

O Problema do Caminho Mínimo com Restrições de Recursos para o Problema de Roteamento de Veículos com Frota Heterogênea

Kellen Dayelle Endler (Grupo de Tecnologia Aplicada à Otimização (GTAO) / PPGMNE-UFPR)

kellen_dayelle@hotmail.com

Cassius Tadeu Scarpin (GTAO / PPGMNE-UFPR) cassiusts@gmail.com

Maria Teresinha Arns Steiner (PPGEPS-PUCPR) maria.steiner@pucpr.br

Alexandre Checoli Choueiri (GTAO / PPGMNE-UFPR) alexandrechecoli@gmail.com

Resumo:

O Problema de Roteamento de Veículos (PRV) consiste em um dos temas mais estudados da Pesquisa Operacional. A frota heterogênea é uma generalização natural do problema com vários tipos de veículos, cada um deles definido de acordo com a capacidade, um custo fixo, um custo variável por unidade de distância além de um número limitado ou ilimitado de veículos disponíveis por tipo. Este artigo apresenta o problema do caminho mínimo com restrições de recursos aplicado ao problema de roteamento de veículos com frota heterogênea. Além disso, uma possível implementação é descrita com algoritmos e metodologia de cálculo apresentados de forma didática e detalhada. Algumas adaptações foram realizadas como a permissão de soluções infactíveis incluindo-se penalizações para as mesmas. Essa técnica é utilizada em algoritmos de alta performance que buscam produzir métodos competitivos e escalonáveis. Seu estudo é, portanto, fundamental, visto que uma pequena melhoria nas soluções pode acarretar em significativas consequências econômicas e ambientais na prática.

Palavras chave: Problema do Caminho Mínimo. Problema de Roteamento de Veículos. Frota Heterogênea. Roteirização.

The Minimum Path Problem with Resource Constraints for the Problem of Vehicle Routing with Heterogeneous Fleet

Abstract

The Vehicle Routing Problem (VRP) consists one of the most studied topics of Operational Research. The heterogeneous fleet is a natural generalization of the problem with several types of vehicles, each defined according to capacity, a fixed cost, a variable cost per unit of distance and a limited or unlimited number of vehicles available by type. This paper presents the problem of the minimum path with resource constraints for the VRP with heterogeneous fleet. In addition, a possible implementation is described with algorithms and calculation methodology presented in didactic and detailed form. Some adaptations have been made such as permitting infeasible solutions including penalties for them. This technique is used in high performance algorithms that seek to produce competitive and scalable methods. Its study is therefore fundamental, since a small improvement in solutions can have enormous economic and environmental consequences in practice.

Key-words: Minimum Path Problem. Vehicle Routing Problem. Heterogeneous Fleet. VRP.

1. Introdução

A distribuição física efetiva constitui-se em um papel chave dos sistemas logísticos e envolve o fluxo de produtos desde as fábricas ou centros de distribuição através da rede de transportes para os consumidores. Um estudo realizado pela Fundação Dom Cabral mostra que as companhias gastaram, em média, 12,37% do seu faturamento bruto com custos logísticos no Brasil desde 2015 até o ano de publicação deste artigo (SANTOS, 2018).

A literatura da Pesquisa Operacional aborda este problema como o de Problema de Roteamento de Veículos (PRV) ou *Vehicle Routing Problem* (VRP), um dos temas mais estudados da área. O PRV é um nome genérico para uma classe de problemas de otimização combinatória em que os clientes devem ser atendidos por vários veículos. Os veículos deixam o depósito, servem os clientes na rede e retornam ao depósito após completar suas rotas. Cada cliente é descrito por uma certa demanda.

O PRV clássico, segundo Prins (2004), é definido em uma rede completa não direcionada $G = (V, A)$ com um conjunto de nós $V = \{0, 1, \dots, n\}$ e um conjunto de arestas A . O nó 0 é um depósito com m veículos idênticos de capacidade W , m pode ser definido *a priori* ou deixado como uma variável de decisão. Cada outro nó $i > 0$ representa um cliente com um q_i de demanda não negativa e cada borda (i, j) tem um custo de viagem não negativo $c_{ij} = c_{ji}$.

