

# Uma nova abordagem para o problema *dial-a-ride*

Geraldo Regis Mauri UFES

Luiz Antonio Nogueira Lorena INPE

## RESUMO

Este trabalho descreve um modelo matemático geral e multiobjetivo para o problema *dial-a-ride* e uma aplicação do *simulated annealing* para resolvê-lo. O modelo trata a forma estática do problema e abrange vários casos distintos dos modelos mais comuns, tais como frota homogênea e heterogênea, garagens múltiplas ou únicas, e uma função de minimização multiobjetivo que trata os custos de transporte e a inconveniência dos clientes por meio de penalizações. A aplicação do *simulated annealing* é simples, porém, para a geração de novas soluções vizinhas, são utilizados três movimentos de troca selecionados de forma aleatória e uniformemente distribuída, e as rotas são roteirizadas e programadas separadamente por outros métodos heurísticos. Os resultados computacionais são obtidos com o uso de instâncias públicas disponíveis e comparados com outros métodos que apresentam o atual estado da arte em que o problema se encontra.

## PALAVRAS-CHAVE

*Dial-a-ride*, *simulated annealing*, modelo multiobjetivo.

# *A new approach for the dial-a-ride problem*

## ABSTRACT

*This paper describes a general multi-objective mathematical model for the dial-a-ride problem approximately solved by Simulated Annealing. The model deals with a static problem and includes several distinct cases such as heterogeneous or homogeneous fleet of vehicles, multi or single depot and a multi-objective function that treats transportation costs and customer inconveniences by using penalties. The simulated annealing application is straightforward with three types of neighbors' moves that are randomly selected and equally distributed. The routes are clustered and scheduled in a separate way using specific heuristic methods. Computational results are performed over instances of the literature and the results are compared against current state of the art methods.*

## KEY WORDS

*Dial-a-ride*, *simulated annealing*, *multi-objective model*.

## 1. INTRODUÇÃO

Nos países ocidentais, principalmente na Europa e na América do Norte, várias autoridades locais vêm se dedicando à implantação de serviços de coleta e entrega ou à revisão dos sistemas já existentes em resposta à crescente demanda. No que diz respeito ao transporte de pessoas, este fenômeno pode ser atribuído em parte ao envelhecimento da população, mas também a uma tendência para o desenvolvimento de serviços de cuidado com a saúde (CORDEAU; LAPORTE, 2007).

De uma forma geral, o problema *dial-a-ride* compartilha várias características de problemas com coleta e entrega, como por exemplo os serviços de mensagens (correios). Porém, considerando o transporte de pessoas, o nível do critério de serviço (a “qualidade” do serviço) se torna mais importante e complexo. Assim, pontualidade, redução do tempo de espera e duração das rotas são mais críticas nesse caso do que em outros problemas correlatos (CORDEAU; LAPORTE, 2007).

No Brasil, apesar desse tipo de serviço ainda ser pouco comum, nota-se certo crescimento em sua utilização nos últimos anos. Fato este evidenciado pela operação em algumas regiões, como por exemplo nos municípios de Vitória e Vila Velha – ES, São José dos Campos e São Paulo – SP.

**N**o Brasil, apesar desse tipo de serviço ainda ser pouco comum, nota-se certo crescimento em sua utilização nos últimos anos.

No município de São Paulo, tal serviço (apenas para deficientes) opera desde 1996, por iniciativa de uma empresa chamada São Paulo Transportes S/A, por meio de um sistema denominado ATENDE. Embora o sistema utilizado não consiga acompanhar o nível da demanda atual, não se pode negar que tal serviço constitui um significativo avanço na melhoria das condições de mobilidade e integração dos deficientes. Além disso, trata-se de um serviço de caráter social, pois destina-se prioritariamente a famílias carentes, que não têm condições de prover o deslocamento regular dos deficientes, na sua maioria crianças, até os centros especializados de atendimento (ZNAMENSKY; CUNHA, 1999).

Assim, considerando o nível de demanda atual, não só no Brasil, mas como em todo o mundo, bem como a sua esperada tendência de aumento, faz-se imperativo o uso de métodos automatizados de roteirização e programação dos veículos para otimizar esse tipo de serviço.

Esse problema é conhecido por ser *NP-Hard* (BAUGH et al., 1998), e várias metodologias vêm sendo adotadas para resolvê-lo. Como pode ser visto em Cordeau (2006) e Ropke et al. (2007), uma formulação matemática exata para solucionar tal problema é muito complexa, e incapaz de ser resolvida em um tempo aceitável para problemas de “tamanho real”.

O uso de heurísticas e meta-heurísticas para solucionar esse problema vem aumentando significativamente, pois tais métodos, apesar de não garantirem a obtenção de soluções ótimas, permitem a inserção das inúmeras restrições de uma forma mais amena. Dentre esses métodos, meta-heurísticas como *Algoritmos Genéticos*, *Busca Tabu*, *Simulated Annealing*, entre outras, permitem incluir com facilidade vários tipos de condições que tratam simultaneamente a satisfação dos clientes e os custos operacionais (BAUGH et al., 1998; BERGVINSDOTTIR, 2004; ZNAMENSKY; CUNHA, 1999). Além disso, existem também inúmeras heurísticas específicas para esse problema (CALVO; COLORNI, 2007; MADSEN et al., 1995; TOTH; VIGO, 1996, 1997; XIANG et al., 2006).

Este trabalho apresenta uma alternativa simples e eficaz para resolver o problema em questão. É proposto um modelo matemático geral e multiobjetivo para representar o problema, e a meta-heurística *Simulated Annealing* é utilizada juntamente com outras heurísticas para tratá-lo, ou seja, gerar roteiros que sejam econômicos do ponto de vista operacional e que satisfaçam à demanda dos clientes, mantendo um “bom” nível de qualidade no serviço prestado.

O artigo está organizado como segue. A Seção 2 apresenta brevemente o estado da arte relacionado ao problema em questão. O modelo proposto é detalhado e analisado na Seção 3, enquanto a Seção 4 descreve os métodos utilizados para resolvê-lo. Os resultados computacionais são apresentados na Seção 5, e as conclusões são resumidas na Seção 6.

## 2. ESTADO DA ARTE

Dentre a classificação dos problemas de roteirização e programação de veículos, aqueles que envolvem o transporte de passageiros de seus locais de origem a seus destinos são conhecidos genericamente na literatura como problemas do tipo “*dial-a-ride*” (ZNAMENSKY; CUNHA, 1999).

O *Dial-a-Ride* – DARP (CORDEAU; LAPORTE, 2007) consiste no desenvolvimento de rotas e escalas de veículos para transportar diversos usuários, os quais especificam requisições de embarque e desembarque (coleta e entrega) entre locais de origem e destino específicos. O objetivo

desse processo é planejar um conjunto de rotas para alguns veículos, com “custo mínimo”, capaz de acomodar o maior número possível de usuários, e sempre obedecendo a um conjunto de restrições.

Em se tratando especificamente do caso de transporte de deficientes, um ponto de embarque (coleta) corresponde ao endereço onde um determinado deficiente deve ser buscado, e o ponto de desembarque (entrega) associado corresponde ao endereço onde o mesmo deficiente deve ser entregue. Cada ponto de embarque e seu respectivo ponto de desembarque, juntamente com suas respectivas “janelas de tempo”, formam uma *Requisição de Transporte* de um cliente.

Uma tendência comum nos modelos do DARP é deixar que os usuários determinem uma “janela de tempo” (isto é, janelas ou intervalos de horários para seu atendimento) para sua partida e sua chegada, pois os usuários devem estar aptos a especificar um intervalo de horários para seu embarque e desembarque, ambos em locais específicos, possibilitando assim um melhor planejamento da programação e roteirização dos veículos como um todo.

Cada veículo possui uma capacidade, medida normalmente em número de assentos convencionais e número de cadeiras de rodas, por exemplo. Analogamente, a cada solicitação de atendimento está associada uma ocupação em termos de quantidade de assentos convencionais (eventuais acompanhantes) e em quantidade de cadeiras de rodas (ZNAMENSKY; CUNHA, 1999).

