Capítulo 13

Metodologia para o PRV: um Estudo de Caso para a Distribuição de Água Mineral

Sheila Margot Gonçalves*, Maria Teresinha Arns Steiner e Luzia Vidal de Souza

Resumo: A metodologia proposta é constituída pelas seguintes etapas: 1) algoritmo de Teitz & Bart, para a determinação de medianas, cuja quantidade varia de acordo com a quantidade de pontos de demanda a serem atendidos e com a capacidade dos veículos que fazem a entrega; 2) algoritmo de Gillett e Johnson modificado, para a definição dos agrupamentos de pontos de demanda a serem atendidos pelos veículos; 3) algoritmos heurísticos dos *Savings* de Clarke e Wright, da Inserção do Mais Próximo, da meta-heurística Busca Tabu e algumas de suas combinações, comparativamente, para a obtenção da sequência de atendimento aos pontos de demanda.

Palavras-chave: Problemas de roteamento de veículos (PRV), Algoritmos heurísticos e Meta-heurístico, Estudo de caso.

Abstract: *The methodology proposed is constituted by the following phases: 1) Teitz & Bart's algorithm, in order to determine the medians which quantity varies according to the quantity of demand points as well as the capacity of the vehicles used on the delivery; 2) modified Gillet and Johnson's algorithm, in order to define the clusters of demand points to be supplied by the vehicles; 3) Clarke & Wright's savings algorithm as well as the Nearest Insertion heuristics, the Tabu Search metaheuristic and by their combination, comparatively, in order to get the sequence in which the demand points will be supplied.*

Keywords: *Vehicle routing problems, Heuristics and metaheuristic algorithms, Case study.*

1. Introdução

É notório que o nível de competitividade tem aumentado sensivelmente em todos os setores da economia brasileira em função das transformações observadas a partir dos anos 90. Do ponto de vista macroeconômico, neste período tem-se vivenciado a estabilização monetária, a abertura de mercado, a desregulamentação e a privatização de setores chave. No dia-a-dia das empresas, em virtude da sua importância cada vez maior no contexto logístico, os Problemas de Roteamento de Veículos (PRV), que estão diretamente relacionados com os seus custos operacionais, vêm recebendo uma atenção especial por parte de planejadores e pesquisadores. A impossibilidade de obter soluções ótimas para instâncias encontradas no mundo real, devido as diferentes características e restrições vem trazendo o desafio da busca de novos procedimentos heurísticos de maneira a produzir resultados próximos do ótimo.

O objetivo principal deste capítulo é apresentar uma metodologia para realizar a tarefa de roteamento no serviço de entrega, aplicado aos mais diversos problemas de distribuição e, mais especificamente, na distribuição de água mineral para o município de Itú, SP. Esta metodologia faz uso de ferramentas da Programação Matemática, através das quais possa se apresentar uma solução para o atendimento a todos os clientes minimizando as distâncias percorridas pela frota de veículos.

São centenas de trabalhos na literatura que tratam dos PRV, alguns dos quais são citados a seguir, dando desta forma, um respaldo adicional à metodologia apresentada na seção 3 que pode resolver, dentre outros, o problema apresentado na Seção 2. Os resultados da metodologia aplicada ao problema real são mostrados na seção 4 e, finalmente, na seção 5 são apresentadas as conclusões.

Breedam (2001) apresenta uma comparação entre as meta-heurísticas *Simulated Annealing* e Busca Tabu para solucionar o PRV. Seus estudos mostram que, em condições semelhantes, os referidos algoritmos apresentam resultados semelhantes, sendo que o que determina a escolha de uma ou outra meta-heurística

*Autor para contato: sheilamargot@up.com.br

 DOI: 10.7436/2013.mhpo.13 ISBN 978-85-64619-10-4

é o tempo computacional. O autor propõe também, uma combinação dos resultados obtidos pelas duas heurísticas.

Já Toth & Vigo (2003) implementaram uma modificação no algoritmo da Busca Tabu e a denominaram de Busca Tabu granular. Em sua metodologia realizam uma restrição na busca pela vizinhança, analisando apenas os vizinhos potencialmente capazes de oferecer uma boa solução para o problema. Os testes computacionais mostraram a importância de se utilizar uma estratégia para criar uma lista apropriada de candidatos promissores e seu impacto na criação de métodos mais eficientes.

Boulduc et al. (2010) propõem uma heurística baseada em Busca Tabu para o integrar o PRV com a programação da produção e a data de entrega. Este novo problema consiste em determinar quais clientes serão atendidos por um transportador comum, bem como os que serão atendidos por uma frota privada, de forma a minimizar os custos com transporte. O problema foi modelado como um Problema de Programação Inteiro Misto e foi resolvido utilizando a meta-heurística Busca Tabu com divisão de entregas. Os resultados demonstram que é possível obter uma excelente solução para atendimento aos clientes dentro do prazo estabelecido. Foram utilizadas estratégias de redução da vizinhança para a redução do tempo computacional.

Leung et al. (2011) apresentam um algoritmo de Busca Tabu estendido para combinar o PRV com o problema do empacotamento de cargas bi-dimensionais. No algoritmo denominado *Extended Guided Tabu Search* (EGTS), é proposta uma melhoria para guiar a Busca Tabu utilizando a ideia de busca local guiada. Além disto, é implementado um critério de aspiração no sentido de evitar a perda de boas soluções. Por outro lado, propõem um algoritmo para o carregamento das cargas que aumenta a probabilidade de apresentar soluções viáveis, adotando uma estratégia híbrida de empacotamento para verificar a factibilidade do carregamento. Foram realizados testes para 180 bancos de dados e a eficiência da metodologia ficou comprovada.

Choi & Tcha (2007) apresentam uma metodologia baseada na técnica de geração de colunas para o PRV com frota heterogênea. Um modelo de Programação Inteira é apresentado e o relaxamento da Programação Linear é resolvido através da geração de colunas. Alguns problemas de Programação Dinâmica são desenvolvidos para o PRV clássico e são propostas algumas modificações para a geração de colunas viáveis; em seguida foi aplicado o algoritmo *branch-and-bound* com as colunas geradas. A metodologia foi testada para 36 problemas da biblioteca de Golden *et al*, 2003, *apud* Choi & Tcha (2007), de forma comparativa. As soluções apresentadas confirmam a eficiência da técnica.

Valle & Cunha (2009) apresentam um estudo sobre o PRV no qual, nem todos os clientes precisam ser visitados, sendo que o objetivo é minimizar o comprimento da maior rota. O problema é formulado como um Problema de Programação Inteira e os métodos *branch-and-cut* e *local branching* são utilizados para resolver o problema. Os limites superiores foram obtidos utilizando a meta-heurística GRASP (*Greedy Randomized Adaptative Search Procedure*), usada na inicialização dos métodos. O resultado ótimo foi obtido em diversas instâncias testadas e o tempo computacional foi limitado em 4 horas.

Du & He (2012) combinam duas meta-heurísticas: vizinho mais próximo e Busca Tabu para a solução do PRV. Na primeira etapa, o algoritmo do vizinho mais próximo é utilizado para a construção das rotas iniciais e a Busca Tabu é utilizada para implementar melhorias nas rotas obtidas. Os resultados demonstram que a metodologia apresenta resultados satisfatórios.

