

UM ESTUDO DE CASO APLICADO AO PROBLEMA REAL DE LARGA ESCALA DE ROTEAMENTO DE VEÍCULOS COM FROTA HETEROGÊNEA E MÚLTIPLAS VIAGENS

Vitor Nazário Coelho¹, Helena Ramalinho Lourenço², Alex Grasas²,
Igor Machado Coelho⁴, Marccone Jamilson Freitas Souza³ e Raphael Carlos Cruz¹

¹ Departamento de Engenharia de Controle e Automação, UFOP, Ouro Preto/MG, Brazil

² Departamento de Economia e Administração, UPF, Barcelona, Espanha

³ Departamento de Ciências da Computação, UFOP, Ouro Preto/MG, Brazil

⁴ Instituto de Computação, UFF, Niterói/RJ, Brazil

vncoelho@gmail.com, helena.ramalinho@upf.edu, alex.grasas@upf.edu

imcoelho@ic.uff.br, marccone@iceb.ufop.br e phaelcarlos@gmail.com

Resumo. Este trabalho tem seu foco no problema de roteamento de veículos (PRV) com frota heterogênea e múltiplas viagens. Dada a natureza combinatória do problema, que praticamente inviabiliza a utilização exclusiva de métodos de programação matemática, dito exatos, a solução de casos reais normalmente é feita por meio de métodos heurísticos. Os problemas reais de roteamento de veículos envolvem a utilização de diferentes tipos de caminhões e, em alguns casos, suporte para múltiplas rotas. Como contribuição para esse campo de pesquisa, descreve-se neste trabalho um algoritmo, inspirado na heurística Iterated Local Search, aplicado a um problema real de roteamento de veículos, considerando uma frota heterogênea e múltiplas viagens opcionais. O estudo de caso é baseado em um conjunto de instâncias de uma empresa europeia de grande porte, envolvendo aproximadamente 400 clientes. Resultados computacionais comprovam a eficiência do algoritmo proposto, mostrando-se capaz de reduzir os gastos da empresa em até 51100,00 euros.

PALAVRAS-CHAVE: Roteamento de Veículos com Frota Heterogênea, Múltiplas Viagens, Aplicações Reais e Iterated Local Search.

Abstract. This paper has its focus on the vehicle routing problem (VRP) with heterogeneous fleet multiple trips. Given the combinatorial nature of the problem, which practically precludes the exclusive use of mathematical programming methods, said exact, solution of real cases is typically done through heuristic methods. The real vehicle routing problems involve the use of different types of trucks and, in some cases, support for multiple routes. As a contribution to this field of research, this paper describes an algorithm inspired on the Iterated Local Search heuristic applied on a real-life case problem, considering a heterogeneous fleet with optional multiple trips. The case study is based on a set of instances provided by a large European company, involving approximately 400 stores. Computational results show the effectiveness of the proposed algorithm, being able to reduce company expenses by up to 51100,00 EUR.

KEYWORDS: Heterogeneous Fleet Vehicle Routing Problem, Multiple Trips, Real-life applications and Iterated Local Search.

1 Introdução

No problema de roteamento de veículos (PRV) procura-se determinar as rotas de menor custo de entrega, desde um depósito até um conjunto de clientes. Esta classe de problemas é essencial nos setores distribuição e logística, devido à importância da obtenção de rotas eficientes para reduzir custos e aumentar a satisfação do cliente.

Uma versão clássica é o PRV capacitado, proposto por Dantzig e Ramser (1959). Neste problema, existe um conjunto de clientes que devem ser atendidos, minimizando a distância total e respeitando a restrição da capacidade do caminhão. Embora este problema tenha sido estudado há mais de cinco décadas (Laporte, 2009), as aplicações reais apresentam uma variedade de restrições e regras operacionais que devem ser levadas em consideração em qualquer implementação focada em problemas reais da classe do PRV. Estas restrições adicionais e as regras de operação podem ser relacionadas com os clientes, depósitos, veículos e, além de tornar o problema mais complexo, implicam em um impacto significativo sobre as soluções. No presente trabalho, propõe-se tratar uma variante do PRV baseada em uma aplicação real de uma importante empresa de distribuição que atende mais de 380 clientes no nordeste da Espanha. Em particular, a variante abordada tem como objetivo principal o tratamento das restrições com relação aos veículos, em relação ao tipo e capacidade, bem como a possibilidade da realização de múltiplas viagens. Deste modo, designa-se o problema descrito aqui como o Problema de Roteamento de Veículos com Frota Heterogênea e Múltiplas Viagens (PRVFH MV). Este problema também é conhecido na literatura como *Heterogeneous Fleet Multitrip VRP* (Prins, 2002), pertencente à classe \mathcal{NP} -Difícil, e, como tal, métodos exatos de solução têm aplicabilidade restrita. A abordagem mais comum é por meio de procedimentos heurísticos.

O Problema de Roteamento de Veículos com Frota Heterogênea ainda pode ser dividido em classes que dizem respeito à frota ser limitada ou ilimitada, e também quanto aos custos dos veículos, que podem ser fixos ou variáveis (Penna *et al.*, 2013). Quando a frota é limitada, o número de veículos e a capacidade de cada veículo são conhecidos previamente, cabendo ao sistema encontrar rotas para os clientes utilizando os veículos disponíveis. No caso da frota ilimitada não se sabe a priori a quantidade necessária de veículos para atender a demanda dos clientes, assim a frota deve ser dimensionada levando em conta a capacidade e custos de cada veículo.

