

O problema da roteirização periódica de veículos

Luciele Wu¹; Cláudio Barbieri da Cunha²

Resumo: Este artigo trata do problema de roteirização periódica, uma generalização do problema clássico de roteirização de veículos. Neste problema, os pontos a serem atendidos podem ter diferentes frequências de visitas e a roteirização deve considerar um período de planejamento. Portanto, é preciso decidir quais são os melhores dias de visita para cada ponto (respeitando sua frequência de visitas), para que a roteirização seja otimizada para todo o horizonte de planejamento. Este problema ocorre no contexto da logística *inbound* na indústria automobilística. Neste caso, não só os roteiros de coleta devem ser determinados como também os dias em que cada fornecedor pode ser visitado, dado que os diferentes fornecedores podem ter diferentes frequências. Com a finalidade de explorar novas abordagens para este tipo de problema de maneira que houvesse um aumento na qualidade de soluções e diminuição do tempo de processamento computacional, são propostas três estratégias heurísticas de solução, baseadas em inserção, GRASP e algoritmos genéticos. As estratégias foram implementadas e testadas para problemas encontrados na literatura, bem como problemas gerados aleatoriamente.

Abstract: This article deals with the period vehicle routing problem, which can be viewed as a generalization of the classical vehicle routing problem. In this problem, points to be serviced may have different frequencies and the routing is accomplished over a given planning horizon. Thus, one must decide which are the best days to visit each point (respecting its given frequency), so that the routing over the overall planning horizon is optimized. This problem arises in the context of inbound logistics in the automotive industry. In this case, pick-up routes should be determined as well as the days each supplier must be visited, given that the different suppliers may have different frequencies. To explore new approaches to this problem, aiming to improved solutions and reduced CPU time, three new heuristics are proposed, based on GRASP and genetic algorithm. The heuristics were implemented and tested against problems found in the literature, as well as in randomly generated problems.

1. INTRODUÇÃO

O termo roteirização (ou roteamento) de veículos é a forma que vem sendo utilizada como equivalente ao inglês “routing” (ou “routeing”) para designar o processo para a determinação de um ou mais roteiros, ou seqüências de paradas a serem cumpridos por veículos de uma frota, objetivando visitar um conjunto de pontos geograficamente dispersos, em locais pré-determinados, que necessitam de atendimento.

Em linhas gerais, o problema de roteirização periódica tem, em sua essência, os mesmos conceitos básicos de um problema de roteirização clássico. Ele pode ser considerado como uma extensão do problema original por ter como característica principal um período de planejamento superior a um dia. Neste caso, a cada ponto de atendimento está associada uma frequência de atendimento (por exemplo, diária, em dias alternados, uma vez por semana, etc.), sendo necessário definir também em que dias cada ponto será visitado, respeitando sua frequência, de tal forma que o custo total

da roteirização em todo o período de planejamento seja minimizado. É evidente que uma má designação dos pontos a serem atendidos em cada dia pode levar a necessidades de tamanhos de frota muito distintos nos diferentes dias, ou ainda distâncias percorridas muito diferentes, ambos indesejáveis.

Uma aplicação prática bastante relevante e atual da roteirização periódica é o abastecimento de uma planta de uma montadora de veículos. Atualmente, a maioria das montadoras no Brasil atua no sistema *just-in-time*, ou seja, não mantém e nem armazena estoque de insumos e matérias-primas utilizadas na produção dos veículos. Usualmente um operador logístico é contratado para efetuar as coletas, nos fornecedores, das quantidades exatas necessárias para o próximo dia ou turno de trabalho, de acordo com uma programação de produção definida previamente. Com isso, a montadora evita a complexa tarefa de recebimento e movimentação de peças e componentes de um número elevado de fornecedores que entregam diretamente na planta, transferindo a responsabilidade pela coleta, conferência e entrega para um operador logístico. Racionaliza-se ainda o sistema de recebimento e as áreas necessárias, evitando-se as filas para descarga de um número elevado de fornecedores. Nesse sistema de coleta, alguns fornecedores são visitados diariamente, enquanto outros são coletados apenas uma vez por semana (nesse caso porque a demanda e os volumes envolvidos podem não justificar uma frequência de coleta maior).

¹ Luciele Wu, Departamento de Engenharia de Transportes, Escola Politécnica da Universidade de São Paulo. São Paulo, SP, Brasil (e-mail: luciele.wu@gmail.com).

² Cláudio Barbieri da Cunha, Departamento de Engenharia de Transportes, Escola Politécnica da Universidade de São Paulo. São Paulo, SP, Brasil (e-mail: cbcunha@sc.usp.br).

Tal sistema de abastecimento da indústria automobilística é conhecido como coleta *milk-run*, uma analogia ao antigo sistema de coleta de leite nas fazendas por cooperativas de laticínios para pasteurizar o produto.

Outras aplicações de problemas de roteirização periódica incluem, entre outros, problemas de limpeza (varrição) de ruas ou de coleta de lixo urbano ou industrial.

Segundo Cunha (2000), problemas que envolvem a roteirização de veículos correspondem a problemas combinatórios, do tipo NP-difícil, o que torna impossível a obtenção de soluções ótimas através de pacotes de otimização para instâncias encontradas no mundo real, trazendo o desafio da busca de novas heurísticas mais eficientes que possibilitem a sua aplicação a problemas práticos do dia-a-dia das empresas. O problema de roteirização periódica apresenta a complexidade adicional de ser necessário definir ainda os melhores dias de atendimento para cada ponto, de tal forma que a roteirização para o conjunto de dias que compõem um período de planejamento seja otimizada.

Nesse contexto, pretende-se, neste artigo, explorar novas estratégias heurísticas com a finalidade de obter melhores qualidades de resultados e tempos computacionais menores. Três estratégias são propostas sendo elas baseadas em heurísticas simples de inserção, GRASP e Algoritmos Genéticos. São também apresentados resultados de experimentos computacionais realizados para alguns problemas clássicos da literatura e problemas gerados aleatoriamente.

2. REVISÃO BIBLIOGRÁFICA

Antes de receber a denominação de roteirização periódica, o problema proposto era considerado como um problema de designação (*assignment*) de dias de visita. Beltrami e Bodin (1974) fazem um estudo com essa denominação para a coleta de lixo urbano na cidade de Nova Iorque. Os autores discutem algumas estratégias de solução em que (i) os pontos são agrupados e depois roteirizados, ou (ii) são criados os roteiros, e com base neles são definidos os dias de visita. Em ambos os casos as heurísticas se baseiam numa versão modificada do método de economias de Clarke e Wright (1964).