2. Descrição do Problema Real

O trabalho de roteamento em uma empresa é um dos fatores que determinam a sua eficiência, face à competitividade do mercado. Sendo assim, todos os pontos de demanda referentes a uma determinada empresa devem estar roteirizados, prevendo em alguns casos, inclusive, a possibilidade de pedidos extras a serem adicionados em alguma rota. Definir um bom roteiro considerando problemas de trânsito, acesso, circulação e estacionamento de veículos nos centros urbanos não é um problema de fácil solução, no entanto, com a utilização de técnicas da área de Pesquisa Operacional, esta tarefa pode tornar-se mais segura e eficiente, podendo ser realizada em poucos minutos.

O interesse e a demanda pela aplicação de modelos de roteamento para problemas reais têm crescido muito nos últimos anos. As principais razões para a procura deste tipo de trabalho relacionam-se às exigências dos clientes com relação a prazos de entrega, datas e horários de atendimento; a competitividade do mercado e a busca pela eficiência, acarretando a redução de custos e melhor desempenho na prestação dos serviços oferecidos.

Este estudo de caso foi realizado na Empresa de Mineração Clarita Ltda, concessionária da extração e comercialização da água mineral Clarita, localizada na cidade de Itú, no estado de São Paulo, fundada em 26 de fevereiro de 1991. O envasamento desta água mineral é feito em galões de 10 e 20 litros. A empresa vende em torno de 200.000 litros por mês. Por ocasião da pesquisa, a distribuição era terceirizada e atendia, além de Itú, também as cidades de Salto, Porto Feliz, Sorocaba, Jundiaí e Indaiatuba.

O objetivo deste estudo é apresentar uma metodologia para o problema de roteamento de veículos com as características aqui apresentadas, ilustrando-a na distribuição dos garrafões de água mineral para a referida empresa. Na cidade de Itú, as ruas são, de uma forma geral, estreitas e geralmente de mão dupla e com estacionamento nas laterais, o que dificulta o tráfego de caminhões de grande porte e, por este motivo, o caminhão a ser utilizado é de pequeno porte, sendo que a sua capacidade é de 59 garrafões de 20 litros por viagem.

A empresa realiza a distribuição dos galões de água aos 46 clientes, aqui tratados como pontos de demanda, entregando em torno de 14 garrafões de 20 litros por semana para cada cliente, sendo que esta entrega deverá ser realizada duas vezes na semana. Desta forma, os 46 pontos de demanda são responsáveis pela comercialização de aproximadamente 3.000 garrafões de 20 litros por mês.

A entrega é realizada, atualmente, através de nove rotas, sendo cada uma delas percorrida duas vezes na semana. A quantidade de pontos em cada rota varia de 4 a 6 pontos atendidos, sendo que a distância total percorrida pelos veículos é de cerca de 90.360 metros diários. Devido ao fato da distribuição ser realizada duas vezes na semana, ficou definido neste trabalho que cada ponto de demanda de cada rota receberá 7 garrafões a cada entrega, totalizando 14 garrafões por semana. Como a capacidade do transporte utilizado é de 59 garrafões, para entregar sete garrafões por ponto, a quantidade de pontos atendidos por veículo será de oito pontos (59 ÷ 7), com uma folga de 3 garrafões.

Tendo sido definida a quantidade de pontos de demanda a serem atendidos por veículo, determinou-se o número de rotas necessárias. Como o número de clientes a serem atendidos é de 46 e como só podem ser efetuadas oito visitas a cada viagem, teremos que ter, então, a formação de seis rotas. Destas, cinco atenderão a oito pontos de demanda e uma atenderá a seis pontos, satisfazendo assim os 46 pontos de demanda existentes. Desta maneira, contabilizando os garrafões das folgas, para os quais quase sempre há uma demanda também, com a percorrida das seis rotas duas vezes por semana, a quantidade entregue por semana será de 708 garrafões por semana, e, portanto, de 2.832 garrafões por mês.

Na Figura 1, a seguir, são apresentadas as nove rotas desenvolvidas pela empresa por ocasião da pesquisa e na Tabela 1, as distâncias percorridas por cada uma das nove rotas.

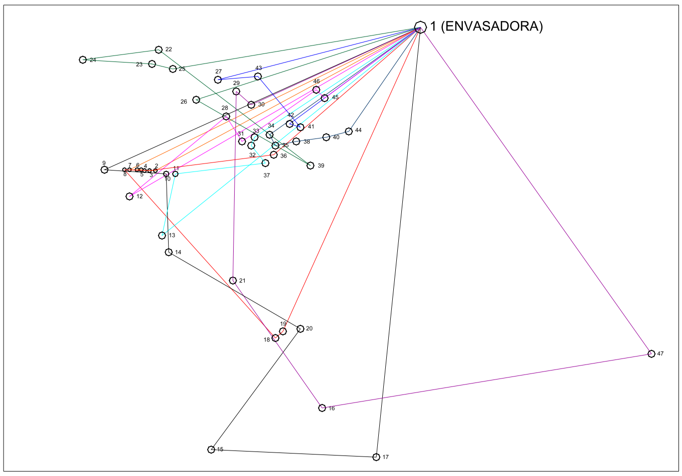

Figura 1. Roteamento desenvolvido pela empresa por ocasião da pesquisa (9 rotas).

3. Metodologia Proposta para a Solução do PRV

É aqui apresentada uma metodologia para realizar o roteamento de veículos, sendo que a mesma é ilustrada a um estudo de caso – problema de distribuição de água mineral na cidade de Itú. Embora a quantidade de pontos de demanda a ser atendida neste problema específico não seja muito grande, optou-se pela utilização dos algoritmos heurísticos e meta-heurísticos, afim de que esta metodologia possa, de fato, ser aplicada em problemas de qualquer ordem.

A metodologia está dividida em quatro etapas: na primeira etapa é aplicado o Algoritmo de Teitz & Bart (Teitz & Bart, 1968) para a determinação de medianas, cuja quantidade varia de acordo com a quantidade de pontos a serem atendidos e, também com a capacidade dos veículos que fazem a entrega; na segunda etapa, propõe-se a aplicação do algoritmo de Gillett e Johnson modificado (Bodin et al., 1983; Corrêa, 2000), definindo os diversos *clusters* de pontos a serem atendidos por cada um dos veículos. Na terceira etapa é definido o roteamento para cada veículo, ou seja, a sequência em que os pontos deverão ser atendidos e, para

Tabela 1. Distâncias percorridas em cada uma das rotas apresentadas na Figura 1.

Rotas	Distância (m)	Sequência de atendimento							
1	16.639	1	9	10	14	20	15	17	1
2	10.972	1	36	3	8	18	19	1	
3	8.593	1	33	32	37	11	13	1	
4	5.875	1	42	41	43	27	1		
5	11.717	1	25	23	24	22	39	26	1
6	8.554	1	45	46	31	28	12	1	
7	7.812	1	2	4	5	6	7	1	
8	4.785	1	34	35	38	40	44	1	
9	15.414	1	47	16	21	29	30	1	
Total	**90.361**								

isto, faz-se uso dos algoritmos heurísticos dos *Savings* de Clarke & Wright (Bodin et al., 1983), da Inserção do Ponto Mais Próximo (Bodin et al., 1983), da meta-heurística Busca Tabu (Glover & Laguna, 1997) e de algumas de suas combinações, comparativamente. Finalmente, na quarta e última etapa é realizada a análise e comparação entre as rotas obtidas verificando-se a melhor opção de resolução para o caso aqui abordado.