Segundo Cordeau e Laporte (2007), e outros diversos autores, o DARP pode ser abordado de forma estática ou dinâmica. No primeiro caso, todas as requisições de transporte são conhecidas *a priori*, enquanto no segundo caso são consideradas requisições feitas ao longo do dia (normalmente por telefone), e as rotas dos veículos são ajustadas em tempo real de acordo com a demanda. Porém, na prática, raramente existem DARPs dinâmicos “puros”, pois normalmente um subconjunto de requisições é conhecido com antecedência.

A maioria dos estudos acerca do DARP assume a disponibilidade de um conjunto de veículos homogêneos instalados em um único depósito, entretanto, é importante perceber que na prática existem situações diferentes, como por exemplo: pode haver vários depósitos, especialmente em grandes áreas geográficas, e o conjunto de veículos às vezes é heterogêneo, sendo alguns veículos projetados para só transportar cadeiras de rodas, outros só para transportar passageiros para ambulatórios e ainda alguns capazes de acomodar ambos os tipos de passageiro (CORDEAU; LAPORTE, 2007).

O DARP pode ser considerado com vários objetivos distintos, como por exemplo: minimizar os custos operacionais sujeito à satisfação de toda a demanda; maximizar a satisfação da demanda, sujeito à disponibilidade dos veículos,

ou outras combinações destes, que geralmente buscam um equilíbrio entre custo operacional e qualidade do serviço prestado.

Em termos gerais, o DARP representa um caso particular de vários problemas de roteamento de veículos como o *Pick-up and Delivery Vehicle Routing Problem (PDVRP)* e o *Vehicle Routing Problem with Time Windows (VRPTW)*. O que faz o DARP diferente da maioria de tais problemas é a perspectiva humana, ou seja, no transporte de passageiros, a redução da “inconveniência” dos usuários (qualidade do serviço) deve ser equilibrada com a minimização dos custos operacionais (CORDEAU; LAPORTE, 2007).

O DARP é um problema muito comum, porém relativamente pouco difundido dentro da Pesquisa Operacional e da Otimização Combinatória devido a sua alta complexidade. Entretanto, alguns pesquisadores o vêm estudando, e conseqüentemente apresentando métodos variados de solução e trabalhos realizados segundo seus preceitos e aplicações.

Cordeau e Laporte (2003) apresentam uma abordagem que utiliza a meta-heurística *Busca Tabu* para resolver o DARP. Nessa abordagem o problema é tratado em sua forma estática, com múltiplos veículos, frota homogênea e garagem única. Além disso, são comparados três métodos heurísticos, *P1*, *P2* e *P3*, que juntamente com a Busca Tabu são utilizados para resolver o problema. O primeiro (*P1*) visa apenas à minimização das violações nas janelas de tempo, enquanto o segundo (*P2*), além de tentar minimizar essas violações, visa também à minimização da duração das rotas. Já o terceiro (*P3*) tem esses mesmos objetivos, porém visando também a redução nos tempos de viagem dos clientes. Os resultados obtidos mostram que *P1* e *P2*, por serem mais “simples”, são mais rápidos que *P3*, porém as soluções obtidas por *P3* são expressivamente melhores. Todos os resultados são obtidos a partir de instâncias geradas aleatoriamente baseadas em informações cedidas pela *Montreal Transit Commission (MTC)* de Montreal – Canadá. Essas instâncias variam de 24 a 144 clientes. Além disso, eles apresentam outros resultados obtidos para instâncias de problemas reais (com 200 e 295 clientes) cedidas por uma transportadora dinamarquesa.

Jorgensen et al. (2007) apresentam uma abordagem que utiliza um *Algoritmo Genético* para resolver o DARP. Nessa abordagem o problema é tratado na sua forma estática, com múltiplos veículos, frota heterogênea e garagens múltiplas. A resolução do problema é obtida por uma abordagem clássica de “agrupar primeiro” e “rotear depois”. O Algoritmo Genético é utilizado para agrupar (distribuir) os clientes nos veículos, ou seja, determinar quais serão os clientes atendidos por cada veículo. Já o roteamento (sequência de atendimento) e a programação (determinação dos horários) são determinados independentemente para cada veículo por

meio de uma heurística específica. Os resultados são obtidos a partir das instâncias geradas aleatoriamente por Cordeau e Laporte (2003). Essa mesma abordagem é apresentada com mais detalhes em Bergvinsdottir (2004).

Cordeau (2006) apresenta uma abordagem que utiliza o algoritmo *Branch-and-Cut* para resolver o problema. Nessa abordagem o problema é tratado na sua forma estática, com múltiplos veículos, frota homogênea e garagem única. A resolução do problema é dada por meio de uma abordagem exata, garantindo assim a obtenção da solução ótima para o problema. Os resultados são obtidos a partir de instâncias geradas aleatoriamente com no máximo 32 clientes.

### 3. MODELAGEM PROPOSTA

O modelo proposto neste trabalho trata o DARP de forma estática (todas as requisições são conhecidas *a priori*), com múltiplos veículos, com frota heterogênea (cada veículo possui uma capacidade distinta) e garagens múltiplas (cada veículo inicia e termina sua rota em garagens específicas). Este modelo é novo, porém adaptado dos mesmos apresentados por Bergvinsdottir (2004) e Cordeau (2006), e é capaz de representar situações reais do problema.

## Este trabalho apresenta uma alternativa simples e eficaz para resolver o problema em questão.

Nesse modelo, assume-se inicialmente a existência de  $n$  clientes (requisições de transporte) a serem atendidos por  $m$  veículos. Cada requisição de transporte especifica um local de embarque  $i$  e um de desembarque  $n+i$ . Para representar o problema, então, são definidos os seguintes conjuntos:

- $K$ : conjunto dos veículos disponíveis ( $|K| = m$ ).
- $G^-$ : conjunto de garagens de origem.
- $G^+$ : conjunto de garagens de destino.
- $P$ : conjunto dos locais de embarque.
- $U$ : conjunto dos locais de desembarque.
- $N = G^- \cup P \cup U \cup G^+$ : conjunto com todos os locais (pontos).

Cada cliente  $i$  ( $\forall i \in P$ ) especifica a “carga”  $q_i$  necessária para seu transporte, ou seja, o número de assentos no veículo que ele irá ocupar, e também dois intervalos de horários, um em que ele gostaria de embarcar na sua origem  $[e_i, l_i]$ , e outro em que ele gostaria de desembarcar no seu destino  $[e_{n+i}, l_{n+i}]$  (janelas de tempo). A carga  $q_i$  deverá ser um valor positivo nos locais de embarque e o mesmo valor, porém negativo, nos respectivos locais de desembarque.

Cada veículo  $k$  ( $\forall k \in K$ ) tem uma capacidade conhecida  $Q_k$ , que é definida como sendo a quantidade de assentos disponíveis, e cada veículo é específico para determinados tipos de atendimento, como por exemplo: só cadeiras de rodas, só macas, etc. Além disso, cada veículo ainda possui um tempo máximo de duração associado a sua rota  $T_k$  e inicia sua rota em uma garagem específica  $g_k^-$  e termina em uma outra  $g_k^+$ , podendo ser a mesma ou não. Todas as garagens ainda têm suas próprias janelas de tempo para partida (garagem de origem) e chegada (garagem de destino).

A cada cliente  $i$  ( $\forall i \in P$ ) é associado um tempo máximo de viagem  $R_i$ , ou seja, o tempo máximo em que o cliente poderá permanecer dentro do veículo. A cada local  $i$  ( $\forall i \in \{P \cup U\}$ ) é associado um tempo máximo de espera  $W_i$ , ou seja, o tempo máximo em que os veículos poderão ficar esperando até iniciar o “serviço”. Considera-se como serviço o embarque ou desembarque de um cliente em um determinado local, sendo atribuído um tempo  $s_i$  necessário para sua completude. Por fim, dados os locais referentes às garagens e aos pontos pertencentes às requisições, têm-se as distâncias  $d_{ij}$  e as durações das viagens  $t_{ij}$  entre os pontos  $i$  e  $j$ ,  $\forall i, j \in N$  e  $i \neq j$ .

O objetivo, então, é minimizar os custos operacionais e a “insatisfação” dos clientes, ou seja, minimizar os requisitos “não-essenciais” do problema. Estes requisitos estão relacionados à distância total percorrida pelos veículos, ao número de veículos utilizados na solução do problema, à duração das rotas, ao tempo de viagem dos clientes e ao tempo de espera nos locais de embarque e desembarque.

