em relação ao SGT).

Para permitir as análises que serão apresentadas a seguir, os problemas relacionados no ítem 7.1 foram resolvidos na seguinte sequência:

I. Resolução de Golden. E a resolução do problema de roteirização conforme Golden et alii., sem custos variáveis diferenciados, ou seja o custo variável é a própria distância.

II. WGT Livre. E a resolução do problema com custos variáveis específicos para cada tipo de veículo utilizado.

III. WGT Restrito sem Prêmio. E a resolução do problema com custos variáveis e com restrição de duração máxima de jornada.

IV. WGT Restrito com Prêmio. Com custos variáveis, restrição de jornada e prêmio de incentivo à realização de viagens mais rápidas.

A análise dos efeitos dos diversos custos sobre os resultados do PRV está condicionada à estrutura de demanda e geometria dos problemas selecionados, razão pela qual conclusões do ponto de vista quantitativo não podem ser generalizadas. O primeiro aspecto enfocado e que será apresentado agora foi a questão da importância dos custos variáveis sobre os resultados da roteirização e será apresentado a seguir.

## 7.3 Influência dos Custos Variáveis

Quanto maior for o custo variavel, maior é o seu peso na decisão de escolha da frota e, consequentemente, menor é a participação do custo fixo no problema. Nesses casos, procura-se utilizar veículos maiores para consolidar as entregas reduzindo o percurso ocioso de ida e volta ao depósito. Por outro lado, quanto menor for o custo variável, maior é a tendência de se escolher veículos com menor custo fixo unitário (custo fixo por unidade de capacidade). A tabela 7.4 mostra um resultado obtido com o modelo que confirma o que foi dito acima . Em todas as situações foi designado um mesmo custo variável para todos os veículos e variou-se esse valor de modo a modificar a importância relativa do custo variável total no problema. (O veículo de menor custo fixo unitário corresponde ao A).

TABELA 7.4

EFEITO DOS CUSTOS VARIAVEIS CONSTANTES

PROBLEMA CW13 #1

| C.VARIAVEL | FROTA | DISTANCIA |
|---|---|---|
| 0.05 | $A^4B^4$ | 508 |
| 0.2 | $B^2C^2$ | 314 |
| 1 | $B^2C^2$ | 314 |
| 5 | $B^1C^3$ | 300 |
| 10 | $B^1C^3$ | 300 |
| 100 | $B^1C^3$ | 300 |
| 1000 | $B^1C^3$ | 300 |

Pode-se observar que a partir de um certo ponto o problema se torna insensível à variação do custo variável.

Adotando-se uma composição de custos variáveis diferentes para cada tipo de veículo, obteve-se uma série de resultados que permitem comparar as soluções entre o modelo com custo variável independente do tipo de veículo e o modelo proposto. Para efetuar uma comparação justa entre o modelo aqui proposto e o de Golden et alii. onde os custos variáveis são iguais para todos os veículos, utilizou-se o seguinte procedimento: as distâncias das rotas são multiplicadas pleos custos variáveis adotados para o modelo proposto obtendo-se o custo variável total. Os resultados desses cálculos são apresentados na tabela 7.5.

Pode-se constatar através destes resultados que a especificação de custos variáveis para cada tipo de veículo no processo de roteirização gera resultados melhores do que quando isto não ocorre.

TABELA 7.5

EFEITO DOS CUSTOS VARIAVEIS SOBRE DIVERSOS PROBLEMAS

| PROBLEMA | C.VARIAVEL | FROTA | DISTANCIA | C.TOTAL |
|---|---|---|---|---|
| 1 CW13 #1 | A=1;B=1;C=1 | $A^1B^2C^2$ | 322 | $823 |
| | A=0,5;B=1;<br>C=2 | $A^2B^5$ | 454 | $700 |
| 2 CE21-51A #3 | A=1;B=1;C=1;<br>D=1;E=1 | $A^1B^1C^2E^2$ | 378 | $2166 |
| | A=0,5;B=1;<br>C=2;D=3;E=6 | $A^9B^6D^1$ | 728 | $1103 |
| 3 CW31 #7 | A=1;B=1;C=1<br>D=1;E=1 | $A^2B^2$ | 754 | $4062 |
| | A=1;B=5;<br>C=8;D=12;E=20 | $A^8$ | 1382 | $2582 |
| 4 CE51-76 #13 | A=1;B=1;C=1;<br>D=1;E=1;F=1 | $A^1B^2C^2D^1E^5F^2$ | 685 | $6483 |
| | A=0,49;B=0,87;<br>C=1,24;D=3,18<br>E=6,36;F=11,30 | $A^{17}B^{19}C^5$ | 1849 | $2729 |
| 5 CE76 #17 | A=1;B=1;C=1;<br>D=1 | $A^5B^3C^4$ | 886 | $2034 |
| | A=0,5;B=1;<br>C=1,5;D=2 | $A^{30}B^1$ | 1741 | $1743 |
| 6 CE101 #19 | A=1;B=1;C=1 | $A^{16}$ | 1349 | $21490 |
| | A=10;B=12;<br>C=15 | $A^6B^5$ | 1029 | $20588 |

Um outro aspecto observado é que quanto maior a relação entre os custos variáveis do maior veículo em relação ao menor veículo, maior é a tendência de se utilizar o veículo menor, com menor custo variável relativo. Isto pode ser constatado através dos resultados apresentados na tabela 7.6.

TABELA 7.6

EFEITO DOS CUSTOS VARIAVEIS RELATIVOS ENTRE VEICULOS

PROBLEMA CE51 #15

| CUSTO VARIAVEL | FROTA | DISTANCIA |
|---|---|---|
| A=1 B=1 C=1 | $A^4B^2C^2$ | 734 |
| A=1 B=1.2 C=1.5 | $A^{12}C^1C^1$ | 876 |
| A=2 B=3 C=4 | $A^{10}$ | 1100 |
| A=2 B=4 C=6 | $A^{10}$ | 1100 |

Uma outra análise foi feita mantendo-se constante o custo variável por veículo e variando-se os seus custos fixos de cada veículo, de modo a ir aumentando a diferença entre eles. Os resultados são apresentados na tabela 7.7 e confirmam que a escolha tende para o veículo com menor custo fixo unitário. A solução do problema não é, porém muito sensível à variação desses custos, como se verifica na passagem do problema 1 ao 2 e do problema 3 ao 4.

TABELA 7.7

EFEITO DA VARIAÇÃO DOS CUSTOS FIXOS

| PROBL. | C.FIXO | CF/CAPAC | FROTA | DISTANCIA |
|---|---|---|---|---|
| | A=80 | 4 | | |
| 1 | B=355 | 4 | $A^1B^2C^4$ | 786 |
| | C=1216 | 4 | | |
| | A=104 | 5,2 | | |
| 2 | B=355 | 4 | $A^1B^2C^4$ | 786 |
| | C=851 | 2,8 | | |
| | A=160 | 8 | | |
| 3 | B=355 | 4 | $C^5$ | 721 |
| | C=604 | 2 | | |
| | A=216 | 11 | | |
| 4 | B=355 | 4 | $C^5$ | 721 |
| | C=434 | 1 | | |

Obs.: os custos variáveis da frota são:
A=0,67; B=1,15; C=1,56

Para se ter uma ideia do comportamento da roteirização em relação à importância relativa dos custos variáveis sobre o custo total, foram determinados três conjuntos de dados de custos de modo a se obter relações de custo variável total sobre o custo total de 20%, 50% e 80%. Basicamente, os problemas de entrega urbana em cidades densas apresentam menor relação de custo variável sobre custo total do que outras cidades mais esparsas. Os resultados da análise estão na tabela 7.8. Observou-se que existe uma certa insensibilidade da faixa de 80% em relação a faixa de 50%, mostrando um aspecto do problema que os processos empíricos de análise de frotas não permitem avaliar.

TABELA 7.8

PARTICIPAÇÃO RELATIVA DOS CUSTOS VARIAVEIS
SOBRE O CUSTO TOTAL
PROBLEMA CW13 #1

| C.VAR. | FROTA | DISTANCIA | C.VAR.TOTAL | C.TOTAL | CV/CT |
|---|---|---|---|---|---|
| A=0,07<br>B=0,13<br>C=0,20 | $B^4C^1$ | 392 | $55 | $275 | 20% |
| A=0,26<br>B=0,53<br>C=0,79 | $A^1B^2C^2$ | 332 | $218 | $438 | 50% |
| A=1,06<br>B=2,12<br>C=3,18 | $A^1B^2C^2$ | 332 | $878 | $1098 | 80% |

Obs.: Os custos fixos da frota são A=20;B=40;C=60.

O mesmo procedimento foi realizado para o CE21-51A não se obtendo nenhuma alteração dos resultados, mostrando que este problema insensível à relação entre custos fixos e variáveis na faixa pesquisada.

## 7.4 Influência da Restrição de Jornada

A restrição de tempo de duração do roteiro, ou jornada máxima de trabalho diária, tem influência sobre a escolha da frota e dos roteiros, elevando o custo total da operação. Para o estudo dos efeitos das restrições de jornada, usou-se os valores apresentados na tabela 7.9, estabelecidos arbitrariamente, de modo a gerar rotas de duração adequada para a análise.

TABELA 7.9

TEMPOS UTILIZADOS

Duração máxima da jornada ................. 9 ou 10h

Tempo de corte para premiação ........... 4,5 ou 5h

Tempo constante de atendimento em cada ponto ... 20'

| | | |
|---|---|---|
| Taxa de descarga | #1 ................. | 4'/unidade |
| | #2 ................. | 2'/unidade |
| | #7 ................. | 0.8'/unidade |
| | #13 ................. | 1,2'/unidade |
| | #15 ................. | 1,5'/unidade |
| | #17 ................. | 0,69/unidade |
| | #19 ................. | 0,8'/unidade |

Estes dados serão utilizados na resolução dos problemas relacionados na tabela 7.10.

TABELA 7.10

DADOS DOS PROBLEMAS 1 A 8 DA TABELA 7.11

| PROBLEMA | REFERENCIA | CUSTOS VARIAVEIS |
|---|---|---|
| 1 | CW13 #1 | A=0,5;B=1;C=2 |
| 2 | CW31 #7 | A=1;B=1;C=1;D=1;E=1 |
| 3 | CE51-76 #13 | A=1;B=1;C=1;D=1;E=1;F=1 |
| 4 | CE51 #15 | A=1;B=1;C=1 |
| 5 | CE51 #15 | A=1;B=1,2;C=1,5 |
| 6 | CE51 #15 | A=2;B=4;C=6 |
| 7 | CE76 #17 | A=1;B=1;C=1;D=1 |
| 8 | CE101 #19 | A=10;B=12;C=15 |

O efeito da restrição de jornada depende dos dados de tempos utilizados. Num problema resolvido sem essa restrição,quanto maior for o número de rotas com tempo superior à jornada máxima de trabalho, maiores serão as alterações necessárias para se atender a essa restrição. Via de regra, essas alterações levam ao aumento do custo total em decorrência do acréscimo da distância total percorrida pela frota acoplado ao aumento do tamanho da frota. O aumento da distância percorrida se dá pelo aumento do número de rotas gerando uma elevação dos percursos ociosos ou mortos. O aumento da frota se dá geralmente através de uma mudança da composição de veículos, aumentando a participação de veículos menores, conforme pode-se observar pelos casos apresentados na tabela 7.11.

TABELA 7.11

EFEITO DA RESTRIÇÃO DE JORNADA

| PROBLEMA | CONDIÇÃO | FROTA | DISTANCIA | C.TOTAL |
|---|---|---|---|---|
| 1 | sem restrição | $A^{2}B^{2}$ | 454 | $700 |
| | jornada 9h | $A^{4}B^{6}$ | 672 | $924 |
| 2 | sem restrição | $A^{1}B^{1}C^{1}$ | 600 | $2050 |
| | jornada 10h | $A^{8}$ | 1377 | $2427 |
| 3 | sem restrição | $A^{1}B^{2}C^{2}D^{1}E^{3}F^{2}$ | 685 | $2470 |
| | jornada 10h | $A^{3}B^{5}C^{2}D^{1}E^{2}F^{2}$ | 790 | $2495 |
| 4 | sem restrição | $A^{6}B^{2}C^{2}$ | 734 | $2734 |
| | jornada 10h | $A^{18}$ | 1130 | $2930 |
| 5 | sem restrição | $A^{12}B^{1}C^{1}$ | 876 | $2865 |
| | jornada 10h | $A^{18}$ | 1130 | $2930 |
| 6 | sem restrição | $A^{18}$ | 1100 | $4000 |
| | jornada 9h | $A^{21}$ | 1269 | $4637 |
| 7 | sem restrição | $A^{9}B^{3}C^{4}$ | 886 | $1851 |
| | jornada 10h | $A^{21}B^{9}$ | 1652 | $2577 |
| 8 | sem restricào | $A^{6}B^{9}$ | 1029 | $20588 |
| | jornada 10h | $A^{28}$ | 1952 | $33523 |

Obs.: os problemas são indicados na tabela 7.10.

A colocação de restrição de jornada eleva o custo total de roteirização devido ao aumento do número de rotas criadas pela quebra das rotas mais demoradas. Podem ocorrer casos em que é possível se compor rotas pequenas (ou rápidas) de modo que um mesmo veículos faça as duas (ou mais viagens) reduzindo assim o custo fixo. Isto é feito através do pós-processamento das rotas obtidas.

## 7.5 Influência da Fusão de Rotas

Neste trabalho se considerou, para efeito de simplificação, apenas a fusão de rotas alocadas a veículos de mesma capacidade, na realização de no máximo duas viagens por dia. Nota-se que é possível a fusão de rotas alocadas a veículos de capacidades diferentes, contanto que o veículo que irá ser alocado para a realização de viagens múltiplas tenha capacidade compatível. A tabela 7.2 apresenta alguns casos de pós-processamento realizados a partir de resultados dos processamentos feitos anteriormente.

TABELA 7.12

EFEITO DO POS-PROCESSAMENTO DOS RESULTADOS

| PROBLEMA | | FROTA ANTES | FROTA DEPOIS |
|---|---|---|---|
| 1 | CE21-51A | $B^2C^2D^4$ | $B^1C^1D^2$ |
| 2 | CW31 | $A^4B^6$ | $A^3B^6$ |
| 3 | " | $A^8$ | $A^7$ |
| 4 | CE51-76 | $A^3B^5C^2D^1E^2F^2$ | $A^1B^1C^1D^1E^2F^2$ |
| 5 | CE51 | $A^{21}$ | $A^{20}$ |
| 6 | " | $A^6B^2C^2$ | $A^3B^2C^1$ |
| 7 | CE76 | $A^{21}B^5$ | $A^{19}B^5$ |

Deve-se observar que não se considerou o tempo adicional de embarque das cargas no depósito para as viagens múltiplas, nos casos em que ocorrem, por uma questão de simplificação dos dados necessários para caracterização dos problemas. Este aspecto é facilmente implementável, bastando-se para isso somar uma constante ao tempo de duração de cada rota realizada.

A partir desta concepção de ganhos no custo fixo obtidos pela fusão de rotas, desenvolveu-se uma política de incentivo, ou premiação dos veículos, para a realização de viagens mais curtas. O incentivo é feito através de descontos do custo fixo, no processo de determinação de roteiros, procurando-se deste modo obter reduções no custo total.

## 7.6 Incentivo à Utilização de Veículos Menores

A política adotada para a análise foi a de se conceder um desconto de 50% do custo fixo para as rotas que se realizam num tempo igual ou inferior à metade da duração máxima da jornada. A adoção desta política permitiu, na maioria dos casos, se obter resultados melhores através da realização de duas viagens diárias por determinados veículos, conforme exemplos mostrados na tabela 7.13.

TABELA 7.13

EFEITO DO INCENTIVO A UTILIZAçãO DE VEICULOS MENORES

| PROBLEMA | SITUAçãO | FROTA | | DISTANCIA | CUSTO TOTAL |
|---|---|---|---|---|---|
| 1 | CE21-51A | ANTES | $B^1C^1D^1E^2$ | 345 | $1000 |
| | | DEPOIS | $B^1C^1D^2$ | 450 | $775 |
| 2 | CE76 | ANTES | $A^{19}B^5$ | 1652 | $2527 |
| | | DEPOIS | $A^{21}B^3$ | 1712 | $2477 |
| 3 | CE101 | ANTES | $A^{20}$ | 1952 | $33523 |
| | | DEPOIS | $A^{27}$ | 1960 | $33105 |

Uma política alternativa de premiação seria a de se conceder um desconto do custo fixo, para as rotas com duração igual ou inferior ao tempo limite de duraçao da primeira viagem, proporcional ao tempo de duração da rota. O custo fixo seria igual ao custo fixo do veículo multiplicado pela duração do roteiro dividida pela jornada máxima. Esta política foi testada em dois problemas (CE76 e CE101) e seus resultados foram idênticos aos obtidos pela política anterior.

Cabe ressaltar que não se pretendeu buscar a quantificação das conclusões obtidas pelas análises. A generalização de conclusões envolvendo relações de custo ou demais parâmetros foge ao escopo do trabalho por exigirem um processamento exaustivo de dados, além de terem, provavelmente, aplicabilidade muito restrita em função da diversidade de casos existentes na prática. Cabe apontar também que os resultados obtidos atestam comportamentos que seriam esperados na prática, pela lógica ou experiência pessoal. O que se pretendeu foi mostrar que a abordagem adotada possibilita predizer com maior precisão os resultados esperados em função dos parâmetros e dados do problema.