Russel e Igo (1979) propõem um modelo matemático para o problema de atribuição de dias e roteirização que, segundo os autores, é intratável, mesmo para instâncias de menor porte. Os autores desenvolvem três heurísticas de solução, inspiradas no trabalho de Beltrami e Bodin (1974). Numa delas, os pontos são agrupados de forma a se obter grupos compactos; nesse caso, os pontos são alocados sequencialmente aos dias de visita, considerando-se a distância a cada um dos

grupos de pontos segundo uma ordem que depende da frequência de visita (os pontos que requerem apenas uma visita no período são os últimos a serem alocados). As outras duas heurísticas buscam melhorar os resultados da heurística anterior através da consideração explícita dos roteiros, sendo que, numa delas também é utilizada uma versão modificada do algoritmo de economias de Clarke e Wright (1964). Os autores também tratam do problema de coleta de resíduos. A medida básica de avaliação foi a distância total percorrida no período.

O primeiro artigo que adota a denominação “roteirização periódica” é o de Christofides e Beasley (1984), que caracterizam o problema e apresentam o modelo matemático completo para um caso genérico. Os autores decompõem e modelam o problema em duas partes, sendo a primeira uma escolha da combinação de dias para cada cliente com o objetivo de minimizar o custo total no período. Cada dia do período é analisado como um subproblema de roteirização. Os autores concluem ser este um problema muito complexo, que envolve um enorme número de variáveis. Para auxiliar na solução, são propostas duas relaxações que transformam os subproblemas de roteirizações diárias em um *mediam problem* (soma das distâncias totais de todos os pontos visitados em um certo dia a um ponto central) e um problema de caixeiro viajante.

A heurística proposta pelos autores é baseada primeiro na escolha inicial das combinações de dias de entrega para os clientes e, em seguida, uma troca (*interchange*) da combinação dos dias dos clientes, na tentativa em reduzir o custo total. A dificuldade dessa estratégia é avaliar o efeito da mudança dos clientes nas combinações dos dias de visita, uma vez que para isso é necessário realizar a roteirização diária com as mudanças, e comparar o resultado final com o resultado anterior.

Chao *et al.* (1995) propõem uma heurística para resolver o problema de caixeiro viajante periódico. Com a finalidade de facilitar a etapa de alocação de clientes em dias de visitas, um algoritmo de designação busca equilibrar o número de clientes que são atendidos em cada dia no período, permitindo obter uma solução inicial. Busca-se, dessa forma, minimizar o número máximo de clientes visitados em um único dia, evitando a concentração deles em alguns dias. Uma vez obtida essa solução inicial, executa-se uma fase de melhoria da solução, baseada na realocação de cada ponto em dias diferentes, um por vez. Após a escolha final dos dias de visitas para cada cliente, executa-se uma etapa de limpeza (*clean-up*), em que se busca diminuir a distância total percorrida em cada dia, trocando os clientes de roteiros, além da utilização de um algoritmo de melhoria do tipo *2-opt*.

Cordeau *et al.* (1997) propõem uma estratégia base-

ada em busca tabu para resolver três problemas bastante conhecidos: (i) roteirização periódica, (ii) caixeiro viajante periódico, e (iii) roteirização de veículos com múltiplos depósitos. Uma contribuição deste trabalho é a comprovação que o problema de múltiplos depósitos pode ser formulado como um caso especial de roteirização periódica. As três heurísticas baseadas em busca tabu são semelhantes entre si, exceto pela construção da solução inicial, podendo a mesma ser viável ou não. A diversificação do método é baseada em uma função de penalidade dos atributos mais recorrentes nas soluções que implicam em aumento na função objetivo. Os resultados obtidos através dos testes com instâncias retiradas da literatura mostraram que os algoritmos desenvolvidos são melhores que os encontrados anteriormente.

Vianna *et al.* (1999) propõem uma metaheurística híbrida paralela para o problema de roteirização periódica; a mesma se baseia em algoritmo genético paralelo, *scatter search* e uma heurística de busca local, utilizando o modelo de ilha (*island model*) na qual a população de cromossomos é repartida em várias subpopulações, sendo que a frequência de migração entre subpopulações dos cromossomos é baixa, somente executada quando a renovação da subpopulação é necessária. Os indivíduos são representados por genes que caracterizam os pontos de demanda que devem ser atendidos, sendo que o código que está contido em cada um deles representa as combinações de dias de visitas permitidas. A roteirização diária é realizada através do algoritmo de economias de Clarke e Wright (1964). A fase de reprodução é feita por cruzamento e mutação, sendo que nesta fase a capacidade máxima diária da frota também tem que ser respeitada. A etapa de diversificação é realizada através de um operador de migração (*island model*) com uma taxa de renovação pequena. Os resultados obtidos são comparados com outros da literatura, verificando-se resultados superiores em termos de tempo de processamento computacional e qualidade de solução, o que mostra que a utilização de algoritmos genéticos para este tipo de problema traz benefícios em otimização.

Tortelly e Occhi (2006) apresentam uma metaheurística híbrida baseada em conceitos de GRASP e busca tabu para resolver o problema de roteirização periódica. A construção da solução inicial é feita de duas maneiras, ambas utilizando GRASP para a designação aleatória de dias de visitas aos pontos. A diferença está na utilização ou não de um filtro para manter apenas as soluções melhores ao invés de apenas construir as soluções sem critério de escolha. A roteirização diária é conseguida pelo método de roteirização em pétalas. Uma busca local utilizando a busca tabu a cada iteração do GRASP é realizada para melhorar a solução inicial, sendo esta solução inicial utilizada

como semente na busca tabu. Testes foram realizados com instâncias retiradas da literatura e geradas aleatoriamente, utilizando também a variante de busca tabu encontrada no estudo de Courdeau *et al.* (1997). A utilização de GRASP com filtro apresenta resultados bastante competitivos comparados à literatura sem aumentar o tempo de processamento, considerando a estratégia sem filtro.

3. CARACTERIZAÇÃO DO PROBLEMA

Em linhas gerais, dado um horizonte de planejamento, o problema de roteirização periódica objetiva a definição da melhor combinação de dias de visitas e de roteiros para cada dia, com o intuito de atender a demanda total dos clientes no período e minimizar o custo total do sistema.

Para cada ponto de atendimento são conhecidas a uma demanda, localização e frequência de visitas. Esta última característica do cliente pode ser feita em intervalos regulares ou não e é descrita como o número de dias que um cliente tem que ser visitado no período, podendo ser escolhida de acordo com a demanda requerida do ponto, pela disponibilidade de estoque no ponto ou no depósito, pela disponibilidade de espaço no depósito, pela quantidade mínima requerida de produtos por dia, pela possibilidade de fabricação da quantidade a ser coletada, entre outros.

Tanto as visitas quanto os roteiros têm que respeitar restrições operacionais para serem efetuadas. Tais restrições dependem de cada problema em particular e podem estar relacionadas a janelas de tempo, jornada de trabalho, diferenciação de produtos de coleta, etc.