Existe ainda um conjunto de requisitos essenciais que devem ser obedecidos, ou seja, requisitos que obrigatoriamente devem ser atendidos para que se obtenha uma solução válida (viável em termos práticos) para o problema. Dentre esses requisitos estão:

- A duração da rota executada pelo veículo  $k$  ( $\forall k \in K$ ) não deve exceder o tempo máximo permitido  $T_k$ .
- O tempo de viagem do cliente  $i$  ( $\forall i \in P$ ) não deve exceder o tempo máximo de viagem permitido  $R_i$ ;
- O tempo de espera no local  $i$  ( $\forall i \in \{P \cup U\}$ ) não deve exceder o tempo máximo de espera permitido  $W_i$ ;
- A capacidade  $Q_k$  ( $\forall k \in K$ ) dos veículos não pode ser excedida em nenhum local;
- O início do serviço em todos os locais  $i$  ( $\forall i \in N$ ) deve estar dentro dos intervalos preestabelecidos  $[e_i, l_i]$ .

De posse dessas informações, podem-se definir as variáveis de decisão que irão fornecer toda a programação dos veículos:

- $A_i$ : é o horário de chegada no local  $i$  ( $\forall i \in N$ ) pelo veículo que irá atender tal local, sendo  $A_i = 0$  se  $i \in G^-$  e  $A_i = D_{i-1} + t_{i-1,i}$  se  $i \in \{P \cup U \cup G^+\}$ ;
- $D_i$ : é o horário de partida do local  $i$  ( $\forall i \in N$ ), novamente pelo veículo que irá atendê-lo, sendo  $D_i = 0$  se  $i \in G^+$  e  $D_i = B_i + s_i$  se  $i \in \{P \cup U\}$  e  $D_i = B_i$  se  $i \in \{G^-\}$ ;
- $B_i$ : é o horário de início do serviço no local  $i$  ( $\forall i \in N$ ), sendo  $B_i = D_i$  se  $i \in G^-$  e  $B_i = \max\{e_i, A_i\}$  se  $i \in \{P \cup U \cup G^+\}$ ;
- $W_i$ : é o tempo de espera antes do início do serviço no local  $i$  ( $\forall i \in N$ ), sendo  $W_i = 0$  se  $i \in G^-$  e  $W_i = B_i - A_i$  se  $i \in \{P \cup U \cup G^+\}$ ;
- $Q_i$ : é a carga (número de assentos ocupados) do veículo que atende o local  $i$  ( $\forall i \in N$ ), após o término do serviço, sendo  $Q_i = 0$  se  $i \in \{G^- \cup G^+\}$  e  $Q_i = Q_{i-1} + q_i$  se  $i \in \{P \cup U\}$ ;
- $R_i$ : é o tempo de viagem do cliente  $i$  ( $\forall i \in P$ ), sendo  $R_i = B_{n+i} - D_i$ .

A partir de então, considerando  $x_{ij}^k = 1$  caso o veículo  $k$  vá do local  $i$  para o local  $j$  e  $x_{ij}^k = 0$  caso contrário, tem-se o seguinte modelo matemático:

**Minimizar:**

$$\omega_0 \sum_{k \in K} \sum_{i \in N} \sum_{j \in N; j \neq i} (d_{i,j} x_{i,j}^k) + \omega_1 \sum_{k \in K} \sum_{j \in P} x_{g_k^-, j}^k + \omega_2 \sum_{k \in K} (B_{g_k^+} - D_{g_k^-}) + \omega_3 \sum_{i \in P} R_i + \omega_4 \sum_{i \in \{P \cup U\}} W_i + \quad (1)$$

$$\beta_0 \sum_{k \in K} \max\{0, (B_{g_k^+} - D_{g_k^-}) - T''_k\} + \beta_1 \sum_{i \in P} \max\{0, R_i - R''_i\} + \beta_2 \sum_{i \in \{P \cup U\}} \max\{0, W_i - W''_i\} + \quad (2)$$

$$\beta_3 \sum_{k \in K} \max\left\{0, \left(Q_i \sum_{i \in \{P \cup U\}} \sum_{j \in \{P \cup U\}; j \neq i; j \neq n-i} x_{i,j}^k\right) - \tilde{Q}_k\right\} + \beta_4 \sum_{i \in N} (\max\{0, e_i - B_i\} + \max\{0, B_i - l_i\}) \quad (3)$$

**Sujeito a:**

$$\sum_{j \in \{P \cup \{g_k^+\}\}} x_{g_k^-, j}^k = 1 \quad \forall k \in K \quad (4)$$

$$\sum_{i \in \{U \cup \{g_k^+\}\}} x_{i, g_k^+}^k = 1 \quad \forall k \in K \quad (5)$$

$$\sum_{k \in K} \sum_{j \in \{P \cup U\}; j \neq i} x_{i,j}^k = 1 \quad \forall i \in P \quad (6)$$

$$\sum_{j \in \{P \cup U\}; j \neq i} x_{i,j}^k - \sum_{j \in \{P \cup U \cup \{g_k^+\}\}; j \neq i; j \neq n+i} x_{n+i,j}^k = 0 \quad \forall k \in K; i \in P \quad (7)$$

$$\sum_{j \in \{P \cup U \cup \{g_k^+\}\}; j \neq i; j \neq n+i} x_{j,i}^k - \sum_{j \in \{P \cup U\}; j \neq i} x_{i,j}^k = 0 \quad \forall k \in K; i \in P \quad (8)$$

$$\sum_{j \in \{P \cup U\}; j \neq i} x_{j,i}^k - \sum_{j \in \{P \cup U \cup \{g_k^+\}\}; j \neq i; j \neq n-i} x_{i,j}^k = 0 \quad \forall k \in K; i \in U \quad (9)$$

$$B_j = (B_i + s_i + t_{i,j} + W_j) \sum_{k \in K} x_{i,j}^k \quad \forall i, j \in N; i \neq j \quad (10)$$

$$Q_j = (Q_i + q_j) \sum_{k \in K} x_{i,j}^k \quad \forall i, j \in N; i \neq j \quad (11)$$

$$A_i = B_i - W_i \quad \forall i \in \{P \cup U \cup G^+\} \quad (12)$$

$$D_i = B_i + s_i \quad \forall i \in \{P \cup U \cup G^-\} \quad (13)$$

$$R_i = B_{n+i} - D_i \quad \forall i \in P \quad (14)$$

$$A_{g_k^-} = D_{g_k^+} = Q_{g_k^-} = Q_{g_k^+} = W_{g_k^-} = 0 \quad \forall k \in K \quad (15)$$

$$A_i, W_i, B_i, D_i, Q_i \text{ irrestritas} \quad \forall i \in N \quad (16)$$

$$R_i \text{ irrestritas} \quad \forall i \in P \quad (17)$$

$$x_{i,j}^k \in \{0,1\} \quad \forall k \in K; \forall i, j \in N; i \neq j \quad (18)$$

A função objetivo é dividida em duas partes (termo (1) e termos (2) e (3)). A primeira (1) visa minimizar os requisitos não-essenciais do problema. Os termos dessa equação representam, respectivamente, a distância total percorrida pelos veículos, o número de veículos utilizados para atender a todos os clientes, o tempo total de duração das rotas, o tempo total de viagem dos clientes e o tempo total de espera dos veículos. Já a segunda parte, (2) e (3), visa à minimização das violações nos requisitos essenciais. Nessas equações são representados, respectivamente, o tempo que excede o tempo máximo de duração das rotas, o tempo total que excede os tempos máximos de viagem permitidos para os clientes, o tempo total que excede os tempos máximos de espera permitidos para cada local nas rotas, o excesso na capacidade dos veículos e o total dos tempos que violam as janelas de tempo.

Além disso, os requisitos não-essenciais são “penalizados” por um vetor de números inteiros positivos (pesos)  $\omega = [\omega_0, \omega_1, \omega_2, \omega_3, \omega_4]$ , e os essenciais com um vetor semelhante  $\beta = [\beta_0, \beta_1, \beta_2, \beta_3, \beta_4]$ .