# CAPITULO 8

# CONCLUSÕES E RECOMENDAÇÕES

Muito pouco se tem feito para se considerar a influência dos custos fixos e variáveis dos veículos na resolução dos problemas de roteirização de veículos (PRV). O trabalho mostrou a importância do seu efeito na escolha da frota e na formação dos roteiros.

Este estudo teve por objetivo pesquisar as diversas abordagens existentes para o PRV com vistas à análise da influência dos diferentes custos veiculares na determinação de frotas de diferentes capacidades para a entrega de produtos em zonas urbanas. Através do desenvolvimento de um modelo (WGT), resolveu-se uma série de problemas com diferentes composições de frota e custos. Os resultados foram análisados quanto aos seguintes fatores: (1) efeito do custo variável sobre a formação de rotas e escolha de veículos, (2) influência da restrição de jornada máxima de trabalho na rota sobre a frota, (3) reflexo do incentivo à formação de rotas mais rápidas por veículos menores (ao invés de se ter rotas mais longas realizadas por veículos maiores) e, finalmente, (4) analisar o efeito da fusão das rotas formadas para obtenção de economias na roteirização.

## 8.1 Conclusões

As principais conclusões deste estudo são de natureza qualitativa e são relativas à estratégia para resolução de problemas de roteirização com a escolha da frota por tipo de veículo. Estas conclusões são as seguintes:

(1) Os roteiros formados mediante o cômputo conjunto dos custos fixos e variáveis da frota apresentaram, sem exceção,

resultados melhores que os gerados sem esta consideração.

(2) A restrição da jornada de trabalho tem papel decisivo sobre os resultados e deve ser adequadamente conjugada com as diversas alternativas de veículos através dos seus custos fixos e variáveis.

(3) O estudo mostrou que o incentivo à formação de roteiros mais rápidos com veículos menores é uma estratégia muito eficaz para a redução do custo total de roteirização. Esta abordagem está condicionada à utilização de políticas de redução dos custos fixos durante o processo de criação dos roteiros e dos algorítmos de pós-processamento para fusão das rotas assim obtidas.

Não se pôde traçar conclusões de cunho quantitativo relacionando parâmetros do problema ou da frota com o custo total da roteirização pelo seguinte motivo: o PRV é um problema de grande complexidade e que tem uma enorme gama de parâmetros envolvidos, conforme apresentado no trabalho. Qualquer tentativa de sintetizar conclusões quantitativas levaria à particularização escessiva da uso dessas conclusões fugindo ao escopo do trabalho.

## 8.2 Recomendações

Tendo em vista a experiência adquirida através deste trabalho julga-se oportuno estabelecer algumas recomendações com a finalidade de dar prosseguimento a esta linha de pesquisa. São elas:

(1) Avaliar a qualidade dos resultados de possíveis implementações computacionais do modelo proposto através dos seguintes procedimentos: para um mesmo problema, aplicar o modelo desenvolvido para diversos roteiros gigantes criados aleatoriamente e melhorados pelo procedimento 2-opt (a exemplo do SGT de Golden et alii.), pesquisar o modelo com M

cópias de pontos (a exemplo do MGT) e com pós-processamento através do procedimento de troca de arcos entre rotas diferentes (a exemplo do MGT+2-opt).

(2) Estudar outras políticas de premiação para incentivo à realização de viagens mais rápidas por veículos menores, como por exemplo variando o tempo de corte, ou concedendo prêmios decrescentes com o tmepo de viagem.

(3) Implementar o modelo ROS $\hat{\gamma}$ de Golden et alii. para comportar custos variáveis por veículo (além do custo fixo) e avaliar o seu desempenho em relação ao modelo utilizado neste trabalho.

(4) Além do recomendado no ítem anterior, o ROS-g pode ser implementado para a realização de múltiplas viagens por dia, através da redução do custo fixo das rotas.

(5) Finalmente, o modelo desenvolvido para o presente estudo considera a frota ilimitada. Seria muito interessante se implementar a restrição de disponibilidade de veículo por tipo para o uso rotineiro, uma vez estando a frota estabelecida. Para isto, seria necessário se implementar um contador de veículos por tipo no cálculo da distância a cada estágio, no algorítimo de determinação da distância mínima de modo a torná-la infinita toda vez que todos os veículos de um determinado tipo já tiverem sido utilizados. Eventualmente, este método pode sofrer de miopia por tender a utilizar os melhores veículos logo no começo, o que poderia ser compensado por um fator de incentivo ou penalização, após um primeiro processamento.

# REFERENCIAS BIBLIOGRAFICAS


1. Balas, E. e Christofides, N.. "A Restricted Lagrangean Approach to the Traveling-Salesman Problem". Math. Prog.. 21,19-46 (1981).
2. Barachet, L.L.. "Graphic Solution of the Traveling-Salesman Problem". Oper. Res..12, 300-4 (1964).
3. Bellmore, M. e Malone, J.C.. "Pathology of Traveling-Salesman Subtour-Elimination Algorithms". Oper. Res.. 19, 278-307 (1971).
4. Bellmore, M. e Nemhauser, G.L.. "The Traveling-Salesman Problem: A Survey". Oper. Res.. 16, 538-58 (1974).
5. Bodin, L., Golden, B., Assad, A. e Ball, M.. "Routing and Scheduling of Vehicles and Crews, The State of the Art". Comp. Oper. Res.. 10(2), 63-211 (1983).
6. Bodin, L.D. e Golden, B.. "Classification in Vehicle Routing and Scheduling". Networks. 11(2), 97-108 (1981).
7. Christofides, N. e Eilon, S.. "An Algorithm for the Vehicle-Dispatching Problem". Oper. Res. Quart.. 20,309-18 (1969).
8. Christofides, N., Mingozzi, A. e Toth, P.. "State-Space Relaxation Procedures for the Computation of Bounds to Routing Problems. Networks. 11(2), 145-64 (1981).
9. Christofides, N.. "Bounds for the Traveling-Salesman Problem". Oper. Res.. 20, 1044-56 (1972).
10. Clarke, G. e Wright, J.W.. "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points". Oper. Res.. 12, 568-81 (1964).
11. Crowder, H. e Padberg, M.W.. "Solving Large-Scale Symmetric Traveling-Salesman Problems to Optimality". Manag. Sci.. 26(5), 495-509 (1980).
12. Cullen, F.H., Jarvis, J.J. e Ratliff, H.D.. "Set Partitioning Based Heuristics for Interactive Routing".

Networks. 11(2), 125-44 (1981).

13. Dantzig, G., Fulkerson, R. e Johnson, S.. "Solution of a Large-Scale Traveling-Salesman Problem". Oper. Res.. 2(4), 393-410 (1954).

14. Dantzig, G.B. e Ramser, J.H.. "The Truck Dispatching Problem". Manag. Sci.. 6,81-91 (1959).

15. Dror, M. e Levy, L.. "A Vehicle Routing Improvement Algorithm Comparison of a Greedy and a Matching Implementation for Inventory Routing". Comp. Oper. Res.. 13(1), 33-45 (1986).

16. Fisher, M.. "The Lagrangian Relaxation Method for Solving Integer Programming Problems". Manag. Sci.. 27, 1-12 (1981).

17. Fisher, M.L. e Jaikumar, R.. "A Generalized Assignment Heuristic for Vehicle Routing". Networks. Networks. 11(2), 109-24 (1981).

18. Flood, M.M.. "The Traveling-Salesman Problem". Oper. Res.. 4, 61-75 (1956).

19. Gaskell, T.J.. "Bases for Vehicle Fleet Scheduling". Oper. Res. Quart.. 18, 281-95 (1967).

20. Gillet, B.E. e Miller, L.. "A Heuristic Algorithm for the Vehicle-Dispatch Problem". Oper. Res.. 22, 340-49 (1974).

21. Golden, B., Assad, A., Levy, L. e Gheysens, F.. "The Fleet Size and Mix Vehicle Routing Problem". Comp. Oper. Res.. 11(1), 49-66 (1984).

22. Golden, B., Bodin, L., Doyle, T. e Stewart, W.Jr.."Approximate Traveling Salesman Algorithms".Oper. Res.. 28(3), 694-711 (1980).

23. Golden, B.L., Magnanti, T.L. e Nguyen, H.Q.. "Implementing Vehicle Routing Algorithms". Networks. 7, 113-48 (1977).

24. Golden, B.L.. "A Statistical Approach to the TSP". Networks. 7, 209-25 (1977).

25. Harverly, R.C. e Seber, J.J.. "Physical Distribution Software 1986 Edition". Arthur Andersen & Co. for The Council of Logistics Management. Stamford, USA. (1986).

26. Held, M. e Karp, R.M.. "The Traveling-Salesman Problem and

Minimum Spanning Trees". Oper. Res.. 18, 1138-62 (1970).

27. Held, M. e Karp, R.M.. "The Traveling-Salesman Problem and Minimum Spanning Trees: Part II". Math. Prog.. 1, 6-25 (1971).

28. IBM Corporation. "IBM Vehicle Scheduling Program-Extended Program Description Manual". Stuttgart, Germany. (1970).

29. Lawler, E.L. e Wood, D.E.. "Branch-and-Bound Methods: A Survey". Oper. Res.. (1966).

30. Lenstra, J.K. e Rinnooy Kan, A.H.G.. "Complexity of Vehicle Routing and Scheduling Problems". Networks. 11(2), 221-7 (1981).

31. Lin, S. e Kernighan, B.W.. "An Effective Heuristic Alghorithm for the Traveling-Salesman Problem". Oper. Res.. 21, 498-516 (1973).

32. Little, J.D.C., Murty, K.G., Sweeney, D.W. e Karel, C.. "An Algorithm for the Traveling-Salesman Problem". Oper. Res.. 11(6), 972-89 (1963).

33. Magnanti, T.L.. "Combinatorial Optimization and Vehicle Fleet Planning: Perspectives and Prospectcs". Networks. 11(2), 179-214 (1981).

34. Miliotis, P.. "Integer Programming Approaches to the Traveling-Salesman Problem". Math. Prog.. 10, 367-78 (1976).

35. Miliotis, P.. "Using Cutting Planes to Solve the Symmetric Traveling-Salesman Problem". Math. Prog.. 15, 177-88 (1978).

36. Norback, J.P. e Love, R.F.. "Geometric Approaches to Solving the Traveling-Salesman Problem". Manag. Sci.. 23, 1208-23 (1977).

37. Novaes, A.G.N.. "Designing Aspects of a Retail Delivery Service". Rev. Bras. Tecnolog.. 6, 155-67 (1975).

38. Parker, R.G. e Rardin, R.L.. "The Traveling-Salesman Problem: An Update of Research". Nav. Res. Log. Quart.. (1983).

39. Russell, R. e Igo, W.. "An Assignment Routing Problem". Networks. 9(1), 1-17 (1979).

40. Russell, R.A.. "An Effective Heuristic for the M-Tour

Traveling-Salesman Problem with Some Side Conditions". Oper. Res.. 25(3), 517-24 (1977).

41. Saigal, R.. "A Constrained Shortest Route Problem". Oper. Res.. (1967).

42. Schrage, L.. "Formulation and Structure of More Complex/Realistic Routing and Scheduling Problems". Networks. 11(2), 229-32 (1981).

43. Shimizu, T.. Simulação em Computador Digital. 80-3. Ed. Edgard Blucher & Universidade de São Paulo. São Paulo. (1975).

44. Woolsey, R.E.D. e Swanson, H.S.. "Operations Research for Immediate Application, A Quick & Dirty Manual". Harper & Row, U.S.A. 159-61.

45. Yellow, P.. "A Computational Modification to the Savings Method of Vehicle Scheduling". Oper. Res. Quart.. 21, 281-3 (1970).

# APENDICE A

## ANALISE DA REDE

São duas as abordagens básicas para a definição da rede viária: o método das distâncias reais, e o método das coordenadas. A decisão de escolha é, segundo o manual da IBM [28], influenciada pelos seguintes fatores:

. Grau de precisão desejado
. Volume de dados de entrada
. Levantamento e preparação dos dados
. Tempo de processamento
. Extensão e frequência das mudanças da rede.

### 1. Método das Distâncias Reais

O método das distâncias reais gera um resultado com maior nível de precisão, uma vez que determina as distâncias reais, e permite a utilização de velocidades específicas para cada via da rede. O uso deste método é particularmente indicado quando se tem janelas de tempo que demandam tempos de viagem mais precisos. Apesar de ser mais indicado para redes estáticas, é perfeitamente possível de ser utilizado em redes em expansão, contanto que sejam feitas previsões de novas zonas.

Uma rede definida pelo método das distâncias reais é descrita por arcos - ou "links" - e pontos. Um link é um trecho de via urbana conectando dois pontos. Um ponto pode ser a representação de uma zona de entregas, ou um nó, que é meramente uma intersecção ou uma junção de ruas. Um detalhe que cabe mencionar é que a rede não necessariamente deve conter todas as vias do mapa, mas somente aquelas, que como links, permitirão encontrar os menores caminhos entre todas as zonas.

O processamento do programa de análise da rede dispende um tempo sensivelmente maior devido a maior quantidade de dados envolvidos. Além disso, o programa precisa determinar as menores distâncias entre cada par de pontos não adjacentes da rede.

O uso da decomposição no método das distâncias reais pode reduzir consideralvelmente o tempo de processamento, sem afetar a precisão dos resultados. Isto é feito dividindo-se o conjunto de todos os pontos em três segmentos: dois grupos de pontos aproximadamente iguais em número, chamados de segmento Y e segmento Z, e um terceiro grupo separando os dois anteriores, que é chamado de segmento X. Isto permite o programa de análise da rede manipular cada segmento separadamente. A decomposição deve obedecer aos seguintes critérios:

(1) O segmento X precisa conter o depósito, assim como as zonas ou nós que servem como conectores de viages entre pontos do segmento Y e pontos do segmento Z.

(2) Nenhum caminho deve existir, de um ponto em Y a outro em Z, sem passar por ao menos um ponto em X.

(3) O menor caminho entre todos dos pares de pontos em X deve estar completamente imerso no segmento X.

Segundo o manual da IBM, a decomposiçào torna-se eficiente para redes com mais de 400 pontos.

## 2. Método das Coordenadas

O método das coordenadas determina as distâncias de forma aproximada. Ao analisar a rede, considera somente a distância em linha reta entre pontos, ajustando-o em seguida por um fator para compensar o desvio em relação à distância real. Na média os resultados são razoavelmente aceitáveis. Algumas distâncias serão um pouco menores e outras um pouco maiores.

Para representar irregularidades no traçado ou percurso pelas vias são utilizadas barreiras e áreas de congestionamento.

As barreiras são segmentos de reta definidas pelas coordenadas das suas extremidades, e servem para representar rios, fundos de vale, contornos de morro, ou vias expressas, transponíveis apenas por pontes ou viadutos. Nenhuma linha reta ligando dois pontos quaisquer pode passar através dela. A distância neste caso é calculada considerando-se o desvio que tem que ser feito, por uma das extremidades da barreira (a que for livre ou mais próxima).

Uma área congestionada é representada por um circulo e descrita pelas coordenadas do seu centro e pelo comprimento do seu raio. O trecho da linha reta, que liga um par de pontos quaisquer, que se encontra no interior do círculo, tem um tempo de percurso ajustado pela velocidade vigente na área.

A extensão e frequência das mudanças da rede podem ter um considerável efeito sobre o tempo total dispendido na atualização da rede, tanto na manipulação como no processamento de dados.

O método das coordenadas requer um trabalho menor de preparação de dados e geralmente demanda um tempo de execução menor que o método das distâncias reais. E indicado para situações onde a simplicidade de levantamento de dados e a rapidez com que é feita a análise da rede são aspectos importantes. Uma rede que muda frequentemente e que não justifica o tempo adicional e esforço requerido para o método das distâncias reais é um candidato natural ao método das coordenadas. Entretanto, se um número muito grande de barreiras ou áreas congestionadas for usado, o tempo de processamento pode exceder ao requerido pelo método da distâncias reais, sem necessariamente fornecer o mesmo nível de precisão.

# APENDICE B

# IMPLEMENTAÇÃO EM COMPUTADOR

Este apêndice tem o objetivo de complementar a relação de procedimentos heurísticos apresentados no Capítulo 5 no que tange a sua implementação em computador.

## 1. Dificuldade Computacional

Segundo Bodin et alii. [5], uma consideração bastante importante que se tem que fazer é a análise da carga computacional associada à técnica de solução a ser empregada para o problema. Geralmente, o esforço computacional cresce à medida que aumenta o tamanho do problema. Se este crescimento é muito rápido, a capacidade computacional logo torna-se proibitiva para problemas de porte moderado, limitando a aplicabilidade da técnica num contexto prático, onde os problemas encontrados são típicamente grandes.

Lenstra e Kan [30] chama de problema de dificuldade Nào Polinomial ("NP-hard") a aquele que não pode ser resolvido num tempo limitado polinomialmente em relação ao seu tamanho, no caso pelo número de nós. A resolução exata dos problemas do caixeiro viajante, de roteirização e de programação de veículos são considerados problemas desta categoria, dada a complexidade computacional envolvida. Estes problemas, segundo Bodin et alii., apresentam tempos de processamento exponenciais em função do seu tamanho, o que leva as pessoas a procurarem métodos de abordagem heurística ou procedimentos aproximados para obterem soluções aproximadas, ao invés da solução ótima.