O PRV consiste em determinar um conjunto de m viagens de custo total mínimo, de modo que cada viagem comece e termine no depósito, cada cliente seja visitado exatamente uma vez e a demanda total tratada por qualquer veículo não exceda a sua capacidade Q . Na prática, os clientes estão espalhados geograficamente por uma grande rede rodoviária e o custo em cada aresta é geralmente medido em distância ou tempo de viagem. Esse problema real pode ser compactado em um Problema do Caminho Elementar Mínimo com Restrições de Recursos entre os clientes mantendo-se os nós de depósitos, usando para sua resolução, o algoritmo de Dijkstra, por exemplo.

Este artigo apresenta o método do Caminho Mínimo para o PRV com frota heterogênea. A importância desse estudo atribui-se primeiramente, a essa técnica fazer parte das estratégias consideradas “vencedoras” utilizadas por algoritmos recentes e de alto desempenho em PRV que combinam múltiplos atributos juntos, que são os que mais se aproximam da realidade. Podemos citar os artigos de Boudia, Prins e Reghioui (2007), Liu, Huang e Ma (2009), Prins (2009), Ngueveu, Prins e Calvo (2010), Vidal et al. (2012), Vidal, et al. (2013), Vidal et al. (2014), para aplicações em Algoritmo Genético (AG), e Duhamel, Potvin e Rousseau (2011) em *Greedy Randomized Adaptive Search Procedure + Evolutionary Local Search* (GRASP + ELS). Estes artigos utilizam a representação de Prins (2004), de um *giant tour* (roteiro gigante) sem delimitadores de rota. Em segundo lugar, a utilização deste método é apresentada de forma integrada e unificada para o PRV com frota heterogênea, o que torna o problema mais aplicável, visto que a maioria das empresas apresenta esta característica. Além disso, uma possível implementação é descrita com algoritmos e metodologia de cálculo apresentados de forma didática e detalhada.

O presente artigo está estruturado da seguinte forma: a seção 2 apresenta conceitos importantes sobre o problema do caminho mínimo sob restrições relacionado ao PRV com frota heterogênea. A seção 3 apresenta a metodologia utilizada, a seção 4 apresenta os algoritmos para a utilização do método do caminho mínimo para PRV com frota heterogênea. Por fim, a seção 5 traz as conclusões e sugere trabalhos futuros.

2. Referencial Teórico

O Problema do Caminho Mínimo (SPP - *Shortest Path Problem*), é provavelmente um dos mais

conhecidos na Teoria dos Grafos. Neste problema, deseja-se encontrar um caminho entre dois vértices dados, de modo que a soma dos valores associados às arestas que o constituem seja a menor possível. Esses valores podem ter vários significados, como o tempo que se leva para percorrer tal caminho, a quantidade de vértices no mesmo ou a distância total (COSTA e LONGO, 2011).

O problema tem uma grande variedade de aplicações, principalmente em transportes e telecomunicações. O SPP tem sido estudado por um longo período, o que possibilitou o surgimento de técnicas eficientes que o solucionam, como os algoritmos de Dijkstra (1959), Bellman (1958), Ford e Randolph (1956), Floyd (1962) e Ulusoy (1985) os quais tem complexidade de tempo polinomial no número de vértices e arestas.

A adição de restrições de recursos ao SPP pode aumentar bastante a dificuldade de resolução do mesmo, geralmente transformando-o em um problema NP-hard. Um exemplo consiste no Problema do Caminho Mínimo com Restrição de Capacidade (SPPCC - *Shortest Path Problem with a Capacity Constraint*), no qual são associados valores aos vértices e a soma dos valores associados aos vértices que fazem parte da solução não podem ser superior a um certo limite.