3.1 Etapa 1: localização das medianas

De acordo com Minieka (1978), dado um grafo, a mediana é o vértice cuja distância dele a todos os outros é a menor possível. Historicamente, os estudos relativos à aplicação de medianas na resolução de problemas, tiveram início com Weber que, em 1929, estudou a localização de uma fábrica visando minimizar a distância desta aos diversos pontos de matéria-prima e o mercado consumidor. A extensão deste problema deu origem à metodologia das p-medianas. A busca de p-medianas em um grafo é um problema clássico de localização, que tem por objetivo localizar p facilidades (medianas) de forma a minimizar a soma das distâncias de cada vértice a sua facilidade mais próxima, sendo comumente chamados de "problemas de localização de soma mínima (*minisum*)".

3.1.1 Formulação matemática para o problema das p-medianas:

A formulação proposta por Christofides (1975), resolve o problema de localização das p-medianas como um Problema de Programação Linear Inteiro Binário (PPLIB).

Seja $[\xi_{ij}]$ uma matriz de alocações, onde:

$$\xi_{ij} = \begin{cases} 1, & \text{se o vértice } x_j \text{ é alocado ao vértice } x_i; \\ 0, & \text{caso contrário.} \end{cases} \tag{1}$$

Além disto,

$$\xi_{ii} = \begin{cases} 1, & \text{se o vértice } x_i \text{ é uma mediana;} \\ 0, & \text{caso contrário.} \end{cases} \tag{2}$$

O problema das p-medianas pode então ser formulado como segue.

$$\min \quad Z = \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij}\xi_{ij} \tag{3}$$

$$\text{sa} \quad \sum_{i=1}^{n} \xi_{ij} = 1 \qquad \text{para } j = 1..n \tag{4}$$

$$\sum_{i=1}^{n} \xi_{ii} = p \tag{5}$$

$$\xi_{ij} \leq \xi_{ii} = 1 \qquad \text{para todo } i, j = 1..n \tag{6}$$

$$\xi_{ij} \in \{0, 1\} \tag{7}$$

onde $[dij]$ é a matriz de distâncias ponderadas, isto é, a matriz de distâncias com cada coluna j multiplicada pelo peso v_j, cujos valores serão determinados por cada problema. A função objetivo (3) minimiza a soma das distâncias dos vértices de demanda até o conjunto de medianas. As restrições (4) asseguram que todo vértice x_j é alocado a um, e somente um, vértice mediana x_i. As restrições (5) asseguram que existem exatamente p

vértices medianas. As restrições (6) garantem que as alocações só podem ser feitas a vértices-medianas. As restrições (7) garantem a integralidade, ou seja, ξ_{ij} é variável binária, podendo assumir o valor "0" ou "1".

Se as restrições (7) forem reformuladas para $\xi_{ij} \geq 0$, então o problema resultante é um Problema de Programação Linear. Em função de o problema ser formulado como um PPLIB, por si só já é bastante complexo, dependendo da cardinalidade de nós do grafo, sua resolução torna-se inviável. Desta forma, são propostas heurísticas para resolver o problema.

Diversas heurísticas têm sido desenvolvidas para o problema das p-medianas. Apresenta-se a seguir o Algoritmo de Teitz & Bart aplicado na primeira etapa da resolução do caso real, objeto deste estudo.

3.1.2 Descrição do algoritmo de Teitz & Bart

Entre as heurísticas desenvolvidas para o problema das p-medianas, a mais conhecida foi desenvolvida por Teitz & Bart (1968), denominada de Algoritmo das p-medianas de Teitz & Bart. Esta heurística baseia-se na substituição de vértices. Sua finalidade é melhorar o valor da função objetivo a cada iteração, partindo-se de uma solução inicial. Além de produzir boas soluções para problemas pequenos, principalmente quando aplicada várias vezes ao mesmo problema com diferentes soluções iniciais, esta heurística é de fácil implementação.

O Algoritmo de Teitz & Bart para o problema das p-medianas pode ser definido como segue: seja $G(V, A)$ um grafo não direcionado onde V são os seus vértices e A as suas arestas. Seja v_i um vértice qualquer pertencente a V e sejam conhecidas as distâncias euclidianas entre todos os vértices de V. Denomina-se número de transmissão do vértice v_i, $\sigma(v_i)$, a soma do produto entre as menores distâncias existentes do vértice v_i a todos os demais vértices v_j do grafo e o peso w_j associado a cada vértice v_j. Sendo n o número total de vértices do grafo, o número de transmissão é dado pela equação (8) a seguir.

$$\sigma(v_i) = \sum_{j=1}^{n} w_j d(v_i, v_j), \quad v_i, v_j \in V \tag{8}$$

Assim sendo, v_m é uma mediana se (9) for atendida,

$$\sigma(v_m) = \min\{\sigma(v_i)\}, \forall v_i \in V \tag{9}$$

ou seja, entre todos os vértices do grafo, v_m é aquele que produz a menor soma total das distâncias desde si próprio até cada um dos demais vértices do grafo.

Para o problema de encontrar p-medianas ($p > 1$), toma-se um conjunto de vértices (V_p) pertencentes a V escolhidos aleatoriamente, onde a quantidade de vértices em V_p é igual ao número de medianas procuradas ($|V_p| = p$).

$$d(V_p, v_j) = \min\{d(v_i, v_j)\} \forall v_i \in V_p, v_j \in V \tag{10}$$

$$\sigma(V_p) = \sum_{j=1}^{n} w_j d(V_p, v_j), \forall v_j \in V. \tag{11}$$

Desta maneira, um conjunto de p vértices é a solução ótima para o problema das p-medianas se, entre todos os outros conjuntos de p vértices do grafo, é aquele que produz a menor distância total desde si próprio até todos os outros vértices do grafo. Portanto, deve-se ter 12.

$$\sigma(V_{p_{\text{solução ótima}}}) = \min\{\sigma(V_p)\}, \forall V_p \subset V. \tag{12}$$

O objetivo do algoritmo das p-medianas de Teitz & Bart é, portanto, encontrar um conjunto V_p em V para o qual o número de transmissão seja mínimo. A seguir é descrito o procedimento básico executado pelo algoritmo das p-medianas de Teitz & Bart (Teitz & Bart, 1968).

Algoritmo de Teitz & Bart:

Passo 1: Selecione aleatoriamente um conjunto $V_p \subset V$ com $|V_p| = p$ para formar uma solução inicial para o problema.

Passo 2: Rotule todos os vértices $v_i \in \{V - V_p\}$ como "não analisados".

Passo 3: Enquanto existirem vértices não analisados em $\{V - V_p\}$ faça: Selecione um vértice não analisado $v_i \in \{V - V_p\}$, e calcule a redução Δ_{ij} do número de transmissão, para todos os vértices v_j pertencentes a V_p, conforme equação (13) a seguir.

$$\Delta_{ij} = \sigma(V_p) - \sigma(V_p \cup \{v_i\} - \{v_j\}), \forall v_j \in V_p \tag{13}$$

Faça $\Delta_{ij_{\text{máximo}}} = \max[\Delta_{ij}]$, $\forall \Delta_{ij}$ calculado anteriormente.

Se $\Delta_{ij_{\text{máximo}}} > 0$ então:

Faça $V_p = (V_p \cup \{v_i\} - \{v_j\})$ e insira v_j em $\{V - V_p\}$.

Rotule v_j como "analisado" e inicie o **Passo 3** novamente

Caso contrário , rotule v_i como "analisado" e inicie o **Passo 3** novamente.

Não existindo mais vértices a serem analisados, apresente o conjunto V_p obtido.