## 2. A Abordagem de Yellow

Yellow [45] descobriu que a montagem incial de um arquivo de economias (para a utilização do procedimento de economia de Clarke e Wright) pode ser eliminada usando-se uma técnica geométrica simples através de uma lista ordenada de coordenadas polares dos pontos de entrega. Modificando a formulação feita por Gaskell para o cálculo da economia [19], Yellow definiu o parâmetro $\gamma$ de formato da rota segundo

$$S_{ij} = d_{ii} + d_{ij} - \gamma d_{ij} \qquad \text{(A.1)}$$

onde: $S_{ij}$ = economia por inserção,
$d_{ij}$ = distância entre os pontos i e j.

Conforme $\gamma$ cresce, maior ênfase é colocada sobre a distância entre os pontos i e j, ao invés da sua distância em relação ao depósito.

Yellow prova matematicamente que existe um local geométrico "loc-i" onde os pontos geram economias iguais quando ligados ao ponto i. Yelow comparou seu procedimento em 6 problemas de 21 a 100 pontos com os algorítmos de Gaskell (método de economias múltiplas), Christofides e Eilon (3-opt) e conclui que os tempos de processamento obtidos foram significativamente melhores, apesar de perder um pouco na qualidade dos resultados principalmente para o algorítmo de Christofides e Eilon (3-opt).

## 3. A Abordagem de Golden

Golden et alii. [23] realizaram diversas implementações a partir do procedimento de Clarke e Wright modificado por Yellow, para a roteirização de um sistema de distribuição de jornais contendo cerca de 600 pontos de entrega.

Foi pesquisada, dentre outras coisas, a influência do parâmetro b na roterização de grandes problemas.

Por estar se considerando uma matriz simétrica de custos, se precisaria, em princípio, somente de meia matriz para armazenar as distâncias simétricas, deixando as economias na outra metade. A rigor ainda, nem todas as distâncias precisariam ser armazenadas. Apenas aquelas necessárias para a variação de b seriam guardadas.

Percebendo que existem economias relativas a ligações que obviamente nunca serão realizadas, foi idealizado o seguinte artifício: sobrepondo uma grade com sub-divisões retangulares sobre o mapa, definiu-se que seriam armazenados somente os arcos que ligam o depósito a todos os demais nós, bem como os arcos que ligam todos os pares de nós situados no mesmo retângulo ou no máximo em retângulos vizinhos. Quanto menor o tamanho dos retângulos, para um mesmo problema, maior é o número de arcos a serem considerados.

Um problema de 200 pontos (e 4 depósitos) necessitou de 200Kb de memória, um problema de 300 nós de 400 Kb, e um de 600 nós de 1.5Mb. Os resultados em tempo de processamento foram considerados encorajadores. Um problema de 50 nós foi resolvido em menos de 1 segundo num IBM 370/168. Para o problema real de 600 nós, o tempo de processamento foi de 20 segundos.

Quanto ao parâmetro de formato de rota b, Golden et al. concluiram que a variação de b permite a busca de melhores soluções que as obtidas pelo algorítmo tradicional onde b = 1. Além disso, confirmaram a suspeita de que ele é dependente de cada problema em particular.

## 4. A Abordagem do VSPX da IBM

A IBM em seu VSPX [28] utiliza um outro critério para redução da matriz de economias obtendo em média um ganho de 40 a 60% dependendo do formato da rede e da localização do depósito. E realizado um procedimento matemático antes da formação do arquivo de economias, que considera o ângulo medido a partir do depósito entre todos os pares de pontos. São eliminados todos aqueles que tiverem um ângulo maior que um valor limite.

Mas o critério mais interessante e prático estabelecido no VSPX para a redução do problema é a utilização do conceito de "zona" para a montagem da rede. Analisando-se a localização dos clientes sobre um mapa, é sempre possível se encontrar clientes que podem ser agrupados devido a sua relativa proximidade. Uma zona deve representar não somente um único cliente, mas sempre que razoável, uma área contendo diversos clientes. Como entrada na rede, uma zona é representada simplesmente por um ponto, que pode ser alocado em qualquer ponto razoável da zona. Uma vez que uma zona é uma área onde ocorrem pequenas viagens de um ponto a outro, assume-se no programa um tempo médio de viagem constante entre pontos.

Somente as ligações de zona para zona, ou depósito para zona (e vice-versa), são consideradas no cálculo das economias. A economia para uma ligação de pontos de uma mesma zona será sempre maior que a economia de ligação de pares de pontos de zonas diferentes, sendo facilmente demonstrável por geometria.

# APENDICE C

## DETERMINAçÃO DO MENOR CAMINHO

O algorítmo utilizado de determinação do menor caminho entre dois pontos (1 e N+1) através de uma rede foi o de Dijkstra e foi baseado na representação de Woolsey e Swanson [44]. Seus passos são os seguintes:

1. Definir $d_{i,j}$ = distância do nó i ao nó j se o arco existir, ou $d_{i,j} = \infty$ caso contrário.

2. Fazer $f_{1,j} = d_{1,j}$ para o estágio 1.

3. Fazer $f_{k,j} = \min_i \{ f_{k-1,i} + d_{i,j} \}$

4. Depois de ka = n estágios parar.

5. O menor caminho é dado pelos arcos (i,j) tais que:

$$d_{i,j} = f_{ka,j} - f_{ka,i}$$

e a distância do menor caminho é dada por $f_{ki}$.

onde: $f_{1,j}$ = menor distância do ponto de origem 1 ao ponto j, no primeiro estágio,
$f_{k,j}$ = menor distância do ponto 1 ao ponto j, no k-ésimo estágio,

# APENDICE D
## BATERIA DE DADOS

Os dados utilizados neste trabalho são baseados nos mesmos usados por Golden et alii. [21] salvo quando devidamente indicado no texto. A tabela seguinte apresenta os dados relativos as diversas frotas, que são identificadas pelo sinal #.

TABELA D.1
FROTAS DE GOLDEN ET AlII.

| FROTA | VEICULO | CAPACIDADE | CUSTO FIXO |
|---|---|---|---|
| #1 | A | 15 | $20 |
| | B | 35 | $50 |
| | C | 60 | $100 |
| #2 | A | 30 | $60 |
| | B | 40 | $90 |
| | C | 110 | $300 |
| #3 | A | 20 | $20 |
| | B | 30 | $35 |
| | C | 40 | $50 |
| | D | 70 | $120 |
| | E | 120 | $225 |
| #4 | A | 60 | $1000 |
| | B | 80 | $1500 |
| | C | 150 | $3000 |

TABELA D.1

FROTAS DE GOLDEN ET ALII.

(CONTINUAÇAO)

| FROTA | VEICULO | CAPACIDADE | CUSTO FIXO |
|---|---|---|---|
| #5 | A | 20 | $20 |
| | B | 30 | $35 |
| | C | 40 | $50 |
| | D | 70 | $120 |
| | E | 120 | $225 |
| #6 | A | 60 | $1000 |
| | B | 80 | $1500 |
| | C | 150 | $3000 |
| #7 | A | 40 | $150 |
| | B | 100 | $500 |
| | C | 140 | $800 |
| | D | 200 | $1200 |
| | E | 300 | $200 |
| #8 | A | 10 | $15 |
| | B | 50 | $50 |
| | C | 150 | $200 |
| | D | 400 | $600 |
| #9 | A | 40 | $30 |
| | B | 100 | $100 |
| | C | 140 | $160 |
| | D | 200 | $240 |
| | E | 300 | $400 |

TABELA D.1

FROTAS DE GOLDEN ET ALII.

(CONTINUAÇAO)

| FROTA | VEICULO | CAPACIDADE | CUSTO FIXO |
|---|---|---|---|
| #10 | A | 40 | $30 |
| | B | 100 | $100 |
| | C | 140 | $160 |
| | D | 200 | $240 |
| #11 | A | 30 | $60 |
| | B | 80 | $200 |
| | C | 200 | $700 |
| | D | 350 | $1500 |
| #12 | A | 30 | $40 |
| | B | 50 | $80 |
| | C | 75 | $150 |
| | D | 120 | $300 |
| | E | 180 | $500 |
| | F | 250 | $800 |
| #13 | A | 20 | $20 |
| | B | 30 | $35 |
| | C | 40 | $50 |
| | D | 70 | $120 |
| | E | 120 | $225 |
| | F | 200 | $400 |
| #14 | A | 120 | $1000 |
| | B | 160 | $1500 |
| | C | 300 | $3500 |

TABELA D.1

FROTAS DE GOLDEN ET ALII.

(CONTINUAÇÃO)

| FROTA | VEICULO | CAPACIDADE | CUSTO FIXO |
|---|---|---|---|
| #15 | A | 50 | $100 |
| | B | 100 | $250 |
| | C | 160 | $450 |
| #16 | A | 40 | $100 |
| | B | 80 | $200 |
| | C | 140 | $400 |
| #17 | A | 50 | $25 |
| | B | 120 | $80 |
| | C | 200 | $150 |
| | D | 350 | $320 |
| #18 | A | 20 | $10 |
| | B | 50 | $35 |
| | C | 100 | $100 |
| | D | 150 | $180 |
| | E | 250 | $400 |
| | F | 400 | $800 |
| #19 | A | 100 | $500 |
| | B | 120 | $1200 |
| | C | 300 | $2100 |
| #20 | A | 60 | $100 |
| | B | 140 | $300 |
| | C | 200 | $500 |

As tabelas seguintes apresentam os problemas utilizados neste trabalho. Estes foram criados por Clarke e Wright [10] (problemas CW13 e CW31), Christofides e Eilon [7] (problemas CE51, CE76 e CE101) e Golden et alii. [21] (problemas CE21-51A e CE51-76) e encontram-se organizados na ordem apresentada abaixo:

1. CW13: com 13 pontos (considerando-se o depósito).

2. CE21-51A: primeiros 21 pontos do problema CE51.

3. CW31: com 31 pontos.

4. CE51-76: primeiros 51 pontos do problema CE76.

5. CE51: com 51 pontos.

6. CE76: com 76 pontos.

7. CE101: com 100 pontos.

CW 13

| I | X | Y | P |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 12 |
| 3 | 0 | 0 | 17 |
| 4 | 0 | 0 | 15 |
| 5 | 0 | 0 | 14 |
| 6 | 0 | 0 | 17 |
| 7 | 0 | 0 | 14 |
| 8 | 0 | 0 | 12 |
| 9 | 0 | 0 | 19 |
| 10 | 0 | 0 | 18 |
| 11 | 0 | 0 | 16 |
| 12 | 0 | 0 | 17 |
| 13 | 0 | 0 | 11 |

| I | J | DIST |
|---|---|---|
| 1 | 2 | 9 |
| 1 | 3 | 14 |
| 1 | 4 | 21 |
| 1 | 5 | 23 |
| 1 | 6 | 22 |
| 1 | 7 | 25 |
| 1 | 8 | 32 |
| 1 | 9 | 36 |
| 1 | 10 | 38 |
| 1 | 11 | 42 |
| 1 | 12 | 50 |
| 1 | 13 | 52 |

| I | J | DIST |
|---|---|---|
| 2 | 3 | 5 |
| 2 | 4 | 12 |
| 2 | 5 | 22 |
| 2 | 6 | 21 |
| 2 | 7 | 24 |
| 2 | 8 | 31 |
| 2 | 9 | 35 |
| 2 | 10 | 37 |
| 2 | 11 | 41 |
| 2 | 12 | 49 |
| 2 | 13 | 51 |

| I | J | DIST |
|---|---|---|
| 3 | 4 | 7 |
| 3 | 5 | 17 |
| 3 | 6 | 16 |
| 3 | 7 | 23 |
| 3 | 8 | 26 |
| 3 | 9 | 30 |
| 3 | 10 | 36 |
| 3 | 11 | 36 |
| 3 | 12 | 44 |
| 3 | 13 | 46 |

| I | J | DIST |
|---|---|---|
| 4 | 5 | 10 |
| 4 | 6 | 21 |
| 4 | 7 | 30 |
| 4 | 8 | 27 |
| 4 | 9 | 37 |
| 4 | 10 | 43 |
| 4 | 11 | 31 |
| 4 | 12 | 37 |
| 4 | 13 | 39 |

| I | J | DIST |
|---|---|---|
| 5 | 6 | 19 |
| 5 | 7 | 28 |
| 5 | 8 | 25 |
| 5 | 9 | 35 |
| 5 | 10 | 41 |
| 5 | 11 | 29 |
| 5 | 12 | 31 |
| 5 | 13 | 29 |

| I | J | DIST |
|---|---|---|
| 6 | 7 | 9 |
| 6 | 8 | 10 |
| 6 | 9 | 16 |
| 6 | 10 | 22 |
| 6 | 11 | 20 |
| 6 | 12 | 28 |
| 6 | 13 | 30 |

| I | J | DIST |
|---|---|---|
| 7 | 8 | 7 |
| 7 | 9 | 11 |
| 7 | 10 | 13 |
| 7 | 11 | 17 |
| 7 | 12 | 25 |
| 7 | 13 | 27 |

| I | J | DIST |
|---|---|---|
| 8 | 9 | 10 |
| 8 | 10 | 16 |
| 8 | 11 | 10 |
| 8 | 12 | 18 |
| 8 | 13 | 20 |

| I | J | DIST |
|---|---|---|
| 9 | 10 | 6 |
| 9 | 11 | 6 |
| 9 | 12 | 14 |
| 9 | 13 | 16 |

| I | J | DIST |
|---|---|---|
| 10 | 11 | 12 |
| 10 | 12 | 12 |
| 10 | 13 | 20 |

| I | J | DIST |
|---|---|---|
| 11 | 12 | 8 |
| 11 | 13 | 10 |

| I | J | DIST |
|---|---|---|
| 12 | 13 | 10 |

CW 31

| I | X | Y | P |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 4 |
| 3 | 0 | 0 | 14 |
| 4 | 0 | 0 | 11 |
| 5 | 0 | 0 | 15 |
| 6 | 0 | 0 | 11 |
| 7 | 0 | 0 | 1 |
| 8 | 0 | 0 | 3 |
| 9 | 0 | 0 | 9 |
| 10 | 0 | 0 | 6 |
| 11 | 0 | 0 | 5 |
| 12 | 0 | 0 | 6 |
| 13 | 0 | 0 | 5 |
| 14 | 0 | 0 | 2 |
| 15 | 0 | 0 | 8 |
| 16 | 0 | 0 | 8 |
| 17 | 0 | 0 | 10 |
| 18 | 0 | 0 | 18 |
| 19 | 0 | 0 | 5 |
| 20 | 0 | 0 | 13 |
| 21 | 0 | 0 | 17 |
| 22 | 0 | 0 | 9 |
| 23 | 0 | 0 | 16 |
| 24 | 0 | 0 | 15 |
| 25 | 0 | 0 | 5 |
| 26 | 0 | 0 | 0 |
| 27 | 0 | 0 | 0 |
| 28 | 0 | 0 | 19 |
| 29 | 0 | 0 | 15 |
| 30 | 0 | 0 | 10 |
| 31 | 0 | 0 | 3 |

| I | J | DIST |
|---|---|---|
| 1 | 2 | 41 |
| 1 | 3 | 38 |
| 1 | 4 | 80 |
| 1 | 5 | 80 |
| 1 | 6 | 97 |
| 1 | 7 | 92 |
| 1 | 8 | 96 |
| 1 | 9 | 78 |
| 1 | 10 | 98 |
| 1 | 11 | 87 |
| 1 | 12 | 95 |
| 1 | 13 | 77 |
| 1 | 14 | 93 |
| 1 | 15 | 91 |
| 1 | 16 | 98 |
| 1 | 17 | 96 |
| 1 | 18 | 40 |
| 1 | 19 | 73 |
| 1 | 20 | 82 |
| 1 | 21 | 55 |
| 1 | 22 | 52 |
| 1 | 23 | 76 |
| 1 | 24 | 76 |
| 1 | 25 | 76 |
| 1 | 26 | 72 |
| 1 | 27 | 98 |
| 1 | 28 | 98 |
| 1 | 29 | 93 |
| 1 | 30 | 89 |
| 1 | 31 | 68 |

| I | J | DIST |
|---|---|---|
| 2 | 3 | 3 |
| 2 | 4 | 54 |
| 2 | 5 | 54 |
| 2 | 6 | 64 |
| 2 | 7 | 59 |
| 2 | 8 | 56 |
| 2 | 9 | 39 |
| 2 | 10 | 59 |
| 2 | 11 | 52 |
| 2 | 12 | 58 |
| 2 | 13 | 38 |
| 2 | 14 | 55 |
| 2 | 15 | 52 |
| 2 | 16 | 58 |
| 2 | 17 | 59 |
| 2 | 18 | 45 |
| 2 | 19 | 34 |
| 2 | 20 | 48 |
| 2 | 21 | 16 |
| 2 | 22 | 16 |
| 2 | 23 | 46 |
| 2 | 24 | 44 |
| 2 | 25 | 50 |
| 2 | 26 | 33 |
| 2 | 27 | 58 |
| 2 | 28 | 58 |
| 2 | 29 | 66 |
| 2 | 30 | 55 |
| 2 | 31 | 32 |