As restrições (4) e (5) garantem, respectivamente, que cada veículo sairá de sua garagem de origem e chegará à

sua garagem de destino uma única vez, ou seja, cada rota será iniciada em sua garagem de origem e terminará em sua garagem de destino. A restrição (6) garante que cada cliente será atendido uma única vez, ou seja, por apenas um veículo. A restrição (7) garante que um local de embarque estará sempre na mesma rota que seu respectivo local de desembarque, enquanto que a contenção do fluxo, ou seja, tudo que entra é igual a tudo que sai, é garantida pelas restrições (8) e (9). A restrição (10) determina o horário de início do serviço e o tempo de espera em cada local e também o veículo que o atenderá. Como mostrado em Cordeau (2006), essas

**A** flexibilidade na escolha do objetivo final é dada pela utilização de uma função multi-objetivo, que utiliza pesos para determinar a “importância” de cada requisito na solução do problema.

restrições, em conjunto, garantem a eliminação de sub-rotas (*sub-tours*) no caso do DARP. A restrição (11) determina a carga dos veículos em cada local, enquanto as restrições (12), (13) e (14) garantem, respectivamente, um cálculo correto dos horários de chegada e partida nos locais e dos tempos de viagem dos clientes. A restrição (15) “inicializa” algumas variáveis referentes às garagens. Por fim, a restrição (18) apenas garante que as variáveis de decisão  $x$  são binárias, enquanto as restrições (16) e (17) garantem que as demais variáveis sejam irrestritas.

### 3.1 Análise do modelo

Esse modelo pode ser facilmente adaptado para representar matematicamente diversos outros problemas, como por exemplo modelos com garagem única, com apenas um veículo, com frota homogênea, e o mais interessante, com vários objetivos distintos. A flexibilidade na escolha do objetivo final é dada pela utilização de uma função multiobjetivo, que utiliza pesos para determinar a “importância” de cada requisito na solução do problema.

É interessante destacar também que esse modelo é baseado em vários outros encontrados na literatura (BERGVINSDOTTIR, 2004; CORDEAU, 2006) e, como em alguns desses, ele representa o problema de forma relaxada, tratando os requisitos essenciais como parte da função a ser minimizada, e não como restrições do modelo. Essa é uma abordagem interessante que torna a resolução do problema menos árdua, porém permite que soluções inválidas sejam obtidas. Entretanto, soluções inválidas podem ser interessantes em alguns casos, pois como é sabido (CORDEAU,

2006; ROPKE et al., 2007) os modelos exatos são incapazes de resolver de forma satisfatória um problema de tamanho “real”, e em alguns casos uma solução com violações mínimas nos requisitos essenciais pode ser aceitável.

Como sugerido em Cordeau (2006), nesse modelo é feita uma redução no número de arcos (caminhos) existentes entre os locais (como pode ser visto na formulação matemática). Tal fato se deve às características do DARP, que, por exemplo, não permite um arco ligando um local a si próprio, não permite que um desembarque seja realizado antes de seu respectivo embarque, etc. Além disso, ao contrário do modelo proposto por Bergvinsdottir (2004), que utiliza as variáveis referentes aos locais de forma exclusiva para cada veículo ( $A_i^k, B_i^k, D_i^k, W_i^k, Q_i^k, R_i^k$ ), nesse modelo são utilizadas variáveis únicas ( $A_i, B_i, D_i, W_i, Q_i, R_i$ ), o que reduz significativamente o número de variáveis de decisão do modelo.

Por fim, deve-se ressaltar que o *software* CPLEX 9.1 (ILOG, 2003), que é uma referência para resolver problemas de otimização, não foi capaz de resolver, em 60 minutos, esse modelo na sua forma exata (tratando os requisitos essenciais como restrições) para as instâncias utilizadas neste trabalho.

## 4. SIMULATED ANNEALING APLICADO AO DARP

O *Simulated Annealing* – SA é um método de busca local que aceita movimentos de piora para escapar de ótimos locais. Ele foi proposto originalmente por Kirkpatrick et al. (1983) e se fundamenta em uma analogia com a termodinâmica ao simular o resfriamento de um conjunto de átomos aquecidos. Esta técnica começa sua busca a partir de uma solução inicial qualquer.

O procedimento principal consiste em um laço que gera aleatoriamente, em cada iteração, um único vizinho  $S'$  da solução corrente  $S$ . A cada geração de um vizinho  $S'$  de  $S$  é testada a variação do valor da função objetivo (custo), isto é,  $\Delta = f(S') - f(S)$ .

Para um problema de minimização, se  $\Delta < 0$ , o método aceita a solução e  $S'$  passa a ser a nova solução corrente. Caso  $\Delta \geq 0$  a solução vizinha candidata também poderá ser aceita, mas neste caso com uma probabilidade  $e^{-\Delta/T}$ , onde  $T$  é um parâmetro do método, chamado de temperatura, que regula a probabilidade de aceitação de soluções de pior custo.

A temperatura  $T$  assume, inicialmente, um valor elevado  $T_0$ . Após um número fixo de iterações:  $SA_{max}$  (o qual representa o número de iterações necessárias para o sistema

atingir o equilíbrio térmico em uma dada temperatura), a temperatura é gradativamente diminuída por uma razão de resfriamento  $\alpha$  ( $0 < \alpha < 1$ ). Com esse procedimento, no início, dá-se uma chance maior para escapar de mínimos locais e, à medida que  $T$  se aproxima de zero, o algoritmo comporta-se como o método de descida, uma vez que diminui a probabilidade de se aceitar movimentos de piora ( $T \rightarrow 0 \Rightarrow e^{-\Delta T} \rightarrow 0$ ).

O procedimento pára quando a temperatura chega a um valor próximo de zero (temperatura de congelamento:  $T_c$ ) e nenhuma solução que piore o valor da melhor solução é mais aceita, isto é, quando o sistema está estável. A solução obtida quando o sistema encontra-se nesta situação evidencia o encontro de um mínimo local, que em alguns casos também pode representar um mínimo global. Os parâmetros de controle do procedimento são a razão de resfriamento  $\alpha$ , o número de iterações para cada temperatura  $S_{max}$ , a temperatura inicial  $T_0$  e a temperatura de congelamento  $T_c$ .

Para a utilização do SA, deve-se definir *a priori*, um método para geração de uma solução inicial  $S$ , um método para geração das soluções vizinhas  $S'$  (estrutura de vizinhança), e uma função objetivo  $f(S)$  a ser otimizada.

#### 4.1 Solução inicial

Neste trabalho a solução inicial é gerada por uma *heurística de distribuição* (Figura 1), que é responsável pela roteirização dos veículos, ou seja, pela formação dos agrupamentos dos locais nas rotas e da sequência de atendimento destes.

1. **CRIAR** ( $m$  rotas vazias, e atribuí-las aos  $m$  veículos);
2. **CRIAR** (uma lista  $L$  com todas as requisições de transporte dos clientes);
3. **PARA** (cada rota  $k$ ,  $k = 1, \dots, m$ ) **FAÇA**
4.     **SELECIONAR** ( $\lfloor n/m \rfloor$  requisições de  $L$ );
5.     **PARA** (cada requisição selecionada) **FAÇA**
6.         Pos1  $\leftarrow$  posição qualquer da rota  $k$ ;
7.         Pos2  $\leftarrow$  posição qualquer da rota  $k$ , porém posterior a Pos1;
8.         **INSERIR** (o ponto de embarque em Pos1);
9.         **INSERIR** (o ponto de desembarque em Pos2);
10.     **FIM-PARA**;
11.     **INSERIR** (a garagem de origem do veículo  $k$  no início da rota  $k$ );
12.     **INSERIR** (a garagem de destino do veículo  $k$  no fim da rota  $k$ );
13.     **REMOVER** (as requisições selecionadas de  $L$ );
14. **FIM-PARA**;

**Figura 1: Heurística de distribuição.**

Nessa heurística são criadas inicialmente  $m$  rotas vazias, sendo atribuído, a cada uma delas, um veículo específico. Posteriormente, todas as requisições de transporte dos

clientes (pontos de embarque e seus respectivos pontos de desembarque) são distribuídas aleatoriamente, porém de forma uniforme, a essas rotas, ou seja, as  $n$  requisições são divididas igualmente entre os  $m$  veículos. Obviamente, em alguns casos a divisão de  $n$  por  $m$  não será inteira, e nesses casos a última rota deverá atender a um número superior de requisições. A seleção das requisições e das rotas que as atenderão também é aleatória.