Passo 4: Utilize o conjunto V_p obtido no **Passo 3** e inicie novamente pelo **Passo 2**. Se durante a execução do **Passo 3**, houver alguma modificação no conjunto V_p, então volte ao **Passo 2** e continue a execução do algoritmo. Caso contrário, pare e apresente o conjunto V_p como uma solução aproximada para o problema das p-medianas.

3.2 Etapa 2: formação dos *clusters* (agrupamentos)

Após a obtenção das medianas, a próxima etapa é a determinação dos diversos *clusters* (grupos) de pontos de demanda a serem atendidos por cada um dos veículos. Esta etapa é realizada utilizando o algoritmo de Gillett e Johnson modificado (Bodin et al., 1983; Corrêa, 2000). Trata-se de um procedimento heurístico para designar um ponto de demanda a depósitos, que neste problema são representados pelas medianas.

Inicialmente, todos os pontos de demanda encontram-se sem designação. Para cada ponto i sejam $L_1(i)$ e $L_2(i)$ o primeiro e o segundo depósitos mais próximos a i respectivamente.

Para cada ponto i, a diferença: $d(i) = L_2(i) - L_1(i)$ é calculada e todos os pontos são colocados numa "lista de designação" em ordem decrescente pelos valores de $d(i)$. A designação começa pelos primeiros elementos da lista (pontos com maior diferença $d(i)$), e é feita obedecendo-se a capacidade dos depósitos. Durante a designação, sempre que a última vaga restante em um depósito for preenchida, a diferença $d(i)$ é recalculada para todos os pontos de demanda que ainda não foram designados, considerando-se apenas os depósitos cuja capacidade não esteja esgotada. Estes pontos de demanda são novamente colocados na "lista de designação", em ordem decrescente pelos valores de $d(i)$, e a designação continuará até que todos os pontos de demanda tenham sido designados para algum depósito.

Algoritmo de Gillett e Johnson modificado:

Passo 1: Para todo ponto i não designado, faça: Encontre $L_1(i)$ e $L_2(i)$ respectivamente o primeiro e o segundo depósitos (medianas) mais próximos do ponto de demanda i cujas capacidades não estejam esgotadas. Calcule a diferença $d(i) = L_2(i) - L_1(i)$ e coloque o ponto i na "lista de designação" pela ordem decrescente dos valores de $d(i)$. Iniciando pelo topo da "lista de designação" designe o ponto de demanda i para o depósito (mediana) mais próximo.

Passo 2: Enquanto existirem pontos de demanda sem designação, faça: designe o ponto de demanda atual para o depósito (mediana) mais próxima de sua localização que dispuser de capacidade; diminua uma unidade da capacidade do depósito (mediana) para o qual o ponto de demanda atual foi designado; se a capacidade do depósito (mediana) que recebeu o ponto de demanda atual ficou esgotada (igual a zero), então, volte ao Passo 1 e recalcule a "lista de designação"; caso contrário, continue.

3.3 Etapa 3: roteamento de veículos

A etapa de roteamento para cada veículo, ou seja, a seqüência em que os pontos de demanda pertencentes a um mesmo *cluster* devem ser atendidos foi realizada utilizando os algoritmos heurísticos: *Savings* de Clarke e Wright (Bodin et al., 1983; Clarke & Wright, 1964); Inserção do Ponto de Demanda mais Próximo (Bodin et al., 1983); da meta-heurística Busca Tabu (Glover & Laguna, 1997) e através de algumas de suas combinações, comparativamente.

3.3.1 Descrição do algoritmo dos *savings* de Clarke e Wright

O algoritmo dos *savings* (ecomomias) de Clarke e Wright é um dos algoritmos heurísticos clássicos de construção de rotas, sendo muito aplicado na resolução tanto do Problema do Caixeiro Viajante Simples, como na dos Múltiplos Caixeiros Viajantes. O procedimento deste algoritmo é a construção de rotas simultaneamente, sendo que a sua característica principal é o conceito de "economia" obtido através da ligação de dois nós de forma sucessiva num grafo, ou seja, a cada iteração vai-se efetuando trocas nestas rotas objetivando um conjunto melhor de rotas.

O processo é inicializado com tantas rotas quantas forem o número de pontos de demanda, supondo-se assim, que existam n veículos a fim de atender as demandas de cada um dos pontos individualmente. O cálculo da economia de distâncias resultantes da inserção de pontos de demanda, entre o depósito e outro ponto de demanda é determinado para todos os pontos de demanda. Verifica-se, assim, a possibilidade de trocar dois veículos que atendem aos pontos de demanda i e j por apenas um, visando a maximizar as economias.

Algoritmo dos *savings* de Clarke e Wright:

Passo 1: Selecione o nó que será definido como depósito (nó 1), inicialize as n rotas ligando cada ponto de demanda exclusivamente ao depósito, obtendo-se as rotas $(1 - i - 1)$, para $i = 2, ..., n$.

Passo 2: Calcule os *savings* para todos os pares (i, j) da seguinte maneira: $s_{ij} = 0$ se $i = j$; $i = 1$; $j = 1$. Nos demais caso calcule: $s_{ij} = c_{1i} + c_{1j} - c_{ij}$, para $i, j = 2, 3, ..., n$.

Passo 3: Ordene os *savings* dos pares (i, j) em ordem decrescente.

Passo 4: Iniciando do topo da lista dos *savings* forme sub-rotas maiores ligando os pontos de demanda i e j apropriados. Os pontos de demanda só podem ser incluídos "nas pontas" das sub-rotas. Repetir este procedimento até que todos os pontos de demanda tenham sido "designados".

3.3.2 Descrição do algoritmo de inserção do mais próximo:

Segundo Goldbarg & Luna (2000), o algoritmo da inserção do mais próximo é uma heurística que possui um processo onde três níveis de decisão são envolvidos: a escolha do vértice a ser inserido na solução; a posição de inserção deste novo vértice; a decisão de um ciclo inicial.

Normalmente este tipo de heurística parte de uma sub-rota (*subtour*) inicial (um ciclo normalmente de comprimento igual a três) e vai selecionando e inserindo vértices ainda não incluídos na solução até completar um ciclo. No algoritmo aplicado neste trabalho, o critério utilizado para a seleção dos vértices a serem acrescidos ao *subtour* é a inserção do vértice mais próximo.

Algoritmo de inserção do mais próximo:

Passo 1: Comece com um sub-grafo consistindo somente do nó i.

Passo 2: Ache o nó k tal que c_{ik} seja mínimo e forme a sub-rota $i - k - i$.

Passo 3: (Passo de seleção): Dada uma sub-rota, encontre o nó k que não esteja na sub-rota e que esteja mais próximo de qualquer nó da sub-rota.

Passo 4: (Passo de inserção): Encontre o arco (i, j) da sub-rota que minimize: $c_{ik} + c_{kj} - c_{ij}$. Insira k entre i e j.

Passo 5: Volte ao Passo 3 até obter um ciclo Hamiltoniano.

3.3.3 Busca tabu

A meta-heurística Busca Tabu (*Tabu Search*), segundo Viana (1998), teve origem a partir de uma solução de Glover (1986) para problemas de Programação Inteira. Glover é considerado o criador do Algoritmo *Tabu Search* devido aos seus inúmeros trabalhos publicados.

Segundo Glover & Laguna (1997) em seu livro especializado neste assunto, Busca Tabu é uma meta-heurística que guia um procedimento heurístico de busca local para explorar o espaço solução além do "ótimo local".