| I | J | DIST |
|---|---|---|
| 3 | 4 | 56 |
| 3 | 5 | 56 |
| 3 | 6 | 67 |
| 3 | 7 | 62 |
| 3 | 8 | 59 |
| 3 | 9 | 41 |
| 3 | 10 | 62 |
| 3 | 11 | 50 |
| 3 | 12 | 61 |
| 3 | 13 | 41 |
| 3 | 14 | 58 |
| 3 | 15 | 53 |
| 3 | 16 | 61 |
| 3 | 17 | 62 |
| 3 | 18 | 5 |
| 3 | 19 | 37 |
| 3 | 20 | 46 |
| 3 | 21 | 19 |
| 3 | 22 | 17 |
| 3 | 23 | 49 |
| 3 | 24 | 46 |
| 3 | 25 | 53 |
| 3 | 26 | 34 |
| 3 | 27 | 61 |
| 3 | 28 | 61 |
| 3 | 29 | 68 |
| 3 | 30 | 58 |
| 3 | 31 | 33 |

| I | J | DIST |
|---|---|---|
| 4 | 5 | 3 |
| 4 | 6 | 19 |
| 4 | 7 | 13 |
| 4 | 8 | 16 |
| 4 | 9 | 54 |
| 4 | 10 | 20 |
| 4 | 11 | 47 |
| 4 | 12 | 15 |
| 4 | 13 | 30 |
| 4 | 14 | 15 |
| 4 | 15 | 25 |
| 4 | 16 | 19 |
| 4 | 17 | 17 |
| 4 | 18 | 60 |
| 4 | 19 | 46 |
| 4 | 20 | 44 |
| 4 | 21 | 54 |
| 4 | 22 | 68 |
| 4 | 23 | 8 |
| 4 | 24 | 11 |
| 4 | 25 | 4 |
| 4 | 26 | 53 |
| 4 | 27 | 33 |
| 4 | 28 | 32 |
| 4 | 29 | 14 |
| 4 | 30 | 10 |
| 4 | 31 | 64 |

| I | J | DIST |
|---|---|---|
| 5 | 6 | 16 |
| 5 | 7 | 10 |
| 5 | 8 | 14 |
| 5 | 9 | 54 |
| 5 | 10 | 17 |
| 5 | 11 | 46 |
| 5 | 12 | 12 |
| 5 | 13 | 29 |
| 5 | 14 | 12 |
| 5 | 15 | 22 |
| 5 | 16 | 16 |
| 5 | 17 | 14 |
| 5 | 18 | 61 |
| 5 | 19 | 46 |
| 5 | 20 | 44 |
| 5 | 21 | 54 |
| 5 | 22 | 68 |
| 5 | 23 | 9 |
| 5 | 24 | 11 |
| 5 | 25 | 4 |
| 5 | 26 | 54 |
| 5 | 27 | 30 |
| 5 | 28 | 29 |
| 5 | 29 | 12 |
| 5 | 30 | 9 |
| 5 | 31 | 64 |

| I | J | DIST |
|---|---|---|
| 6 | 7 | 7 |
| 6 | 8 | 11 |
| 6 | 9 | 53 |
| 6 | 10 | 12 |
| 6 | 11 | 46 |
| 6 | 12 | 8 |
| 6 | 13 | 34 |
| 6 | 14 | 10 |
| 6 | 15 | 24 |
| 6 | 16 | 10 |
| 6 | 17 | 8 |
| 6 | 18 | 71 |
| 6 | 19 | 50 |
| 6 | 20 | 45 |
| 6 | 21 | 58 |
| 6 | 22 | 77 |
| 6 | 23 | 19 |
| 6 | 24 | 20 |
| 6 | 25 | 20 |
| 6 | 26 | 57 |
| 6 | 27 | 27 |
| 6 | 28 | 26 |
| 6 | 29 | 23 |
| 6 | 30 | 8 |
| 6 | 31 | 67 |

| I | J | DIST |
|---|---|---|
| 7 | 8 | 10 |
| 7 | 9 | 57 |
| 7 | 10 | 13 |
| 7 | 11 | 42 |
| 7 | 12 | 8 |
| 7 | 13 | 32 |
| 7 | 14 | 14 |
| 7 | 15 | 19 |
| 7 | 16 | 10 |
| 7 | 17 | 8 |
| 7 | 18 | 65 |
| 7 | 19 | 46 |
| 7 | 20 | 42 |
| 7 | 21 | 55 |
| 7 | 22 | 72 |
| 7 | 23 | 15 |
| 7 | 24 | 15 |
| 7 | 25 | 14 |
| 7 | 26 | 55 |
| 7 | 27 | 30 |
| 7 | 28 | 29 |
| 7 | 29 | 18 |
| 7 | 30 | 5 |
| 7 | 31 | 66 |

| I | J | DIST |
|---|---|---|
| 8 | 9 | 48 |
| 8 | 10 | 4 |
| 8 | 11 | 35 |
| 8 | 12 | 3 |
| 8 | 13 | 25 |
| 8 | 14 | 3 |
| 8 | 15 | 12 |
| 8 | 16 | 4 |
| 8 | 17 | 4 |
| 8 | 18 | 63 |
| 8 | 19 | 39 |
| 8 | 20 | 33 |
| 8 | 21 | 48 |
| 8 | 22 | 69 |
| 8 | 23 | 18 |
| 8 | 24 | 15 |
| 8 | 25 | 18 |
| 8 | 26 | 47 |
| 8 | 27 | 21 |
| 8 | 28 | 20 |
| 8 | 29 | 22 |
| 8 | 30 | 7 |
| 8 | 31 | 57 |

| I | J | DIST |
|---|---|---|
| 9 | 10 | 39 |
| 9 | 11 | 12 |
| 9 | 12 | 45 |
| 9 | 13 | 24 |
| 9 | 14 | 47 |
| 9 | 15 | 30 |
| 9 | 16 | 42 |
| 9 | 17 | 44 |
| 9 | 18 | 40 |
| 9 | 19 | 8 |
| 9 | 20 | 9 |
| 9 | 21 | 22 |
| 9 | 22 | 36 |
| 9 | 23 | 44 |
| 9 | 24 | 42 |
| 9 | 25 | 50 |
| 9 | 26 | 6 |
| 9 | 27 | 27 |
| 9 | 28 | 28 |
| 9 | 29 | 65 |
| 9 | 30 | 48 |
| 9 | 31 | 22 |

| I | J | DIST |
|---|---|---|
| 10 | 11 | 33 |
| 10 | 12 | 6 |
| 10 | 13 | 21 |
| 10 | 14 | 7 |
| 10 | 15 | 9 |
| 10 | 16 | 3 |
| 10 | 17 | 5 |
| 10 | 18 | 66 |
| 10 | 19 | 39 |
| 10 | 20 | 31 |
| 10 | 21 | 45 |
| 10 | 22 | 65 |
| 10 | 23 | 22 |
| 10 | 24 | 19 |
| 10 | 25 | 21 |
| 10 | 26 | 45 |
| 10 | 27 | 15 |
| 10 | 28 | 15 |
| 10 | 29 | 25 |
| 10 | 30 | 10 |
| 10 | 31 | 55 |

| I | J | DIST |
|---|---|---|
| 11 | 12 | 39 |
| 11 | 13 | 18 |
| 11 | 14 | 39 |
| 11 | 15 | 24 |
| 11 | 16 | 36 |
| 11 | 17 | 38 |
| 11 | 18 | 49 |
| 11 | 19 | 12 |
| 11 | 20 | 4 |
| 11 | 21 | 30 |
| 11 | 22 | 46 |
| 11 | 23 | 40 |
| 11 | 24 | 36 |
| 11 | 25 | 43 |
| 11 | 26 | 15 |
| 11 | 27 | 18 |
| 11 | 28 | 20 |
| 11 | 29 | 54 |
| 11 | 30 | 39 |
| 11 | 31 | 38 |

| I | J | DIST |
|---|---|---|
| 12 | 13 | 28 |
| 12 | 14 | 3 |
| 12 | 15 | 15 |
| 12 | 16 | 4 |
| 12 | 17 | 2 |
| 12 | 18 | 65 |
| 12 | 19 | 43 |
| 12 | 20 | 36 |
| 12 | 21 | 53 |
| 12 | 22 | 71 |
| 12 | 23 | 16 |
| 12 | 24 | 18 |
| 12 | 25 | 17 |
| 12 | 26 | 49 |
| 12 | 27 | 19 |
| 12 | 28 | 18 |
| 12 | 29 | 20 |
| 12 | 30 | 5 |
| 12 | 31 | 63 |

| I | J | DIST |
|---|---|---|
| 13 | 14 | 26 |
| 13 | 15 | 14 |
| 13 | 16 | 24 |
| 13 | 17 | 26 |
| 13 | 18 | 40 |
| 13 | 19 | 16 |
| 13 | 20 | 18 |
| 13 | 21 | 24 |
| 13 | 22 | 44 |
| 13 | 23 | 20 |
| 13 | 24 | 18 |
| 13 | 25 | 25 |
| 13 | 26 | 22 |
| 13 | 27 | 19 |
| 13 | 28 | 19 |
| 13 | 29 | 41 |
| 13 | 30 | 29 |
| 13 | 31 | 34 |

| I | J | DIST |
|---|---|---|
| 14 | 15 | 14 |
| 14 | 16 | 6 |
| 14 | 17 | 4 |
| 14 | 18 | 62 |
| 14 | 19 | 41 |
| 14 | 20 | 36 |
| 14 | 21 | 51 |
| 14 | 22 | 68 |
| 14 | 23 | 17 |
| 14 | 24 | 14 |
| 14 | 25 | 16 |
| 14 | 26 | 49 |
| 14 | 27 | 21 |
| 14 | 28 | 20 |
| 14 | 29 | 20 |
| 14 | 30 | 5 |
| 14 | 31 | 60 |

| I | J | DIST |
|---|---|---|
| 15 | 16 | 12 |
| 15 | 17 | 14 |
| 15 | 18 | 54 |
| 15 | 19 | 28 |
| 15 | 20 | 21 |
| 15 | 21 | 38 |
| 15 | 22 | 57 |
| 15 | 23 | 24 |
| 15 | 24 | 18 |
| 15 | 25 | 28 |
| 15 | 26 | 34 |
| 15 | 27 | 8 |
| 15 | 28 | 7 |
| 15 | 29 | 32 |
| 15 | 30 | 18 |
| 15 | 31 | 47 |

| I | J | DIST |
|---|---|---|
| 16 | 17 | 2 |
| 16 | 18 | 65 |
| 16 | 19 | 42 |
| 16 | 20 | 34 |
| 16 | 21 | 48 |
| 16 | 22 | 67 |
| 16 | 23 | 20 |
| 16 | 24 | 20 |
| 16 | 25 | 20 |
| 16 | 26 | 46 |
| 16 | 27 | 17 |
| 16 | 28 | 16 |
| 16 | 29 | 24 |
| 16 | 30 | 9 |
| 16 | 31 | 58 |

| I | J | DIST |
|---|---|---|
| 17 | 18 | 66 |
| 17 | 19 | 44 |
| 17 | 20 | 35 |
| 17 | 21 | 50 |
| 17 | 22 | 69 |
| 17 | 23 | 18 |
| 17 | 24 | 18 |
| 17 | 25 | 19 |
| 17 | 26 | 48 |
| 17 | 27 | 19 |
| 17 | 28 | 18 |
| 17 | 29 | 22 |
| 17 | 30 | 7 |
| 17 | 31 | 60 |

| I | J | DIST |
|---|---|---|
| 18 | 19 | 36 |
| 18 | 20 | 45 |
| 18 | 21 | 18 |
| 18 | 22 | 14 |
| 18 | 23 | 52 |
| 18 | 24 | 47 |
| 18 | 25 | 57 |
| 18 | 26 | 34 |
| 18 | 27 | 60 |
| 18 | 28 | 60 |
| 18 | 29 | 72 |
| 18 | 30 | 62 |
| 18 | 31 | 32 |

| I | J | DIST |
|---|---|---|
| 19 | 20 | 9 |
| 19 | 21 | 22 |
| 19 | 22 | 36 |
| 19 | 23 | 37 |
| 19 | 24 | 35 |
| 19 | 25 | 41 |
| 19 | 26 | 6 |
| 19 | 27 | 26 |
| 19 | 28 | 26 |
| 19 | 29 | 57 |
| 19 | 30 | 44 |
| 19 | 31 | 26 |

| I | J | DIST |
|---|---|---|
| 20 | 21 | 31 |
| 20 | 22 | 45 |
| 20 | 23 | 35 |
| 20 | 24 | 33 |
| 20 | 25 | 40 |
| 20 | 26 | 15 |
| 20 | 27 | 18 |
| 20 | 28 | 19 |
| 20 | 29 | 54 |
| 20 | 30 | 39 |
| 20 | 31 | 33 |

| I | J | DIST |
|---|---|---|
| 21 | 22 | 21 |
| 21 | 23 | 45 |
| 21 | 24 | 39 |
| 21 | 25 | 50 |
| 21 | 26 | 16 |
| 21 | 27 | 44 |
| 21 | 28 | 44 |
| 21 | 29 | 61 |
| 21 | 30 | 51 |
| 21 | 31 | 21 |

| I | J | DIST |
|---|---|---|
| 22 | 23 | 59 |
| 22 | 24 | 57 |
| 22 | 25 | 64 |
| 22 | 26 | 30 |
| 22 | 27 | 61 |
| 22 | 28 | 61 |
| 22 | 29 | 79 |
| 22 | 30 | 69 |
| 22 | 31 | 18 |

| I | J | DIST |
|---|---|---|
| 23 | 24 | 6 |
| 23 | 25 | 5 |
| 23 | 26 | 47 |
| 23 | 27 | 34 |
| 23 | 28 | 34 |
| 23 | 29 | 20 |
| 23 | 30 | 15 |
| 23 | 31 | 66 |

| I | J | DIST |
|---|---|---|
| 24 | 25 | 10 |
| 24 | 26 | 42 |
| 24 | 27 | 28 |
| 24 | 28 | 28 |
| 24 | 29 | 26 |
| 24 | 30 | 12 |
| 24 | 31 | 53 |

| I | J | DIST |
|---|---|---|
| 25 | 26 | 50 |
| 25 | 27 | 35 |
| 25 | 28 | 34 |
| 25 | 29 | 15 |
| 25 | 30 | 11 |
| 25 | 31 | 60 |

| I | J | DIST |
|---|---|---|
| 26 | 27 | 32 |
| 26 | 28 | 34 |
| 26 | 29 | 64 |
| 26 | 30 | 52 |
| 26 | 31 | 18 |

| I | J | DIST |
|---|---|---|
| 27 | 28 | 3 |
| 27 | 29 | 39 |
| 27 | 30 | 24 |
| 27 | 31 | 51 |

| I | J | DIST |
|---|---|---|
| 28 | 29 | 39 |
| 28 | 30 | 23 |
| 28 | 31 | 52 |

| I | J | DIST |
|---|---|---|
| 29 | 30 | 15 |
| 29 | 31 | 76 |

| I | J | DIST |
|---|---|---|
| 30 | 31 | 65 |

CE 51

| I | X | Y | P |
|---|---|---|---|
| 1 | 30 | 40 | 0 |
| 2 | 37 | 52 | 7 |
| 3 | 49 | 49 | 30 |
| 4 | 52 | 64 | 16 |
| 5 | 20 | 26 | 9 |
| 6 | 40 | 30 | 21 |
| 7 | 21 | 47 | 15 |
| 8 | 17 | 63 | 19 |
| 9 | 31 | 62 | 23 |
| 10 | 52 | 33 | 11 |
| 11 | 51 | 21 | 5 |
| 12 | 42 | 41 | 19 |
| 13 | 31 | 32 | 29 |
| 14 | 5 | 25 | 23 |
| 15 | 12 | 42 | 21 |
| 16 | 36 | 16 | 10 |
| 17 | 52 | 41 | 15 |
| 18 | 27 | 23 | 3 |
| 19 | 17 | 33 | 41 |
| 20 | 13 | 13 | 9 |

| I | X | Y | P |
|---|---|---|---|
| 21 | 57 | 58 | 28 |
| 22 | 62 | 42 | 8 |
| 23 | 42 | 57 | 8 |
| 24 | 16 | 57 | 16 |
| 25 | 8 | 52 | 10 |
| 26 | 7 | 38 | 28 |
| 27 | 27 | 68 | 7 |
| 28 | 30 | 48 | 15 |
| 29 | 43 | 67 | 14 |
| 30 | 58 | 48 | 6 |
| 31 | 58 | 27 | 19 |
| 32 | 37 | 69 | 11 |
| 33 | 38 | 46 | 12 |
| 34 | 46 | 10 | 23 |
| 35 | 61 | 33 | 26 |
| 36 | 62 | 63 | 17 |
| 37 | 63 | 69 | 6 |
| 38 | 32 | 22 | 9 |
| 39 | 45 | 35 | 15 |
| 40 | 59 | 15 | 14 |

| I | X | Y | P |
|---|---|---|---|
| 41 | 5 | 6 | 7 |
| 42 | 10 | 17 | 27 |
| 43 | 21 | 10 | 13 |
| 44 | 5 | 64 | 11 |
| 45 | 30 | 15 | 16 |
| 46 | 39 | 10 | 10 |
| 47 | 32 | 39 | 5 |
| 48 | 25 | 32 | 25 |
| 49 | 25 | 55 | 17 |
| 50 | 48 | 28 | 18 |
| 51 | 56 | 37 | 10 |