3.1. Definição do problema

No caso do estudo, o período de planejamento escolhido é de seis dias por configurar atendimentos de segunda-feira a sábado, o que normalmente acontece em instâncias reais. As frequências de visita seguem um determinado padrão pré-estabelecido, o que facilita o planejamento e coordenação da operação. As visitas ocorrem com intervalos regulares no período. Caso haja, por exemplo, um período de planejamento correspondendo a seis dias (2ª feira a sábado) e um ponto requer duas visitas no período, as seguintes combinações são observadas: (2ª e 4ª), (3ª e 5ª), (4ª e 6ª) e (5ª e sábado).

Não foi levado em consideração o fator de tempo, no caso de janelas de tempo, por exemplo, pois o foco do estudo está na avaliação da qualidade da solução das estratégias propostas levando em consideração apenas restrições operacionais básicas. A frota não é pré-estabelecida, sendo que não configura uma restrição o uso de um certo número pré-fixado de veículos. Considera-se ainda uma frota homogênea de veículos com restrição de capacidade de carga. Sendo o pro-

blema, apenas de coleta.

O custo total, função objetivo da otimização, é formado por duas parcelas: custo fixo e custo variável. O custo fixo é determinado pelo dia que requer o maior número de veículos dentro do período. O custo variável está relacionado à distância total percorrida por todos os veículos dentro do período de planejamento. Logo, o resultado que conseguir obter menor número de veículos e menor distância percorrida total é considerado o melhor resultado do problema.

3.2. Formulação matemática

Baseado no exposto acima, o modelo matemático deve determinar simultaneamente (i) a atribuição dos melhores dias de visitas para cada ponto de atendimento de acordo com as respectivas frequências de visitas e (ii) a definição de roteiros para cada dia, de modo que o custo total no período seja minimizado.

Seja $t = 1, 2, 3, \dots, T$, cada um dos dias de um período de planejamento com duração de T dias e $i = 1, 2, 3, \dots, N$, o conjunto de pontos (ou nós) a serem atendidos até T . A cada ponto i está associada uma quantidade q_i a ser coletada (ou entregue) em cada visita necessária durante o período T . O ponto 0 corresponde ao depósito central, de onde partem e para onde retornam os veículos ao final dos roteiros em cada um dos dias t . Sejam $k = 1, 2, \dots, K_i$, o conjunto de combinações permitidas de dias de visitas que cada ponto i requer em T , como ilustrado no Quadro 1 para um horizonte de planejamento de $T = 6$ dias. Consequentemente, define-se a constante a^{kt} que assume o valor um para os dias t em que deve ocorrer coleta na combinação k , e zero, caso contrário.

A distância entre todos os nós do conjunto, incluindo o depósito, recebe a denominação de d_{ij} , que corresponde à menor distância partindo do ponto i e alcançando o ponto j .

Seja $v = 1, 2, \dots, NV$ o índice de cada um dos veículos disponíveis de uma frota homogênea para ser utilizada ao longo do período T , sendo que NV indica o número de veículos disponíveis, apesar de, no caso estudado, não haver uma frota pré-determinada. A estimativa de NV depende diretamente das combinações selecionadas de k , pois caso em um determinado dia haja um acúmulo de pontos, o NV será maior que em dias de menor demanda. O limitante superior de

Quadro 1. Exemplo de combinações permitidas de dias de visitas para $T = 6$ dias

Frequência de visitas	Combinação	Dias de visitas
1	1	Seg
	2	Ter
	3	Qua
	4	Qui
	5	Sex
	6	Sáb
2	1	Seg - Qua
	2	Ter - Qui
	3	Qua - Sex
	4	Qui - Sáb
3	1	Seg - Qua - Sex
	2	Ter - Qui - Sáb

NV é quando ele tem o mesmo valor de N , ou seja, um veículo para cada cliente i . A capacidade de cada veículo é dada por Q , o custo variável com a distância percorrida é dado por C_v e o custo fixo de cada veículo no período T é dado por C_f .

As variáveis de decisão do problema são as seguintes:

- $u_i^k = 1$, se a combinação k é escolhida para o ponto i ;
0, caso contrário.
- $y_i^t = 1$, se ponto i é visitado no dia t ;
0, caso contrário.
- $x_{ij}^{vt} = 1$, se o veículo v vai do ponto i para o ponto j no dia t ;
0, caso contrário.
- $f_u =$ número de veículos da frota

A formulação proposta a seguir está baseada na formulação original de Christofides e Beasley (1984) com algumas modificações que seguem o que foi exposto anteriormente.

[Minimizar]

$$Z = C_f \cdot f_u + C_v \cdot \sum_{t=1}^T \sum_{v=1}^{NV} \sum_{i=0}^n \sum_{j=0}^n d_{ij} \cdot x_{ij}^{vt} \quad (1)$$

sujeito às seguintes restrições:

$$\sum_{k \in K_i} u_i^k, \quad i = 1, 2, \dots, n \quad (2)$$

$$y_i^t = \sum_{k \in K_i} u_i^k \cdot a^{kt}, \quad t = 1, 2, \dots, T; i = 1, 2, \dots, n \quad (3)$$

$$\sum_{v=1}^{NV} x_{ij}^{vt} \leq \frac{y_i^t + y_j^t}{2}, \quad t = 1, 2, \dots, T; i = 1, 2, \dots, n; j = 1, 2, \dots, n; (i \neq j) \quad (4)$$

$$\sum_{i=0}^n x_{ip}^{vt} = \sum_{j=0}^n x_{pj}^{vt}, \quad \begin{cases} i = 1, 2, \dots, n; j = 1, 2, \dots, n; p = 1, 2, \dots, n \\ t = 1, 2, \dots, T; v = 1, 2, \dots, NV; (i \neq j) \end{cases} \quad (5)$$

$$\sum_{v=1}^{NV} \sum_{i=0}^n x_{ij}^{vt} = \begin{cases} y_j^t, & t = 1, 2, \dots, T, j \neq 0 \\ f_u, & t = 1, 2, \dots, T, j = 0 \end{cases} \quad (6)$$

$$\sum_{i \in H} \sum_{j \in H} x_{ij}^{vt} \leq |H| - 1, \quad t = 1, 2, \dots, T; v = 1, 2, \dots, NV; \forall H \subseteq \{1, 2, \dots, n\} \quad (7)$$

$$\sum_{j=1}^n x_{oj}^{vt}, \quad t = 1, 2, \dots, T; v = 1, 2, \dots, NV \quad (8)$$

$$\sum_{i=1}^n q_i^t \cdot \left(\sum_{j=0}^n x_{ij}^{vt} \right) \leq Q, \quad t = 1, 2, \dots, T; v = 1, 2, \dots, NV \quad (9)$$

$$f_u \geq \sum_{v=1}^{NV} \sum_{j=1}^n x_{oj}^{vt}, \quad t = 1, 2, \dots, T \quad (10)$$

$$y_i^t \in \{0, 1\}, \quad i = 1, 2, \dots, n; t = 1, 2, \dots, T \quad (11)$$

$$u_i^k \in \{0, 1\}, \quad i = 1, 2, \dots, n; k = 1, 2, \dots, K_i \quad (12)$$

$$x_{ij}^{vt} \in \{0, 1\}, \quad i = 1, 2, \dots, n; t = 1, 2, \dots, T; k = 1, 2, \dots, K_i; v = 1, 2, \dots, NV \quad (13)$$