A Busca Tabu pode ser caracterizada como sendo uma busca através das soluções vizinhas. Dada uma solução $x \in X$ em um conjunto associado de soluções vizinhas $V(x) \subset X$ chamadas soluções vizinhas a x, toda solução $x' \in V(x)$ pode ser gerada a partir de x por certo tipo de operação denominada movimento. Normalmente em Busca Tabu, soluções vizinhas são simétricas, ou seja, x' é solução vizinha a x se, e somente se, x é solução vizinha a x'.

Este método tem como base as noções de "movimento" e "vizinhança". O "movimento" é uma função que transforma uma solução em outra. A "vizinhança" é o conjunto de soluções obtidas aplicando a uma determinada solução um subconjunto dos movimentos possíveis.

A Busca Tabu, que foi projetada para encontrar boas aproximações para a solução ótima global de qualquer problema de otimização, basicamente norteia-se por três princípios: uso de uma estrutura de dados (fila ou lista) para "guardar" o histórico da evolução do processo de busca; uso de um mecanismo de controle para fazer um balanceamento entre a aceitação, ou não de uma nova configuração, com base nas informações registradas na “lista tabu” referentes às restrições e aspirações desejadas; incorporação de procedimentos que alternam as estratégias de diversificação e oscilação.

O procedimento da meta-heurística Busca Tabu começa a partir de uma solução inicial. Em cada passo gera-se a vizinhança da solução atual e faz-se uma pesquisa para determinar o vizinho com as melhores características. Este vizinho passa a ser a nova solução atual e efetua-se um novo passo.

Para prevenir ciclos e guiar o algoritmo para regiões “boas” do espaço de procura, mantém-se um histórico das soluções já visitadas em memória de curta duração designado por “lista tabu” que tem um papel fundamental. Esta lista não permite que o algoritmo volte para trás e revisite uma solução pela qual tenha passado nos passos anteriores. Na prática, a lista tabu contém os atributos de soluções revisitadas ou os movimentos que lhe deram origem em vez das soluções propriamente dita. O tempo que um movimento deve permanecer nesta lista, em geral, está relacionado com o número de iterações do algoritmo e com o número de movimentos possíveis a partir da solução que está sendo analisada.

Pelo fato de se ter os movimentos proibidos, podendo-se proibir por vezes soluções interessantes, define-se também os critérios de aspiração, que têm o papel de avaliar se, apesar de proibido, vale a pena (ou não) realizar um movimento proibido. Os critérios de aspiração são introduzidos em Busca Tabu a fim de determinar quando uma restrição tabu pode ser “quebrada”, ou seja, quando a restrição pode ser ignorada e o movimento, mesmo classificado como proibido, pode ser executado. Um critério de aspiração bastante utilizado é o de ignorar a restrição tabu sempre que a solução formada por um determinado movimento proibido for melhor do que a melhor solução encontrada até o momento. A aplicação adequada deste procedimento é fundamental para se atingir altos níveis de desempenho em Busca Tabu.

O algoritmo termina quando é atingido um critério de parada. Normalmente o critério de parada é um número pré-especificado de iterações ou um número pré-especificado de iterações após a última melhoria ter sido encontrada.

Como na maior parte dos outros algoritmos, é importante que o algoritmo esteja bem adaptado ao problema a resolver e, para tal, a implementação dos conceitos definidos anteriormente deve ser feita com o cuidado de possuir a estrutura particular do problema em questão.

4. Implementação da Metodologia ao Problema Real e Análise dos Resultados

Objetivando a solução para o caso real apresentado na seção 2, os algoritmos descritos na seção 3 foram devidamente modelados ao mesmo. O procedimento desenvolveu-se segundo as seguintes fases:

1^a **fase:** Nesta fase realizou-se uma visita a empresa, objetivando conhecer os procedimentos atualmente realizados em relação ao roteamento e o levantamento dos dados necessários para o estudo pretendido.

2^a **fase:** O objetivo desta fase foi a determinação, dentre os pontos de demanda, de seis medianas de tal forma que a soma das distâncias de cada ponto até a mediana mais próxima fosse a menor possível. Dentre as várias heurísticas disponíveis, optou-se pela aplicação do Algoritmo das p-medianas de Teitz & Bart.

3^a **fase:** A partir da determinação das medianas, o próximo passo executado foi a formação de agrupamentos (*clusters*) em torno das medianas, de maneira que haja uma distribuição uniforme dos pontos de demanda entre as medianas. A finalidade deste procedimento é obter um melhor roteamento (4^a. Fase). Aplicou-se para tal, o procedimento heurístico de Gillett e Johnson modificado.

4^a **fase:** Esta fase é a de roteamento. Objetivando uma solução que resulte numa maior eficiência com o menor custo possível. Aplicou-se para tal, duas heurísticas e uma meta-heurística: Algoritmo dos *savings* de Clarke e Wright, Algoritmo de Inserção do Mais Próximo e Busca Tabu. A Figura 2, a seguir, apresenta a esquematização destas quatro fases.

Em virtude de não haver *software* disponível no mercado que permitisse a resolução deste problema, todos os algoritmos utilizados para resolver o problema apresentado foram implementados em *Visual Basic*.

4.1 Levantamento de dados: 1^a fase

Inicialmente, procedeu-se a um trabalho manual de localização dos pontos de demanda e da envasadora no mapa digitalizado/*Google Earth*, determinando-se as coordenadas de cada um deles. Ainda nesta fase, obtiveram-se os dados referentes à quantidade de garrafões entregues em cada ponto de demanda, a frequência desta entrega e a sequência de atendimento dos pontos de demanda, como já descrito na seção 2. A localização da envasadora, denotada como ponto 1, e dos pontos de demanda, numerados de 2 a 47, são mostrados na Figura 3.

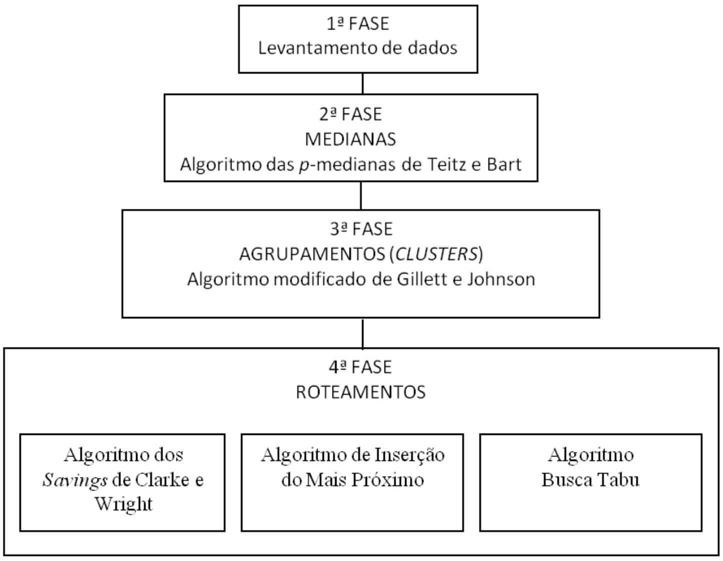

Figura 2. Esquema da metodologia para a resolução do problema de distribuição abordado.