Esse problema foi utilizado por Prins (2004), que apresentou um AG híbrido para solucionar o PRV que se mostrou superior a todos os outros AGs até então publicados, comparando-se as instâncias de Christofides (1979) e Golden et al. (1998). Um dos fatores fundamentais para o sucesso de seu algoritmo se deve a codificação do cromossomo sem delimitadores de rotas (vide Figura 1) que permite a utilização do operador de *crossover* OX (*order crossover*), que é o método de cruzamento utilizado com mais sucesso nos AGs aplicados a problemas do tipo caixeiro-viajante e de roteamento de veículos segundo Prins (2004).

Outro fator importante foi a utilização de uma implementação eficiente de um algoritmo que avaliasse o caminho mínimo com restrições de carga e duração de rota. Esse método de particionamento do *roteiro gigante* – chamado de *Split procedure* – garante que, para a sequência de clientes determinada pelo cromossomo, seja obtida a rota ótima.

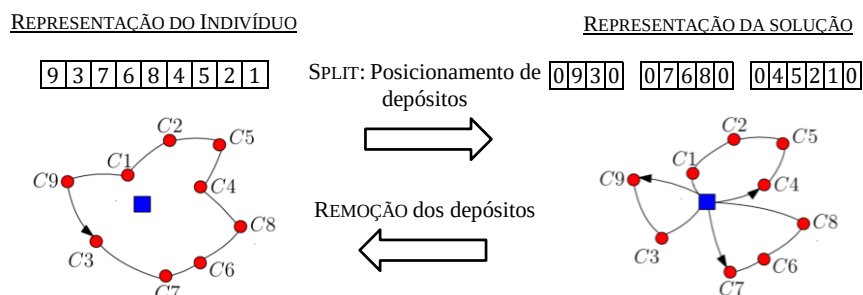


Figura 1 – Representação do indivíduo da solução segundo Prins (2004). Fonte: Os autores.

Vale mencionar que anteriormente, autores como Beasley (1983) e Schmitt (1995) haviam trabalhado com representações sem delimitadores de rotas. Os resultados do primeiro autor, no entanto, nunca superaram os das mais tradicionais heurísticas. Schmitt elaborou um AG em que as rotas são criadas por cortes sequenciais da solução. A diferença deste algoritmo, para o algoritmo *Split* apresentado por Prins (2004) é que neste, mesmo que um cliente possa ser inserido na rota anterior (pela capacidade do veículo ou duração de rota não terem ultrapassado limites permitidos), ele também pode iniciar uma nova rota, caso isto reduza o custo total. Ou seja, *Split* trata-se de procedimento que resulta em um roteiro ótimo de determinada sequência de pontos de entregas.

O exemplo da Figura 2 mostra a diferença no funcionamento do método de particionamento proposto por Prins (2004) com o de Schmitt (1995) aplicado ao PRV clássico. Supondo que os

veículos que estão no depósito possam atender no máximo três clientes, e estão em ordem 1, 2, 3, 4 e 5. O algoritmo de Schmitt (1995) formaria uma rota com os pontos 1, 2 e 3 e uma segunda rota com os pontos 4 e 5, conforme mostra a Figura 2(a). Já o algoritmo de Prins (2004) não incluiria o ponto 3 na primeira rota, já que este está localizado mais próximo dos pontos 4 e 5, construindo rotas mais econômicas Figura 2(b).

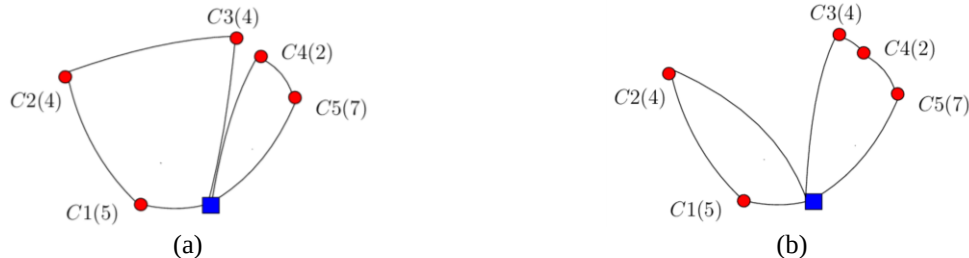


Figura 2 – (a) Rotas formadas por Schmitt (1995). (b) Rotas formadas pelo método proposto por Prins (2004).
Fonte: Adaptado de: Araújo (2008).