A função objetivo (1) representa a minimização do custo total, dada pela soma das parcelas de custo fixo da utilização dos veículos e custo variável com a distância total percorrida no período. A restrição (2) assegura que apenas uma combinação permitida de dias de visitas seja escolhida para cada ponto i , enquanto que a restrição (3) assegura que, para cada ponto i , as visitas só ocorram nos dias t que lhe foram designados, baseada na combinação escolhida. Já a restrição (4) garante que um veículo só vai de um ponto i para um ponto j em um dia t , se ambos os pontos i e j estiverem alocados para o dia t . A restrição (5) assegura a continuidade de fluxo, ou seja, todo veículo que chega a um ponto i em um dia t sai desse ponto. A restrição (6) garante que cada ponto i seja visitado somente nos dias de visitas selecionados, e que todos os veículos retornam ao depósito central.

A impossibilidade de ocorrência de *subtours* é assegurada pela restrição (7), sendo H qualquer subconjunto de pontos alocados a um veículo, excluindo-se o depósito, que não se repetem e que fazem parte de um mesmo roteiro. O número máximo de arcos que podem existir nesse roteiro não pode ser maior que o número de pontos menos uma unidade, evitando as-

sim, fechar o ciclo entre os pontos.

A restrição (8) impõe que cada veículo v seja utilizado apenas uma única vez por dia, enquanto que a capacidade de cada veículo em cada dia t é assegurada pela restrição (9). A restrição (10) determina a máxima frota utilizada no período. Por fim, as restrições (11), (12) e (13) estão relacionadas à integralidade das variáveis de decisão.

Trata-se de um problema de programação linear inteira-mista extremamente complexo. Assim, torna-se necessário o desenvolvimento de estratégias de solução heurísticas, a fim de permitir a resolução de instâncias de problema encontradas na prática.

4. ESTRATÉGIA DA SOLUÇÃO

Os problemas de roteirização de veículos são conhecidos pela sua complexidade e natureza combinatoria. Sendo assim, é impraticável a utilização de métodos exatos para a resolução desses problemas para instâncias reais em tempos e esforços computacionais razoáveis. Da mesma forma, também não é possível solucionar as extensões do problema de roteirização de veículos utilizando algoritmos exatos sem algum tipo

de relaxação ou simplificação da formulação matemática, como Russel e Igo (1979) demonstraram. Diversos autores observaram que o problema de roteirização de veículos é NP-difícil; como o problema de roteirização periódica reduz-se ao problema de roteirização quando o período T é unitário, pode-se afirmar que o problema de roteirização periódica também é NP-difícil.

Torna-se necessário utilizar métodos heurísticos, que não asseguram a obtenção de soluções ótimas, porém que explorem a região das possíveis soluções, de modo a obter uma boa solução viável, em tempos de processamentos reduzidos. É também importante que esses métodos consigam diversificar a exploração por novas soluções, a fim de não se aprisionarem em um ótimo local.

Uma vez obtido um primeiro resultado com os dias de visitas escolhidos e os roteiros montados, é necessário melhorar a solução atual. Alguns autores sugerem um mecanismo de troca para a melhoria da solução, semelhante aos mecanismos do tipo k -opt, como Psaraftis (1983) descreveu em seus estudos, devidamente adaptados para um horizonte de planejamento. Nesse caso, as trocas envolvem não só pontos entre roteiros de um dia, como num problema de roteirização tradicional, mas também, e principalmente, trocas de dias de visita para os diferentes pontos. Assim, a estrutura de vizinhança pode se tornar bastante complexa para permitir, no caso do problema de roteirização periódica, a proposição de uma estratégia de solução baseada em Busca Tabu, a qual vem sendo aplicada com sucesso em inúmeros problemas de roteirização de veículos.

A primeira estratégia proposta baseia-se em uma heurística simples para selecionar os dias de coleta para cada ponto, buscando um equilíbrio do esforço ao longo do período sem levar em consideração restrições de distância e tempo. Duas medidas foram consideradas para essa finalidade: minimizar a máxima fro-

ta necessária e equilibrar a demanda ao longo dos dias do período de planejamento; as heurísticas correspondentes são denominadas ALOCF e ALOCD, respectivamente.

Essas regras permitem selecionar rapidamente melhores dias de visitas dos pontos, além de serem simples. A idéia principal é evitar a concentração de atendimento dos pontos em certos dias do período, o que ocasionaria um aumento da máxima frota utilizada; assim nos dias em que poucos veículos são necessários para o atendimento, os demais ficariam ociosos, o que pode ser indesejável. Isso é semelhante à idéia utilizada por Chao *et al.* (1995).

Uma vez tendo sido definidos os dias de atendimento para todos os pontos, a roteirização diária é efetuada através do método de economias paralelo (Clarke e Wright, 1964). As melhorias são obtidas através da mudança dos dias de visitas dos pontos da seguinte forma: primeiramente, seleciona-se um subconjunto de pontos para os quais serão definidas novas combinações de dias de visitas, sendo que os pontos que formam esse subconjunto mudam a cada iteração. Uma vez definido este subconjunto de pontos candidatos à mudança, os mesmos são removidos dos dias em que foram alocados para, assim, serem realocados. Para realizar essa realocação, optou-se pelo uso da heurística inicial de designação de dias de visitas, repetida diversas vezes até que se atinja um número máximo de iterações. Procura-se assegurar, dessa forma, que essa heurística de designação seqüencial produza resultados diferentes a partir de sua aplicação a conjuntos de pontos distintos, tendo em vista que alguns pontos permanecem nos mesmo dias, enquanto outros são realocados, sendo que esses dois grupos de pontos mudam a cada iteração. O melhor custo obtido dentre todas iterações é selecionado como a melhor solução da heurística.

A Figura 1 apresenta o pseudocódigo das estratégias heurísticas descritas, ressaltando-se que a diferença

1	Ler dados externos.
2	Ordenação dos pontos através do <i>quicksort</i> .
3	Para todos os pontos, faça.
4	Designação de combinação de visitas por Frota ou Demanda.
5	Fim dos pontos.
6	Montar roteiros para os pontos alocados pelo método de Clarke e Wright paralelo.
7	Enquanto (iterações < número máximo de iterações), faça.
8	Para todos os pontos, faça.
9	Sortear aleatoriamente uma probabilidade.
10	Se (probabilidade do ponto $i \geq$ probabilidade limite), faça.
11	Realocar o ponto i pelo método de Frota ou Demanda.
12	Se (probabilidade do ponto $i <$ probabilidade limite), permaneça.
13	Fim do pontos.
14	Montar roteiros pelo método de Clarke e Wright paralelo.
15	Comparar resultado e guardar o melhor.
16	Fim do enquanto iterações (passo 7).
17	Fim da heurística ALOCF ou ALOCD. Reportar o melhor resultado (frota, distância percorrida, roteiros e custos).