As posições de inserção dos pontos (locais) na rota também são selecionadas de forma aleatória, mas sempre atendendo a restrição de precedência oriunda da requisição de transporte do cliente em questão, ou seja, o local de embarque sempre deverá ser anterior ao local de desembarque. Por fim, devido ao fato de toda rota começar e terminar em garagens específicas (referentes ao veículo que irá executá-la) pode-se considerar que os locais referentes a estas garagens sempre serão alocados como primeiro e último pontos em cada uma das rotas criadas.

Como descrito anteriormente, essa heurística trata apenas a roteirização dos veículos, entretanto, ainda se deve fazer a programação destes, ou seja, determinar os horários de chegada nos locais, os horários de partida, etc. Para isso, é utilizada uma outra heurística, denominada *heurística de programação* (Figura 2), que é adaptada da apresentada em Cordeau e Laporte (2003), e realiza a programação de uma forma a tentar reduzir as violações nas janelas de tempo, a duração das rotas e os tempos de viagem dos clientes.

1.  $B_0 \leftarrow e_0$ ;  $D_0 \leftarrow B_0$ ;
2. **CALCULAR** ( $A_i, B_i, W_i, D_i, Q_i$  para cada ponto  $v_i \in V_k$  e  $v_i \neq v_0$ );
3. **CALCULAR** ( $F_0$ );
4.  $B_0 \leftarrow e_0 + \min \left\{ F_0, \sum_{0 < p \leq z} W_p \right\}$ ;  $D_0 \leftarrow B_0$ ;
5. **ATUALIZAR** ( $A_i, B_i, W_i, D_i$  para cada ponto  $v_i \in V_k$  e  $v_i \neq v_0$ );
6. **CALCULAR** ( $R_i$  para cada ponto  $v_i \in V_k$  e  $v_i \in P$ );
7. **PARA** (cada ponto  $v_i \in V_k$  e  $v_i \in P$ ) **FAÇA**
8.     **CALCULAR** ( $F_i$ );
9.      $B_i \leftarrow B_i + \min \left\{ F_i, \sum_{i < p \leq z} W_p \right\}$ ;
10.      $D_i \leftarrow B_i + s_i$ ;  $W_i \leftarrow B_i - A_i$ ;
11.     **ATUALIZAR** ( $A_j, B_j, W_j, D_j$  para cada ponto  $v_j \in V_k$  e  $v_j$  posterior a  $v_i$ );
12.     **ATUALIZAR** ( $R_j$  para cada ponto  $v_j \in V_k$ ,  $v_j \in P$  e  $v_{n+j}$  posterior a  $v_i$ );
13. **FIM-PARA**;

**Figura 2: Heurística de programação.**

Nessa heurística, é utilizado um conceito de “atraso”, proposto inicialmente por Savelsbergh (1992), que consiste

basicamente em atrasar, na medida do possível, o horário de partida da garagem de origem e o início do serviço nos locais de embarque.

Inicialmente, o horário de partida da garagem de origem é marcado para o horário de início da respectiva janela de tempo. A partir de então, os demais cálculos (horários de chegada, início do serviço, tempo de espera, horário de par-

(que atende o ponto  $i$ ) poderá “aguardar” antes de iniciar o serviço no ponto  $i$ . Na equação (19), o somatório de  $W_p$  indica o tempo total de espera nos locais atendidos pela rota em questão, já o termo seguinte indica a “menor” violação entre a janela de tempo e o tempo de viagem dos clientes. Essa equação é definida apenas para os pontos de embarque. Na equação (20), não é considerada a violação

no tempo de viagem, pois tal equação é definida apenas para as garagens de origem, e a equação (21) é definida apenas para os pontos de desembarque e para as garagens de destino, onde não são considerados atrasos.

Como apresentado por Savelsbergh (1992), esse “atraso” busca adiar, na medida

do possível, o horário de partida da garagem de origem e o início do serviço nos locais de embarque, sem que haja violações nas janelas de tempo, na duração das rotas e nos tempos de viagem dos clientes.

## Como estrutura de vizinhança foram utilizados três diferentes movimentos de troca: *Re-ordenar rota, Re-alocar ponto e Trocar pontos.*

tida e carga do veículo) são realizados para todos os pontos seguintes na rota. Posteriormente, é calculado o atraso para a partida da garagem de origem, e então o horário de partida é ajustado de forma a não aumentar as violações nas janelas de tempo. É feita então uma atualização nos horários para todos os pontos posteriores a garagem de origem, e também é calculado o tempo de viagem dos clientes. Por fim, para cada ponto de embarque pertencente à rota em questão é calculado o seu respectivo atraso (sendo o horário de início do serviço ajustado de forma a reduzir a duração da rota e o tempo de viagem dos clientes e não aumentar as violações nas janelas de tempo), e os horários de todos os pontos posteriores são atualizados, assim como o tempo de viagem dos clientes cujo desembarque é posterior ao ponto de embarque em questão.

Considerando então uma rota  $V_k = \{v_0, v_p, v_{n+p}, v_{n+p+1}, \dots, v_z\}$ , executada pelo veículo  $k$  ( $\forall k \in K$ ), tem-se que  $v_0$  e  $v_z$  representam, respectivamente, as garagens de origem e de destino de  $k$  ( $v_0 \in G^-$ ;  $v_z \in G^+$ ), e os demais pontos representam os pontos de embarque e desembarque, sempre lembrando que o ponto  $v_{n+i}$  representa o ponto de desembarque referente ao ponto de embarque  $v_p$ , ou seja  $v_i \in P$  e  $v_{n+i} \in U$ . Para uma rota  $V_k$  qualquer, o atraso é calculado da maneira apresentada nas equações (19), (20) e (21).

$$F_i = \min_{i \leq j \leq z} \left\{ \sum_{i < p \leq j} W_p + (\min\{1_j - B_j, R''_i - R_j\})^+ \right\} \quad \text{se } i \in P \quad (19)$$

$$F_i = \min_{i \leq j \leq z} \left\{ \sum_{i < p \leq j} W_p + (1_j - B_j) \right\} \quad \text{se } i \in G^- \quad (20)$$

$$F_i = 0 \quad \text{se } i \in \{U \cup G^+\} \quad (21)$$

Nas equações anteriores ((19) (20) e (21)), nota-se que a variável  $F_i$  representa o tempo máximo em que o veículo

### 4.2 Estrutura de vizinhança

Como estrutura de vizinhança foram utilizados três diferentes movimentos de troca: *Reordenar rota, Re-alocar ponto e Trocar pontos*. Esses movimentos são baseados em outros encontrados frequentemente nos trabalhos referentes ao DARP (CORDEAU; LAPORTE, 2003; JORGENSEN, 2007; SAVELSBERGH, 1992; TOTH; VIGO, 1996, 1997).

É interessante destacar que nesses movimentos não são consideradas as garagens, pois essas são “fixas” em todas as rotas, e conseqüentemente suas posições não poderão ser alteradas.

O movimento *Reordenar rota* consiste basicamente em selecionar uma rota qualquer pertencente à solução, selecionar um ponto qualquer nessa rota, selecionar uma nova posição para esse ponto e trocar sua posição para a nova. Esse movimento é ilustrado na Figura 3.

Como é selecionado apenas um ponto na rota, este pode ser de embarque ou desembarque. No primeiro caso (Figura 3a), a nova posição deverá, obrigatoriamente, ser **anterior** ao seu respectivo ponto de desembarque. Já no segundo caso (Figura 3b), a nova posição deverá ser **posterior** ao respectivo ponto de embarque. Esses “limites” são apresentados pelas linhas pontilhadas na Figura 3.