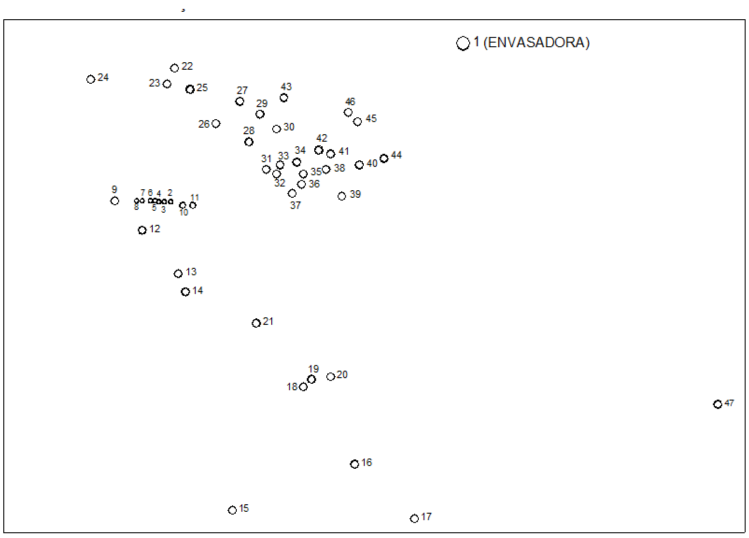

Figura 3. Localização dos pontos de demanda.

4.2 Determinação das medianas: 2^a fase

Em virtude de serem 6 rotas, como já mencionado, buscou-se encontrar 6 medianas dentre os 46 pontos de demanda existentes. Após uma análise dos resultados obtidos com a aplicação do Algoritmo de Teitz & Bart contendo todos os pontos de demanda, seguido dos procedimentos para agrupamento e roteamento, entendeu-se que para uma melhor eficiência, seria necessário que o ponto de número 47, devido a sua localização em relação aos demais pontos, fosse excluído da lista de candidatos à mediana. Procedeu-se, então, a uma nova aplicação do Algoritmo de Teitz & Bart, obtendo-se como medianas, os pontos: 23, 29, 19, 4, 16, 38, mostradas na Figura 4.

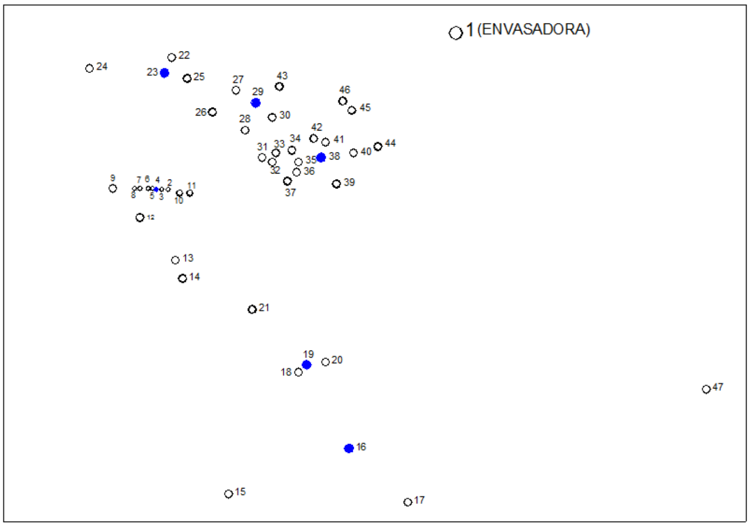

Figura 4. Localização das medianas (obtidas através do algoritmo de Teitz & Bart).

4.3 Formação dos agrupamentos *clusters*: 3^a fase

Após a determinação das 6 medianas, procedeu-se a aplicação do algoritmo modificado de Gillett e Johnson modificado para a formação dos agrupamentos em torno destas medianas, objetivando uma alocação uniforme dos pontos de demanda a serem atendidos. Os resultados obtidos podem ser visualizados na Figura 5, sendo que os pontos designados a cada rota estão no Tabela 2.

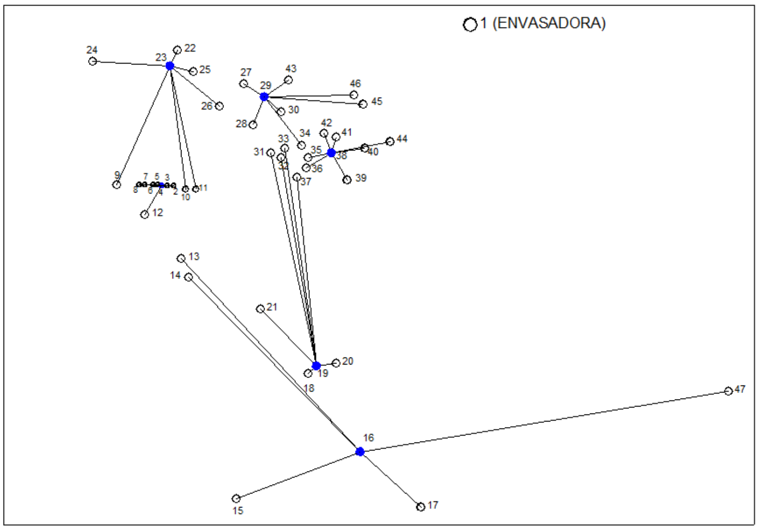

Figura 5. Obtenção dos agrupamentos (algoritmo de Gillett e Johnson).

Tabela 2. Agrupamentos obtidos através do algoritmo de Gillettt e Johnson.

Cluster	Pontos de demanda designados a cada *cluster*							
1	4	12	5	3	6	2	7	8
2	23	24	22	25	26	9	10	11
3	16	17	13	14	47	15		
4	19	21	18	20	31	32	33	37
5	38	39	40	44	41	36	35	42
6	29	43	28	27	45	30	46	34

4.4 Roteamento: 4^a fase

Após a formação dos agrupamentos, procedeu-se a fase de roteamento onde foram aplicados os algoritmos dos *Savings* de Clarke e Wright, inserção do mais próximo e busca Tabu.

A. Algoritmo dos *savings* de Clarke e Wright

A Figura 6 apresenta as rotas determinadas pelo algoritmo dos *savings* de Clarke e Wright. Já a Tabela 3 apresenta para cada rota a distância percorrida e a sequência dos pontos de demanda a serem atendidos, além da distância total percorrida.

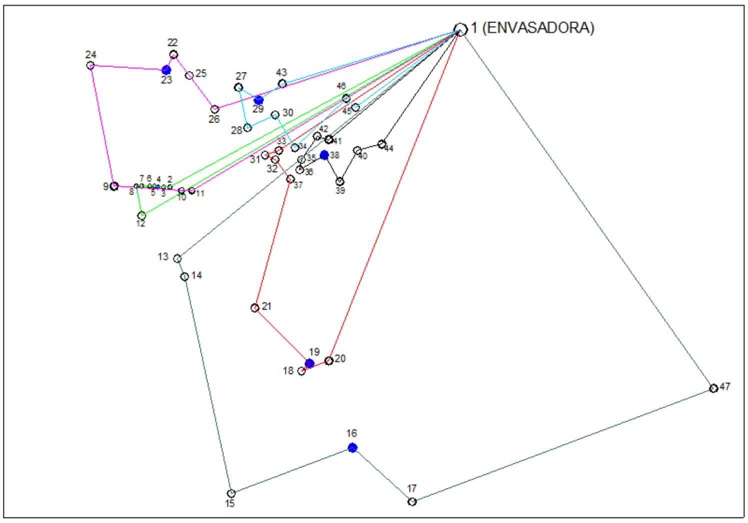

Figura 6. Ilustração do roteamento (algoritmo dos *savings* de Clarke e Wright).