CE 76

| I | X | Y | P |
|---|---|---|---|
| 1 | 40 | 40 | 0 |
| 2 | 22 | 22 | 18 |
| 3 | 36 | 26 | 26 |
| 4 | 21 | 45 | 11 |
| 5 | 45 | 35 | 30 |
| 6 | 55 | 20 | 21 |
| 7 | 33 | 34 | 19 |
| 8 | 50 | 50 | 15 |
| 9 | 55 | 45 | 16 |
| 10 | 26 | 59 | 29 |
| 11 | 40 | 66 | 26 |
| 12 | 55 | 65 | 37 |
| 13 | 35 | 51 | 16 |
| 14 | 62 | 35 | 12 |
| 15 | 62 | 57 | 31 |
| 16 | 62 | 24 | 8 |
| 17 | 21 | 36 | 19 |
| 18 | 33 | 44 | 20 |
| 19 | 9 | 56 | 13 |
| 20 | 62 | 48 | 15 |

| I | X | Y | P |
|---|---|---|---|
| 21 | 66 | 14 | 22 |
| 22 | 44 | 13 | 28 |
| 23 | 26 | 13 | 12 |
| 24 | 11 | 28 | 6 |
| 25 | 7 | 43 | 27 |
| 26 | 17 | 64 | 14 |
| 27 | 41 | 46 | 18 |
| 28 | 55 | 34 | 17 |
| 29 | 35 | 16 | 29 |
| 30 | 52 | 26 | 13 |
| 31 | 43 | 26 | 22 |
| 32 | 31 | 76 | 25 |
| 33 | 22 | 53 | 28 |
| 34 | 26 | 29 | 27 |
| 35 | 50 | 40 | 19 |
| 36 | 55 | 50 | 10 |
| 37 | 54 | 10 | 12 |
| 38 | 60 | 15 | 14 |
| 39 | 47 | 66 | 24 |
| 40 | 30 | 60 | 16 |

| I | X | Y | P |
|---|---|---|---|
| 41 | 30 | 50 | 33 |
| 42 | 12 | 17 | 15 |
| 43 | 15 | 14 | 11 |
| 44 | 16 | 19 | 18 |
| 45 | 21 | 48 | 17 |
| 46 | 50 | 30 | 21 |
| 47 | 51 | 42 | 27 |
| 48 | 50 | 15 | 19 |
| 49 | 48 | 21 | 20 |
| 50 | 12 | 38 | 5 |
| 51 | 15 | 56 | 22 |
| 52 | 29 | 39 | 12 |
| 53 | 54 | 38 | 19 |
| 54 | 55 | 57 | 22 |
| 55 | 67 | 41 | 16 |
| 56 | 10 | 70 | 7 |
| 57 | 6 | 25 | 26 |
| 58 | 65 | 27 | 14 |
| 59 | 40 | 60 | 21 |
| 60 | 70 | 64 | 24 |

| I | X | Y | P |
|---|---|---|---|
| 61 | 64 | 4 | 13 |
| 62 | 36 | 6 | 15 |
| 63 | 30 | 20 | 18 |
| 64 | 20 | 30 | 11 |
| 65 | 15 | 5 | 28 |
| 66 | 50 | 70 | 9 |
| 67 | 57 | 72 | 37 |
| 68 | 45 | 42 | 30 |
| 69 | 38 | 33 | 10 |
| 70 | 50 | 4 | 8 |
| 71 | 66 | 8 | 11 |
| 72 | 59 | 5 | 3 |
| 73 | 35 | 60 | 1 |
| 74 | 27 | 24 | 6 |
| 75 | 40 | 20 | 10 |
| 76 | 40 | 37 | 20 |

| I | X | Y | P |
|---|---|---|---|
| 1 | 35 | 35 | 0 |
| 2 | 41 | 49 | 10 |
| 3 | 35 | 17 | 7 |
| 4 | 55 | 45 | 13 |
| 5 | 55 | 20 | 19 |
| 6 | 15 | 30 | 26 |
| 7 | 25 | 30 | 3 |
| 8 | 20 | 50 | 5 |
| 9 | 10 | 43 | 9 |
| 10 | 55 | 60 | 16 |
| 11 | 30 | 60 | 16 |
| 12 | 20 | 65 | 12 |
| 13 | 50 | 35 | 19 |
| 14 | 30 | 25 | 23 |
| 15 | 15 | 10 | 20 |
| 16 | 30 | 5 | 8 |
| 17 | 10 | 20 | 19 |
| 18 | 5 | 30 | 2 |
| 19 | 20 | 40 | 12 |
| 20 | 15 | 60 | 17 |
| 21 | 45 | 65 | 9 |
| 22 | 45 | 20 | 11 |
| 23 | 45 | 10 | 18 |
| 24 | 55 | 5 | 29 |
| 25 | 65 | 35 | 3 |
| 26 | 65 | 20 | 6 |
| 27 | 45 | 30 | 17 |
| 28 | 35 | 40 | 16 |
| 29 | 41 | 37 | 16 |
| 30 | 64 | 42 | 9 |
| 31 | 40 | 60 | 21 |
| 32 | 31 | 52 | 27 |
| 33 | 35 | 69 | 23 |
| 34 | 53 | 52 | 11 |
| 35 | 65 | 55 | 14 |
| 36 | 63 | 65 | 8 |
| 37 | 2 | 60 | 5 |
| 38 | 20 | 20 | 8 |
| 39 | 5 | 5 | 16 |
| 40 | 60 | 12 | 31 |
| 41 | 40 | 25 | 9 |
| 42 | 42 | 7 | 5 |
| 43 | 24 | 12 | 5 |
| 44 | 23 | 3 | 7 |
| 45 | 11 | 14 | 18 |
| 46 | 6 | 38 | 16 |
| 47 | 2 | 48 | 1 |
| 48 | 8 | 56 | 27 |
| 49 | 13 | 52 | 36 |
| 50 | 6 | 68 | 30 |
| 51 | 47 | 47 | 13 |
| 52 | 49 | 58 | 10 |
| 53 | 27 | 43 | 9 |
| 54 | 37 | 31 | 14 |
| 55 | 57 | 29 | 18 |
| 56 | 63 | 23 | 2 |
| 57 | 53 | 12 | 6 |
| 58 | 32 | 12 | 7 |
| 59 | 36 | 26 | 18 |
| 60 | 21 | 24 | 28 |
| 61 | 17 | 34 | 3 |
| 62 | 12 | 24 | 13 |
| 63 | 24 | 58 | 19 |
| 64 | 27 | 69 | 10 |
| 65 | 15 | 77 | 9 |
| 66 | 62 | 77 | 20 |
| 67 | 49 | 73 | 25 |
| 68 | 67 | 5 | 25 |
| 69 | 56 | 39 | 36 |
| 70 | 37 | 47 | 6 |
| 71 | 37 | 56 | 5 |
| 72 | 57 | 68 | 15 |
| 73 | 47 | 16 | 25 |
| 74 | 44 | 17 | 9 |
| 75 | 46 | 13 | 8 |
| 76 | 49 | 11 | 18 |
| 77 | 49 | 42 | 13 |
| 78 | 53 | 43 | 14 |
| 79 | 61 | 52 | 3 |
| 80 | 57 | 48 | 23 |
| 81 | 56 | 37 | 6 |
| 82 | 55 | 54 | 26 |
| 83 | 15 | 47 | 16 |
| 84 | 14 | 37 | 11 |
| 85 | 11 | 31 | 7 |
| 86 | 16 | 22 | 41 |
| 87 | 4 | 18 | 35 |
| 88 | 28 | 18 | 26 |
| 89 | 26 | 52 | 9 |
| 90 | 26 | 35 | 15 |
| 91 | 31 | 67 | 3 |
| 92 | 15 | 19 | 1 |
| 93 | 22 | 22 | 2 |
| 94 | 18 | 24 | 22 |
| 95 | 26 | 27 | 27 |
| 96 | 25 | 24 | 20 |
| 97 | 22 | 27 | 11 |
| 98 | 25 | 21 | 12 |
| 99 | 19 | 21 | 10 |
| 100 | 20 | 26 | 9 |
| 101 | 18 | 18 | 17 |

# APENDICE E

'MAIN Program:

## LISTAGEM DO PROGRAMA DE MICROCOMPUTADOR

```
10 '-----------------------------------------------
20 ' Este programa foi desenvolvido por Wan Yu Chih,
30 ' Sao Paulo, em junho de 1987.
40 '
50 ' O WGT faz a determinacao de roteiros
60 ' e tem as seguintes caracteristicas:
70 '
80 ' . veiculos de diversas capacidades
90 ' . diferentes custos fixos e variaveis
100 ' . restricao de jornada de trabalho
110 '
120 ' Seus dados de entrada sao restritos a:
130 '
140 ' . Coordenadas x,y,p : 100 pontos
150 ' . Frota : 10 tipos de veiculos
160 '-----------------------------------------------
170 CLS
180 COLOR BORDER BROWN
190 COLOR 0,7
200 LOCATE 10,15
210 PRINT"                                    "
220 LOCATE 11,15
230 PRINT"                 W G T              "
240 LOCATE 12,15
250 PRINT"                                    "
260 LOCATE 13,15
270 PRINT"  Programa de Roteirizacao de Frotas  "
280 LOCATE 14,15
290 PRINT"                                    "
300 LOCATE 15,15
310 PRINT"                                    "
320 COLOR 7,0
330 LOCATE 17,15
340 PRINT"  (c) 1987 Wan Yu Chih, Sao Paulo.     "
350 COLOR 7,0
360 LOCATE 23,5
370 INPUT"Tecle Enter para continuar";Z$
380 COLOR BORDER BLACK
390 '-----------------------------------------------
400 ' Procedures:
410 '
420 Xyp
430 Distancias
440 Rotaleatoria
450 Opt2
460 Giro
470 Frota
480 PRINT
490 PRINT
500 PRINT "Tecle S para acessar ou alterar dados de tempos"
510 PRINT "Tecle N caso contrario"
520 INPUT Z$
530 IF Z$="S" OR Z$="s" GOTO 550
540 IF Z$="N"OR Z$="n" GOTO 560 ELSE GOTO 520
550 Tempo
560 Zonas
570 '-----------------------------------------------
590 PRINT
600 PRINT "Tecle S para continuar a utilizacao do programa
610 PRINT "Tecle N para encerrar"
620 INPUT Z$
630 IF Z$="S" OR Z$="s" GOTO 650
640 IF Z$="N"OR Z$="n" GOTO 720 ELSE GOTO 620
650 PRINT
660 PRINT
670 PRINT "Tecle S para alterar dados da frota"
680 PRINT "Tecle N para alterar somente dados de tempos"
690 INPUT Z$
700 IF Z$="S" OR Z$="s" GOTO 470
710 IF Z$="N" OR Z$="n" GOTO 550 ELSE GOTO 690
720 END

ENDFILE
```

```
SOURCE
PRECISION 8
PROCS=31
REAL ARRAY(103): X,Y,P
REAL ARRAY(103,103): Dist/X
INTEGER: N,Ntv
STRING: Z$[16]
REAL: Distrand
INTEGER ARRAY(103): Rotaotim,Randrota,Rota2opt
REAL ARRAY(12): Capv,Cfv
REAL ARRAY(12): Cvv
REAL: Tmax,Tcorte
REAL: Velocidade,Tfixo,Ttaxa

PROCEDURE: Xyp
END PROCEDURE

PROCEDURE: Distancias
END PROCEDURE

PROCEDURE: Opt2
END PROCEDURE

PROCEDURE: Zonas
END PROCEDURE

PROCEDURE: Giro
END PROCEDURE

PROCEDURE: Rotaleatoria
END PROCEDURE

PROCEDURE: Frota
END PROCEDURE

PROCEDURE: Tempo
END PROCEDURE

PROCEDURE: Xyp
   EXTERNAL: X,Y,N,P
   STRING: Z$[16]

   PROCEDURE: Grava
   END PROCEDURE

   PROCEDURE: Le
   END PROCEDURE

   PROCEDURE: Digita
   END PROCEDURE

   PROCEDURE: Corrige
   END PROCEDURE

   PROCEDURE: Mostra
   END PROCEDURE
```

```
PROCEDURE: Grava
   EXTERNAL: X,Y,P,N
   INTEGER: I,J,K
   STRING: Arq[16]
     10 PRINT
     20 COLOR 0,6
     30 PRINT"  Digite o nome do arquivo  ";
     40 COLOR 0,7
     50 INPUT?"   ";Arq
     60 COLOR 7,0
     70 CLS
     80 Arq=Arq+".XYP"
     90 LOCATE 8,10
    100 COLOR 0,6
    110 PRINT" Coordenadas x y p  "
    120 COLOR 7,0
    130 LOCATE 10,10
    140 COLOR 0,15
    150 PRINT" Gravando            "
    160 COLOR 7,0
    170 OPEN Arq AS #1 LEN=6
    180 FOR I=1 TO N
    190   LOCATE 12,10
    200   PRINT" I = ";:PRINT USING "###";I
    210   WRITE RECORD #1 I X(I)
    220   J=N+I
    230   WRITE RECORD #1 J Y(I)
    240   K=2*N+I
    250   WRITE RECORD #1 K P(I)
    260 NEXT
    270 CLOSE #1
    280 LOCATE 10,10
    290 COLOR 0,7
    300 PRINT" Pronto !            "
    310 COLOR 7,0
    320 BEEP
    330 LOCATE 14,10
END PROCEDURE
```

```
PROCEDURE: Le
   EXTERNAL: X,Y,P,N
   INTEGER: I,J,K
   STRING: Arq[16]
      10 PRINT
      20 COLOR 0,6
      30 PRINT"  Digite o nome do arquivo  ";
      40 COLOR 0,7
      50 INPUT?"   ";Arq
      60 COLOR 7,0
      70 CLS
      80 Arq=Arq+".XYP"
      90 LOCATE 8,10
     100 COLOR 0,6
     110 PRINT" Coordenadas x y p  "
     120 COLOR 7,0
     130 LOCATE 10,10
     140 COLOR 0,15
     150 PRINT" Lendo              "
     160 COLOR 7,0
     170 OPEN Arq AS #1 LEN=6
     180 FOR I= 1 TO N
     190   LOCATE 12,10
     200   PRINT" I = ";:PRINT USING "###";I
     210   READ RECORD #1 I X(I)
     220   J=N+I
     230   READ RECORD #1 J Y(I)
     240   K=2*N+I
     250   READ RECORD #1 K P(I)
     260 NEXT
     270 CLOSE #1
     280 LOCATE 10,10
     290 COLOR 0,7
     300 PRINT" Pronto !           "
     310 COLOR 7,0
     320 BEEP
     330 LOCATE 14,10
END PROCEDURE

PROCEDURE: Digita
   EXTERNAL: X,Y,P,N
   INTEGER: I
      10 CLS
      20 COLOR 0,6
      30 PRINT      "  Digitar as coordenadas x,y e p   "
      40 COLOR 7,0
      50 PRINT
      60 COLOR 0,5
      70 PRINT      "  Comecar com o deposito   "
      80 COLOR 7,0
      90 PRINT
     100 PRINT "  x( 1 )  ";
     110 INPUT?X(1)
     120 PRINT "  y( 1 )  ";
     130 INPUT?Y(1)
     140 P(1)=0
     150 FOR I=2 TO N
     160   PRINT
     170   PRINT "  x(";I;")  " ;
     180   INPUT?X(I)
     190   PRINT "  y(";I;")  " ;
     200   INPUT?Y(I)
     210   PRINT "  p(";I;")  " ;
     220   INPUT?P(I)
     230 NEXT
END PROCEDURE

PROCEDURE: Corrige
   EXTERNAL: X,Y,P,N
   INTEGER: I
      10 CLS
      20 COLOR 0,6
      30 PRINT "  Correcao/alteracao        "
      40 COLOR 7,0
      50 PRINT
      60 COLOR 0,6
      70 PRINT " I"," x"," y"," p  "
      80 COLOR 7,0
      90 INPUT?"?";I
     100 IF I>N GOTO 90
     110 LOCATE 4,9
     120 PRINT X(I),Y(I),P(I)
     130 PRINT
     140 COLOR 0,6
     150 PRINT "  x     ";
     160 COLOR 0,7
     170 INPUT?X(I)
     180 COLOR 0,6
     190 PRINT "  y     ";
     200 COLOR 0,7
     210 INPUT?Y(I)
     220 COLOR 0,6
     230 PRINT "  p     ";
     240 COLOR 0,7
     250 INPUT?P(I)
     260 COLOR 7,0
END PROCEDURE

PROCEDURE: Mostra
   EXTERNAL: X,Y,P,N
   INTEGER: I,K
   STRING: Z$[16]
      10 CLS
      20 PRINT
      30 PRINT " I"," x"," y"," p  "
      40 PRINT "-----------------------------"
      50 K=0
      60 FOR I=1 TO N
      70   K=K+1
      80   IF K<>11 GOTO 150
      90   K=0
     100   PRINT
     110   INPUT"Tecle Enter para continuar";Z$
     120   PRINT
     130   PRINT " I"," x"," y"," p  "
     140   PRINT "-----------------------------"
     150   PRINT I,X(I),Y(I),P(I)
     160 NEXT
END PROCEDURE
```

```
10 '-----------------------------------------------------
20 ' Esta procedure faz a entrada de dados do problema
30 ' atraves das coordenadas x e y
40 '
50 '
60 ' O deposito deve ser sempre o ponto numero 1.
70 '
80 '-----------------------------------------------------
90 CLS
100 PRINT
110 COLOR 0,6
120 PRINT "  Entrada de Dados  "
130 COLOR 7,0
140 PRINT
150 COLOR 0,6
160 PRINT "  Numero de pontos  ";
170 COLOR 0,7
180 INPUT?"   ";N
190 COLOR 7,0
200 PRINT
210 PRINT
220 PRINT"Tecle S para ler dados em disco"
230 PRINT"Tecle N para digitar dados     "
240 INPUT Z$
250 IF Z$="S" OR Z$="s" GOTO 390
260 IF Z$="N" OR Z$="n" GOTO 280 ELSE GOTO 240
270 '-----------------------------------------------------
280 Digita
290 Mostra
300 PRINT
310 PRINT
320 PRINT"Tecle S para correcoes"
330 PRINT"Tecle N caso contrario"
340 INPUT Z$
350 IF Z$="S" OR Z$="s" GOTO 550
360 IF Z$="N" OR Z$="n" GOTO 570 ELSE GOTO 340
370 PRINT
380 '-----------------------------------------------------
390 Le
400 Mostra
410 PRINT
420 PRINT"Tecle S para correcoes"
430 PRINT"Tecle N para continuar"
440 INPUT Z$
450 IF Z$="S" OR Z$="s" GOTO 480
460 IF Z$="N" OR Z$="n" GOTO 470 ELSE GOTO 440
470 END
480 Corrige
490 Mostra
500 PRINT
510 PRINT
520 PRINT"Tecle S para correcoes"
530 PRINT"Tecle N para continuar"
540 INPUT Z$
550 IF Z$="S" OR Z$="s" GOTO 480
560 IF Z$="N" OR Z$="n" GOTO 570 ELSE GOTO 540
570 PRINT
580 PRINT"Tecle S para gravar o arquivo em disco"
590 PRINT"Tecle N caso contrario"
600 INPUT Z$
610 IF Z$="S" OR Z$="s" GOTO 630
620 IF Z$="N" OR Z$="n" GOTO 640 ELSE GOTO 600
630 Grava 'Procedure
640 END
END PROCEDURE
```

```
PROCEDURE: Distancias
   EXTERNAL: X,Y,N,Dist
   STRING: Z$[16]

   PROCEDURE: Le
   END PROCEDURE

   PROCEDURE: Grava
   END PROCEDURE

   PROCEDURE: Mostra
   END PROCEDURE

   PROCEDURE: Calcula
   END PROCEDURE

   PROCEDURE: Digita
   END PROCEDURE

   PROCEDURE: Corrige
   END PROCEDURE

   PROCEDURE: Le
      EXTERNAL: Dist,N
      STRING: Arq[16]
      INTEGER: I,J,K
         10 PRINT
         20 COLOR 0,6
         30 PRINT"  Digite o nome do arquivo ";
         40 COLOR 0,7
         50 INPUT?"   ";Arq
         60 COLOR 7,0
         70 Arq=Arq+".DST"
         80 CLS
         90 LOCATE 8,10
        100 COLOR 0,6
        110 PRINT" Matriz de distancias  "
        120 COLOR 7,0
        130 LOCATE 10,10
        140 COLOR 0,15
        150 PRINT" Lendo                    "
        160 COLOR 7,0
        170 OPEN Arq AS #1 LEN=6
        180 FOR I= 1 TO N
        190   LOCATE 12,10
        200   PRINT " I = ";:PRINT USING "###";I
        210   FOR J= 1 TO N
        220     LOCATE 14,10
        230     PRINT " J = ";:PRINT USING "###";J
        240     K=(I-1)*N+J
        250     READ RECORD #1 K Dist(I,J)
        260   NEXT
        270 NEXT
        280 CLOSE #1
        290 COLOR 0,7
        300 LOCATE 10,10
        310 PRINT " Pronto !                  "
        320 COLOR 7,0
        330 BEEP
        340 LOCATE 16,10
    END PROCEDURE
```

```
PROCEDURE: Mostra
  EXTERNAL: Dist,N
  INTEGER: I,J,K,I1,I2
  STRING: Z$[16]
    10 CLS
    20 COLOR 0,6
    30 PRINT"  Matriz de distancias  "
    40 COLOR 7,0
    50 PRINT
    60 COLOR 0,6
    70 PRINT "  Ver de I >=   1";
    80 COLOR 0,7
    90 INPUT?"     ";I1
   100 IF I1>=N GOTO 70
   110 PRINT
   120 COLOR 0,6
   130 PRINT "  Ate    I <=";:PRINT USING "###";N-1;
   140 COLOR 0,7
   150 INPUT?"     ";I2
   160 COLOR 7,0
   170 IF I2>N-1 GOTO 130
   180 COLOR 7,0
   190 FOR I=I1 TO I2
   200   CLS
   210   COLOR 0,6
   220   PRINT"  Matriz de distancias  "
   230   COLOR 7,0
   240   PRINT
   250   PRINT " I"," J","Distancia  "
   260   PRINT "----------------------------"
   270   K=0
   280   FOR J=I+1 TO N
   290     K=K+1
   300     IF K<>18 GOTO 370
   310     K=0
   320     PRINT
   330     INPUT "Tecle Enter para continuar";Z$
   340     PRINT
   350     PRINT " I"," J","Distancia  "
   360     PRINT "----------------------------"
   370     PRINT I,J,Dist(I,J)
   380   NEXT
   390   PRINT
   400   INPUT "Tecle Enter para continuar";Z$
   410   PRINT
   420 NEXT
   430 PRINT
   440 PRINT "Tecle S para ver de novo"
   450 PRINT "Tecle N caso contrario"
   460 INPUT Z$
   470 IF Z$="S" OR Z$="s" GOTO 10
   480 IF Z$="N" OR Z$="n" GOTO 490 ELSE GOTO 460
   490 END
END PROCEDURE
```

```
PROCEDURE: Grava
  EXTERNAL: Dist,N
  INTEGER: I,J,K
  STRING: Arq[16]
    10 PRINT
    20 COLOR 0,6
    30 PRINT"  Digite o nome do arquivo ";
    40 COLOR 0,7
    50 INPUT?"    ";Arq
    60 COLOR 7,0
    70 Arq=Arq+".DST"
    80 CLS
    90 LOCATE 8,10
   100 COLOR 0,6
   110 PRINT " Matriz de distancias  "
   120 COLOR 7,0
   130 LOCATE 10,10
   140 COLOR 0,15
   150 PRINT " Gravando                "
   160 COLOR 7,0
   170 OPEN Arq AS #1 LEN=6
   180 FOR I= 1 TO N
   190   LOCATE 12,10
   200   PRINT" I = ";:PRINT USING "###";I
   210   FOR J= 1 TO N
   220     LOCATE 14,10
   230     PRINT" J = ";:PRINT USING "###";J
   240     K=(I-1)*N+J
   250     WRITE RECORD #1 K Dist(I,J)
   260   NEXT
   270 NEXT
   280 CLOSE #1
   290 LOCATE 10,10
   300 COLOR 0,7
   310 PRINT " Pronto !                "
   320 COLOR 7,0
   330 BEEP
   340 LOCATE 16,10
END PROCEDURE
```

```
PROCEDURE: Calcula
   EXTERNAL: N,Dist,X,Y
   INTEGER: I,J
      10 '  Calculo da matriz de distancias
      20 CLS
      30 Dist(N,N)=9E9
      40 FOR I=1 TO N-1
      50   Dist(I,I)=9E9
      60   PRINT" I"," J","Distancia  "
      70   PRINT "------------------------------"
      80   FOR J=I+1 TO N
      90     Dist(I,J)= ((X(I)-X(J))^2+(Y(I)-Y(J))^2)^.5
     100     PRINT I,J,Dist(I,J)
     110     Dist(J,I)=Dist(I,J)
     120   NEXT
     130   PRINT
     140 NEXT
     150 BEEP
END PROCEDURE

PROCEDURE: Digita
   EXTERNAL: N,Dist
   INTEGER: I,J
      10 CLS
      20 COLOR 0,6
      30 PRINT "  Entrada da matriz de distancias   "
      40 COLOR 7,0
      50 PRINT
      60 Dist(N,N)=9E9
      70 FOR I=1 TO N-1
      80   Dist(I,I)=9E9
      90   FOR J=I+1 TO N
     100     PRINT " Dist(";I;",";J;") ";
     110     INPUT? Dist(I,J)
     120     Dist(J,I)=Dist(I,J)
     130   NEXT
     140   PRINT
     150 NEXT
END PROCEDURE
PROCEDURE: Corrige
   EXTERNAL: Dist
   INTEGER: I,J
      10 CLS
      20 COLOR 0,6
      30 PRINT "  Correcao  "
      40 COLOR 7,0
      50 PRINT
      60 COLOR 0,6
      70 PRINT "  I   ";
      80 COLOR 0,7
      90 INPUT?I
     100 COLOR 0,6
     110 PRINT "  J   ";
     120 COLOR 0,7
     130 INPUT?J
     140 COLOR 7,0
     150 COLOR 0,6
     160 PRINT "  Dist(";I;",";J;") = ";Dist(I,J)
     170 COLOR 7,0
     180 PRINT
     190 COLOR 0,6
     200 PRINT "  Dist(";I;",";J;") = ";
     210 COLOR 0.7
     220 INPUT?Dist(I,J)
     230 COLOR 7,0
END PROCEDURE
```

```
 10 '---------------------------------------------
 20 ' Esta procedure faz a determinacao da matriz
 30 ' de distancias com as seguintes alternativas :
 40 '
 50 ' . calculo analitico para cada par de pontos
 60 ' . entrada direta por digitacao
 70 '
 80 '---------------------------------------------
 90 CLS
100 COLOR 0,6
110 PRINT"  Matriz de distancias  "
120 COLOR 7,0
130 PRINT
140 PRINT
150 PRINT"Tecle S para ler dados arquivados em disc
160 PRINT"Tecle N para calculo ou digitacao"
170 INPUT Z$
180 IF Z$="S" OR Z$="s" GOTO 390
190 IF Z$="N" OR Z$="n" GOTO 210 ELSE GOTO 170
200 '---------------------------------------------
210 PRINT
220 PRINT"Tecle S para calculo"
230 PRINT"Tecle N para digitacao"
240 INPUT Z$
250 IF Z$="S" OR Z$="s" GOTO 280
260 IF Z$="N" OR Z$="n" GOTO 300 ELSE GOTO 240
270 '---------------------------------------------
280 Calcula
290 GOTO 560 'opcao para gravar
300 Digita
310 Mostra
320 PRINT
330 PRINT"Tecle S para correcoes"
340 PRINT"Tecle N caso contrario"
350 INPUT Z$
360 IF Z$="S" OR Z$="s" GOTO 480
370 IF Z$="N" OR Z$="n" GOTO 560 ELSE GOTO 350
380 '---------------------------------------------
390 Le
400 Mostra
410 PRINT
420 PRINT"Tecle S para correcoes"
430 PRINT"Tecle N caso contrario"
440 INPUT Z$
450 IF Z$="S" OR Z$="s" GOTO 480