A primeira aplicação do VRPFH na literatura envolvia uma variante com frota ilimitada e apenas custos fixos (Golden *et al.*, 1984). Foram desenvolvidas duas heurísticas para resolver o problema. A que obteve os melhores resultados era baseada na heurística clássica de Clarke e Wright (1964), conhecida como heurística das economias. Foi, também, proposta uma formulação matemática para este problema e para a variante com custos dependentes, os primeiros limites inferiores foram apresentados.

Gendreau *et al.* (1999) sugerem um algoritmo baseado em Busca Tabu (BT) incorporado ao procedimento GENIUS, considerando custos variáveis, porém, utilizando apenas problemas planos, onde os nós estão localizados em um plano euclidiano. O número de veículos é parcialmente ilimitado, sendo realizada uma aproximação quando o algoritmo inicia sua execução. Mesmo nesse contexto, o Problema de Roteamento com Frota Mista apresentado ainda não é muito realístico, misturam-se os custos de investimento a médio prazo, com custos operacionais de curto prazo que flutuam de acordo com os clientes específicos de um dia.

O VRPFH já foi tratado na literatura por abordagens exatas e heurísticas. Choi e Tcha (2007) obteve limites inferiores para todas as variantes do problema ilimitado a partir de uma abordagem utilizando geração de colunas baseada em uma formulação do problema de cobertura de conjuntos, do inglês *set covering*. Baldacci e Mingozzi (2009) propuseram uma variante baseada no problema de particionamento de conjuntos, do inglês *set partitioning*), que utilizava os limites fornecidos por um procedimento baseado em relaxação linear e Lagrangeana, resolvendo as principais variantes do problema, envolvendo o problema com frota limitada e ilimitada, com custos variáveis e dependentes. O método proposto foi capaz de resolver instâncias com até 100 clientes, apresentando-se como o estado-da-arte para algoritmos exatos aplicados ao problema.

Entre as abordagens heurísticas apresentadas na literatura, destacam-se as baseadas em Algoritmos Evolucionários. Ochi *et al.* (1998) desenvolveram um algoritmo que combina o Algoritmo Genético (AG) com Scatter Search (Glover *et al.*, 2003) para resolver a variante do problema com frota limitada e com custos fixos. Um AG híbrido com procedimento de busca local foi proposto por Liu *et al.* (2009). Em Moscato (2003) foi proposto um Algoritmo Memético (AM) para a variante com custos fixos e dependentes. Prins (2009) apresenta dois AM para resolver o problema. O primeiro deles utiliza um procedimento *giant tour* como codificador dos cromossomos e um procedimento de *split* que realiza a distribuição ótima dos veículos e rotas, sujeito à sequência de visitação na *giant tour*. O segundo algoritmo utiliza estratégias de cálculo de distância de forma a diversificar a busca no espaço de soluções.

Um estudo geral do VRPFH contendo as variantes aqui mencionadas pode ser encontrado em Baldacci *et al.* (2008); Penna *et al.* (2013).

Prins (2002) trata um problema real de grande porte, denominado *Heterogeneous Fleet Multitrip VRP with Application to a Large-Scale Real Case*, com algumas similaridades à variante tratada neste trabalho, associado a um fabricante de móveis francês com 775 lojas. Um algoritmo baseado em Busca Tabu foi desenvolvido a partir de uma abordagem bi-objetivo simples. Considerou-se o tempo de viagem do veículo até o destino, sendo o objetivo principal minimizar a duração total de todas as viagens e o objetivo secundário de minimizar o número de caminhões. Os resultados computacionais validaram a proposta e superaram as soluções utilizadas pela empresa.

No presente trabalho, propõe-se resolver uma variante do problema com frota limitada passível de múltiplas viagens, considerando custos fixos e variáveis. Essa variante revela-se de grande importância para o setor de logística e transporte, buscando-se utilizar a frota disponível da forma mais eficiente possível. Ressalta-se também a restrição de docagem, com a qual determinados clientes não podem ser atendidos por certos tipos de veículos.

No presente trabalho é desenvolvido um algoritmo heurístico baseado nos procedimentos *Iterated Local Search* – ILS (Lourenço *et al.*, 2003), *Variable Neighborhood Descent* – VND (Mladenovic e Hansen, 1997) e *Greedy Randomized Adaptive Search Procedures* – GRASP (Resende e Ribeiro, 2010). Esses métodos foram escolhidos devido aos bons resultados apresentados na literatura, como em Penna *et al.* (2013); Souza *et al.* (2010). Usando essa metodologia, espera-se resolver o PRVFH de forma eficiente e com um razoável tempo computacional, atendendo-se assim, às exigências operacionais da empresa.

O restante deste trabalho está organizado como segue. A Seção 2 define o PRVFHMV. A Seção 3 detalha a metodologia utilizada na resolução deste problema. A Seção 4 mostra os resultados dos experimentos computacionais e a Seção 5 apresenta as considerações finais.