Tabela 3. Distância percorrida em cada uma das rotas da Figura 6.

Rotas	Distância (m)	Sequência de atendimento									
1	8.349	1	2	3	4	5	6	7	8	12	1
2	17.245	1	13	14	15	16	17	47	1		
3	9.592	1	20	18	19	21	37	32	31	33	1
4	10.221	1	26	25	22	23	24	9	10	11	1
5	6.167	1	43	29	27	28	30	34	46	45	1
6	5.268	1	41	42	35	36	38	39	40	44	1
Total	**56.842**										

B. Algoritmo de inserção do mais próximo

A Figura 7 mostra as rotas definidas pleo algoritmo de inserção do mais próximo. A Tabela 4 apresenta para cada rota a sequência de atendimento e a distância percorrida, além de fornecer a distância total percorrida.

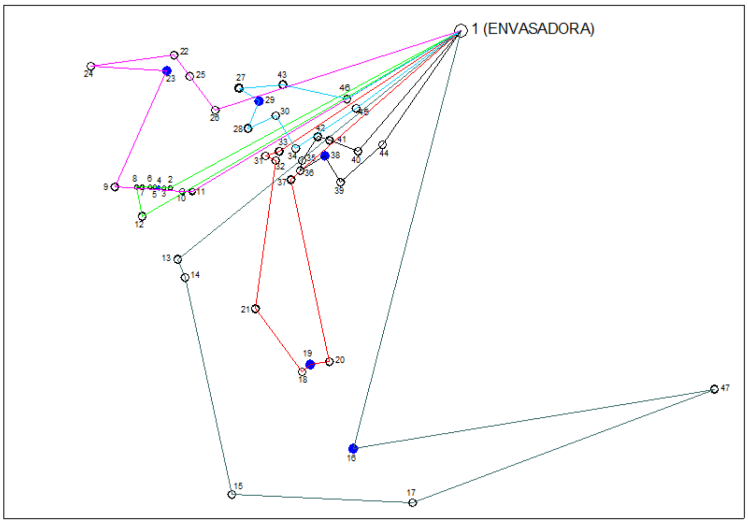

Figura 7. Ilustração do roteamento (algoritmo da inserção do mais próximo).

Tabela 4. Distância percorrida em cada uma das rotas da Figura 7.

Rotas	Distância (m)	Sequência de atendimento									
1	8.349	1	12	8	7	6	5	4	3	2	1
2	20.851	1	16	47	17	15	14	13			
3	9.999	1	33	31	32	21	18	19	20	37	1
4	10.998	1	11	10	9	23	24	22	25	26	1
5	6.218	1	45	34	30	28	29	27	43	46	1
6	5.412	1	44	39	38	36	35	42	41	40	1
Total	**61.827**										

C. Busca tabu

Em seguida procedeu-se o roteamento através de Busca Tabu, também utilizando os agrupamentos formados através do Algoritmo de Gillett e Johnson. Aplicou-se Busca Tabu em 3 rotas iniciais diferentes. Em cada uma das rotas iniciais, aplicou-se Busca Tabu definindo-se 5 trocas em cada iteração. Foram realizados vários testes com relação ao número de iterações, sendo que a melhor solução foi obtida com 300 iterações. A seguir, na Figura 8 e Tabela 5 estão os resultados para uma destas situações.

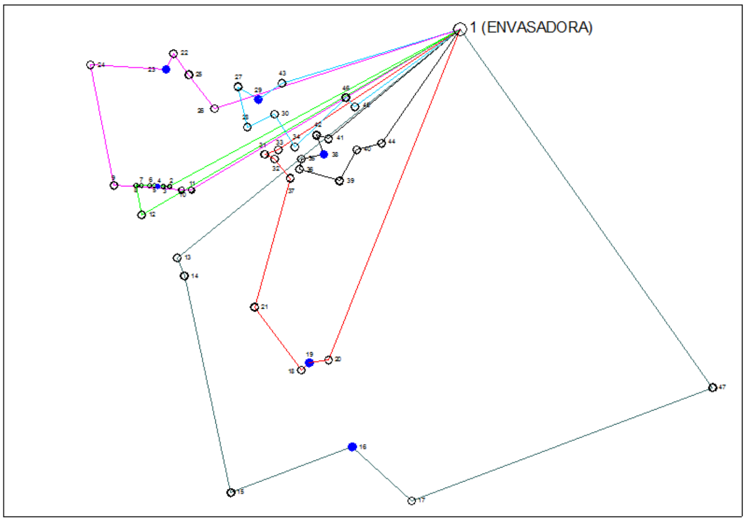

Figura 8. Ilustração do roteamento (busca tabu - 300 iterações).

Diversas simulações adicionais foram realizadas utilizando a busca tabu:

Tabela 5. Distância percorrida em cada uma das rotas da Figura 8.

Rotas	Distância (m)	Sequência de atendimento									
1	8.349	1	2	3	4	5	6	7	8	12	1
2	17.245	1	13	14	15	16	17	47	1		
3	9.486	1	33	31	32	37	21	18	19	20	1
4	10.221	1	26	25	22	23	24	9	10	11	1
5	6.167	1	43	29	27	28	30	34	46	45	1
6	5.246	1	44	40	39	36	35	38	42	41	1
Total	**56.714**										

1. Tomando-se como rotas iniciais, as formadas pelo algoritmo de Clarke e Wright. Da mesma maneira, definiu-se 5 trocas para cada iteração e aplicou-se Busca Tabu variando-se o número de iterações;

2. Tomando-se como rotas iniciais, as formadas pelo algoritmo de Inserção do Mais Próximo. Analogamente, definiu-se 5 trocas para cada iteração e aplicou-se Busca Tabu variando-se o número de iterações.

Após a implementação dos algoritmos pôde-se realizar a análise dos resultados e a comparação entre as diferentes técnicas aplicadas, conforme apresentado na Tabela 6 mais adiante.

Aplicando-se o Algoritmo dos *Savings* de Clarke e Wright, obteve-se uma distância total percorrida de 56.842m, sendo que a economia em relação ao roteamento atual (90.361m) é de 33.519m, os quais em termos percentuais reduz o total percorrido em 37,09%. Esta redução por si só é significativa para qualquer empresa do ramo de distribuição. Com a aplicação do Algoritmo de Inserção do mais Próximo, a distância percorrida obtida foi de 61.827m. A redução em relação ao roteamento atual é de 28.534m, ou 31,58%.

O roteamento através de Busca Tabu foi realizado variando-se as rotas iniciais e o número de iterações. Primeiro as rotas iniciais foram formadas seguindo a ordem de designação obtida para os agrupamentos formados através do algoritmo modificado de Gillett e Johnson. Fixando-se estas rotas iniciais obteve-se uma distância total percorrida de 56.714m após 300 iterações que, dentre diversos testes realizados, apresentou o melhor resultado. Em seguida tomou-se como rotas iniciais as obtidas através do algoritmo dos *Savings* de Clarke e Wright. Da mesma forma, fixando-se estas rotas iniciais, obteve-se após 300 iterações, uma distância total percorrida de 56.714m. Por último, tomou-se como rotas iniciais as obtidas através do algoritmo de Inserção do Mais Próximo. Fixando-se as rotas iniciais, obteve-se uma distância total percorrida de 56.714m após 300 iterações.