460 IF Z$="N" OR Z$="n" GOTO 470 ELSE GOTO 440
470 END
480 Corrige
490 Mostra
500 PRINT
510 PRINT"Tecle S para correcoes"
520 PRINT"Tecle N para continuar"
530 INPUT Z$
540 IF Z$="S" OR Z$="s" GOTO 480
550 IF Z$="N" OR Z$="n" GOTO 560 ELSE GOTO 530
560 PRINT
570 PRINT"Tecle S para gravar os dados em disco"
580 PRINT"Tecle N caso contrario"
590 INPUT Z$
600 IF Z$="S" OR Z$="s" GOTO 620
610 IF Z$="N" OR Z$="n" GOTO 630 ELSE GOTO 590
620 Grava
630 END
END PROCEDURE
```

```
PROCEDURE: Opt2
  EXTERNAL: Dist,N,Randrota,Distrand,Rota2opt
  INTEGER: I,J,A,B
  INTEGER: C,D,E,Ia,Ka,J1,J2,Flag
  INTEGER: Somaflag,K1,K2
  REAL: Dist2opt,Dant,Ddep
  STRING: Z$[16]
     10 '--------------------------------------------------
     20 ' Esta procedure faz a determinacao do roteiro
     30 ' gigante otimo pelo procedimento 2-opt de Lin.
     40 '
     50 '--------------------------------------------------
     60 CLS
     70 COLOR 0,6
     80 LOCATE 10,10
     90 PRINT"   Roteiro gigante pelo 2-opt   "
    100 COLOR 7,0
    110 PRINT
    120 FOR I=1 TO N
    130   Rota2opt(I)=Randrota(I)
    140 NEXT
    150 Dist2opt=Distrand
    160 Somaflag=0
    170 LOCATE 15,15
    180 PRINT "Somaflag = ";:PRINT USING "###";Somaflag
    190 FOR I=1 TO N
    200   Flag=0   'nenhuma troca realizada
    210   ' Arcos A e B
    220   '
    230   A=Rota2opt(I)
    240   IF I<N THEN B=Rota2opt(I+1) ELSE  B=Rota2opt(1)
    250   IF I<N THEN J1=I+2 ELSE  J1=2
    260   J2=J1+N-4
    270   ' Arcos C e D
    280   '
    290   FOR J= J1 TO J2
    300     IF J <=N THEN K1=J ELSE K1=J-N
    310     C=Rota2opt(K1)
    320     IF (K1+1) <=N THEN K2=K1+1 ELSE K2=K1+1-N
    330     D=Rota2opt(K2)
    340     ' Testa a troca
    350     '
    360     Dant=Dist(A,B)+Dist(C,D)
    370     Ddep=Dist(A,C)+Dist(B,D)
    380     IF Ddep<Dant THEN Flag=1  ELSE GOTO 650
    390     ' Efetiva a troca
    400     '
    410     IF I<N THEN Rota2opt(I+1)=C ELSE Rota2opt(1)=C
    420     Rota2opt(K1)=B
    430     '--------------------------------------------------
    440     ' Redefine a sequencia do ponto I+2 ate'o ponto K1-1
    450     '
    460     IF I<N-1 THEN Ia=I+2 ELSE Ia=I+2-N
    470     Ka=K1
    480     IF Ia=Ka GOTO 580
    490     IF Ka>=2 THEN Ka=Ka-1 ELSE Ka=Ka-1+N
    500     IF Ia=Ka GOTO 580
    510     E= Rota2opt(Ia)
    520     Rota2opt(Ia)=Rota2opt(Ka)
    530     Rota2opt(Ka)=E
    540     IF Ia-Ka=1 OR Ia-Ka=-1 GOTO 580
    550     IF Ia<N THEN Ia=Ia+1 ELSE Ia=1
    560     GOTO 490
    570     '--------------------------------------------------
    580     Somaflag=0
    590     LOCATE 15,15
    600     PRINT "Somaflag = ";:PRINT USING "###";Somaflag
    610     Dist2opt=Dist2opt-Dant+Ddep
    620     LOCATE 13,15
    630     PRINT"Dist2opt = ";:PRINT USING "####.##";Dist2
    640     GOTO 680
    650     ' Nao faz a troca
    660     '
    670     GOTO 690
    680     J=J2
    690   NEXT
    700   IF Flag=1 GOTO 750
    710   Somaflag=Somaflag+1
    720   LOCATE 15,15
    730   PRINT"Somaflag = ";:PRINT USING "###";Somaflag
    740   IF Somaflag=N THEN I=N
    750 NEXT
    760 IF Somaflag<N THEN GOTO 190
    770 BEEP
    780 LOCATE 20,10
    790 INPUT "Tecle Enter para continuar";Z$
END PROCEDURE

PROCEDURE: Zonas
  EXTERNAL: P,Dist,N,Rotaotim,Capv,Cfv,Cvv,Ntv
  REAL ARRAY(51): Pzona,Dzona
  INTEGER ARRAY(51): Npzona,Veiczona
  INTEGER ARRAY(51,51): Rotzona
  REAL: Pesotot,Distot,Pesoacum,Distacum,Cvz,Ctz
  REAL: Ctu
  INTEGER: Nz,Seqz,I,J,K,L,Nzonas
  STRING: Z$[16]
  REAL ARRAY(103,103): Cost/X
  INTEGER ARRAY(103): Ccrit
  INTEGER: Tipoveic
  EXTERNAL: Tmax,Tcorte
  EXTERNAL: Tfixo,Ttaxa,Velocidade
  REAL ARRAY(51): Tzona

  PROCEDURE: Costkm
  END PROCEDURE

  PROCEDURE: Critico
  END PROCEDURE

  PROCEDURE: Resumo
  END PROCEDURE

  PROCEDURE: Veiculo
  END PROCEDURE

  PROCEDURE: Costkm
    EXTERNAL: Dist,N,Capv,Cfv,Cost,Rotaotim,P,Ntv
    INTEGER: K,M,R,Tipoveic
    REAL: Soma,Distancia
    EXTERNAL: Cvv
    INTEGER: L
    EXTERNAL: Tmax,Tcorte,Tfixo,Ttaxa
    EXTERNAL: Velocidade
    REAL: Pesorot,Temporot,Custofixo

    PROCEDURE: Veiculo
    END PROCEDURE
```

```
PROCEDURE: Veiculo
   EXTERNAL: Rotaotim,K,M,P,Ntv,Capv,Tipoveic
   INTEGER: R,I
   EXTERNAL: Pesorot
      10 '------------------------------------------
      20 ' Esta procedure determina o menor veiculo
      30 ' que pode atender a soma de pesos
      40 '
      50 '------------------------------------------
      60 Pesorot=0
      70 FOR R = K TO M-1
      80   Pesorot=Pesorot+P(Rotaotim(R))
      90 NEXT
     100 FOR I= 1 TO Ntv+1
     110   IF Capv(I)>=Pesorot THEN Tipoveic=I ELSE GOTO 130
     120   I=Ntv+1
     130 NEXT
END PROCEDURE
   10 '----------------------------------------------------
   20 '  Esta procedure faz o calculo da matriz Cost(k,m)
   30 '  onde:
   40 '
   50 '  Cost(k,m) = custo do percurso que sai do deposito,
   60 '              vai ao ponto Rotaotim(k), passando por
   65 '              Rotaotim(k+1),Rotaotim(k+2),...
   70 '              ate Rotaotim(m-1),
   80 '              retornando em seguida ao deposito
   90 '----------------------------------------------------
  100 CLS
  110 COLOR 0,6
  120 PRINT "   Matriz de Custos   "
  130 COLOR 7,0
  140 PRINT
  150 FOR K=2 TO N+1
  160   PRINT " K"," M","Cost  "
  170   PRINT "-------------------------"
  180   FOR M= K TO N+1
  190     Soma=0
  200     IF K=M THEN Cost(K,M)=0 ELSE GOTO 220
  210     GOTO 460
  220     IF K=M-1 GOTO 260
  230     FOR R=K TO M-2
  240       Soma=Soma+Dist(Rotaotim(R),Rotaotim(R+1))
  250     NEXT
  260     Veiculo
  270     Custofixo=Cfv(Tipoveic)
  280     Distancia=Dist(1,Rotaotim(K))+Soma+Dist(Rotaotim(M-1),1)
  290     Temporot=(Distancia/Velocidade)+(Tfixo*(M-K))+(Ttaxa*Pesorot)
  300     IF Temporot <=Tcorte GOTO 360
  310     IF Temporot > Tmax THEN Custofixo=9E9
  320     GOTO 380
  330     '-------------------------
  340     ' Politica de premiacao
  350     '
  360     Custofixo=Custofixo/2
  370     '-------------------------
  380     Cost(K,M)=Distancia*Cvv(Tipoveic)+Custofixo
  390     IF Cost(K,M)<9E9 GOTO 460
  400     FOR L=M TO N+1
  410       Cost(K,L)=9E9
  420     NEXT
  430     PRINT K,M,Cost(K,M)
  440     M=N+1
  450     GOTO 470
  460     PRINT K,M,Cost(K,M)
  470   NEXT
  480   PRINT
  490 NEXT
  500 PRINT
END PROCEDURE

PROCEDURE: Critico
   REAL ARRAY(103,103): F/X
   INTEGER: I,J,K,Imin
   REAL: Minimo,Fd,Delta
   EXTERNAL: Ccrit,N,Cost
   EXTERNAL: Nzonas
   INTEGER: Conta,L
     10 '--------------------------------------------------
     20 ' Esta procedure determina o menor caminho do
     30 ' ponto 2 ao N+1, onde:
     40 '
     50 '   F(K,J) = menor distancia do ponto 2 ao J
     60 '            na K-esima etapa
     70 '   Numero de etapas = N-1
     80 '--------------------------------------------------
     90 CLS
    100 COLOR 0,6
    110 PRINT "   Menor caminho   "
    120 COLOR 7,0
    130 PRINT
    140 PRINT " K"," J"," F(K,J)   "
    150 PRINT "--------------------------"
    160 FOR J= 2 TO N+1
    170   F(1,J)=Cost(2,J)
    180   PRINT 1,J,F(1,J)
    190 NEXT
    200 PRINT
    210 '  Menor caminho entre os pontos 2 e N+1
    220 '
    230 FOR K= 2  TO N-1
    240   Conta=1
    250   PRINT " K"," J"," I"," F(K,J)   "
    260   PRINT "-----------------------------------"
    270   FOR J= 2 TO N+1
    280     Minimo=9E9
    290     FOR I= 2 TO N+1
    300       Fd=F(K-1,I)+Cost(I,J)
    310       IF Fd<Minimo THEN Minimo=Fd ELSE GOTO 330
    320       Imin=I
    330     NEXT
    340     F(K,J)=Minimo
    350     IF F(K,J)=F(K-1,J) THEN Conta=Conta+1
    360     IF F(K,J)<9E9 GOTO 430
    370     PRINT K,J,Imin,F(K,J)
    380     FOR L=J TO N+1
    390       F(K,L)=9E9
    400     NEXT
    410     J=N+1
    420     GOTO 440
    430     PRINT K,J,Imin,F(K,J)
    440   NEXT
    450   IF Conta<N+1 GOTO 520
    460   FOR L=K TO N-1
    470     FOR J=2 TO N+1
    480       F(L,J)=F(L-1,J)
    490     NEXT
    500   NEXT
```

```
510   K=N-1
520   PRINT
530 NEXT
540 '---------------------------------------------------
550 CLS
560 COLOR 0,6
570 PRINT "  Menor caminho do ponto 2 ao N+1  "
580 COLOR 7,0
590 PRINT
600 J=N+1
610 K=N
620 Ccrit(K)=N+1
630 PRINT " K"," I"," J"," Cost(I,J)   "
640 PRINT "-----------------------------------"
650 FOR I = 2 TO N+1
660   IF I=J GOTO 820
670   Delta=F(N-1,J)-F(N-1,I)
680   PRINT K,I,J,Cost(I,J)
690   '
700   '   precisao=0.01
710   '
720   IF Cost(I,J)-Delta<-0.01 GOTO 820
730   IF Cost(I,J)-Delta>0.01 GOTO 820
740   K=K-1
750   Ccrit(K)=I
760   J=I
770   PRINT
780   IF J=2 THEN I=N+1 ELSE GOTO 800
790   GOTO 820
800   PRINT " K"," I"," J"," Cost(I,J)   "
810   PRINT "-----------------------------------"
820 NEXT
830 IF J<>2 GOTO 650
840 '---------------------------------------------------
850 CLS
860 COLOR 0,6
870 PRINT "   Sequencia    "
880 COLOR 7,0
890 J=1
900 Ccrit(N)=N+1
910 FOR I= K TO N
920   PRINT TAB(5);Ccrit(I)
930   Ccrit(I-K+1)=Ccrit(I)
940 NEXT
950 Nzonas=N-K
960 PRINT
970 COLOR 0,6
980 PRINT "  Custo total   ";:COLOR 0,7:PRINT F(N-1,N+1)
990 COLOR 7,0
1000 PRINT
1010 PRINT
END PROCEDURE
```

```
PROCEDURE: Resumo
   EXTERNAL: Nzonas,Distot,Pesotot,Ctu,Veiczona,Pzona,Dzona,Cfv
   EXTERNAL: Capv,Cvv
   REAL: Cfz,Cvz,Txutil,Ctz,Cftot,Cvtot,Ctot
   INTEGER: K,I
   STRING: Z$[16]
   EXTERNAL: Tzona
   REAL: Tempotot
   STRING: Texto[50]
     10 CLS
     20 COLOR 0,7
     30 PRINT"  Zoneamento  "
     40 COLOR 0,6
     50 PRINT "  Resumo     "
     60 COLOR 7,0
     70 PRINT
     80 COLOR 0,6
     90 PRINT " Zona";:PRINT USING"\        \";"   Peso";" Capveic";"   Dist";"  Tempo";"  CFixo";"   CVar";" CTotal"
    100 COLOR 7,0
    110 Cftot=0
    120 Cvtot=0
    130 Tempotot=0
    140 K=0
    150 FOR I= 1 TO Nzonas
    160   K= K+1
    170   IF K<>11 GOTO 310
    180   K=0
    190   PRINT
    200   INPUT "Tecle Enter para continuar";Z$
    210   CLS
    220   COLOR 0,7
    230   PRINT"  Zoneamento  "
    240   COLOR 0,6
    250   PRINT "  Resumo     "
    260   COLOR 7,0
    270   PRINT
    280   COLOR 0,6
    290   PRINT " Zona";:PRINT USING"\        \";"   Peso";" Capveic";"   Dist";"  Tempo";"  CFixo";"   CVar";" CTotal"
    300   COLOR 7,0
    310   Cfz=Cfv(Veiczona(I))
    320   Cvz=Cvv(Veiczona(I))*Dzona(I)
    330   Ctz=Cfz+Cvz
    340   PRINT USING "##";I;:PRINT USING "######.#";Pzona(I);Capv(Veiczona(I));Dzona(I);Tzona(I);Cfz;Cvz;Ctz
    350   Cftot=Cftot+Cfz
    360   Cvtot=Cvtot+Cvz
    370   Tempotot=Tempotot+Tzona(I)
    380 NEXT
    390 Ctot=Cftot+Cvtot
    400 PRINT
    410 PRINT
    420 COLOR 0,7
    430 PRINT "  Totais      "
    440 COLOR 7,0
    450 PRINT
    460 COLOR 0,6
    470 PRINT " Zona";:PRINT USING"\        \";"   Peso";" Capveic";"   Dist";"  Tempo";"  CFixo";"   CVar";" CTotal"
    480 COLOR 7,0
    490 PRINT Nzonas;:PRINT USING "#######.#";Pesotot;Ctu;Distot;Tempotot;Cftot;Cvtot;Ctot
    500 Txutil=Pesotot/Ctu*100
```

```
        510 PRINT
        520 COLOR 0,6
        530 PRINT "  Taxa de utilizacao   ";:COLOR 0,7:PRINT USING "###.## %";Tx
util;"   "
        540 COLOR 7,0
        550 PRINT
        560 PRINT
        570 PRINT "Tecle S para imprimir resumo"
        580 PRINT "Tecle N caso contrario"
        590 INPUT Z$
        600 IF Z$="S" OR Z$="s" GOTO 650
        610 IF Z$="N" OR Z$="n" GOTO 1070 ELSE GOTO 590
        620 '-------------------------------------------------------------
        630 ' Impressao do resumo
        640 '
        650 CLS
        660 COLOR 0,6
        670 PRINT " Impressao do resumo  "
        680 COLOR 7,0
        690 PRINT
        700 COLOR 0,6
        710 PRINT " Digite o titulo da listagem  ";
        720 COLOR 0,7
        730 INPUT?"   ";Texto
        740 COLOR 7,0
        750 PRINT
        760 LPRINT SPACE$(5);"  ";Texto
        770 LPRINT
        780 LPRINT SPACE$(5);"  Resumo      "
        790 LPRINT
        800 LPRINT SPACE$(5);"Zona";"     Peso";"   CapPACE$(5);:LPRINT USING "##";I;:LPRINT USING "#######.#";Pz
ona(I);Capv(Veiczona(I));Dzona(I);Tzona(I);Cfz;Cvz;Ctz
        900   Cftot=Cftot+Cfz
        910   Cvtot=Cvtot+Cvz
        920   Tempotot=Tempotot+Tzona(I)
        930 NEXT
        940 Ctot=Cftot+Cvtot
        950 LPRINT
        960 LPRINT SPACE$(5);"  Totais      "
        970 LPRINT
        980 LPRINT SPACE$(5);"Zona";"     Peso";"   Capveic";"      Dist";"
Tempo";"     CFixo";"      CVar";"    CTotal"
        990 LPRINT SPACE$(5);"-----------------------------------------------------
------------------------"
        1000 LPRINT SPACE$(5);:LPRINT USING "##";Nzonas;:LPRINT USING "#######.#"
;Pesotot;Ctu;Distot;Tempotot;Cftot;Cvtot;Ctot
        1010 Txutil=Pesotot/Ctu*100
        1020 LPRINT
        1030 LPRINT SPACE$(5);"  Taxa de utilizacao   ";:COLOR 0,7:LPRINT USING "
###.## %";Txutil;"   "
        1040 LPRINT
        1050 LPRINT
        1060 LPRINT
        1070 END
    END PROCEDURE
```

```
PROCEDURE: Veiculo
   EXTERNAL: Pzona
   INTEGER: I
   EXTERNAL: Ntv,Capv,Tipoveic,Nz
      10 '-------------------------------------------------
      20 ' Esta procedure determina o menor veiculo
      30 ' que pode atender a demanda Pzona(Nz).
      40 '
      50 '-------------------------------------------------
      60 FOR I= 1 TO Ntv+1
      70   IF Capv(I)>=Pzona(Nz) THEN Tipoveic=I ELSE GOTO 90
      80   I=Ntv+1
      90 NEXT
END PROCEDURE
   10 '----------------------------------------------------
   20 ' Esta procedure faz o zoneamento da frota atraves
   30 ' da particao do roteiro gigante gerado pelo Opt2
   40 '
   50 ' As variaveis utilizadas sao as seguintes:
   60 '
   70 '   Rotaotim(i)       = roteiro gigante da procedure Opt2
   80 '                       modificado pela procedure giro
   90 '   Rotzona(Nz,Seqz) = roteiro otimo da zona Nz
  100 '   Pzona(Nz)        = peso da zona Nz
  110 '   Dzona(Nz)        = distancia da zona Nz
  120 '   Npzona(Nz)       = numero de pontos da zona Nz
  130 '   Seqz             = sequencia de cada zona
  140 '   Nz               = numero da zona
  150 '   Nzonas           = numero de zonas
  160 '   Ccrit(Nz)        = primeiro ponto da zona Nz
  170 '                      aops o deposito
  180 '----------------------------------------------------
  190 Costkm
  200 Critico
  201 BEEP
  210 CLS
  220 Distot=0
  230 Pesotot=0
  240 Ctu=0
  250 '----------------------------------------------------
  260 '  Detalhamento das zonas
  270 '