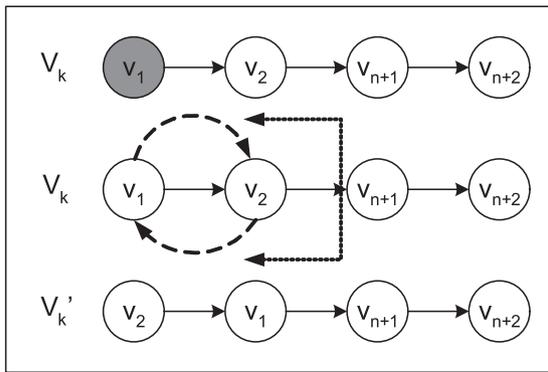
O movimento *Realocar pontos* consiste basicamente em também selecionar duas rotas quaisquer pertencentes à solução ( $V_{k1}$  e  $V_{k2}$ ), selecionar uma requisição qualquer ( $v_i$  e  $v_{n+i}$ ) em apenas uma das duas rotas, extraí-la (seus pontos de embarque e desembarque) de sua rota de origem ( $V_{k1}$ ) e adicioná-la na outra rota ( $V_{k2}$ ), em posições quaisquer, selecionadas aleatoriamente. Esse movimento é ilustrado na

Figura 4. O ponto de embarque ( $v_i$ ) e seu respectivo ponto de desembarque ( $v_{n+1}$ ) são extraídos simultaneamente, porém sua inserção na “outra” rota pode ser feita de forma separada, ou seja, esses pontos são alocados individualmente em qualquer posição da rota, porém sempre mantendo a condição de que o ponto de embarque fique antes do ponto de desembarque (restrição de precedência).

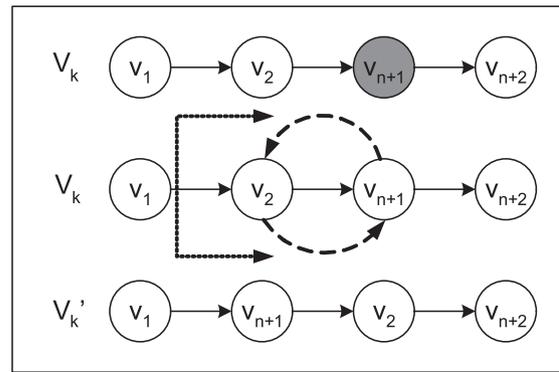
Já o movimento *Trocar pontos* consiste basicamente em selecionar duas rotas quaisquer pertencentes à solução, selecionar uma “requisição” (ponto de embarque e seu respectivo ponto de desembarque) qualquer em cada uma das duas rotas, e trocá-las. Esse movimento é ilustrado na

Figura 5. Nesse caso, como são trocadas requisições, ou seja, seus pontos de embarque e desembarque são trocados simultaneamente (trocados em pares), pode-se garantir que o ponto de embarque será sempre anterior ao seu respectivo ponto de desembarque.

A partir dessa estrutura de vizinhança o SA foi implementado de uma forma em que cada solução vizinha é gerada por apenas um desses movimentos, sendo a sua escolha feita de forma aleatória, porém uniformemente distribuída, possibilitando assim uma boa diversidade entre as soluções intermediárias geradas, e conseqüentemente uma boa exploração do espaço de soluções.

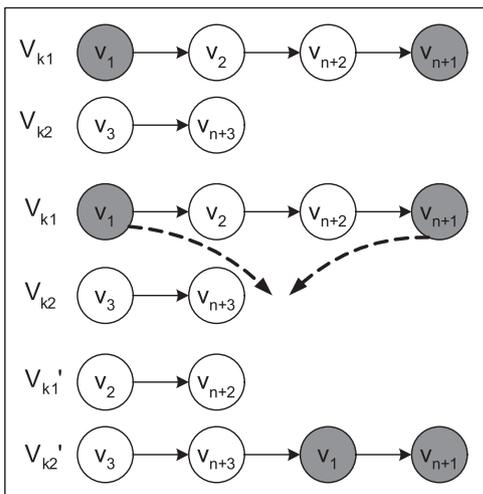


(a)

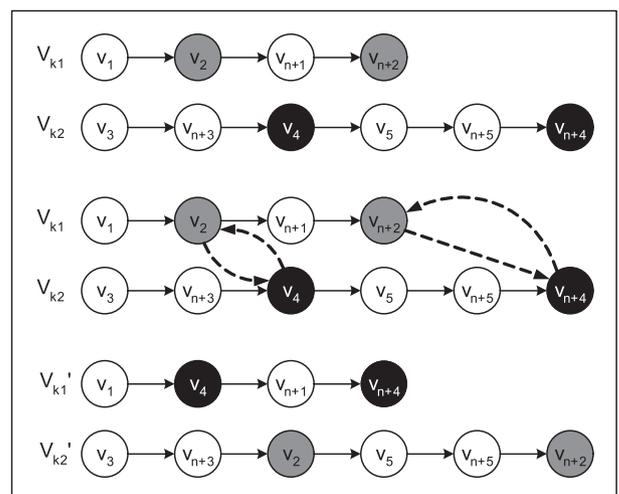


(b)

**Figura 3: Movimento Reordenar rota.**



**Figura 4: Movimento Realocar pontos.**



**Figura 5: Movimento Trocar pontos.**

1. DADO ( $\alpha$ ,  $S_{Max}$ ,  $T_0$  e  $T_c$ ) FAÇA
2. GERAR (uma solução  $S$  pela heurística de distribuição);
3. APLICAR (a heurística de programação em todas as rotas de  $S$ );
4.  $S^* \leftarrow S$ ; {Melhor solução obtida até então}
5.  $IterT \leftarrow 0$ ; {Número de iterações na temperatura  $T$ }
6.  $T \leftarrow T_0$ ; {Temperatura corrente}
7. ENQUANTO ( $T > T_c$ ) FAÇA
8. ENQUANTO ( $IterT < S_{Max}$ ) FAÇA
9.  $IterT \leftarrow IterT + 1$ ;
10. GERAR (um vizinho  $S'$  por um dos mov. de troca);
11. APLICAR (a heurística de programação em todas as rotas de  $S'$ );
12.  $\Delta \leftarrow f(S') - f(S)$ ;
13. SE ( $\Delta < 0$ )  $S \leftarrow S'$ ;
14. SE ( $f(S') < f(S^*)$ )  $S^* \leftarrow S'$ ; FIM-SE;
15. SENÃO
16. TOMAR ( $x \in [0,1]$ );
17. SE ( $x < e^{-\Delta/T}$ )  $S \leftarrow S'$ ; FIM-SE;
18. FIM-SE;
19. FIM-ENQUANTO;
20.  $T \leftarrow \alpha * T$ ;  $IterT \leftarrow 0$ ;
21. FIM-ENQUANTO;
22.  $S \leftarrow S^*$ ;
23. RETORNAR ( $S$ );

**Figura 6 - Algoritmo Simulated Annealing implementado.**

A função objetivo  $f(S)$  utilizada para avaliar as soluções é aquela descrita pelos termos (1), (2) e (3), e as restrições apresentadas no modelo proposto na ((4) a (18)) são atendidas implicitamente nas heurísticas de distribuição e programação e nos movimentos de troca aqui descritos. Um pseudocódigo do SA implementado é apresentado na Figura 6.

## 5. RESULTADOS COMPUTACIONAIS

Visando avaliar o potencial da abordagem apresentada neste trabalho, foram realizados vários experimentos nos quais foram utilizadas as instâncias apresentadas por Cordeau e Laporte (2003) (disponíveis em: <<http://www.hec.ca/chairedistributique/data/darp/>>), que são referências em inúmeros trabalhos de considerável importância para resolução do DARP (BERGVINSDOTTIR, 2004; CORDEAU, 2006; JORGENSEN et al., 2007). Estão disponíveis 20 instâncias, combinadas entre 24 a 144 requisições de transporte (48

a 288 pontos) e 3 a 13 veículos, sendo as primeiras instâncias (R1a – R10a) formadas por janelas de tempo “largas” e o restante (R1b – R10b) formado por janelas de tempo mais “estreitas”.

Essas instâncias representam problemas com garagem única e frota homogênea, e não adotam o conceito de tempo máximo de espera. Entretanto, o modelo proposto (Seção 3) se adapta facilmente a elas. Para isso, basta tratar essa única garagem como garagem de origem e destino de todos os veículos, e um valor muito grande como tempo máximo de espera (que irá eliminar a possibilidade de ocorrer qualquer violação).