Em relação à distância total percorrida, o melhor roteamento encontrado (56.714m) foi obtido através da técnica de Busca Tabu; do Algoritmo dos *Savings* de Clarke e Wright seguido da Busca Tabu e do Algoritmo de Inserção do Mais Próximo seguido da Busca Tabu (hachuriado na Tabela 5). A economia resultante do roteamento obtido através de Busca Tabu em relação ao do Algoritmo dos Savings de Clarke e Wright (puro, ou seja, sem a aplicação da Busca Tabu) foi de 128 m, ou 0,23%, enquanto que em relação ao Algoritmo de Inserção do mais Próximo (puro) foi de 5.113, ou 8,27%.

5. Conclusões

O objetivo principal deste trabalho foi apresentar uma metodologia, composta de vários algoritmos, que permita a resolução de um problema real de roteamento de veículos de forma rápida e eficiente.

Aplicou-se para tal propósito os seguinte algoritmos: o Algoritmo das p-medianas de Teitz & Bart (para a determinação de p-medianas), o Algoritmo modificado de Gillett e Johnson (para a *clusterização* dos pontos de demanda em torno das p-medianas), e os algoritmos dos *Savings* de Clarke e Wright, da Inserção do Mais Próximo, da Busca Tabu e algumas de suas combinações (para a obtenção das rotas, ou seja, da seqüência dos pontos dentro de cada *cluster*, de distância mínima) comparativamente.

A melhor solução obtida para o caso real abordado (distribuição de água mineral) foi através da aplicação da Busca Tabu e, também, através do algoritmo dos *Savings* de Clarke e Wright seguido da Busca Tabu e, ainda, através do algoritmo da Inserção do mais Próximo seguido da Busca Tabu. Todas estas três alternativas forneceram a distância total percorrida igual a 56.714m. Têm-se, assim, fortes indicativos de que, além do fato da meta-heurística Busca Tabu (pura) fornecer a solução ótima para o problema de pequeno porte aqui abordado, também o fato de que ela pode ser bastante promissora ao utilizá-la em conjunto com os procedimentos heurísticos aqui contemplados para problemas de maior porte.

Tabela 6. Resultados comparativos (distâncias em m) para o problema real abordado.

	Algoritmos				
	A	**B**	**C**	**A+C**	**B+C**
1	8.349	8.349	8.349	8.349	8.349
2	17.245	20.851	17.245	17.245	17.245
3	9.592	9.999	9.486	9.592	9.592
4	10.221	10.998	10.221	10.221	10.221
5	6.167	6.218	6.167	6.167	6.167
6	5.267	5.412	5.246	5.267	5.267
Total	**56.842**	**61.827**	**56.714**	**56.714**	**56.714**

A: *Savings* de Clarke e Wright;
B: Inserção do Ponto Mais Próximo;
C: Busca Tabu.

A solução ótima para o caso abordado, obtida através da resolução de um modelo matemático de Programação Inteira, com o uso do software comercial LINGO (*Language for Interactive General Optimizer*) forneceu exatamente a solução obtida através da metodologia apresentada, mostrando com isto a eficácia da mesma.

A vantagem da metodologia em relação a solução exata está na eficiência da mesma, ou seja, enquanto o software precisa de algumas horas para resolver o problema de roteamento de 46 pontos de demanda, a metodologia o faz em poucos segundos. Vale salientar que para problemas maiores, com um maior número de pontos de demanda, a adoção do modelo matemático seria totalmente inviável, enquanto que a metodologia aqui apresentada pode tolerar o número de pontos que for necessário.

Referências

Bodin, L.; Golden, B.; Assad, A. & Ball, M., Routing and scheduling of vehicles and crews: the state of the art. *Computers & Operations Research*, 10(2):63–211, 1983.

Boulduc, M.C.; Laporte, G.; Renaud, J. & Boctor, F.F., A tabu search heuristic for the split delivery vehicle routing problem with production and demand calendars. *European Journal of Operations Research*, 202(1):122–130, 2010.

Breedam, A.V., Comparing descent heuristics and metaheuristics for the vehicle routing problem. *Computers & Operations Research*, 28(4):289–315, 2001.

Choi, E. & Tcha, D.W., A column generation approach to the heterougeneous fleet vehicle routing problem. *Computers & Operations Research*, 34(7):2080–2095, 2007.

Christofides, N., *Graph Theory - An Algorithmic Approach*. New York, USA: Academic Press, 1975.

Clarke, G. & Wright, J.W., Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4):568–581, 1964.

Corrêa, E.S., *Algoritmos Genéticos e Busca Tabu Aplicados ao Problema das p-medianas*. Dissertação de mestrado, Programa de Pós-Graduação em Métodos Numéricos em Engenharia, Universidade Federal do Paraná, Curitiba, PR, 2000.

Du, L. & He, R., Combining nearest neighbor search with tabu search for large scale vehicle routing problem. *Physics Procedia*, 25:1536–1546, 2012.

Glover, F., Future paths for integer programming and links to artificial inteligence. *Computers & Operations Research*, 13(5):533–549, 1986.

Glover, F. & Laguna, M., *Tabu Search*. University of Colorado, USA: Kluwer, Boulder, Academic Publishers, 1997.

Goldbarg, M.C. & Luna, H.P.L., *Otimização Combinatória e Programação Linear - Modelos e Algoritmos*. Rio de Janeiro, RJ: Editora Campus, 2000.

Leung, S.C.H.; Zhou, X.; Zhang, D. & Zheng, J., Extended guided tabu search and a new packing algorithm for two-dimensional loading vehicle routing problem. *Computers & Operations Research*, 38:205–215, 2011.

Minieka, E., *Optimization Algorithms for Networks and Graphs*. New York, USA: Marcel Dekker, Inc., 1978.

Teitz, M.B. & Bart, P., Heuristics methods for estimating the generalized vertex median of a weighted graph. operations research. *Operations Research*, 16:955–961, 1968.

Toth, P. & Vigo, D., The granular tabu search and its application to the vehicle-routing problem. *Journal on Computing*, 15(4):333–346, 2003.

Valle, A. & Cunha, A.S., Exact algorithms for a selective vehicle routing problem where the longest route is minimized. *Discrete Mathematics*, 35:133–138, 2009.

Viana, G.V.R., *Meta-Heurísticas e Programação Paralela em Otimização Combinatória*. Fortaleza, CE: UFC Edições, 1998.

Notas Biográficas

Sheila Margot Gonçalves é graduada em Licenciatura em Matemática (UFPR, 1990); especialista em Matemática (UFPR, 1999); mestre em Métodos Numéricos em Engenharia (UFPR, 2003). Atualmente é docente da Universidade Positivo (UP) e da Faculdade Educacional Araucária (FACEAR).

Maria Teresinha Arns Steiner é graduada em Licenciatura em Matemática e em Engenharia Civil (UFPR, 1978 e 1981, respectivamente), mestre e doutor em Engenharia de Produção, na área de Pesquisa Operacional (UFSC, 1988 e 1995, respectivamente). Tem pós-doutorado em Pesquisa Operacional (ITA, 2005). De 1978 a 2010 foi docente dos departamentos de Matemática e de Engenharia de Produção da UFPR. Desde 2011 é docente do programa de Engenharia de Produção e Sistemas da PUCPR.

Luzia Vidal de Souza Zamboni é graduada em Licenciatura em Matemática (UFPR, 1988), mestre e doutor em Métodos Numéricos em Engenharia (UFPR, 1997 e 2006, respectivamente). Atualmente é docente da UFPR, do departamento de Expressão Gráfica e do programa de Pós-Graduação em Métodos Numéricos em Engenharia.