  280 FOR Nz=1 TO Nzonas
  290   K=Ccrit(Nz)
  300   J=Rotaotim(K)
  310   Rotzona(Nz,1)=J
  320   Npzona(Nz)=Ccrit(Nz+1)-Ccrit(Nz)
  330   Pesoacum=P(J)
  340   Distacum=Dist(1,J)
  350   IF Npzona(Nz)=1 THEN I=J ELSE GOTO 370
  360   GOTO 460
  370   FOR Seqz= 2 TO Npzona(Nz)
  380     I=J
  390     K=K+1
  400     J=Rotaotim(K)
  410     Rotzona(Nz,Seqz)=J
  420     Pesoacum=Pesoacum+P(J)
  430     Distacum=Distacum+Dist(I,J)
  440     I=J
  450   NEXT
  460   Dzona(Nz)=Distacum+Dist(I,1)
  470   Pzona(Nz)=Pesoacum
  480   Tzona(Nz)=(Dzona(Nz)/Velocidade)+(Pzona(Nz)*Ttaxa)+(Npzona(Nz)*Tfixo)
```

```
490   Pesotot=Pesotot+Pzona(Nz)
500   Distot=Distot+Dzona(Nz)
510   Veiculo
520   Veiczona(Nz)=Tipoveic
530   Ctu=Ctu+Capv(Tipoveic)
540 NEXT
550 '---------------------------------------------
560 Resumo
570 PRINT
580 COLOR 7,0
590 PRINT "Tecle S para ver os resultados por Zona"
600 PRINT "Tecle N caso contrario"
610 INPUT Z$
620 IF Z$="S" OR Z$="s" GOTO 670
630 IF Z$="N" OR Z$="n" GOTO 1270 ELSE GOTO 610
640 '---------------------------------------------
650 ' Resultados por Zona
660 '
670 FOR Nz=1 TO Nzonas
680   CLS
690   COLOR 0,7
700   PRINT"  Zoneamento  "
710   COLOR 0,6
720   PRINT   "  Resultados  "
730   COLOR 7,0
740   PRINT
750   COLOR 0,6
760   PRINT"  Zona = ";Nz;SPACE$(3)
770   COLOR 7,0
780   PRINT
790   PRINT "  Seq","Ponto","Peso"," I"," J","Dist  "
800   PRINT "-----------------------------------------------"
810   J=1
820   K=0
830   FOR L=1 TO Npzona(Nz)
840     K=K+1
850     IF K<>9 GOTO 970
860     K=0
870     PRINT
880     INPUT "Tecle Enter para continuar  ";Z$
890     CLS
900     PRINT
910     COLOR 0,6
920     PRINT"  Zona = ";Nz;SPACE$(3)
930     COLOR 7,0
940     PRINT
950     PRINT "  Seq","Ponto","Peso"," I"," J","Dist  "
960     PRINT "-----------------------------------------------"
970     I=J
980     J=Rotzona(Nz,L)
990     IF L<>Npzona(K) GOTO 1000
1000    PRINT TAB(2);L,J,P(J),I,J,Dist(I,J)
1010  NEXT
1020  PRINT TAB(2);L," -"," -",J,1,Dist(J,1)
1030  PRINT
1040  COLOR 0,6
1050  PRINT " Peso total",:COLOR 0,7:PRINT Pzona(Nz),:COLOR 0,6:PRINT " Dis
t total",:COLOR 0,7:PRINT Dzona(Nz);SPACE$(3)
1060  COLOR 7,0
1070  PRINT
1080  Cvz=Dzona(Nz)*Cvv(Veiczona(Nz))
1090  COLOR 0,6
1100  PRINT " Capac veic",:COLOR 0,7:PRINT Capv(Veiczona(Nz)),:COLOR 0,6:PR
INT " Custo variavel",:COLOR 0,7:PRINT Cvz;SPACE$(3)
1110  COLOR 7,0
1120  PRINT
1130  COLOR 0,6
1140  Ctz=Cfv(Veiczona(Nz))+Cvz
1150  PRINT " Custo fixo",:COLOR 0,7:PRINT Cfv(Veiczona(Nz)),:COLOR 0,6:PRI
NT " Custo total",:COLOR 0,7:PRINT Ctz;SPACE$(3)
1160  COLOR 7,0
1170  PRINT
1180  INPUT"Tecle Enter para continuar";Z$
1190 NEXT
1200 PRINT
1210 PRINT"Tecle S para ver novamente o resumo"
1220 PRINT"Tecle N caso contrario"
1230 INPUT Z$
1240 IF Z$="S" OR Z$="s" GOTO 1260
1250 IF Z$="N" OR Z$="n" GOTO 1270 ELSE GOTO 1230
1260 Resumo
1270 END
END PROCEDURE
```

```
PROCEDURE: Giro
  EXTERNAL: N,Rotaotim,Rota2opt
  INTEGER: I,J,K,I1,I2
  STRING: Z$[16]
    10 '---------------------------------------------------
    20 ' Esta procedure faz a conversao da sequencia
    30 ' Rota2opt(I) gerada pela procedure Opt2
    40 ' de modo a se ter Rota2opt(1) = 1
    50 '
    60 '---------------------------------------------------
    70 FOR J=1 TO N
    80   IF Rota2opt(J) <>1 THEN GOTO 100
    90   K=J
   100 NEXT
   110 I=0
   120 FOR J=K TO N
   130   I=I+1
   140   Rotaotim(I)=Rota2opt(J)
   150 NEXT
   160 K=0
   170 FOR J=(I+1) TO N
   180   K=K+1
   190   Rotaotim(J)=Rota2opt(K)
   200 NEXT
   210 CLS
   220 COLOR 0,6
   230 PRINT,"Roteiro 2-opt       "
   240 COLOR 7,0
   250 PRINT
   260 PRINT
   270 PRINT ,"Seq","Ponto      "
   280 PRINT "    ---------------------"
   290 PRINT ,1,Rotaotim(1)
   300 I1=Rotaotim(1)
   310 K=1
   320 FOR I=2 TO N
   330   K=K+1
   340   IF K<>11 GOTO 410
   350   K=1
   360   PRINT
   370   INPUT "Tecle Enter para continuar";Z$
   380   PRINT
   390   PRINT ,"Seq","Ponto      "
   400   PRINT "    ---------------------"
   410   PRINT,I,Rotaotim(I)
   420   I2=Rotaotim(I)
   430   I1=I2
   440 NEXT
   450 BEEP
   460 PRINT
   470 INPUT "Tecle Enter para continuar";Z$
END PROCEDURE
```

```
PROCEDURE: Rotaleatoria
  EXTERNAL: N,Dist,Randrota,Distrand
  INTEGER: I,J,K
  INTEGER ARRAY(103): Pr
  STRING: Z$[16]
    10 '----------------------------------------------------
    20 ' Esta procedure faz a geracao do roteiro gigante
    30 ' aleatorio.
    40 '
    50 '----------------------------------------------------
    60 FOR K=1 TO N
    70   Pr(K)=0
    80 NEXT
    90 Distrand=0
   100 I=1+INT(RND(1)*N) 'Geracao de numero aleatorio de 1 a
   110 Randrota(1)=I
   120 Pr(I)=1
   130 FOR K=2 TO N
   140   J=1+INT(RND(1)*N)
   150   IF Pr(J)=1 THEN GOTO 140
   160   Randrota(K)=J
   170   Distrand=Distrand+Dist(I,J)
   180   Pr(J)=1
   190   I=J
   200 NEXT
   210 Distrand=Distrand+Dist(Randrota(N),Randrota(1))
   220 ' Impressao
   230 CLS
   240 COLOR 0,6
   250 PRINT ,"Rota aleatoria       "
   260 COLOR 7,0
   270 PRINT
   280 PRINT ,"Seq","Ponto      "
   290 PRINT "       -----------------"
   300 K=0
   310 FOR I= 1 TO N
   320   K=K+1
   330   IF K<>11 GOTO 400
   340   K=0
   350   PRINT
   360   INPUT "Tecle Enter para continuar";Z$
   370   PRINT
   380   PRINT ,"Seq","Ponto      "
   390   PRINT "       -----------------"
   400   PRINT,I,Randrota(I)
   410 NEXT
   420 BEEP
   430 PRINT
   440 INPUT "Tecle Enter para continuar";Z$
END PROCEDURE
```

```
PROCEDURE: Frota
  STRING: Z$[16]
  EXTERNAL: Ntv,Cfv,Capv,Cvv

  PROCEDURE: Le
  END PROCEDURE

  PROCEDURE: Grava
  END PROCEDURE

  PROCEDURE: Digita
  END PROCEDURE

  PROCEDURE: Corrige
  END PROCEDURE

  PROCEDURE: Mostra
  END PROCEDURE
  PROCEDURE: Le
    STRING: Arq[16]
    INTEGER: I
    EXTERNAL: Ntv,Capv,Cfv,Cvv
    INTEGER: J,K
      10 PRINT
      20 PRINT
      30 COLOR 0,6
      40 PRINT "  Tipos de veiculos  ";
      50 COLOR 0,7
      60 INPUT?"    ";Ntv
      70 COLOR 7,0
      80 PRINT
      90 COLOR 0,6
     100 PRINT  "  Digite o nome do arquivo  ";
     110 COLOR 0,7
     120 INPUT?"    ";Arq
     130 COLOR 7,0
     140 CLS
     150 Arq=Arq+".frt"
     160 LOCATE 8,10
     170 COLOR 0,6
     180 PRINT" Dados da frota      "
     190 COLOR 7,0
     200 LOCATE 10,10
     210 COLOR 0,15
     220 PRINT" Lendo               "
     230 COLOR 7,0
     240 OPEN Arq AS #1 LEN=6
     250 FOR I= 1 TO Ntv
     260   LOCATE 12,10
     270   PRINT" I = ";:PRINT USING "###";I
     280   READ RECORD #1 I Capv(I)
     290   J=Ntv+I
     300   READ RECORD #1 J Cfv(I)
     310   K=2*Ntv+I
     320   READ RECORD #1 K Cvv(I)
     330 NEXT
     340 CLOSE #1
     350 Capv(Ntv+1)=9E9
     360 Cfv(Ntv+1)=9E9
     370 Cvv(Ntv+1)=9E9
     380 LOCATE 10,10
     390 COLOR 0,7
     400 PRINT" Pronto !            "
     410 COLOR 7,0
     420 BEEP
     430 LOCATE 14,10
  END PROCEDURE
```

```
PROCEDURE: Grava
  STRING: Arq[16]
  INTEGER: I,J,K
  EXTERNAL: Ntv,Capv,Cfv,Cvv
     10 PRINT
     20 COLOR 0,6
     30 PRINT"  Digite o nome do arquivo  ";
     40 COLOR 0,7
     50 INPUT?"    ";Arq
     60 COLOR 7,0
     70 CLS
     80 Arq=Arq+".frt"
     90 LOCATE 8,10
    100 COLOR 0,6
    110 PRINT" Dados da frota      "
    120 COLOR 7,0
    130 LOCATE 10,10
    140 COLOR 0,15
    150 PRINT" Gravando            "
    160 COLOR 7,0
    170 OPEN Arq AS #1 LEN=6
    180 FOR I= 1 TO Ntv
    190   LOCATE 12,10
    200   PRINT" I = ";:PRINT USING "###";I
    210   WRITE RECORD #1 I Capv(I)
    220   J=Ntv+I
    230   WRITE RECORD #1 J Cfv(I)
    240   K=2*Ntv+I
    250   WRITE RECORD #1 K Cvv(I)
    260 NEXT
    270 CLOSE #1
    280 LOCATE 10,10
    290 COLOR 0,7
    300 PRINT" Pronto !            "
    310 COLOR 7,0
    320 BEEP
END PROCEDURE

PROCEDURE: Digita
  EXTERNAL: Ntv,Capv,Cfv,Cvv
  INTEGER: I
    10 CLS
    20 PRINT
    30 INPUT?" Tipos de veiculos = ";Ntv
    40 PRINT
    50 PRINT " Entrar com capacidades na ordem crescent
    60 PRINT
    70 FOR I= 1 TO Ntv
    80   PRINT " Capacidade(";I;") ";
    90   INPUT?Capv(I)
   100   PRINT " C Fixo    (";I;") ";
   110   INPUT?Cfv(I)
   120   PRINT " C Variavel(";I;") ";
   130   INPUT?Cvv(I)
   140   PRINT
   150 NEXT
   160 Capv(Ntv+1)=9E9
   170 Cfv(Ntv+1)=9E9
   180 Cvv(Ntv+1)=9E9
END PROCEDURE
```

```
PROCEDURE: Corrige
   EXTERNAL: Capv,Cfv,Cvv
   INTEGER: I
      10 PRINT
      20 PRINT
      30 COLOR 0,7
      40 INPUT?" Veiculo tipo   ";I
      50 COLOR 0,6
      60 INPUT? " Capacidade   ";Capv(I)
      70 INPUT? " C Fixo       ";Cfv(I)
      80 INPUT? " C Variavel   ";Cvv(I)
      90 COLOR 7,0
     100 PRINT
END PROCEDURE

PROCEDURE: Mostra
   EXTERNAL: Ntv,Capv,Cfv,Cvv
   INTEGER: I
      10 CLS
      20 COLOR 0,6
      30 PRINT "  Dados da frota  "
      40 COLOR 7,0
      50 PRINT
      60 COLOR 0,6
      70 PRINT " Tipo",,"Capacidade","Custo fixo","Custo variavel  "
      80 COLOR 7,0
      90 FOR I= 1 TO Ntv
     100    PRINT TAB(2);I,,Capv(I),,Cfv(I),,Cvv(I);:COLOR 7,0:PRINT
SPACE$(25)
     110 NEXT
END PROCEDURE
   10 '-----------------------------------------------
   20 ' Esta procedure faz a entrada de dados da frota
   30 '
   40 '-----------------------------------------------
   50 CLS
   60 COLOR 0,6
   70 PRINT "  Dados da frota  "
   80 COLOR 7,0
   90 PRINT
  100 PRINT
  110 PRINT "Tecle S para usar dados arquivados em disco"
  120 PRINT "Tecle N para digitar ou corrigir"
  130 INPUT?Z$
  140 IF Z$="S" OR Z$="s" GOTO 160
  150 IF Z$="N" OR Z$="n" GOTO 400 ELSE GOTO 130
  160 Le
  170 Mostra
  180 PRINT
  190 PRINT "Tecle S para correcoes"
  200 PRINT "Tecle N para continuar"
  210 INPUT?Z$
  220 IF Z$="S" OR Z$="s" GOTO 240
  230 IF Z$="N" OR Z$="n" GOTO 390 ELSE GOTO 210 'End
  240 Corrige
  250 Mostra
  260 PRINT
  270 PRINT "Tecle S para correcoes"
  280 PRINT "Tecle N caso contrario"
  290 INPUT?Z$
  300 IF Z$="S" OR Z$="s" GOTO 240
  310 IF Z$="N" OR Z$="n" GOTO 320 ELSE GOTO 290
  320 PRINT
  330 PRINT"Tecle S para gravar"
  340 PRINT"Tecle N caso contrario"
  350 INPUT Z$
  360 IF Z$="S" OR Z$="s" THEN GOTO 380
  370 IF Z$="N" OR Z$="n" THEN GOTO 390 ELSE GOTO 350
  380 Grava
  390 END
  400 PRINT
  410 PRINT
  420 PRINT"Tecle S para correcoes"
  430 PRINT"Tecle N para entrada de dados por digitacao"
  440 INPUT Z$
  450 IF Z$="S" OR Z$="s" THEN GOTO 470
  460 IF Z$="N" OR Z$="n" THEN GOTO 510 ELSE GOTO 440
  470 Mostra
  480 Corrige
  490 Mostra
  500 GOTO 260
  510 Digita
  520 Mostra
  530 PRINT
  540 PRINT
  550 PRINT"Tecle S para correcoes"
  560 PRINT"Tecle N caso contrario"
  570 INPUT Z$
  580 IF Z$="S" OR Z$="s" THEN GOTO 240
  590 IF Z$="N" OR Z$="n" THEN GOTO 600 ELSE GOTO 570
  600 PRINT
  610 PRINT"Tecle S para gravar os dados em disco"
  620 PRINT"Tecle N caso contrario"
  630 INPUT Z$
  640 IF Z$="S" OR Z$="s" THEN GOTO 660
  650 IF Z$="N" OR Z$="n" THEN GOTO 670 ELSE GOTO 630
  660 Grava
  670 END
END PROCEDURE

PROCEDURE: Tempo
   EXTERNAL: Tmax,Tcorte,Tfixo,Ttaxa,Velocidade
   STRING: Z$[16]

   PROCEDURE: Entrada
   END PROCEDURE

   PROCEDURE: Mostra
   END PROCEDURE

   PROCEDURE: Entrada
      EXTERNAL: Tmax,Tcorte,Tfixo,Ttaxa,Velocidade
         10 CLS
         20 COLOR 0,6
         30 PRINT "  Dados de Tempos    "
         40 COLOR 7,0
         50 PRINT
         60 PRINT
         70 COLOR 0,6
         80 PRINT " Jornada diaria de trabalho (h)
         90 COLOR 0,7
        100 INPUT?"     ";Tmax
        110 COLOR 7,0
        120 PRINT
        130 COLOR 0,6
        140 PRINT " Duracao maxima da primeira viagem (h)
```

```
150 COLOR 0,7
160 INPUT?"     ";Tcorte
170 COLOR 7,0
180 PRINT
190 COLOR 0,6
200 PRINT " Tempo fixo por cliente (min)            ";
210 COLOR 0,7
220 INPUT?"     ";Tfixo
230 COLOR 7,0
240 COLOR 0,7
250 PRINT
260 COLOR 0,6
270 PRINT " Taxa de descarga (min/t)                ";
280 COLOR 0,7
290 INPUT?"     "; Ttaxa
300 COLOR 7,0
310 PRINT
320 COLOR 0,6
330 PRINT " Velocidade media (km/h)                 ";
340 COLOR 0,7
350 INPUT?"     ";Velocidade
360 COLOR 7,0
390 PRINT
END PROCEDURE

PROCEDURE: Mostra
  EXTERNAL: Tmax,Tcorte,Tfixo,Ttaxa,Velocidade
   10 CLS
   20 COLOR 0,6
   30 PRINT " Dados de tempos  "
   40 COLOR 7,0
   50 PRINT
   60 PRINT
   70 COLOR 0,6
   80 PRINT " Jornada diaria de trabalho (h)           ";
   90 COLOR 0,7
  100 PRINT USING "#####";Tmax
  110 COLOR 7,0
  120 PRINT
  130 COLOR 0,6
  140 PRINT " Duracao maxima da primeira viagem (h)   ";
  150 COLOR 0,7
  160 PRINT USING "#####";Tcorte
  170 COLOR 7,0
  180 PRINT
  190 COLOR 0,6
  200 PRINT " Tempo fixo por cliente (min)            ";
  210 COLOR 0,7
  220 PRINT USING "#####";Tfixo
  230 COLOR 7,0
  240 PRINT
  250 COLOR 0,6
  260 PRINT " Taxa de descarga (min/t)                ";
  270 COLOR 0,7
  280 PRINT USING "###.#";Ttaxa
  290 COLOR 7,0
  300 PRINT
  310 COLOR 0,6
  320 PRINT " Velocidade media (km/h)                 ";
  330 COLOR 0,7
  340 PRINT USING "#####";Velocidade
  350 COLOR 7,0
```

```
END PROCEDURE
  10 '--------------------------------------------------------
  20 ' Esta procedure recebe os dados de tempos
  30 ' para a analise de viagens multiplas por dia.
  40 '
  50 ' Os parametros solicitados sao os seguintes:
  60 '
  70 ' . tmax = jornada de trabalho
  80 ' . tcorte = tempo limite maximo para premiacao
  90 ' . tfixo = tempo constante de atendimento por cliente
 100 ' . ttaxa = taxa de descarga
 110 ' . velocidade = velocidade media no sistema viario
 120 '--------------------------------------------------------
 130 CLS
 140 COLOR 0,6
 150 PRINT "  Dados de Tempos     "
 160 COLOR 7,0
 170 PRINT
 180 PRINT
 190 PRINT "Tecle S para utilizar restricoes de tempo"
 200 PRINT "Tecle N caso contrario"
 210 INPUT?Z$
 220 IF Z$="S" OR Z$="s" GOTO 300
 230 IF Z$="N" OR Z$="n" GOTO 240 ELSE GOTO 210
 240 Tmax=9E9
 250 Tcorte=0
 260 Tfixo=0
 270 Ttaxa=0
 280 Velocidade=9E9
 290 END
 300 PRINT
 310 PRINT "Tecle S para entrada de dados "
 320 PRINT "Tecle N caso contrario "
 330 INPUT?Z$
 340 IF Z$="S" OR Z$="s" GOTO 360
 350 IF Z$="N" OR Z$="n" GOTO 400 ELSE GOTO 330
 360 Entrada
 370 Tfixo=Tfixo/60
 380 Ttaxa=Ttaxa/60
 390 GOTO 450
 400 Tfixo=Tfixo*60
 410 Ttaxa=Ttaxa*60
 420 Mostra
 430 Tfixo=Tfixo/60
 440 Ttaxa=Ttaxa/60
 450 PRINT
 460 PRINT
 470 PRINT "Tecle S para correcoes "
 480 PRINT "Tecle N caso contrario "
 490 INPUT Z$
 500 IF Z$="S" OR Z$="s" GOTO 360
 510 IF Z$="N" OR Z$="n" GOTO 520 ELSE GOTO 490
 520 END
END PROCEDURE
```