Os parâmetros utilizados pelo SA, em todos os experimentos foram  $T_0 = 20000$ ,  $\alpha = 0.975$ ,  $T_c = 0.01$  e  $S_{Max} = 1000$ . Já a escolha dos valores das penalizações (vetores  $\omega$  e  $\beta$ ) aplicadas na função objetivo ((1), (2) e (3)) foi baseada em uma análise apresentada por Jorgensen et al. (2007). Porém, neste trabalho, a penalização para as violações nos requisitos essenciais foi mais “pesada”, visando evitar a obtenção de soluções inválidas para o problema. Os pesos utilizados foram:  $\omega = [8,0,1,3,1]$  e  $\beta = [1500,1500,1500,1500,1500]$ .

Todos os experimentos foram realizados em um *laptop Toshiba A10 S127* com processador *Intel Celeron®* de 2.0 GHz e 256MB de memória RAM. Toda a implementação foi desenvolvida na linguagem C++. As soluções obtidas por Cordeau e Laporte (2003) foram executadas em um *PC Intel Pentium 4®* de 2.0 GHz, e as obtidas por Jorgensen et al. (2007) em um *PC Celeron®* de 2.0 GHz.

$$Desvio = \left( \frac{f(S) \text{ média} - \text{Melhor } f(S)}{\text{Melhor } f(S)} \right) * 100 \quad (22)$$

Para validação do modelo proposto, e da aplicação do SA ao DARP, foram realizados cinco testes para cada instância. A Tabela 1 apresenta um resumo dos resultados obtidos nesses testes ( $f(S)$  é o valor da função objetivo). Nessa tabela, a coluna *Melhor  $f(S)$*  indica o valor da função objetivo da melhor solução encontrada nos cinco testes (para cada instância). A coluna  *$f(S)$  média* apresenta a média aritmética das cinco funções objetivo encontradas, e a coluna *Desvio* é obtida pela equação (22). Em todos esses testes o número de veículos utilizados foi igual ao disponível, todas as requisições de transportes foram atendidas, e o mais importante, todas as soluções obtidas foram válidas, ou seja, todas as restrições essenciais foram satisfeitas.

Os melhores resultados obtidos (Tabela 2) ainda são comparados aos obtidos por Cordeau e Laporte (2003), com o uso da meta-heurística Busca Tabu (Tabela 3), e por Jorgensen et al. (2007), utilizando um Algoritmo Genético

**Tabela 1: Resumo dos experimentos realizados.**

Instância	Nº de veículos	Nº de requisições	f(S) média	Melhor f(S)	Desvio (%)
R1a	3	24	3721,57	3677,91	1,19
R2a	5	48	7101,87	7017,34	1,20
R3a	7	72	11982,18	11873,76	0,91
R4a	9	96	13982,52	13725,92	1,87
R5a	11	120	16006,82	15736,66	1,72
R6a	13	144	20607,77	20465,39	0,70
R7a	4	36	5718,46	5610,05	1,93
R8a	6	72	11554,71	11343,19	1,86
R9a	8	108	17345,60	15632,09	10,96
R10a	10	144	23104,61	22430,00	3,01
R1b	3	24	3407,87	3379,74	0,83
R2b	5	48	5925,35	5889,56	0,61
R3b	7	72	11045,26	11006,12	0,36
R4b	9	96	12856,75	12807,87	0,38
R5b	11	120	14874,48	14544,13	2,27
R6b	13	144	18795,79	18518,82	1,50
R7b	4	36	5202,93	5136,37	1,30
R8b	6	72	10791,67	10703,17	0,83
R9b	8	108	15180,91	15013,71	1,11
R10b	10	144	20492,56	19969,15	2,62

**Tabela 2: Melhores soluções obtidas pelo *Simulated Annealing*.**

Instância	Distância percorrida	Duração das rotas (min)	Tempo de espera (min)		Tempo de viagem (min)		CPU (min)
			total	médio	total	médio	
R1a	252,79	831,3	98,51	2,05	241,93	10,08	1,00
R2a	437,45	1992,34	594,9	6,2	310,17	6,46	1,20
R3a	831,74	2404,67	132,93	0,92	894,08	12,42	1,46
R5a	1085,45	3920,25	434,81	1,81	899,35	7,49	1,79
R9a	1064,23	3258,66	34,42	0,16	1275,06	11,81	2,28
R10a	1392,09	4475,42	203,33	0,71	2204,85	15,31	2,72
R1b	251,85	738,42	6,57	0,14	206,66	8,61	0,92
R2b	436,69	1428,44	31,75	0,33	311,95	6,5	1,30
R5b	1010,09	3654,02	243,94	1,02	855,16	7,13	1,95
R6b	1289,31	4318,33	149,02	0,52	1245,66	8,65	1,94
R7b	375,67	1095,67	0	0	345,1	9,59	1,05
R9b	1041,09	3315,28	114,19	0,53	1085,18	10,05	2,26
R10b	1414,65	4332,69	38,04	0,13	1427,08	9,91	2,77
<b>TOTAL</b>	<b>10883,10</b>	<b>35765,49</b>	<b>2082,41</b>	<b>14,52</b>	<b>11302,23</b>	<b>124,01</b>	<b>22,64</b>

(Tabela 4). Essas referências foram escolhidas devido ao fato de representarem o atual estado da arte em que o problema se encontra. Apenas os resultados obtidos para algumas instâncias são comparados a essas referências, pois Jorgensen et al. (2007) não apresentam os resultados para as demais.

## O agrupamento e o sequenciamento das rotas foram realizados separadamente da programação em si, porém ambos utilizando a estrutura de vizinhança e as heurísticas propostas.

Comparando os resultados obtidos com os apresentados por Cordeau e Laporte (2003), pode-se notar que a abordagem proposta neste trabalho apresentou um aumento de 63,53% na distância total percorrida pelos veículos, porém no tempo de duração das rotas houve um aumento de apenas 0,64%. Já no tempo médio de espera dos veículos esta abordagem foi capaz de obter uma redução de 66,17%, no tempo médio de viagem dos clientes a redução foi de 80,46%, e o mais interessante, uma redução

de 95,45% no tempo de processamento necessário para obter as soluções.

Em relação aos resultados obtidos por Jorgensen et al. (2007), a abordagem proposta neste trabalho foi capaz de reduzir 11,71% no tempo de duração das rotas, 48,53% no tempo médio de espera dos veículos, 75,33% no tempo médio de viagem dos clientes, e 95,37% no tempo de processamento. Nesse caso, como as distâncias percorridas pelos veículos não são apresentadas em Jorgensen et al. (2007), não é feita nenhuma comparação. Já as distâncias apresentadas em Cordeau e Laporte (2003), e neste trabalho, são distâncias Euclidianas entre os pontos, e consequentemente não apresentam unidade de medida.

## 6. CONCLUSÕES

Este trabalho apresentou uma nova abordagem para resolver o problema *dial-a-ride*. O modelo proposto foi capaz de representar o problema de uma forma generalizada, sendo facilmente adaptada a outros modelos já conhecidos.

**Tabela 3: Melhores soluções obtidas por Cordeau e Laporte (2003).**

Instância	Distância percorrida	Duração das rotas (min)	Tempo de espera (min)		Tempo de viagem (min)		CPU (min)
			total	médio	total	médio	
<b>R1a</b>	190,02	881,16	211,15	4,40	1094,99	45,62	1,90
<b>R2a</b>	302,08	1985,94	723,87	7,54	1976,73	41,18	8,06
<b>R3a</b>	532,08	2579,35	607,27	4,22	3586,68	49,82	17,18
<b>R5a</b>	636,97	3869,95	832,98	3,47	6156,48	51,30	46,24
<b>R9a</b>	672,44	3155,49	323,05	1,50	5621,77	52,05	50,51
<b>R10a</b>	878,76	4480,10	721,33	2,50	7163,71	49,75	87,53
<b>R1b</b>	164,46	965,06	320,60	6,68	1041,50	43,40	1,93
<b>R2b</b>	296,06	1564,74	308,68	3,22	2393,18	49,86	8,29
<b>R5b</b>	589,74	3595,63	605,89	2,52	6104,72	50,87	54,33
<b>R6b</b>	743,60	4072,47	448,88	1,56	7347,39	51,02	73,70
<b>R7b</b>	248,21	1097,25	129,03	1,79	1761,99	48,94	4,23
<b>R9b</b>	601,96	3249,29	487,33	2,26	5581,02	51,68	51,28
<b>R10b</b>	798,63	4040,99	362,37	1,26	7072,29	49,11	92,41
<b>TOTAL</b>	<b>6655,01</b>	<b>35537,42</b>	<b>6082,43</b>	<b>42,92</b>	<b>56902,45</b>	<b>634,60</b>	<b>497,59</b>

