

DESIGNANDO ENTREGADORES EXTRAS NO ROTEAMENTO DE VEÍCULOS COM JANELAS DE TEMPO

Vitória Pureza

Departamento de Engenharia de Produção, Universidade Federal de São Carlos
Via Washington Luiz, km 235 – São Carlos, SP – CEP: 13565-905

vpureza@dep.ufscar.br

Reinaldo Morabito

Departamento de Engenharia de Produção, Universidade Federal de São Carlos
Via Washington Luiz, km 235 – São Carlos, SP – CEP: 13565-905

morabito@ufscar.br

RESUMO

Em atividades de entrega de mercadorias, muitas vezes observam-se dificuldades em atender a todos os pedidos durante a jornada de trabalho devido a tempos de serviço relativamente altos em cada ponto de demanda. Estas dificuldades são ainda maiores caso existam janelas de tempo para o início do atendimento dos clientes. Neste trabalho é proposta uma heurística de busca tabu, que além das decisões usuais de roteamento, define uma quantidade de entregadores extras a cada rota gerada. A utilização destes entregadores geralmente reduz tempos de serviço e permite que um maior número de pontos de demanda sejam atendidos durante a jornada de trabalho. Utilizando-se instâncias geradas a partir de um conjunto de exemplos clássicos da literatura, o desempenho do algoritmo é avaliado e suas soluções comparadas com as obtidas com um único entregador por rota e com limitantes inferiores derivados de um modelo de roteamento de veículos.

PALAVRAS CHAVE. Roteamento de veículos com janelas de tempo. Métodos Heurísticos. Busca Tabu. Aplicações a Logística e Transporte.

ABSTRACT

In activities of delivery of goods, it is often difficult to serve all requests during regular working hours due to relatively long service times in the demand sites. These difficulties are even greater if the beginning of the service in each site must occur within a time window. We propose a tabu search heuristic that in addition to the usual routing decisions, assigns a number of extra deliverymen to each route. Using more than one deliveryman usually reduces service times and allows a larger number of demand sites to be served during the regular working hours. Using instances generated from a set of classic examples from the literature, the performance of the algorithm is evaluated and its solutions compared to those obtained with a single deliveryman per route and with lower bounds derived from a vehicle routing model.

KEYWORDS. Vehicle routing with time windows. Heuristics. Tabu Search. Applications to Logistics and Transportation.

1. Introdução

Em problemas de roteamento de veículos busca-se construir um conjunto de rotas de atendimento de pontos de demandas com vistas à otimização de um ou mais objetivos. A literatura sobre o assunto endereça situações com diferentes características operacionais, tais como tipo de distribuição (coleta e/ou entrega), natureza do serviço (transporte de produtos/pessoas ou prestação de serviços), restrições de horário de visita aos clientes, limitações de capacidade dos veículos, duração máxima dos roteiros, restrições de tipos de veículos para atendimento de determinados clientes, entre várias outras. A grande maioria dos problemas de roteamento de veículos tem natureza combinatória, e é considerada de difícil resolução, o que explica os esforços de pesquisa e de desenvolvimento de métodos para o seu tratamento.

Neste trabalho, considera-se a situação enfrentada por empresas que entregam bens em áreas urbanas com frequência regular. Um exemplo típico são as empresas do setor de bebidas e tabaco onde a maioria dos clientes é composta por pequenos estabelecimentos que comercializam esses produtos. Devido à dificuldade de estacionamento dos veículos, pontos de demanda próximos entre si (*cluster*) têm associado um único ponto de parada para o veículo que irá servi-los. A entrega das cargas de cada ponto de demanda de um dado *cluster* é então realizada pelo motorista que visita os clientes a pé. Como os tempos de serviço em cada ponto de demanda são relativamente altos quando comparados ao tempo de deslocamento do veículo, é comum que parte da demanda diária não possa ser atendida durante a jornada de trabalho. Como estes mercados são geralmente competitivos, uma política usualmente adotada é de atendimento de toda a carteira diária de pedidos, ainda que isso resulte no pagamento de horas extras e possíveis multas impostas por órgãos fiscalizadores.

Esta realidade motivou o estudo do problema de roteamento de veículos com designação de entregadores extras. Note que a utilização de entregadores extras em cada rota pode tanto afetar os tempos de serviço individuais de cada nó do *cluster* (caso mais de um entregador atenda cada nó por vez) como o tempo de serviço total associado ao *cluster* (caso ocorra o atendimento simultâneo dos nós, cada qual com um ou mais entregadores). Em ambos os casos, a redução dos tempos de serviço afeta os tempos de conclusão do serviço, os quais, por sua vez, se refletem no número de clientes que podem ser atendidos em uma dada jornada de serviço. Desta forma, além das decisões usuais de roteamento e programação de veículos, a resolução do problema deve também definir o número de entregadores extras em cada rota (em geral, limitado devido a restrições de espaço na cabine do veículo), de maneira a atender o maior número de pontos de parada durante a jornada de trabalho a um mínimo custo. Tanto quanto sabemos, não existem registros da referida situação prática na literatura, embora o problema surja em algumas situações práticas de distribuição de produtos em áreas com grande densidade de clientes, como, por exemplo, na entrega de bebidas em regiões centrais de grandes cidades (Pureza e Morabito, 2008).