Figura 1: Pseudocódigo das estratégias heurísticas ALOCF e ALOCD

entre ALOCF e ALOCd reside no método de designação de combinação de visitas.

Na segunda estratégia de solução, decidiu-se por uma abordagem inspirada nos conceitos do GRASP (*Greedy Randomised Adaptative Search Procedure*), devido à sua simplicidade e rapidez em gerar resultados de boa qualidade. Na etapa de melhoria, os pontos também são reinseridos nos dias de visitas, como na estratégia de solução anterior, porém, ao invés de reinserir os pontos através das heurísticas de inserção, os pontos são realocados através de um sorteio aleatório que tem seus fundamentos em GRASP. Ao final das iterações de melhoria, o melhor resultado obtido é a solução do método. O pseudocódigo mostrado na Figura 2 detalha as estratégias heurísticas GRASPF e GRASPd.

A terceira e última estratégia baseia-se em algoritmos genéticos, inspirada no trabalho de Galvão e Cunha (2003). Os conceitos básicos do algoritmo genético foram desenvolvidos por J. H. Holland, em 1975, e se baseiam na teoria de evolução darwiniana que descreve mecanismos de seleção natural, hereditariedade e evolução genética dos seres vivos. Na natureza, os seres vivos competem entre si em busca de recursos para sobrevivência e parceiros para reprodução. Os mais aptos e mais adaptados ao meio ambiente são aqueles com maior probabilidade de sobrevivência e maior quantidade de herdeiros.

Cada indivíduo da população corresponde a uma solução em potencial. No entanto, para que um indivíduo configure uma solução, seus cromossomos têm que ter um significado, e uma vez que eles são interpretados, um resultado é obtido. Logo, é necessário que haja uma decodificação do significado dos valores que constam nos cromossomos. A interpretação dessa codificação é obtida através de uma função aptidão, como forma de avaliação dos indivíduos da população, ou seja, quão bem o indivíduo resolve o proble-

ma. O algoritmo genético mais simples envolve três operadores básicos: seleção, cruzamento e mutação.

O conjunto de cromossomos de uma mesma cadeia genética, que conseqüentemente formam um indivíduo, representam, na situação estudada, o conjunto de pontos a serem atendidos no período de planejamento pelos veículos da frota. Ou seja, por exemplo, o cromossomo que está na posição 10 identifica uma característica para o ponto 10. É essa característica que é denominada de código do cromossomo, que dá uma característica ao indivíduo. A codificação de cada indivíduo é utilizada da mesma forma por Vianna *et al.* (1999).

Um aspecto crucial para o desempenho e o sucesso de uma estratégia baseada em algoritmo genético, como dito anteriormente, é a função que avalia a aptidão de cada indivíduo, ou seja, uma medida da qualidade da solução. Essa função tem que ser rápida, em virtude do elevado número de avaliações necessárias para todos os indivíduos da população em cada geração e, ao mesmo tempo, eficiente, em termos de representar adequadamente a solução do problema. No caso deste artigo, a aptidão é o custo total gerado pelo indivíduo, sendo composto por custo fixo (frota) e custo variável (distância).

A avaliação de aptidão dos indivíduos é realizada através da roteirização diária de cada indivíduo, de acordo com as combinações escolhidas para cada ponto que consta nos cromossomos. Foi escolhido novamente o método de economias de Clarke e Wright (1964) paralelo.

Ao término da avaliação de todos os indivíduos da primeira população, inicia-se a fase seguinte do algoritmo genético: a seleção. O método de seleção de indivíduos é baseado na idéia de *ranking*. Uma vez que dois indivíduos irão realizar o cruzamento, um ou mais pontos de cruzamento (ou pontos de *crossover*) têm que ser selecionados para gerar a diversificação

1	Ler dados externos.
2	Ordenação dos pontos através do <i>Quicksort</i> .
3	Para todos os pontos, faça.
4	Designação de combinação de visitas por Frota ou Demanda.
5	Fim dos pontos.
6	Montar roteiros para os pontos alocados pelo método de Clarke e Wright paralelo.
7	Enquanto (iterações 1 < número máximo de iterações 1), faça.
8	Para todos os pontos, faça.
9	Sortear aleatoriamente uma probabilidade.
10	Enquanto (iterações 2 < número máximo de iterações 2), faça.
11	Se (probabilidade do ponto $i \geq$ probabilidade limite), faça.
12	Sortear uma nova combinação para o ponto i , de modo aleatório .
13	Se (probabilidade do ponto $i <$ probabilidade limite), permanece.
14	Montar roteiros pelo método de Clarke e Wright paralelo.
15	Comparar resultados e guardar melhor.
16	Fim enquanto iterações 2 (passo 10).
17	Fim enquanto iterações 1 (passo 7).
18	Fim da heurística GRASPF ou GRASPd. Reportar o melhor resultado (frota, distância percorrida, roteiros e custos).

Figura 2: Pseudocódigo das estratégias heurísticas GRASPF e GRASPd

genética. No caso, os pontos de cruzamento são escolhidos de forma aleatória para cada um dos cruzamentos, criando maior diversificação. O valor escolhido identifica entre quais cromossomos haverá a troca da cadeia genética, caso haja mais de um ponto de cruzamento, dois pontos são escolhidos para o par de indivíduos.

A mutação, outro operador do algoritmo genético, é realizada depois da fase de cruzamento dos indivíduos. A frequência com que a mutação ocorre é baixa (ou seja, ela acontece raramente), porém a sua taxa pode começar baixa e, se for necessário, aumentar até chegar a patamares altos (no início é 1%, podendo chegar a 32%, no caso deste estudo). O aumento na taxa de mutação ocorre quando, depois de certo número de iterações, não há uma melhoria na população de

indivíduos. Isso foi feito com a finalidade de aumentar a diversificação e evitar que a população fique estagnada em um ótimo local.

Foram desenvolvidas, para essa estratégia baseada em algoritmos genéticos, dez variantes que se diferenciam em determinados parâmetros: número de pontos de cruzamento, reinício da renovação da população inicial, manutenção ou não dos melhores indivíduos de populações anteriores, e mudança da taxa de mutação dos indivíduos. As Figuras 3, 4 e 5 apresentam o pseudocódigo das variantes da estratégia AG.

Um sumário das principais características das estratégias heurísticas propostas encontra-se no Quadro 2. Já o Quadro 3 apresenta uma síntese das variantes do algoritmo genético estudadas.