O PRV com frota heterogênea (HVRP – *Heterogeneous Vehicle Routing Problem*), por sua vez, é uma generalização do PRV clássico, por uma disponibilidade limitada a_k para cada tipo de veículo k . Cada um deles definidos com uma determinada capacidade, um custo fixo e um custo por unidade de distância. O PRV com frota mista de veículos (VFMP - *Vehicle Fleet Mixed Problem*) é compatível com um número ilimitado de veículos por tipo. Assim, pode ser visto como um HVRP particular com $a_k = n$, uma vez que há no máximo n viagens (com um cliente em cada uma). O VFMP e o HVRP são obviamente *NP-hard*, já que adicionam o PRV a um caso especial.

Golden, et al. (1984) apresentaram o conjunto de instâncias, chamado de “VFMP-F”. Esse conjunto apresenta 20 instâncias com 12 até 100 clientes e de 3 até 6 tipos de veículos. Devido ao fato das instâncias 1, 2, 7 e 12 possuírem somente a matriz de distância entre os clientes e não suas coordenadas, muitos autores passaram a utilizar somente as instâncias com coordenadas em seus experimentos. TAILLARD (1999) apresentou as variações “VFMP-V”, “VFMP-FV” e “HVRP” a partir do conjunto de Golden, et al. (1984). Como afirma Vieira (2017), as instâncias do HVRP possuem a seguinte nomenclatura:

- _ “HVRP” - indica que é um problema de frota fixa e sem custo fixo de utilização;
- _ “VFMP” - indica que é um problema de dimensionamento de frota;
- _ “F” - indica que cada tipo de veículo possui somente custo fixo de utilização e custo unitário de deslocamento;
- _ “V” - indica que cada tipo de veículo não possui custo fixo de utilização, mas somente custo de deslocamento; e
- _ “FV” - indica que cada tipo de veículo possui ambos os custos de utilização e de deslocamento.

O VFMP (com número de veículos de cada tipo não limitado) aparece em situações em que a frota ainda não foi comprada: combina decisões táticas (selecionando o número de veículos a serem adquiridos) e operacionais (computando as viagens e os veículos atribuídos a eles).

3. Metodologia

Essa pesquisa parte de estudos anteriores sobre as estratégias adotadas por algoritmos “vencedores” em termos de desempenho em tempo computacional e qualidade de solução para o problema PRV e suas variantes. Dentre essas estratégias, está a representação da solução

como um roteiro gigante, o que traz vantagens por exemplo para o procedimento de *crossover* para os AGs.

Essa pesquisa, portanto, é de natureza qualitativa, visto que expõe de forma didática algoritmos para o problema do caminho mínimo, aplicado ao PRV com frota heterogênea. Inicialmente, descreve-se o problema do caminho mínimo para o PRV clássico, e em seguida para os problemas com frota heterogênea o VFMP com custo fixo e variável e o HVRP, apresentando-se algoritmos para implementação.

4. Resultados

4.1 Problema do Caminho Mínimo para o PRV clássico

Como estado da arte, existem diversos métodos de alto desempenho na solução de PRVs, que apostam em uma representação de solução composta apenas por clientes a serem atendidos, não contendo delimitadores de rotas. Essa estratégia foi apresentada de forma eficiente por Prins (2004), que acabou inserindo os AGs no grupo das melhores heurísticas para solução de PRV para as instâncias de Christofides et al. (1979). Essa representação pode ser vista como um roteiro gigante (rota TSP – *Travelling Salesman Problem*) para um veículo com capacidade infinita. Essa solução é avaliada usando um procedimento *Split* que divide idealmente a sequência em rotas viáveis.