Neste trabalho é proposta uma abordagem heurística na qual se alterna aplicações de uma heurística de inserção e de um algoritmo de busca tabu. Utilizando-se um conjunto de instâncias com janelas de tempo propostas por Solomon (1987), o desempenho da abordagem é observado frente a soluções obtidas com um único entregador e por limitantes inferiores para o problema obtidos por um modelo matemático para o problema de roteamento de veículos, resolvido pela linguagem de modelagem GAMS com o otimizador CPLEX.

O restante deste artigo é organizado como se segue. A Seção 2 descreve o problema de roteamento de veículos clássico e com janelas de tempo. A Seção 3 descreve uma aplicação que sugere a relevância prática deste tema, enquanto na Seção 4 é apresentada a abordagem proposta. Na Seção 5 são discutidos os experimentos computacionais e os resultados obtidos para um conjunto de instâncias tratado, seguida de conclusões e próximos passos da pesquisa na Seção 6.

2. O problema de roteamento de veículos clássico e com janelas de tempo

O problema de roteamento de veículos clássico (PRV) consiste na definição de rotas de custo mínimo para uma frota de veículos a fim de satisfazer um conjunto de clientes com localizações espaciais e demandas conhecidas por coleta ou entrega. No PRV, todos os pontos de coleta (ou de entrega) dos pedidos são idênticos e correspondem ao depósito central. Cada cliente deve ser visitado exatamente uma vez, ou seja, sua demanda deve ser atendida por um único veículo. A demanda total dos clientes não deve exceder a capacidade total do veículo (expressa em termos de volume, peso ou número de paletes) designado para a rota, assim como o tempo total de cada rota (composto por tempos de deslocamento e de atendimento de cada cliente) não deve violar um limite pré-estabelecido, correspondente ao fim da jornada de trabalho.

O problema de roteamento de veículos com janelas de tempo (PRVJT) é uma variante do PRV onde se inclui uma restrição operacional sobre a observância do período do dia (janela de tempo) estabelecido pelo cliente para entrega ou coleta. Note que esta restrição está particularmente presente na coleta/entrega de mercadorias em estabelecimentos de grande porte, como hipermercados. Ela também aparece na distribuição em centros urbanos, onde a circulação de veículos de grande porte é muitas vezes limitada aos horários de tráfego leve.

Nestes e demais problemas da classe, o roteamento define o subconjunto de clientes a serem atendidos por cada veículo e a ordem na qual as visitas serão realizadas. A programação, por sua vez, indica os instantes de tempo de ocorrência das visitas as quais devem respeitar restrições temporais ou de precedência (Bodin et al., 1983; Cordeau et al., 2002). Existem muitas de aplicações bem sucedidas de roteamento de veículos no mundo real, como por exemplo, em indústrias de bebidas, de alimentação, coleta de lixo e serviço postal (Toth e Vigo, 2002), roteamento de ônibus escolares, entrega de jornais e correspondências, entrega de óleo, e roteamento e programação de aeronaves. Compilações de métodos de solução, classificação de problemas e aplicações podem ser encontradas, por exemplo, em Bodin et al. (1983), Laporte (1992), Osman (1993), Fisher (1995), Desroisiers et al. (1995), Cordeau et al. (2002) e Bräysy e Gendreau (2005a, 2005b).

Problemas de roteamento de veículos pertencem à área de otimização combinatória e são, na grande maioria, intratáveis quando o número de clientes é apenas mediano, no sentido de que não se conhecem algoritmos exatos que forneçam uma solução ótima em tempo computacional aceitável. Por este motivo, recorre-se a métodos heurísticos de resolução. Parte da motivação em estudá-los e desenvolver algoritmos para sua resolução é justificada pelo impacto nos custos logísticos. Um roteamento eficiente pode reduzir custos de distribuição, os quais se refletem em uma maior flexibilidade na prática dos preços finais dos produtos.

3. Exemplo de aplicação prática

O tipo de problema de roteamento de veículos aqui tratado é bem representado por uma situação prática de uma empresa de porte do setor de bebidas, localizada no interior do estado de São Paulo. Além da produção e engarrafamento, a empresa entrega seus produtos a pontos de venda, segundo o procedimento a seguir.

As rotas são elaboradas em um *software* comercial a partir de pedidos coletados por vendedores até o fim da jornada diária (pré-venda). Após o fim da jornada, novos pedidos são transmitidos por celular para o sistema e uma remessa é gerada. Os pedidos são, então, processados e validados pelo setor de faturamento. Com a validação, uma consulta ao estoque é também realizada para verificar se é possível atender as demandas dos clientes.