Quadro 2: Sumário das estratégias heurísticas de solução propostas

Descrição		Etapas		
Estratégia de Solução	Heurística	Designação de dias de visitas	Melhoria da solução	Construção de roteiros
1 - ALOC	ALOCf	Frota	Frota	Clarke e Wright paralelo
	ALOCd	Demanda	Demanda	Clarke e Wright paralelo
2 - GRASP	GRASPf	Frota	GRASP	Clarke e Wright paralelo
	GRASPd	Demanda	GRASP	Clarke e Wright paralelo
3 - AG	AG's	Algoritmo Genético	Algoritmo genético: seleção, cruzamento e mutação	Clarke e Wright paralelo

Quadro 3: Sumário das variantes do algoritmo genético

Nome da Variante	Pontos de Cruzamento	Reinício da População	Indivíduos Mantidos	Taxa de Mutação
AG1c	1	Não	-	Constante
AG2c	2	Não	-	Constante
AG1cr0	1	Sim	Nenhum	Constante
AG1cr1	1	Sim	O melhor	Constante
AG1cr10	1	Sim	10% melhores	Constante
AG2cr0	2	Sim	Nenhum	Constante
AG2cr1	2	Sim	O melhor	Constante
AG2cr10	2	Sim	10% melhores	Constante
AG1cmv	1	Não	-	Variável
AG2cmv	2	Não	-	Variável

1	Ler dados externos.
2	Gerar população inicial de modo aleatório.
3	Enquanto (iteração < número máximo de iterações), faça.
4	Enquanto houver indivíduos da população, faça.
5	Para cada dia do período, faça.
6	Montar roteiros pelo método de Clarke e Wright paralelo.
7	Fim dos dias do período.
8	Ranking das aptidões dos indivíduos e atualização do melhor indivíduo.
9	Fim do enquanto indivíduos (passo 4).
10	Enquanto (número de cruzamento < número máximo de cruzamento), faça.
11	Seleção de indivíduos para cruzamento.
12	Fazer cruzamento. Ponto de cruzamento escolhido de modo aleatório – um ou dois pontos .
13	Se houver mutação, fazer mutação.
14	Fim do enquanto cruzamentos (passo 10).
15	Fim do enquanto iterações (passo 3).
16	Fim da heurística AG. Reportar o melhor resultado (frota, distância percorrida, roteiros e custo).

Figura 3: Pseudocódigo das heurísticas AG1c e AG2c

```

1  Ler dados externos.
2  Gerar população inicial de modo aleatório.
3  Enquanto (iteração < número máximo de iterações), faça.
4      Enquanto (gerações sem melhoria < número máximo de gerações sem melhoria), faça.
5          Enquanto houver indivíduos da população, faça.
6              Para cada dia do período, faça.
7                  Montar roteiros pelo método de Clarke e Wright paralelo.
8                  Fim dos dias do período.
9                  Ranking das aptidões dos indivíduos e atualização do melhor indivíduo, se houver.
10             Fim do enquanto indivíduos (passo 5).
11             Enquanto (número de cruzamento < número máximo de cruzamento), faça.
12                 Seleção de indivíduos para cruzamento.
13                 Fazer cruzamento. Ponto de cruzamento escolhido de modo aleatório – um ou dois pontos.
14                 Se houver mutação, fazer mutação.
15             Fim do enquanto cruzamentos (passo 10).
16             Repetir os passos 4 a 15.
17             Se gerações sem melhoria for maior que valor máximo, reinício população (total, manter 10% melhores ou manter melhor indivíduo).
18         Fim enquanto gerações sem melhoria.
19     Fim do enquanto iterações (passo 3).
20 Fim da heurística AG. Reportar o melhor resultado (frota, distância percorrida, roteiros e custo).

```

Figura 4: Pseudocódigo das heurísticas AG1cr0, AG1cr1, AG1cr10, AG2cr0, AG2cr1, AG2cr10

```

1  Ler dados externos.
2  Gerar população inicial de modo aleatório.
3  Enquanto (iteração < número máximo de iterações), faça.
4      Enquanto houver indivíduos da população, faça.
5          Para cada dia do período, faça.
6              Montar roteiros pelo método de Clarke e Wright paralelo.
7              Fim dos dias do período.
8              Ranking das aptidões dos indivíduos e atualização do melhor indivíduo.
9          Fim do enquanto indivíduos (passo 4).
10     Enquanto (número de cruzamento < número máximo de cruzamento), faça.
11         Seleção de indivíduos para cruzamento.
12         Fazer cruzamento. Ponto de cruzamento escolhido de modo aleatório – um ou dois pontos.
13         Enquanto (gerações sem melhoria < número máximo de gerações sem melhoria), faça.
14             Se houver mutação, fazer mutação.
15             Se número de gerações sem melhoria for maior que número máximo, aumentar a taxa de mutação
16         Fim enquanto gerações sem melhoria.
17     Fim do enquanto cruzamentos (passo 10).
18 Fim do enquanto iterações (passo 3).
19 Fim da heurística AG. Reportar o melhor resultado (frota, distância percorrida, roteiros e custo).

```

Figura 5: Pseudocódigo das heurísticas AG1cmv e AG2mv

5. EXPERIMENTOS COMPUTACIONAIS

As heurísticas desenvolvidas foram implementadas em linguagem C, através do compilador Microsoft Visual Studio 2003. Utilizou-se um computador com microprocessador Intel Pentium IV operando a 2,40GHz, com 512Kb de memória RAM.

Os testes foram divididos em duas etapas, sendo que na primeira etapa são utilizados os problemas-teste de Christofides e Beasley (1984) modificados e também problemas-teste gerados aleatoriamente. Houve a modificação dos problemas-teste originais para que estes representem a realidade que se julga ser de uma instância real. A fim de verificar a flexibilidade da aplicação das estratégias de solução, dois outros conjuntos de problemas-teste foram gerados aleatoriamente. O Quadro 4 apresenta sumariza os problemas-teste utilizados nesta primeira etapa.

Em uma segunda etapa de testes, a fim de comparação com os artigos mencionados anteriormente, utilizam-se os problemas-teste de Christofides e Beasley (1984) sem modificações em sua parametrização original, como se pode ver no Quadro 5, onde são indicados o tamanho do problema, dado pelo número de pontos de entrega, o tamanho do período e a frota disponível. O número do problema corresponde à identificação da instância conforme referenciado pelos autores.