Primeiramente, revisa-se o princípio do procedimento *Split* para o PRV clássico antes de sua generalização para o VFMP e HVRP. O *Split* para o PRV cria um grafo auxiliar acíclico $\mathcal{H}(X, A)$ com $n + 1$ nós. X contém um nó fictício 0, mais 1 a n nós para clientes C_1 a C_n . Um arco $(i - 1, j)$ é incluído em A para cada subsequência $(C_i, C_{i+1}, \dots, C_j)$ de C se uma viagem servindo a esses clientes for viável (vide (1)), isto é, se a demanda total W_{ij} da subsequência (carga de viagem) não excede a capacidade do veículo (Q).

$$W_{ij} = \sum_{k=i}^j q(S_k) \leq Q \quad (1)$$

A expressão (2) apresenta o custo Z_{ij} da subsequência para o grafo auxiliar, que é o comprimento L_{ij} da viagem associada com um limite de duração L .

$$Z_{ij} = L_{ij} = d(0, S_i) + \sum_{k=i}^{j-1} d(S_k, S_{k+1}) + d(S_j, 0) \leq L \quad (2)$$

Uma divisão ideal de uma sequência é dada por um caminho de custo mínimo do nó 0 ao nó n em $\mathcal{H}$. Para m subsequências possíveis ($m \leq \frac{n(n-1)}{2} = O(n^2)$), o grafo pode ser calculado em $O(m)$ usando dois *loops* para i e j . Para um dado i , a carga e o custo de cada subsequência podem ser computados em $O(1)$ quando j é incrementado, usando os valores já calculados para $j - 1$.

A Figura 3 ilustra o algoritmo *Split* para uma sequência de 5 clientes C_1 até C_5 . O primeiro grafo mostra os custos dos arcos e as demandas de cada cliente. Neste exemplo, assume-se que a capacidade dos veículos é $Q = 6$, sendo que a carga máxima permitida para a inclusão de arcos $(i - 1, j)$ no grafo $\mathcal{H}$ é de $Q_{max} = 2Q$. A duração máxima permitida é de $D=150$. Todos os clientes têm um tempo de serviço idêntico $d_i=10$, e os parâmetros de penalização são $\omega_Q=10$ e $\omega_D=1$.

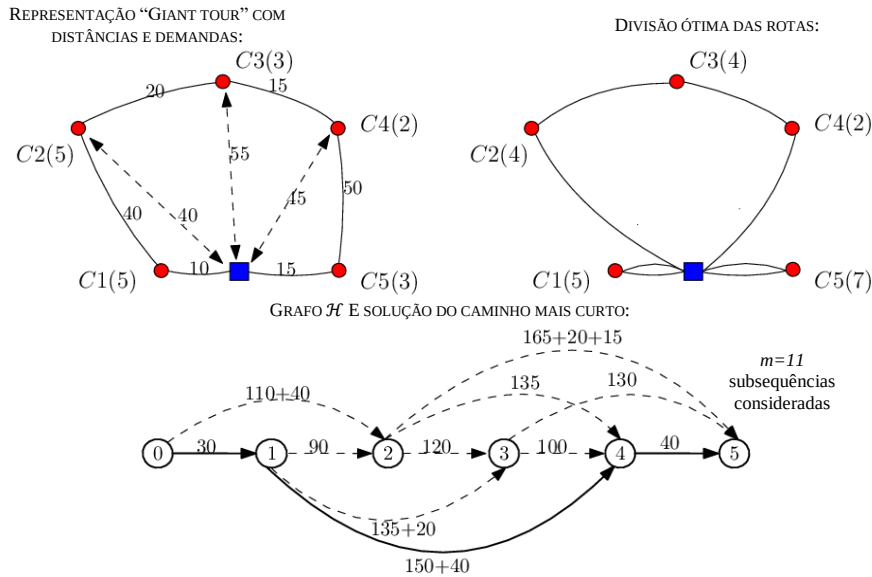


Figura 3 – Ilustração do grafo *Split* com a solução do caminho mais curto

Os arcos do grafo $\mathcal{H}$ são apresentados e o custo da rota incluindo penalidades. Por exemplo, a rota atendendo clientes C_3 , C_4 e C_5 tem um custo de $165 + 20 + 15$, as penalizações de 20 e 15 correspondentes ao excesso de carga de duas unidades e duração, respectivamente. A solução ideal do problema caminho de custo mínimo é de 260. E é composto das três rotas seguintes: rota 1 visita C_1 , rota 2 com as visitas C_2 , C_3 e C_4 e rota 3 visita C_5 .