Os territórios atendidos são pré-estabelecidos, ou seja, as áreas de atendimento são definidas na malha viária de cada cidade e, a cada dia, um grupo de territórios é atendido pelas rotas. Dentre as demais restrições presentes na elaboração das rotas incluem-se:

- Disponibilidade de veículos da frota para cada período de entrega
- Capacidade dos veículos disponíveis
- Disponibilidade no estoque dos produtos solicitados (quantidade e *mix*)
- Disponibilidade de tempo para entrega dos produtos
- Prioridade de entrega do cliente
- Janelas de tempo em um ou mais clientes

A empresa possui frota própria dimensionada para atender a demanda diária total do mercado, considerando a sazonalidade existente para cada tipo de produto. Os veículos utilizados na distribuição são cadastrados no sistema, incluindo características como cubagem (capacidade de carga).

Após a elaboração das rotas, uma ordem de carregamento é emitida para a estiva (expedição), área responsável pela separação das cargas com o *mix* dos produtos e pelo carregamento da frota. O procedimento de montagem das cargas é feito manualmente no fim do dia anterior à distribuição.

A entrega dos produtos é feita por um único funcionário (com funções de motorista do veículo e entregador), geralmente no dia seguinte à coleta dos pedidos. Em regiões com concentração de pontos de venda em *clusters* (por exemplo, áreas urbanas centrais compostas por bares e restaurantes), o funcionário estaciona o veículo em um ponto de parada tal que as distâncias entre os clientes de um *cluster* e o veículo sejam as menores possíveis. Na prática, não há definição de um ponto de parada específico devido às vagas de estacionamento disponíveis não serem sempre as mesmas.

O tempo de serviço em cada cliente é relativamente alto uma vez que envolve os deslocamentos do funcionário, o contato inicial, o recebimento da nota fiscal, e o carregamento do veículo com vasilhames vazios retornados. Como a empresa adota a política de que todas as ordens planejadas para um dado dia sejam atendidas, na maioria das vezes, o tempo total de execução das rotas excede a jornada de trabalho dos funcionários. Tal violação representa custos em horas extras, além de sujeitar a empresa a multas impostas por órgãos fiscalizadores.

Note que nesta situação prática, a empresa poderia considerar alocar um ou dois entregadores extras no veículo (cuja cabine tem capacidade limitada a três indivíduos), para auxiliar o funcionário na distribuição dos produtos para os clientes do *cluster*. Os funcionários realizariam (a pé) o atendimento a cada cliente do *cluster* em questão, podendo atender seqüencialmente apenas um cliente por vez (vários funcionários por cliente), ou simultaneamente mais de um cliente por vez (um funcionário para cada cliente). Note que, desta maneira, o tempo de serviço no *cluster* variaria em função do número de entregadores extras designados à rota (i.e., com 0, 1 ou 2 entregadores extras).

A alternativa de se alocar entregadores extras nas rotas pode permitir que as jornadas de trabalho dos funcionários não sejam violadas. Estudos preliminares que vem sendo realizados na empresa indicam que esta alternativa poderia ser implementada na prática (Ferreira, 2010). Assim, o PRV com entregadores extras da empresa consiste em decidir como roteirizar os veículos para atender todos os clientes, respeitando as jornadas de trabalho, as capacidades dos veículos e as janelas de tempo dos clientes, de maneira a minimizar de forma hierárquica, o número total de veículos necessários, o número total de entregadores extras necessário, e a distância total percorrida pelos veículos. Os tempos de serviço em cada *cluster* de clientes são considerados dados de entrada do problema e variam em função do número de entregadores extras utilizados na rota deste *cluster*. Exceto pelo estudo em Pureza e Morabito (2008), não temos conhecimento de outros trabalhos que tratam este tipo de roteamento com entregadores extras.

4. Abordagem de solução

A discussão a seguir é uma descrição de um procedimento que captura elementos gerais para resolução do PRV com entregadores extras. Conforme mencionado na Seção 1, os

principais componentes do procedimento consistem de uma heurística de construção de rotas com determinação do número de entregadores, e de um algoritmo de busca tabu para redução do número de rotas. O procedimento garante que o tempo máximo de rota não seja excedido, ou seja, que a jornada de trabalho dos entregadores não seja violada, além de atender as restrições de capacidade dos veículos e janelas de tempo dos clientes. Também é considerada uma restrição sobre o número máximo de entregadores por rota devido a limitações de espaço na cabine do veículo.

O objetivo é o de determinar um conjunto de rotas para uma frota homogênea, que de maneira hierárquica minimize: (1) o número de veículos requerido para atender as demandas dos clientes associados ao cada ponto de parada, (2) o número de entregadores requerido no conjunto de rotas, e (3) a distância total percorrida. A minimização do número de veículos reflete a relevância deste custo fixo em relação ao número de entregadores e da distância total (custo variável).

Os passos gerais do procedimento de resolução proposto são descritos na Figura 1. As heurísticas de construção de rotas e de redução do número de rotas são apresentadas nas seções seguintes.