2 Definição do problema

A variante do PRVFHMV descrita neste trabalho pode ser definido sobre um grafo não orientado $G = (V, E)$, em que $V = \{0, 1, \dots, n\}$ representa os vértices do grafo. O depósito é representado pelo valor inteiro 0 e os nós $i \in V \setminus \{0\}$ representam os clientes, cada qual com sua demanda d_i , não negativa. Cada aresta $E = \{(i, j) | i, j \in V, i < j\}$ possui um custo associado c_{ij} , também não negativo. A frota é composta por m diferentes tipos de veículos $T = 1, \dots, m$. Para cada $t \in T$ existe um número m_t de veículos disponíveis com capacidade q_t , custo fixo cf_t e custo variável cv_t dependente da “distância” percorrida c_{ij} .

Seja $r^t = (v_0, v_1, \dots, v_{n(r)+1})$ uma rota válida associada ao veículo $t \in m_t$, onde $v_0 = v_{n(r)+1} = 0$ e $n(r)$ é definido como o número de clientes visitados em r^t . Define-se:

- Demanda total da rota: $Q_r = \sum_{i=v_0}^{v_{n(r)}} d_{v_i}$;
- Custo da rota: $C_r^t = cf_t + (\sum_{i=v_0}^{v_{n(r)}} c_{ij}) \times cv_t$, onde $j = i + 1$;
- Espaço vazio da rota: $CE_r^t = q_t - Q_r$.

Esse conjunto de elementos caracterizam o PRVFHMV. Nesse problema busca-se construir um conjunto de rotas válidas $s = (r_1^{t_1}, \dots, r_n^{t_x})$, onde $n < (2 \times m)$, de forma a minimizar o somatório dos custos totais e o espaço vazio associado a cada rota válida, $Min \sum_{r=r_1}^{r_n} (\alpha \times C_r^t + \beta \times CE_r^t)$, sendo que uma rota válida deve satisfazer os seguintes critérios:

1. Cada rota deve iniciar e terminar no depósito;
2. Cada cliente é visitado exatamente por uma rota;
3. Cada cliente deve possuir compatibilidade com o veículo t alocado para essa rota em que está inserido, ou seja, $(\forall v_i \in r^t : comp(t, v_i) = 1)$, onde $comp(t, v_i)$ possui o valor 1 caso se o veículo t pode atender o cliente v_i (0 caso contrário).
4. O somatório das demandas dos clientes pertencentes a uma rota não deve exceder a capacidade máxima do veículo t associado a essa rota ($q_t \leq Q_r$);
5. Alguns tipos de veículos (ou todos) podem atender uma ou duas rotas.

3 Metodologia

3.1 Algoritmo proposto

O algoritmo proposto, denominado GILS-VND consiste na combinação dos procedimentos heurísticos ILS (Lourenço *et al.*, 2003), GRASP (Feo e Resende, 1995; Resende e Ribeiro, 2008) e VND (Hansen *et al.*, 2008).

O pseudocódigo do algoritmo GILS-VND está esquematizado no Algoritmo 1. Neste algoritmo, $GRASPmax$ representa a quantidade de iterações em que a fase de construção GRASP é aplicada e $IterMax$ indica o número máximo de iterações realizadas em um dado nível de perturbação.

Algoritmo 1: GILS-VND

Entrada: γ , *GRASPmax*, *IterMax*, Função $f(\cdot)$
Saída: Solução s

- 1 $s_0 \leftarrow$ Melhor Solução em *GRASPmax* iterações do procedimento *ConstroiSolucaoEconomias*(γ)
- 2 $s^* \leftarrow$ VND(s_0, f)
- 3 $p \leftarrow 0$
- 4 **enquanto** critério de parada não satisfeito **faça**
- 5 $iter \leftarrow 0$
- 6 **enquanto** $iter < IterMax$ e critério de parada não satisfeito **faça**
- 7 $s' \leftarrow$ Refinamento(s^*, p, f)
- 8 **se** s' for melhor que s^* de acordo com a função f **então**
- 9 $s^* \leftarrow s'$;
- 10 $p \leftarrow 0$;
- 11 $iter \leftarrow 0$
- 12 **fim**
- 13 **senão**
- 14 $iter \leftarrow iter + 1$
- 15 **fim**
- 16 **fim**
- 17 $p \leftarrow p + 1$
- 18 **fim**
- 19 **retorna** s

A solução inicial (linha 1 do Algoritmo 1) é gerada aplicando-se a fase de construção do procedimento GRASP, a partir do procedimento *ConstroiSolucaoSavings*. A solução é gerada com base no procedimento de Economias – *Savings*, proposto por Clarke e Wright (1964), adaptado para o PRVFHVM.

Para cada tipo de veículo $t \in M$ (conforme descrito na Seção 2), o procedimento das *Economias* é executado. Em cada iteração do procedimento GRASP a ordem desses veículos é aleatória. Uma lista restrita de candidatos é criada ordenando as rotas que possuem a demanda total mais próxima da capacidade do veículo do tipo t . O valor do parâmetro γ define o tamanho dessa lista restrita de candidatos, conforme a ideia básica do procedimento GRASP. Todas as m_t melhores rotas são alocadas à solução corrente. Caso ainda reste algum cliente não alocado, um novo procedimento de Economias é executado para os veículos que podem realizar múltiplas viagens.