Foram avaliadas no total 18 variantes dos algoritmos, sendo 10 variantes do AG (definidas segundo diferentes valores adotados para número de pontos de cruzamento, reinício de população, conforme mostrado no Quadro 3) e 2 variantes para cada uma das quatro heurísticas ALOCf, ALOCd, GRASPF e GRASPD, com diferentes probabilidades limite, de 25% e 75% (a fim de que mostrassem um cenário em que muitos pontos são realocados e outro cenário que houvesse

Quadro 4: Sumário dos problemas-teste utilizados na primeira etapa de testes

Conjunto	Procedência	Número de Pontos	Período de Planejamento	Detalhe da Definição da Frequência de Visita
A	Problemas 2, 5, 8 e 10 de Christofides e Beasley (1984) Modificado	50	6	Demanda ≤ 10 uma visita no período; $11 \leq$ demanda ≤ 25 duas visitas no período; demanda ≥ 26 três visitas no período
		75	6	Demanda ≤ 15 uma visita no período; $16 \leq$ demanda ≤ 27 duas visitas no período; demanda ≥ 28 três visitas no período
		100	6	Demanda ≤ 10 uma visita no período; $11 \leq$ demanda ≤ 25 duas visitas no período; demanda ≥ 26 três visitas no período
B	Aleatório (subgrupos)	50	6	Demanda ≤ 10 uma visita no período; $11 \leq$ demanda ≤ 25 duas visitas no período; demanda ≥ 26 três visitas no período
		75	6	Demanda ≤ 15 uma visita no período; $16 \leq$ demanda ≤ 27 duas visitas no período; demanda ≥ 28 três visitas no período
		100	6	Demanda ≤ 10 uma visita no período; $11 \leq$ demanda ≤ 25 duas visitas no período; demanda ≥ 26 três visitas no período
C	Aleatório	50	6	Demanda ≤ 10 uma visita no período; $11 \leq$ demanda ≤ 25 duas visitas no período; demanda ≥ 26 três visitas no período
		75	6	Demanda ≤ 15 uma visita no período; $16 \leq$ demanda ≤ 27 duas visitas no período; demanda ≥ 28 três visitas no período
		100	6	Demanda ≤ 10 uma visita no período; $11 \leq$ demanda ≤ 25 duas visitas no período; demanda ≥ 26 três visitas no período

Quadro 5: Sumário dos problemas-teste de Christofides e Beasley (1984)

Problemas-Teste			
Número do Problema	Pontos	Período	Frota
2	50	5	3
5	75	5	6
8	100	5	5
10	100	5	4

realocação de poucos pontos), de forma que configurassem cenários diferentes e pudessem prover bons resultados.

Para o caso da estratégia de algoritmos genéticos, tem-se que o tamanho da população e o número de pontos têm o mesmo valor. O número de pontos de cruzamento também é diferente entre estas variantes. Utilizam-se um ou dois pontos de cruzamento. E a taxa de mutação varia depois de um certo número de iterações, quando for aplicável, de 1% a 32%, de acordo com uma progressão geométrica.

O número de iterações e de processamentos, porém, continuam os mesmos nas duas etapas de testes, sendo de cinco mil iterações em cada processamento e cinco processamentos para cada variante da estratégia de solução.

Para a primeira parte dos testes, temos alguns parâmetros como o período de planejamento de seis dias e frequências de visitas permitidas de uma, duas e três visitas ao longo do período, bem como as combina-

ções permitidas de visitas. A capacidade veicular escolhida é de cem. Esta definição de parâmetros, como já explicada anteriormente, visa caracterizar um cenário que se julga representar a realidade brasileira para este tipo de problema. Na segunda etapa de testes, o período de planejamento, frequências de visitas e a capacidade veicular são diferentes para cada problema-teste. Os parâmetros utilizados nas duas etapas de testes estão indicados no Quadro 6.

5.1. Resultados: Primeira etapa

Para cada variante das estratégias de solução foram efetuados cinco processamentos com cinco mil iterações em cada processamento. Apesar de as heurísticas utilizarem a frota como variável principal da função objetivo que se deseja minimizar, os resultados analisados são referentes aos melhores resultados obtidos em termos de frota e em termos de distância separadamente dentre os cinco processamentos. Abaixo, encontra-se um resumo das variantes que obtiveram os

Quadro 6: Parâmetros utilizados

Conjunto de Testes	Problema-Teste	Capacidade Veicular	Período de Planejamento	Frequências de Visitas
Primeira Etapa	Problemas 2, 5, 8 e 10 de Christofides e Beasley (1984) modificados e gerados aleatoriamente	100	6	1,2 ou 3
Segunda Etapa	2	160	5	1,2 ou 5
	5	140	5	1,2 ou 5
	8	200	5	1,2 ou 5
	10	200	5	1,2 ou 3

melhores resultados (menor custo total) para cada problema. Informações mais detalhadas sobre os resultados de cada variante podem ser encontradas em Wu (2007).

Ao se comparar os resultados obtidos pelas estratégias de solução, observa-se que, dependendo do que se define como fator de decisão, o que se denomina melhor resultado muda de acordo com este fator. Caso o tempo de processamento seja crucial para a decisão da escolha da estratégia de solução, a heurística GRASP ou ALOC seriam as escolhidas para utilização. Apesar de, em média, obterem resultados 9% piores que AG, caso o tempo seja um fator limitante e uma resposta tenha que ser obtida rapidamente, o gasto em termos de custo total com distâncias maiores talvez não seja muito maior se comparado ao custo do tempo consumido.

Da mesma maneira, pode-se verificar a situação entre frota e distância. A função objetivo utilizada nas estratégias de solução para avaliar a qualidade dos resultados colocou como prioridade o número de veí-

culos utilizados no período de planejamento. Algumas estratégias de solução conseguiram obter frotas e distâncias menores ao mesmo tempo, no entanto, em geral quando se diminui a frota, a distância total percorrida é maior. Este é um *trade-off* a ser estudado, pois depende da característica de cada problema. Como dito em relação ao tempo de processamento, se a distância é um custo mais importante a ser considerado, talvez a heurística AG seja a melhor, apesar de consumir um grande tempo de processamento. Caso a frota seja mais limitante, pode-se considerar que as heurísticas ALOC e GRASP geralmente obtêm menor número de veículos, como alguns casos de AG.

5.2. Resultados: Segunda etapa

A função objetivo nesta etapa de testes visa a minimização das distâncias, uma vez que a frota já é pré-determinada pelos autores dos artigos mencionados. Foi mantido, no entanto, o dimensionamento de frota das estratégias heurísticas desenvolvidas no presente estudo, como outra maneira de avaliação. Para tanto, foram escolhidas para realizar a comparação somente as variantes que obtiveram menores distâncias nos problemas-teste do conjunto A (Christofides e Beasley (1984) modificado) de acordo com o tamanho do problema. As variantes escolhidas são mostradas no Quadro 8.

Foram efetuados cinco processamentos de cada variante, sendo que cada processamento contém cinco mil iterações. Os resultados obtidos são apresentados na Tabela 1.

Nota-se que apenas a heurística híbrida de Vianna *et al.* (1999) conseguiu superar os resultados de Christofides e Beasley (1984) no problema-teste número 2.