1. (**Fase 1**) Seja F o tamanho da frota, e P o conjunto de pontos de parada a serem roteados. Aplique a heurística de construção de rotas (descrita na Figura 2), seguida da heurística de redução do número de rotas (descrita na Figura 4). Seja S a solução resultante.
2. Se houver pontos de parada não atendidos após a fase 1, aplique a heurística de construção de rotas com vistas a inserir nós não roteados em S (**Fase 2**).
3. Enquanto houver pontos de parada não atendidos (**Fase 3**)
 - 3.1. Incremente o tamanho da frota em uma unidade ($F=F + 1$) e aplique a heurística de construção de rotas com vistas a inserir nós não roteados, obtendo uma nova solução S .
 - 3.2. Aplique a heurística de redução do número de rotas.
4. Retorne a solução S .

FIGURA 1: Passos do procedimento para o PRV com Entregadores Extras.

A aplicação da heurística de construção de rotas no passo 1 da Figura 1 (Fase 1) não garante que todos os pontos de parada sejam roteados dentro da jornada de trabalho. Isso se deve a limitações do tamanho de frota F , do número de entregadores em cada veículo, à natureza gulosa da heurística e à existência de janelas de tempo nas instâncias tratadas. Esta é a principal motivação para a utilização de um algoritmo para redução do número de rotas. Em particular, note que tal redução endereça os dois objetivos principais do problema. Ela disponibiliza veículos para a reaplicação da heurística de construção de rotas a fim de incluir pontos de parada ainda não roteados, e a cada veículo eliminado, reduz-se também o número de entregadores utilizados (Fase 2).

Caso ainda hajam pontos de parada não atendidos, permite-se que o tamanho de frota original seja aumentado, seguido da aplicação dos dois procedimentos de maneira alternada (Fase 3). Apesar desta fase ser naturalmente vista como um instrumento de análise do usuário sobre o impacto da expansão da frota, ela pode resultar em soluções que atendam todos os pontos de parada com a frota em seu tamanho original.

4.1 Heurística de construção de rotas

A construção de rotas da abordagem (passos 1, 2, e 3.1 da Figura 1) é primariamente realizada por uma heurística que, no caso de haver nós não roteados com a frota corrente, testa iterativamente incrementos no número de entregadores em cada rota j . O incremento no número de entregadores em um dado j define uma solução tentativa, e é selecionada a opção que

apresenta maiores ganhos segundo indicadores da qualidade da solução ordenados de forma hierárquica (Figura 2).

1. Seja P o conjunto de pontos de parada a serem roteados. Aplique a heurística de inserção (descrita na Figura 3) com um entregador por rota, obtendo a solução S . Seja r o número de rotas resultante, e $crew_j$ o número de entregadores em cada rota j (igual a um para todo j).
2. Se todos os pontos de parada de P foram roteados, vá para o passo 5.
3. Caso contrário, para cada rota j de S , faça $trycrew_j$ (número de entregadores na rota j da solução tentativa a ser testada) = $crew_j$ (número de entregadores na rota j da solução corrente S). Faça $fail=0$ (número de veículos que já contam com a tripulação máxima permitida $maxcrew$), e $trycrewI_j = trycrew_j$.
4. Repita até que $fail \geq r$ ou todos os pontos de parada tenham sido roteados:
 - 4.1. Para cada rota j da solução S :
 - 4.1.1. Se (a) $trycrew_j$ é menor ou igual a $maxcrew$, e (b) um incremento de uma unidade no último número de entregadores de j que não provocou melhorias em S ($lastfail_j + 1$) não excede $maxcrew$:

(Início de geração de solução tentativa)

 - 4.1.1.1. Se não houve falha em melhorar a solução S no último aumento do número de entregadores de j , incremente $trycrew_j$ em uma unidade.
 - 4.1.1.2. Caso contrário, se houve falha em melhorar a solução no último aumento do número de entregadores de j e $lastfail_j + 1$ não excede $maxcrew$, faça $trycrew_j$ igual a $lastfail_j + 1$.
 - 4.1.1.3. Ajuste a programação da rota j com a diminuição dos tempos de serviço decorrentes da adição do novo entregador.
 - 4.1.1.4. Aplique a heurística de inserção à S a fim de incluir pontos de parada não roteados.
 - 4.1.1.5. Se a solução tentativa T resultante melhora indicadores de qualidade (diminuição do número de pontos de parada não roteados > número de rotas > número de entregadores > distância > tempo total de rota), faça T a melhor solução tentativa T^* e atualize os indicadores de referência.
 - 4.1.1.6. Para cada rota j na solução corrente S , faça $trycrew_j = crew_j$.
 - 4.1.2. Caso contrário (a rota j não comporta um aumento no número de entregadores) faça $fail = fail + 1$.

(Fim de geração de solução tentativa)
 - 4.2. Se para nenhuma rota de S , verificam-se benefícios com a adição de um entregador, faça $lastfail_j = trycrewI_j$ para todo j . Caso contrário, faça $S = T^*$, e $crew_j =$ número de entregadores em j de T^* .
5. Retorne a solução S .