A busca local é feita pelo procedimento VND (apresentado no Algoritmo 3), utilizando as estruturas de vizinhança descritas nas Seções 3.3 e 3.4. As linhas 13 e 18 acionam o procedimento *setLocalOtimo*, que marca as vizinhanças como “ótimos-locais”, essa marcação é utilizada pela estrutura de dados auxiliares (descrita na Seção 3.2).

O procedimento Refinamento (linha 7 do Algoritmo 1) é apresentado no Algoritmo 2. Na linha 2 do Algoritmo 2 uma das estruturas de vizinhanças r^{pert} descrita na Seção 3.5 é selecionada aleatoriamente pelo procedimento *SelecionaVizinhança*. Em seguida, a linha 3 realiza uma perturbação na solução corrente de acordo com essa estrutura selecionada.

Assim como em Souza *et al.* (2010), o algoritmo de Refinamento apresentado neste trabalho possui diversos níveis de perturbação. Caso o a solução fique *iterMax* iterações sem apresentar nenhuma melhora (linha 6 do Algoritmo 1), a variável p é incrementada (linha 17). Essa variável regula a intensidade da perturbação (linha 1 do Algoritmo 2), em um determinado nível de perturbação, têm-se $p + 2$ movimentos aleatórios (*shakings*) aplicados na solução corrente.

Algoritmo 2: Refinamento

Entrada: r^{pert} vizinhanças de perturbação em ordem aleatória (ver Seção 3.5)

Entrada: Solução inicial s , Nível p e Função $f(\cdot)$

Saída: Solução s

```

1 para  $i \leftarrow 1$  até  $p + 2$  faça
2   |  $k \leftarrow$  SeleccionaVizinhança( $r^{pert}$ )
3   |  $s' \leftarrow$  Perturbação( $s, k$ )
4 fim
5  $s \leftarrow$  VND( $s', f$ )
6 retorna  $s$ 

```

Algoritmo 3: VND

Entrada: r^{Intra} vizinhanças intra-rotas em ordem aleatória (ver Seção 3.3)

Entrada: r^{Inter} vizinhanças inter-rotas em ordem aleatória (ver Seção 3.4)

Entrada: Solução s_0 e Função $f(\cdot)$

Saída: Solução s de qualidade possivelmente superior à solução inicial s_0 de acordo com a função f

```

1  $s \leftarrow s_0$ 
2  $k^{Inter} \leftarrow 1$ 
3 enquanto  $k^{Inter} \leq r^{Inter}$  faça
4   | Encontre o melhor vizinho  $s' \in N^{(k^{Inter})}(s)$ 
5   | se  $f(s') < f(s)$  então
6     |  $s \leftarrow s'; k^{Inter} \leftarrow 1; k^{Intra} \leftarrow 1$ 
7     | enquanto  $k^{Intra} \leq r^{Intra}$  faça
8       | Encontre o melhor vizinho  $s' \in N^{(k^{Intra})}(s)$ 
9       | se  $f(s') < f(s)$  então
10        |  $s \leftarrow s'; k^{Intra} \leftarrow 1$ 
11        | fim
12        | senão
13          | setLocalOtimo( $s, k$ );  $k^{Intra} \leftarrow k^{Intra} + 1$ 
14          | fim
15        | fim
16      | fim
17      | senão
18        | setLocalOtimo( $s, k$ );  $k^{Inter} \leftarrow k^{Inter} + 1$ ;
19      | fim
20 fim
21 retorna  $s$ 

```

3.2 Estruturas de Dados Auxiliares (ADSs)

De forma a intensificar e otimizar a busca nas estruturas de vizinhanças, algumas ADSs foram desenvolvidas. Foram utilizadas quatro das sete estruturas apresentadas por Penna *et al.* (2013). Uma breve descrição é apresentada a seguir, considera-se $\#nRotas$ como o número total de rotas e $\#nVizinhanças$ indica o número de estruturas de vizinhanças (apresentadas nas Seções 3.3 e 3.4):

- SomaDemanda[#nRotas] – soma das demandas. Este vetor auxiliar guarda, para cada rota, o somatório da demanda de todos os clientes associados a ela. Ex: SomaDemanda[12] = 230, indica que a rota 12 possui um demanda total de 230, logo, a capacidade do veículo deve ser maior ou igual a este valor.
- MinDemanda[#nRotas] – demanda mínima. Indica a demanda mínima de cada rota. Ex: MinDemanda[10] = 30, indica que 30 é a menor demanda associada a rota 10.

- $MaxDemanda[\#nRotas]$ – demanda máxima. Ex: $MaxDemanda[19] = 90$, indica que a maior demanda da rota 19 é 90.
- $StatusVizinhanca[\#nVizinhancas][\#Rotas]$ – indica se a vizinhança está em um ótimo local. Após uma aplicação completa de todas estruturas de vizinhança por um método de busca-local, todas as rotas são marcadas como “ótimos-locais”. Quando a solução é perturbada (linha 3 do Algoritmo 2), os marcadores de “ótimos-locais” são retirados das rotas que foram alteradas nessa perturbação.