**Tabela 4 - Melhores soluções obtidas por Jorgensen et al. (2007).**

Instância	Distância percorrida	Duração das rotas (min)	Tempo de espera (min)		Tempo de viagem (min)		CPU (min)
			total	médio	total	médio	
<b>R1a</b>	-	1039	260	5,42	310	12,90	5,57
<b>R2a</b>	-	1994	514	5,36	1330	27,72	11,43
<b>R3a</b>	-	2781	301	2,09	2894	40,20	21,58
<b>R5a</b>	-	4274	527	2,20	4837	40,30	58,23
<b>R9a</b>	-	3526	32	0,15	6719	62,21	40,78
<b>R10a</b>	-	5025	246	0,86	8341	57,92	65,98
<b>R1b</b>	-	928	164	3,42	549	22,89	5,46
<b>R2b</b>	-	1710	162	1,69	1300	27,07	11,72
<b>R5b</b>	-	4336	568	2,37	4720	39,33	58,93
<b>R6b</b>	-	5227	513	1,78	6397	44,42	81,23
<b>R7b</b>	-	1316	128	1,78	784	21,76	8,29
<b>R9b</b>	-	3676	177	0,82	5358	49,61	44,66
<b>R10b</b>	-	4678	85	0,29	8119	56,38	66,41
<b>TOTAL</b>	-	<b>40508</b>	<b>3678</b>	<b>28,21</b>	<b>51657</b>	<b>502,72</b>	<b>488,61</b>

O agrupamento e o sequenciamento das rotas foram realizados separadamente da programação em si, porém ambos utilizando a estrutura de vizinhança e as heurísticas propostas.

O *Simulated Annealing* integrado com as demais heurísticas apresentadas foi capaz de obter, em todos os casos, e com pouco tempo de processamento, soluções válidas para o problema. Além disso, mostrou-se robusto, pois como pode ser observado na Tabela 1, os desvios apresentados foram satisfatoriamente baixos. A estrutura de vizinhança, por meio dos movimentos de troca, mostrou ser adequada e eficiente para exploração do espaço de soluções.

Os resultados obtidos (ver Tabelas 1 e 2) mostram que o *Simulated Annealing*, juntamente com o modelo proposto (Seção 3) e as demais heurísticas descritas na Seção 4, foi capaz de gerar soluções de boa qualidade para todas as instâncias em tempos computacionais expressivamente baixos. Esses resultados ainda foram comparados com duas outras abordagens recentes encontradas na literatura, e em todos os casos a “qualidade do serviço” foi expressivamente maior, ou seja, a inconveniência dos clientes foi significativamente reduzida, o que na prática,

em se tratando da perspectiva humana, reflete melhores soluções.

Em relação ao trabalho de Cordeau e Laporte (2003), a metodologia proposta apresentou um acréscimo na distância percorrida pelos veículos, apesar da duração das rotas ter sido praticamente igual. Isso é justificado devido ao fato de que nos resultados apresentados por Cordeau e Laporte (2003) os veículos, apesar de “andarem” menos, gastam boa parte de seu tempo “esperando”, e conseqüentemente os clientes também. Já nos resultados obtidos pelo SA, essa espera é menor, pois os veículos priorizam (com o uso de penalizações) o atendimento aos clientes, ou seja, a qualidade do serviço prestado.

Enfim, os resultados mostram claramente o potencial da abordagem apresentada, onde soluções de alta qualidade são obtidas, para problemas relativamente grandes, em tempos de processamento expressivamente baixos.

A partir deste trabalho tem-se um grande campo de pesquisa a ser explorado, como por exemplo a aplicação desta abordagem a problemas reais encontrados nos municípios brasileiros, e também a outros problemas correlatos ao DARP.

**Artigo recebido em 05/2006**

**Aprovado para publicação em 15/12/2008**

## REFERÊNCIAS

- BAUGH, J. W.; KAKIVAYA, D. K. R.; STONE, J. R. Intractability of the dial-a-ride problem and a multiobjective solution using simulated annealing. *Engineering Optimization*, v. 30, n. 2, p. 91-123, 1998.
- BERGVINSDOTTIR, K. B. *The genetic algorithm for solving the dial-a-ride problem*. 2004. 140 p. (IMM 2004-37). Dissertação (Master of Science in Engineering). Technical University of Denmark (DTU), Lyngby, 2004.
- CALVO, R. W.; COLORNI, A. An effective and fast heuristic for the dial-a-ride problem. *4OR: A Quarterly Journal of Operations Research*, v. 5, n. 1, p. 61-73, 2007.
- CORDEAU, J. F. A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research*, v. 54, n. 3, p. 573-586, 2006.
- CORDEAU, J. F.; LAPORTE, G. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, v. 37, n. 6, p. 579-594, 2003.
- \_\_\_\_\_. The dial-a-ride problem: models and algorithms. *Annals of Operations Research*, v. 153, n. 1, p. 29-46, 2007.
- ILOG. *Ilog CPLEX 9.0: user's manual*. France, 2003. 564 p.
- JORGENSEN, R. M.; LARSEN, J.; BERGVINSDOTTIR, K. B. Solving the dial-a-ride problem using genetic algorithms. *Journal of the Operational Research Society*, v. 58, n. 10, p. 1321-1331, 2007.
- KIRKPATRICK, S.; GELLAT, D. C.; VECCHI, M. P. Optimization by simulated annealing. *Science*, v. 220, n. 4598, p. 671-680, 1983.
- MADSEN, O. B. G.; RAVN, H. F.; RYGAARD, J. M. A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. *Annals of Operations Research*, v. 60, n. 1, p. 193-208, 1995.
- ROPKE, S.; CORDEAU, J. F.; LAPORTE, G. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks*, v. 49, n. 4, p. 258-272, 2007.
- SAVELSBERGH, M. W. P. The vehicle routing problem with time windows: minimizing route duration. *ORSA Journal on Computing*, v. 4, n. 2, p. 146-154, 1992.
- TOTH, P.; VIGO, D. Fast local search algorithms for the handicapped persons transportation problem. In: OSMAN, I. H.; KELLY, J. P. (Ed.). *Meta-heuristics: theory and applications*. Berlin: Springer, 1996. p. 677-690.
- \_\_\_\_\_. Heuristic algorithms for the handicapped persons transportation problem. *Transportation Science*, v. 31, n. 1, p. 60-71, 1997.
- XIANG, Z.; CHU, C.; CHEN, H. A fast heuristic for solving a large-scale static dial-a-ride problem under complex constraints. *European Journal of Operational Research*, v. 174, n. 2, p. 1117-1139, 2006.
- ZNAMENSKY, A.; CUNHA, C. B. Um modelo para o problema de roteirização e programação do transporte de deficientes. In: CONGRESSO DE PESQUISA E ENSINO EM TRANSPORTES, 13., 1999, São Carlos. *Anais...* São Carlos: ANPET, 1999. p. 59-62.

## AGRADECIMENTOS

Os autores agradecem aos revisores anônimos pelos valiosos comentários e sugestões, e à FAPESP (processo 04/11053-9) e ao CNPq (processo 304598/2003-8) pelo apoio financeiro parcial dado ao desenvolvimento deste trabalho.

## SOBRE OS AUTORES

### Geraldo Regis Mauri

Universidade Federal do Espírito Santo – UFES  
 End.: Centro de Ciências Agrárias, Alto Universitário s/n – Guararema – 29500-000 – Alegre – ES – Brasil  
 Tel.: (28) 3552-8932 Fax.: (28) 3552-8903  
 E-mail: mauri@cca.ufes.br

### Luiz Antonio Nogueira Lorena

Instituto Nacional de Pesquisas Espaciais – INPE  
 End.: Av. dos Astronautas, 1758 – Jd. Granja – Prédio CTE – sala 23 – 12210-970 – São José dos Campos – SP – Brasil  
 Tel.: (12) 3945-6553 Fax.: (12) 3945-6375  
 E-mail: lorena@lac.inpe.br