FIGURA 2: Passos da heurística de construção de rotas para o PRV com Entregadores Extras.

Note que para cada rota j da solução corrente é mantido um registro do último número de entregadores testado que não resultou em melhorias na solução ($lastfail_j$). Desta forma, no próximo teste em j , o número de entregadores a ser utilizado corresponderá a $lastfail_j + 1$. Os incrementos testados devem respeitar a restrição do tamanho da tripulação $maxcrew$ em cada veículo. Note também que esta implementação não restringe diretamente a quantidade total de entregadores disponibilizados.

Como as instâncias tratadas neste trabalho possuem janelas de tempo, nos passos 1 e 4.1.1.4 da heurística de construção de rotas é utilizada uma heurística de inserção de nós baseada no algoritmo i proposto em Solomon (1987), e descrito na Figura 3. A seleção da melhor posição de inserção de nós utiliza os mesmos critérios do algoritmo original.

1. **Inicialização:** Seja F o tamanho da frota, e P , o conjunto total de pontos de parada a serem inseridos na solução corrente S . Seja I o conjunto de pontos inseridos, e NI o conjunto de pontos não inseridos. Faça $NI = P$ e $I = \emptyset$.
2. Se $r = 0$ ou não houver posições factíveis para inserção dos pontos em NI nas rotas correntes:
 - 2.1 Selecione como ponto semente (p_s) da rota em construção, aquele com o menor limitante superior de janela de tempo (pedido de maior urgência). Caso nenhum ponto possa iniciar a rota, vá para o passo 6.
 - 2.2 Faça $r = r + 1$ e insira p_s em r . Faça $I = I \cup p_s$ e $NI = NI - p_s$.
3. Para cada ponto p em NI , obtenha a melhor posição de inserção de p nas rotas correntes, respeitando a restrição de tempo máximo de rota.
4. Se no passo anterior tiver sido obtida pelo menos uma posição factível de inserção para pelo menos um ponto, selecione e insira o ponto p^* na rota (e posição associada) com a melhor posição. Faça $I = I \cup p^*$ e $NI = NI - p^*$. Seja S a solução resultante.
5. Vá para o passo 3 se $NI \neq \emptyset$ e houver posições factíveis de inserção dos pontos em NI nas rotas correntes. Caso contrário, se $NI \neq \emptyset$ e $r < F$, e não houver posições factíveis de inserção, vá para o passo 2.1.
6. Retorne a solução S .

FIGURA 3: Passos da heurística de inserção para o PRV com Entregadores Extras.

4.2. Algoritmo de busca tabu de redução do número de rotas

Neste trabalho, utilizou-se o algoritmo proposto em Pureza (2006), o qual é caracterizado por mecanismos de intensificação/diversificação que provocam mudanças em elementos de busca tabu (Glover e Laguna, 1997) a partir da análise de padrões de trajetória de busca. Aplicações destes mecanismos são também encontradas nos trabalhos de França et al. (1999), Pureza e França (2001) e Montané (2007).

Em linhas gerais, a abordagem admite que o padrão geral de trajetórias de busca (a curva valor da função objetivo vs iteração) reflete o nível de restritividade imposto pelas condições operacionais, em particular, pelos elementos tabu. A geração de custos de soluções em faixas relativamente estreitas (trajetória estagnada) indica processos de busca aprisionados em uma região do espaço limitada, e estão associados a elementos tabu pouco restritivos. Ela é contrastada com o aumento dos níveis de restritividade, resultando na seleção de movimentos menos atraentes e conseqüente diversificação. Custos de solução crescentes (trajetória ascendente) indicam que em função de condições operacionais muito restritivas, movimentos menos atraentes para o objetivo do problema estão sendo selecionados. Ela é tratada com a relaxação da restritividade. Finalmente, custos de solução decrescentes (trajetória descendente) sugerem que o processo de busca atingiu uma região potencialmente promissora e parece capaz de explorá-la. Neste caso, estabelecem-se níveis de restritividade moderados de forma a permitir uma exploração mais detalhada da região.

A identificação dos padrões de trajetória é feita calculando-se o desvio percentual das médias dos custos de solução observados nos dois últimos estágios de busca. Médias dos dois últimos estágios de busca aproximadamente iguais identificam uma trajetória estagnada. Se a média do último estágio é maior que a do estágio anterior, a busca está descrevendo uma trajetória ascendente. Se a média do último estágio é menor que a do estágio anterior, a busca está descrevendo uma trajetória descendente.

Para o problema de redução do número de rotas na presença de janelas de tempo, Pureza (2006) aplica movimentos convencionais de inserção e troca de nós intra e inter-rota, e propõe

uma função de avaliação lexicográfica para seleção dos movimentos. A função de avaliação segue vários critérios, tais como maior redução no número de veículos, e maior urgência de atendimento do nó a ser exportado de rotas com menores números de nós. Mais detalhes sobre o algoritmo podem ser encontrados em Pureza (2006).