3.3 Estruturas de vizinhança intra-rotas

Para explorar o espaço de soluções do problema, aplicam-se, neste trabalho, cinco diferentes estruturas de vizinhança, três deles são apresentados a seguir e os outras duas são apresentados na Seção 3.4. É importante destacar que não são permitidos movimentos que conduzam a soluções inviáveis.

3.3.1 Movimento 2-opt

O 2-Opt consiste em remover dois arcos não adjacentes e inserir dois novos arcos, de forma que uma nova rota seja formada. A Figura 1 exemplifica a aplicação desse movimento, mostra a remoção dos arcos (1, 6) e (5, 11) da rota 2 e a inserção dos arcos (1, 11) e (6, 5). Ressalta-se que, neste caso ocorreu uma inversão da entre os clientes 6, 16 e 11. A distância total percorrida entre esses clientes não é alterada, caso o problema tratado seja problema simétrico.

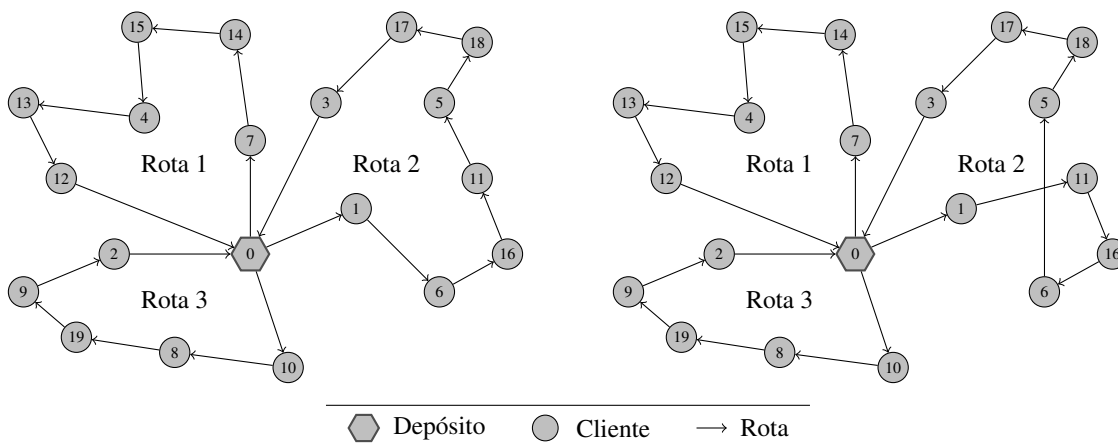


Figura 1. Exemplo do movimento 2-Opt.

3.3.2 Movimento Or-optk

O movimento *Or-optk* consiste em remover k clientes consecutivos de uma rota r e, em seguida, reinseri-los em uma outra posição nessa mesma rota. O valor de k é um parâmetro do movimento. Esse movimento é uma generalização do movimento *Or-opt* proposto por Or (1976), em que é realizado a remoção de no máximo três clientes consecutivos. Quando $k = 1$, a literatura nomeia esse movimento como Reinscrição (Silva *et al.*, 2012).

3.3.3 Movimento *Exchange*

O movimento *Exchange* consiste em trocar um cliente i de uma rota r com um outro cliente j dessa mesma rota.

3.4 Estruturas de vizinhança inter-rotas

3.4.1 Movimento *Shift(1,0)*

O *Shift(1,0)* é um movimento de realocação que consiste em transferir um cliente i de uma rota para outra. A Figura 2 ilustra um exemplo em que o cliente 6 é transferido da Rota 2 para posição 1 da Rota 3.

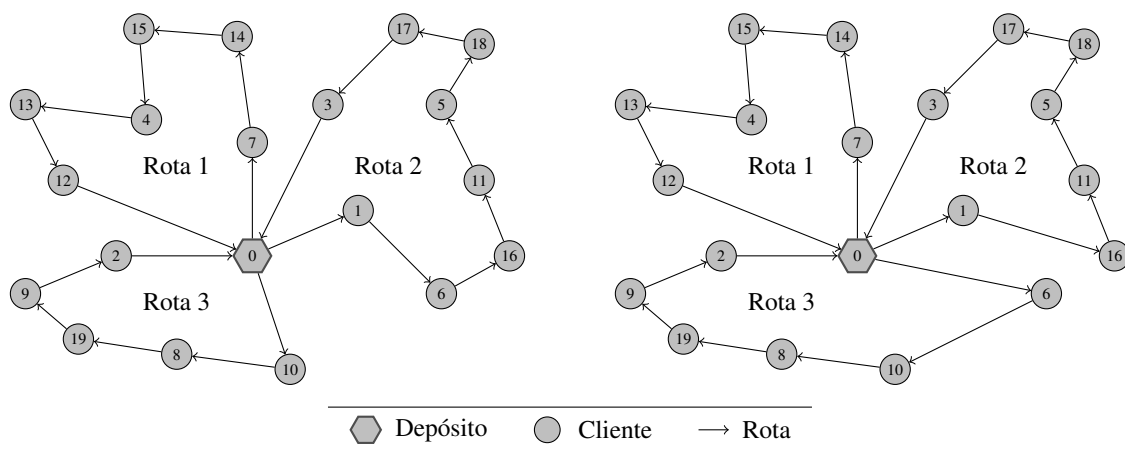


Figura 2. Exemplo do movimento *Shift(1,0)* (Cruz *et al.*, 2012).