Os resultados obtidos pelas variantes de AG desenvolvidas ficaram bem próximos de Christofides e Beasley (1984), com desvios de no máximo 4%, e obtiveram coincidentemente o mesmo número de veículos de frota que o máximo permitido pelos parâmetros

Quadro 7: Resumo dos melhores resultados para cada problema-teste

50 pontos			
Grupo	Tempo de Processamento	Frota	Distância
A	GRASPF25	Todos iguais	AG2cr1
B	GRASPD75	Todos iguais	AG1cr0
C	GRASP	GRASP e ALOCf75	AG2c
75 pontos			
Grupo	Tempo de Processamento	Frota	Distância
A	GRASPD75	AG2cr10	AG1cr10
B	GRASPD25 e GRASPF75	Diversos	AG1cr0
C	GRASPF25	Diversos	AG2cr1
100 pontos			
Grupo	Tempo de Processamento	Frota	Distância
A	GRASPF75	GRASP e ALOCf75	AG2cr0
B	GRASPD25	Todos iguais	AG2cr10
C	GRASPD75	GRASP e ALOCf75	ALOCf75

Quadro 8: Variantes utilizadas na segunda etapa de testes

Problema	Pontos	Melhor Variante	Pontos de Cruzamento	Reinício da População	Indivíduos Mantidos	Taxa de Mutação
2	50	AG2cr1	2	Sim	O melhor	1%
5	75	AG1cr10	1	Sim	10% melhores	1%
8	100	AG2cr0	2	Sim	Nenhum	1%
10	100	AG2cr0	2	Sim	Nenhum	1%

Tabela 1: Resultados da segunda etapa de testes

Problema-teste	Christofides e Beasley (1984)	Courdeau et al. (1997)	Vianna et al. (1999)	Tortelly e Occhi (2006)			Wu (2007)	
				GRASPF	GRASP	VCGL	Distância	Frota
2	1322,87	1330,09	1291,10	1325,77	1340,04	1331,29	1336,73	3
5	2027,99	2061,36	2089,29	2075,42	2106,58	2085,42	2067,85	6
8	2034,15	2054,90	2112,96	2046,45	2100,50	2068,17	2072,09	5
10	1595,84	1629,96	1660,90	1626,27	1671,56	1647,03	1662,61	4

definidos pelos autores, mesmo não se pré-definindo a frota disponível como os demais trabalhos da literatura. Conclui-se que os resultados obtidos pelas variantes de AG não ficaram muito atrás, podendo ser bastante competitivas, pois apenas a distância foi utilizada como medida de comparação e decisão da melhor solução.

6. CONSIDERAÇÕES FINAIS

Com base no levantamento da bibliografia disponível sobre o assunto, verifica-se que o problema de roteirização periódica é um problema ainda pouco estudado na literatura, apesar da sua importância crescente no contexto logístico das empresas e de diversas aplicações que ocorrem na prática. Atualmente, uma demanda importante por este tipo de “serviço” é pelas indústrias automobilísticas que precisam coletar peças e componentes para montar o veículo. Este sistema se baseia no conceito *milk-run* explicado anteriormente. Adicionalmente, pode-se citar entre as aplicações práticas de roteirização periódica a coleta de lixo, tanto industrial como domiciliar, e a limpeza de ruas, embora estes dois últimos casos configurem um problema de roteirização em arcos e não em nós, derivados do problema do carteiro chinês, e que não foi tratado neste artigo.

Em relação aos resultados dos testes, não houve uma estratégia que se mostrou a melhor em todas as instâncias consideradas, talvez por causa das características de cada problema (localização dos pontos, tamanho da demanda, etc.). Cada uma demonstrou ser de grande uso nesse tipo de problema, por terem qualidades que podem ser cruciais na hora da escolha de um método de solução. Porém, houve destaque para as variantes de AG com reinício de população e A-LOC75. Conclui-se que é necessário fazer uma análise mais cuidadosa do *trade-off* entre qualidade de solução e tempo de processamento para a obtenção de um resultado.

Quando as melhores variantes desenvolvidas são comparadas com resultados obtidos por outros autores da literatura, nota-se que os resultados obtidos por Christofides e Beasley (1984) não foram superados, com exceção ao problema número 2 de 50 pontos. Logo, ainda podem ser considerados como *benchmark*. As estratégias AG utilizadas nessa parte dos testes mostraram ser bastante competitivas também comparadas às de outros autores, obtendo resultados de no máximo 4,18% piores que o melhor conhecido.

Em conclusão a estas observações, pode-se dizer que a utilização de ALOC, GRASP ou AG para o problema de roteirização periódica é mais que comprovadamente viável e adequada.

Mais estudos para que se alcancem melhores resul-

tados de problemas de roteirização periódica são necessários, devido à sua ampla aplicação, sendo isto uma grande valia para a otimização de recursos.

REFERÊNCIAS BIBLIOGRÁFICAS

- Beltrami, E.J. e Bodin, L.D. (1974). *Networks and Vehicle Routing for Municipal Waste Collection*. Networks, vol. 4, p. 65-94.
- Christofides, N. e Beasley, J.E. (1984). *The period routing problem*. Networks, vol. 14, pp. 237-256.
- Chao, I.; Golden, B.L.; Wasil, E.A. (1995) *A new heuristic for the period traveling salesman problem*. Computers and Operations Research, v. 22, n. 5, p. 553-565.
- Clarke, G. e Wright, J.W. (1964). *Scheduling of Vehicles From a Central Depot to a Number of Delivery Points*. Operational Research, v. 12, p. 568-581.
- Coudeau, J.F.; Gendreau, M.; Laporte, G. (1997) *A tabu search heuristic for periodic and multi-depot vehicle routing problems*. Networks, v. 30, p. 105-119.
- Cunha, C.B. (1997). *Uma contribuição para o problema de roteirização de veículos com restrições operacionais*. São Paulo: EPUSP, Departamento de Engenharia de Transportes. 222p. (Tese de Doutorado).
- Cunha, C.B. (2000). *Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais*. Transportes, v. 8, n. 2, pp. 51-74.
- Galvão, F.A. e Cunha, C.B. (2003). *Modelagem Matemática do Problema de Coleta de Resíduos de Biomassa de Madeira para Fins Energéticos*. Transportes. nº 1, XI, pp. 5-11.
- Holland, J.H. (1975) *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control and artificial intelligence*. University of Michigan Press.
- Laporte, G.; Geandreau, M.; Potvin, J.Y. e Semet, F. (2000). *Classical and modern heuristics for the vehicle routing problem*. International Transactions in Operational Research, v. 7, n. 4/5, pp. 285-300.
- Psaraftis, H.N. (1983) *An exact algorithm for the single vehicle many-to-many dial-a-ride problem with time windows*. Transportation Science, Maryland, v. 17, n. 3, p. 351-357
- Russel, R. e Igo, W. (1979). *An assignment routing problem*. Networks, vol. 9, pp. 1-17.
- Tortelly Jr., A.; Ochi, L.S. (2006) *Um GRASP eficiente para problemas de roteamento de veículos*. Tendências em matemática aplicada e computacional, v. 7, 9. 149-158.
- Vianna, D.S.; Ochi, L.S.; Drummond, L.M.A. (1999) *A parallel hybrid evolutionary metaheuristic for the period vehicle routing problem*. Lecture notes in computer science, v. 1586, p. 183-191.
- Wu, L. (2007) *O problema de roteirização periódica de veículos*. São Paulo: EPUSP, Departamento de Engenharia de Transportes. 107 p. (Dissertação de Mestrado).