5. Experimentos computacionais

A abordagem heurística descrita na seção anterior (Algoritmo **Hm**) foi implementada em Borland Delphi 7 (Pascal) e os experimentos realizados em um computador Intel Core2 2,4 GHz, e 2,00 GB de RAM. O conjunto R1 de 12 instâncias com janelas de tempo, proposto em Solomon (1987), foi utilizado nos testes. Este conjunto é caracterizado por janelas de tempo estreitas e distribuição geográfica aleatória dos nós.

O parâmetro *maxcrew* (tripulação máxima em cada veículo) foi fixado em 3 e como tamanho de frota *F* adotou-se o número de veículos da melhor solução reportada para o PRVJT. Os demais dados originais de cada instância foram mantidos, com exceção dos tempos de serviço em cada nó (ponto de parada) *i*. Estes últimos foram calculados pela função:

$$t_s(i) = \text{Mínimo} \{ \text{instante máximo de retorno dos veículos ao depósito} - \text{Máximo} \\ (\text{limitante inferior da janela de tempo de } i, \text{ distância entre } i \text{ e o depósito}) - \text{distância entre} \\ i \text{ e o depósito}, 2 \times \text{demanda de } i \}$$

Note que, a princípio, o tempo de serviço $t_s(i)$ é proporcional à demanda do ponto de parada *i* (cada unidade de demanda utiliza 2 unidades de tempo de serviço). Em outras palavras, quanto maior a demanda, maior o tempo de serviço. Entretanto, o valor do tempo de serviço é também limitado pelo tempo necessário para o deslocamento do veículo a partir do depósito e seu retorno, de forma a garantir que qualquer ponto de parada possa ser servido com um veículo exclusivo sem violar a restrição de tempo máximo de rota.

É importante ressaltar que a função acima é válida para pontos de parada servidos em rotas com um único entregador, ou seja, somente com o motorista. Se o número de entregadores em uma rota for maior que um, por simplicidade, o valor de $t_s(i)$ é dividido pelo número de entregadores da rota que serve *i* (note que este valor é um dado de entrada do problema e não precisa necessariamente guardar uma relação linear em função do número de entregadores).

Assim como na heurística *i* de Solomon (1987), a construção da solução inicial utiliza os parâmetros $[\sigma, \delta, \alpha_1, \alpha_2]$ nas decisões de seleção de nós para inserção nas rotas. Nos experimentos, cada instância foi resolvida a partir de três soluções iniciais (conjuntos de valores dos parâmetros). No algoritmo de redução de rotas, os valores dos parâmetros tabu utilizados em Pureza (2006) foram aqui replicados, com exceção do critério de parada: para cada aplicação do algoritmo, adotou-se o limite máximo de 60 segundos de tempo de CPU.

O desempenho da abordagem foi medido em termos dos objetivos descritos na Seção 4 (número de veículos, o número de entregadores, e distância total) e do tempo computacional para finalização da execução. A qualidade das soluções foi comparada à obtida com um único entregador em cada rota (Algoritmo **H1**), a qual corresponde ao resultado do passo 1 da heurística de construção de rotas (Figura 2), assim como pelos limitantes inferiores e superiores do número de veículos obtidos por um modelo de programação matemática de um PRVJT clássico. Utilizou-se o modelo com variáveis 0-1 de apenas dois índices apresentado em Bard et al. (2002), a fim de minimizar o número total de veículos de uma frota homogênea. Na resolução do modelo (**BKY02**), considerou-se que o tempo de serviço de cada ponto de parada é aquele obtido com exatamente três entregadores por rota. Ou seja, soluções ótimas obtidas com BKY02 fornecem o número ótimo de veículos do problema original onde o número de entregadores não é fixo, mas sim, uma decisão. A linguagem de modelagem GAMS com o *solver* CPLEX 11.0 (Brooke et al., 1992) foi utilizada para resolver o modelo, fixando-se um tempo máximo de execução de 3.600 segundos.

5.1. Resultados

A Tabela 1 apresenta os resultados obtidos para os 12 exemplos do conjunto R1. Para o algoritmo Hm proposto são apresentados o número de veículos utilizado (V), o número de entregadores (T), a distância total percorrida (D), e os tempos de CPU em segundos para finalização da execução. Para a heurística de inserção com um entregador (algoritmo H1), o número de entregadores é substituído pelo número de pontos de parada não atendidos durante a jornada de trabalho (NA).

Para o modelo BKY02, são apresentados os limitantes superiores (UB_V) e inferiores (LB_V) do número de veículos e o tempo de execução em segundos. A resolução do modelo foi restrita aos primeiros cinco exemplos do conjunto de instâncias, uma vez que experimentos preliminares mostraram que os *gaps* de otimalidade se mantêm em geral altos, mesmo após várias horas de execução. Isso sugere que os tempos de execução do GAMS/CPLEX são via de regra, excessivos para uma resolução ótima para este porte de instância.

Tabela 1 - Resultados computacionais - conjunto R1 (100 nós).