3.4.2 Movimento *Swap(1,1)*

O movimento *Swap(1,1)* consiste em trocar um cliente i de uma rota r_p com um outro cliente j de uma rota r_q .

3.5 Mecanismo de perturbação

Todas as estruturas de vizinhança descritas nas Seções 3.3 e 3.4 foram utilizadas como estratégia de perturbação. A aplicação dos movimentos ocorre de forma aleatório e nenhuma verificação de melhoria no custo da função objetivo é observada, sendo esse mecanismo utilizado como estratégia fundamental na diversificação da busca no espaço de soluções. Após a aplicação de um movimento, o vetor StatusVizinhanca é modificado, como descrito na Seção 3.2.

4 Experimentos computacionais

O algoritmo proposto GILS-VND foi implementado em C++ com auxílio do *framework* OptFrame 1.5¹ (Coelho *et al.*, 2011, 2010). O framework OptFrame vem sendo aplicado com sucesso na literatura, como pode ser observado em Souza *et al.* (2010); Coelho *et al.* (2012).

¹Disponível em <http://sourceforge.net/projects/optframe/>

Os testes foram executados em um microprocessador Pentium Core 2 Quad(Q6600), 2.4 GHZ com 8 GB de RAM, utilizando o sistema operacional o Ubuntu Linux 10.04 Kernel 2.6.32-33, compilado com o g++ 4.5.2 e utilizando a IDE do Eclipse 3.1.

Foram utilizadas cinco instâncias baseadas no dados reais fornecidos pela empresa europeia de distribuição. Todas as instâncias possuem 382 clientes. Em cada caso, varia-se a demanda associada a cada um destes clientes. Duas dessas instâncias (PRVFH MV_1MT e PRVFH MV_2MT), possuem a demanda total maior que a capacidade de todos os veículos, logo, para essas instâncias, não é possível atender os clientes sem realizar múltiplas viagens. A Tabela 1 apresenta as características frota de veículos utilizadas, bem como o número de viagens permitidas para cada tipo de veículo (coluna “# MV permitidas”). A disponibilidade dos veículos é apresentada na coluna “Disponibilidade (m_t)” e a capacidade teórica é apresentada na coluna “ $CapTeorica (q_t)$ ”.

Tabela 1. Composição da frota de veículos da empresa de distribuição

Veículo	Disponibilidade (m_t)	$CapTeorica (q_t)$	# MV permitidas
A	8	222	1
B	5	414	1
C	139	482	2
D	3	550	1
E	6	616	1
F	3	676	1
G	4	752	1
H	1	1210	1

4.1 Resultados detalhados

Foi realizada uma bateria de teste composta de 10 execuções de 10 minutos. As Tabelas 2 e 3 apresentam as melhores soluções obtidas pelo algoritmo GILS-VND comparadas as soluções geradas e utilizadas pela empresa, foco deste estudo de caso.

Os custo fixos mencionados na Seção 2 foram divididos em três variáveis, CF , CB e CC . A variável CF mensura o custo por utilização do veículo, a variável CB indica a capacidade teórica q_t multiplicado pelo valor 0,0115€, por fim, o custo CC relaciona o número de clientes atendidos multiplicado por 8€(fixo em cada instância). A variável CE mensura a soma de todas as diferenças entre a capacidade real e o número de caixas que o veículo está transportando, ou seja, essa variável mensura o espaço vazio em cada veículo. A Tabela 2 apresenta esses quatro valores, bem como o número de viagens realizadas (coluna #NV).

A Tabela 3 apresenta os custos por deslocamento e os custos totais. O custo de viagem é associado a variável CV , representado a distância que o veículo percorreu multiplicado pelo seu custo variável, conforme descrito na Seção 2. O somatório das distâncias percorridas por cada veículo é apresentado na variável CD . Finalmente, a variável CT indica o custo total, ou seja, $CT = CF + CB + CC + CV$.

Analisando as Tabelas 2 e 3, verifica-se que o algoritmo GILS-VND foi capaz de obter melhores soluções em todas as instâncias utilizadas, encontrando um menor custo em todas as variáveis envolvidas neste estudo de caso.

Ressalta-se a instância PRVFH MV_1MT, em que foi encontrada uma melhora no custo total de 4263,84€. Além disso, foi possível melhorar os custos fixos por utilização

Tabela 2. Comparação de resultados: GILS-VND × Empresa

Instância	GILS-VND	Empresa	GILS-VND	Empresa	GILS-VND	Empresa	GILS-VND	Empresa
	<i>CF</i> (€)	<i>CF</i> (€)	<i>CB</i> (€)	<i>CB</i> (€)	<i>CE</i> (Unit.)	<i>CE</i> (Unit.)	# <i>NV</i>	# <i>NV</i>
SINGLE TRIP								
PRVFHMV_1	9812	10076	6951,13	7146,93	2667	4533	129	134
PRVFHMV_2	4912	5428	3365,31	3688,53	1487	4325	60	71
PRVFHMV_3	5452	6002	3819,54	4063,62	1293	3658	72	80
MULTI-TRIP								
PRVFHMV_1MT	17109	18170	11098,97	12729,51	16870	18837	220	245
PRVFHMV_2MT	14716	15460	9863,12	10841,03	12874	13669	195	208