Exemplo	Heurística H1				Heurística Hm				Modelo BKY02		
	V	NA	D	CPU (s)	V	T	D	CPU (s)	UB_V	LB_V	CPU (s)
R101	19	32	1978,7	0,5	19	50	2130,9	180,6	19	19	5,0
R102	17	27	1627,1	0,4	18	39	2068,0	240,5	19	1	3.600
R103	13	36	1546,0	0,4	13	36	1592,5	240,7	18	1	3.600
R104	9	51	1072,9	0,4	11	30	1321,1	240,4	23	1	3.600
R105	14	42	1718,5	0,5	14	39	1765,1	180,6	16	10	3.600
R106	12	46	1526,7	0,5	12	31	1493,2	240,6	-	-	-
R107	10	49	1332,2	0,4	10	30	1228,9	240,5	-	-	-
R108	9	60	1089,7	0,5	10	28	1181,5	240,5	-	-	-
R109	11	44	1360,0	0,4	12	34	1486,9	300,5	-	-	-
R110	10	50	1106,9	0,4	11	32	1366,7	240,5	-	-	-
R111	10	54	1252,6	0,4	11	30	1298,6	240,5	-	-	-
R112	9	51	1166,7	0,4	10	30	1213,3	240,5	-	-	-

A Tabela 1 indica que o algoritmo proposto Hm produz um aumento na quantidade de pontos de parada servidos durante a jornada de trabalho de cerca 45% em relação às soluções obtidas com um único entregador (algoritmo H1). Juntamente com estas melhorias, verifica-se um aumento médio de 12,6% no número de veículos e a diminuição na distância percorrida de 5,6%. O aumento relativamente pequeno do número de veículos frente ao número de pontos incluídos deve-se naturalmente à adição de entregadores (aumento de 184%) e os esforços de redução do número de rotas. É interessante, entretanto, observar que em 5 das 12 instâncias, o número de veículos se manteve igual ao tamanho da frota. Nota-se também que enquanto o tempo de execução de H1 é menor que 1 segundo, Hm requer em média 4 minutos. Este esforço computacional pode ser reduzido, alterando-se o critério de parada utilizado e com possível degradação da qualidade da solução.

No tocante à resolução do modelo BKY02, o GAMS/CPLEX foi capaz de resolver otimamente apenas o modelo do exemplo R101 dentro do limite de tempo; o resultado obtido indica que Hm foi capaz de produzir uma solução com o número ótimo de veículos. Para os demais exemplos, nota-se que os valores dos limitantes inferiores estão relativamente distantes dos limitantes superiores. As melhores soluções factíveis são inferiores às de Hm, apresentando diferenças que variam entre 1 veículo (exemplo R102) e 12 veículos (exemplo R104).

Em função das dificuldades na aplicação do método exato para resolução de instâncias de 100 nós, novos experimentos foram realizados com Hm e BKY02, tomando-se os 25 e 50 primeiros nós nos exemplos R101-R105. A Tabela 2 ilustra os resultados obtidos com as 10 instâncias produzidas.

Tabela 1 - Resultados computacionais - conjunto R1 (25 e 50 primeiros nós).

Exemplo	Heurística Hm				Modelo BKY02		
	V	T	D	CPU (s)	UB _v	LB _v	CPU (s)
R101_25	8	16	739,0	120,0	8	8	0,03
R102_25	7	11	689,0	120,0	5	1	3.600,0
R103_25	4	10	556,8	120,0	4	2	3.600,0
R104_25	4	8	508,9	120,0	4	1	3.600,0
R105_25	5	12	959,4	120,0	5	5	3,4
R101_50	11	25	1236,2	180,1	11	11	0,2
R102_50	10	21	1182,5	120,1	8	1	3.600,0
R103_50	7	18	1562,1	180,1	9	6	3.600,0
R104_50	6	16	1302,1	180,1	10	1	3.600,0
R105_50	9	19	1849,1	122,5	8	8	2007,5

No tempo de execução disponibilizado, 4 dos 10 exemplos tratados (R101_25, R105_25, R101_50, e R105_5) foram resolvidos otimamente com o modelo, fornecendo um número ótimo de 8, 5, 11 e 8 veículos, respectivamente. Por outro lado, os exemplos R102_25, R103_25, R104_25, R102_50, R103_50, e R104_50 não obtiveram certificados de otimalidade dentro do limite de tempo, resultando em limitantes inferiores no número de veículos de 1, 2, 1, 1, 6 e 1, respectivamente. Em 4 exemplos com 25 nós de demanda, o número de veículos da melhor solução factível é idêntico ao provido por Hm. Nos demais exemplos, o número de veículos da melhor solução factível apresenta diferenças em relação a Hm que variam entre -2 e 4 veículos.

6. Conclusões

Neste trabalho, foram investigados possíveis impactos da utilização de entregadores extras no planejamento de rotas com restrições de tamanho de frota, capacidade dos veículos, janelas de tempo nos nós de demanda, e tempo máximo de rota (jornada de trabalho). Neste sentido, propôs-se um procedimento composto de uma heurística de construção de rotas com determinação do número de entregadores, e de um algoritmo de busca tabu para redução do número de rotas.

Resultados obtidos com instâncias clássicas com janelas de tempo indicam as vantagens do procedimento proposto. Em relação ao aumento na quantidade de pontos servidos durante a jornada de trabalho, ganhos relativos às soluções com um entregador atingiram cerca de 45%. O aumento médio no número de veículos utilizados (12,6%) e na distância total percorrida (5,6%) é consistente com o aumento do número de pontos servidos. Comparações com os limitantes inferiores e superiores para o número ótimo de veículos obtidos por um modelo matemático para o PRV revelam que a abordagem heurística proposta é capaz de produzir boas soluções para as instâncias analisadas. Em 12 das 15 instâncias comparadas, os números de veículos obtidos pela heurística foram menores ou iguais aos números de veículos das melhores soluções (com três entregadores em todas as rotas) obtidas pelo modelo no tempo limite de uma hora de execução.

Agradecimentos

Esta pesquisa teve o apoio do CNPq (processos 303001/2009-7 e 522973/95-4).

Referências

- Bard, J., Kontoravdis, G., Yu, G.** (2002), A branch-and-cut procedure for the vehicle routing problem with time windows, *Transportation Science*, 36:2, 250-269.
- Bräysy, O. e Gendreau, M.** (2005a), Vehicle routing problem with time windows, part I: Route construction and local search algorithms, *Transportation Science*, 39:1, 104-118.
- Bräysy, O. e Gendreau, M.** (2005b), Vehicle routing problem with time windows, part II: Metaheuristics, *Transportation Science*, 39:1, 119-139.
- Bodin, L., Golden, B., Assad, A. e Ball, M.** (1983), Special issue – Routing and scheduling of vehicles and crews – the state of the art, *Computers & Operations Research*, 10.
- Brooke, A., Kendrick, D. e Meeraus, A.** (1992), *GAMS: a user's guide* (release 2.25). San Francisco: The Scientific Press.
- Cordeau, J-F, Desaulniers, G., Desrosiers, J., Solomon, M. M. e F. Soumis** (2002), The VRP with time windows, em Toth, P., e Vigo, D. (Eds.), *The vehicle routing problem. SIAM, Monographs on Discrete Mathematics and applications*, Philadelphia, PA, SIAM, 157-193.
- Desrosiers, J., Dumas, Y., Solomon, M. e Soumis, F.** (1995). Time constrained routing and scheduling, em Ball, M., T. Magnanti, L., Monma, C. L. e Nemhauser, G. L. (Eds.), *Network routing. Handbooks in operations research and management science*, Amsterdam, North-Holland, 35-139.
- França, P. M., Sosa, N. M. e Pureza, V.** (1999), An adaptive tabu search algorithm for the capacitated clustering problem, *International Transactions in Operational Research*, 6, 665-678.
- Ferreira, V. O.** (2010), Uma Abordagem Heurística para o Problema de Roteamento de Veículos com Designação de Entregadores Extras, *Qualificação de Mestrado*, Programa de Pós Graduação – Departamento de Engenharia de Produção, UFSCar.
- Fischer, M. L.** (1995), Vehicle routing, em Ball, M., T. Magnanti, L., Monma, C. L. e Nemhauser, G. L. (Eds.), *Network routing. Handbooks in operations research and management science*, Amsterdam, North-Holland, 1-33.
- Glover, F. e Laguna, M.** (1997), *Tabu Search*. Kluwer Academic Publishers, Massachusetts.
- Laporte, G.** (1992), The vehicle routing problem: an overview of exact and approximate algorithms, *European Journal of Operational Research*, 59, 345-358.
- Montané, F.** (2007), An adaptive tabu search algorithm for the mix fleet vehicle routing problem, *Anais do 7th Metaheuristics International Conference*, Montreal, Canadá
- Osman, I. H.** (1993), Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem, *Annals of Operations Research*, 41, 421-452
- Pureza, V.** (2006), An adaptive tabu search route reduction procedure for the vehicle routing problem with time windows, *Anais do 7th EU/MEeting on Adaptive, Self-Adaptive, and Multi-Level Metaheuristics*, 2006, Málaga, Espanha, 1, 25-28.
- Pureza, V. e França, P.** (2001), Uma abordagem Adaptativa de Busca Tabu aplicada ao Problema de Roteamento de Veículos, *Transportes*, 9:2, 28-47.

Pureza, V. e Morabito, R. (2008). Desenvolvimento de um modelo matemático e um protótipo computacional para planejamento de rotas de atendimento. *Relatório Técnico do Projeto Transmute 863*04*, FAI-UFSCar, São Carlos.

Solomon, M. (1987), Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints, *Operations Research*, 35:2, 254-265.

Toth, P. e Vigo, D. (Eds.) (2002), *The vehicle routing problem*. SIAM, *Monographs on Discrete Mathematics and applications*, Philadelphia, PA, SIAM, 157-193.