Tabela 3. Comparação de resultados 2: GILS-VND × Empresa

Instância	Imp.	GILS-VND	Empresa	GILS-VND	Empresa	GILS-VND	Empresa
		<i>CT</i> (€)	<i>CT</i> (€)	<i>CV</i> (€)	<i>CV</i> (€)	<i>CD</i> (Km)	<i>CD</i> (Km)
SINGLE TRIP							
PRVFHMV_1	1981,76	30490,61	32472,37	10783,47	12305,45	32214,01	36733,11
PRVFHMV_2	2204,36	16792,73	18997,09	5987,42	7352,56	18763,86	22184,74
PRVFHMV_3	2267,67	17999,00	20266,67	6191,46	7664,51	18828,65	23350,67
MULTI-TRIP							
PRVFHMV_1MT	4263,84	49802,54	54066,38	17109,89	19990,88	55982,93	60152,18
PRVFHMV_2MT	2663,46	43796,09	46459,55	16120,97	17062,52	48201,43	51332,48

dos caminhões em até 1061,00€, instância PRVFHMV_1MT, e os custos por caixas entregues foi aprimorado em 1630,53€. Além disso, foi possível realizar uma melhoria na utilização do espaços vazios dos veículos, sendo diminuídas até 2838 unidades de espaço vazio (em cada espaço vazio pode ser transportada uma caixa), instância PRVFHMV_2. Por fim, destaca-se o número de viagens necessárias para atender as demandas de cada instância, foi possível poupar 11 veículos na instância PRVFHMV_2 e 25 múltiplas viagens do veículo do tipo *C* na instância PRVFHMV_1MT.

5 Conclusões e trabalhos futuros

Neste trabalho, foi discutido um Problema de Roteamento de Veículo com suposições realistas. Em particular, este trabalho teve seu foco no Problema de Roteamento de Veículos com Frota Heterogênea e Múltiplas Viagens Opcionais (PRVFHMV). Apesar da sua relevância prática, essa variante tem recebido pouca atenção da literatura. Em virtude de sua dificuldade de solução, foi proposto um algoritmo heurístico de busca em trajetória, denominado GILS-VND, que combina o poderio do GRASP com os métodos ILS e VND, percorrendo um espaço de busca de soluções inteiras viáveis.

Para validar o algoritmo proposto, foi utilizado um conjunto de instâncias reais fornecidas por uma empresa de distribuição localizada na península Ibérica. Os resultados computacionais mostraram que o algoritmo heurístico proposto é superior, apresentando melhores soluções do que aquelas geradas e utilizadas pela empresa. Ao longo de um ano completo, seria possível diminuir os gastos da empresa em até 51166,08€, caso as soluções geradas neste trabalho fossem utilizadas.

O algoritmo proposto possui uma calibragem relativamente simples e não requer qualquer processo complexo de sintonia fina. Portanto, ele pode ser empregado em diversas situações cotidianas a partir de pequenos ajustes. Além disso, dado que a tomada de decisão no problema em pauta tem que ser rápida, os resultados encontrados validam a

utilização do algoritmo proposto enquanto ferramenta de apoio à tomada de decisão.

Para trabalhos futuros propõe-se desenvolver novas estruturas de vizinhanças que realoquem clientes consecutivos, mantendo assim, o atendimento aos clientes que estão localizados em cercanias. Propõe-se, também, a adaptação do presente algoritmo para resolver outra variante do PRVFHVMV já presente na literatura, de forma a validar a robustez e flexibilidade do método proposto. Finalmente, propõe-se a implementação de uma versão paralela do algoritmo GILS-VND visando tirar proveito da tecnologia *multi-core*, já presente nas máquinas atuais e de fácil abstração para algoritmos heurísticos.

Agradecimentos

Os autores agradecem ao CNPq (processo 482765/2010-0), FAPEMIG (processo CEX 01201/09 e 357/09), Ciências sem Fronteiras (processo 202380/2012-2), a Universidade Federal de Ouro Preto e a Universitat Pompeu Fabra, pelo apoio ao presente trabalho.

Referências

- Baldacci, Roberto; Battarra, Maria e Vigo, Daniele. (2008). Routing a heterogeneous fleet of vehicles. *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43, p. 3–27. Springer.
- Baldacci, Roberto e Mingozzi, Aristide. (2009). A unified exact method for solving different classes of vehicle routing problems. *Math. Program.*, v. 120, n. 2, p. 347–380.
- Choi, Eunjeong e Tcha, Dong-Wan. July(2007). A column generation approach to the heterogeneous fleet vehicle routing problem. *Comput. Oper. Res.*, v. 34, n. 7, p. 2080–2095.
- Clarke, G. e Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research*, v. 12, n. 4, p. 568–581.
- Coelho, I. M.; Munhoz, P. L. A.; Haddad, M. N.; Coelho, V. N.; Silva, M. M.; Souza, M. J. F. e Ochi, L. S. (2011). A computational framework for combinatorial optimization problems. *VII ALIO/EURO Workshop on Applied Combinatorial Optimization*, p. 51–54, Porto.
- Coelho, I. M.; Ribas, S.; Perche, M. H. P.; Munhoz, P. L. A.; Souza, M.J. F. e Ochi, L. S. (2010). Optframe: a computational framework for combinatorial optimization problems. *XLII Simpósio Brasileiro de Pesquisa Operacional*, p. 1887 – 1898, Bento Gonçalves, RS.
- Coelho, V.N.; Souza, M.J.F.; Coelho, I.M.; Guimaraes, F.G.; Lust, T. e Cruz, R.C. (2012). Multi-objective approaches for the open-pit mining operational planning problem. *Electronic Notes in Discrete Mathematics*, v. 39, n. 0, p. 233 – 240. ISSN 1571-0653.
- Cruz, R. C.; Souza, M. J. F. e Mine, M. T. (2012). Roteamento de veículos com coleta e entrega simultânea: uma abordagem heurística - parte II. Relatório técnico, Universidade Federal de Ouro Preto, Relatório Técnico do Programa Institucional de Bolsas de Iniciação Científica do CNPq - PIBIC/CNPq.
- Dantzig, George B. e Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, v. 6, p. 80–91.
- Feo, T. A. e Resende, M. G. C. (1995). Greedy randomized adaptive search procedures. *Journal of Global Optimization*, v. 6, p. 109–133.
- Gendreau, Michel; Laporte, Gilbert; Musaraganyi, Christophe e ÉricD. Taillard,. (1999). A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Computers & Operations Research*, v. 26, n. 12, p. 1153 – 1173. ISSN 0305-0548.

- Glover, Fred; Laguna, Manuel e Martí, Rafael. (2003). Scatter search. *Advances in Evolutionary Computing: Theory and Applications*, p. 519–537. Springer-Verlag, (2003).
- Golden, Bruce L.; Assad, Arjang A.; Levy, Larry e Gheysens, Filip. (1984). The fleet size and mix vehicle routing problem. *Computers & OR*, v. , p. 49–66.
- Hansen, P.; Mladenovic, N. e Pérez, J. A. M. (2008). Variable neighborhood search. *European Journal of Operational Research*, v. 191, p. 593–595.
- Laporte, Gilbert. November(2009). Fifty Years of Vehicle Routing. *Transportation Science*, v. 43, p. 408–416. ISSN 1526-5447.
- Liu, Shuguang; Huang, Weilai e Ma, Huiming. (2009). An effective genetic algorithm for the fleet size and mix vehicle routing problems. *Transportation Research Part E: Logistics and Transportation Review*, v. 45, n. 3, p. 434 – 445. ISSN 1366-5545.
- Lourenço, H. R.; Martin, O. C. e Stützle, T. (2003). Iterated local search. Glover, F. e Kochenberger, G., editors, *Handbook of Metaheuristics*. Kluwer Academic Publishers, Boston.
- Mladenovic, N. e Hansen, P. (1997). A variable neighborhood search. *Computers and Operations Research*, v. 24, p. 1097–1100.
- Moscato, Pablo. (2003). A gentle introduction to memetic algorithms. *Handbook of Metaheuristics*, p. 105–144. Kluwer Academic Publishers, (2003).
- Ochi, Luiz Satoru; Vianna, Dalessandro Soares; deA. Drummond, Lúcia Maria e Victor, André O. (1998). An evolutionary hybrid metaheuristic for solving the vehicle routing problem with heterogeneous fleet. *EuroGP*, p. 187–195, (1998).
- Or, I. *Traveling salesman-type combinatorial problems and their relation to the logistics of blood banking*. PhD thesis, Northwestern University, USA, (1976).
- Penna, PucaHuachiVaz; Subramanian, Anand e Ochi, LuizSatoru. (2013). An iterated local search heuristic for the heterogeneous fleet vehicle routing problem. *Journal of Heuristics*, v. 19, n. 2, p. 201–232. ISSN 1381-1231.
- Prins, Christian. (2002). Efficient heuristics for the heterogeneous fleet multitrip vrp with application to a large-scale real case. *Journal of Mathematical Modelling and Algorithms*, v. 1, n. 2, p. 135–150. ISSN 1570-1166.
- Prins, Christian. (2009). Two memetic algorithms for heterogeneous fleet vehicle routing problems. *Eng. Appl. of AI*, v. 22, n. 6, p. 916–928.
- Resende, M. G. C. e Ribeiro, C. C. (2008). Greedy randomized adaptive search procedures: Advances and applications. Gendreau, M. e Potvin, J.Y., editors, *Handbook of Metaheuristics*. Springer, 2 edição.
- Resende, M. G. C. e Ribeiro, C. C. (2010). Greedy randomized adaptive search procedures: Advances, hybridizations, and applications. Gendreau, M. e Potvin, J.Y., editors, *Handbook of Metaheuristics*, p. 283–319. Springer, New York, 2 edição.
- Silva, D. M.; Frota, Y. A. M. e Subramanian, A. (2012). Uma heurística para o problema de roteamento de veículos com múltiplas viagens. *XLIV Simpósio Brasileiro de Pesquisa Operacional*, p. 1–12, Rio de Janeiro, RJ.
- Souza, M. J. F.; Coelho, I. M.; Ribas, S.; Santos, H. G. e Merschmann, L. H. C. (2010). A hybrid heuristic algorithm for the open-pit-mining operational planning problem. *European Journal of Operational Research, EJOR*, v. 207, p. 1041–1051.