4.2 Problema do Caminho Mínimo para o PRV com frota heterogênea

Introduz-se inicialmente o VFMP, no qual a frota é composta por t tipos de veículos. Cada tipo k é definido por uma capacidade Q_k , um custo fixo f_k e um custo por unidade de distância v_k , frequentemente chamado de custo variável. Um percurso de comprimento L feito por um veículo do tipo k tem um custo $f_k + L * v_k$. O objetivo é calcular um conjunto de viagens e atribuir veículos a viagens para minimizar o custo total.

Assim, como para o PRV clássico, a carga Q_{ij} e a duração L_{ij} de cada viagem $(C_i, C_{i+1}, \dots, C_j)$ pode ser determinada em $O(1)$. A viagem é factível se ela pode ser realizada pelo menos com o veículo de maior capacidade, ou seja, $W_{ij} \leq Q_t$. Seja Z_{ijk} o custo da viagem utilizando o veículo k . O custo Z_{ij} para o grafo auxiliar corresponde ao custo mais barato dentre os veículos compatíveis conforme apresenta a expressão (3).

$$Z_{ij} = \min(Z_{ijk} = f_k + v_k L_{ij}; 1 \leq k \leq t \text{ e } W_{ij} \leq Q_k) \quad (3)$$

O melhor tipo de veículo pode ser encontrado em $O(1)$ se as demandas forem inteiras (que é o caso de todas as instâncias da literatura) e se a frota estiver correlacionada (quando custos fixos e variáveis crescem a medida que a capacidade do veículo aumenta). Para tanto, utiliza-se um vetor B indexado de 1 à capacidade do maior veículo, de tal forma que $B(W)$ fornece o melhor tipo de veículo para uma viagem com carga W . Para frotas não correlacionadas, um loop $O(t)$ em tipos de veículos deve ser adicionado e a complexidade do *Split* torna-se $O(mt)$. Para exemplificar numericamente, suponha uma viagem com demanda de 35 e duração de 200. O cálculo do custo dessa rota assim como os parâmetros disponíveis estão disponíveis na Figura 4.

Tipo do veículo (k)	1	2	3	4	Tem-se que: $B(35)=3$. Os veículos 1 e 2 teriam suas capacidades violadas. Assim, o custo dessa viagem seria: $Z = 250 + 30 * 200 = 6250$	W	B(W)
Disponibilidade (a_k)	n	n	n	n		1	1
Carga (Q_k)	20	30	40	50		2	1
Custo Fixo (f_k)	100	200	250	300		⋮	⋮
Custo Variável (v_k)	10	20	30	35		35	3
						⋮	⋮
						50	4

Figura 4 – Exemplo do cálculo de custo para o VFMP

No HVRP, o número de veículos é limitado e o problema pode ser infactível. Note que os dados podem satisfazer (4), que afirma que a demanda de clientes que precisam de um veículo do tipo k ou maior, não deve ser maior que a capacidade total destes veículos. Assume-se que os tipos de veículos estão ordenados de acordo com suas capacidades.

$$\sum_{i=1}^n q_i |q_i| \geq Q_k \leq \sum_{p=k}^t a_p Q_p, 1 \leq k \leq t \quad (4)$$

A Figura 5 mostra um caso de duas soluções com os mesmos clientes. Suponha uma frota com apenas dois veículos com capacidade de 10 e 20. Em S' não existe uma divisão de rota factível, embora a restrição 4 seja satisfeita. Segundo Prins (2009), esses casos são raros na prática, exceto nos casos em que a demanda total estiver muito próxima da capacidade da frota.

Seja $t = 2, a_1 = 1, Q_1 = 10, a_2 = 1, Q_2 = 20$			
S : [1 2 3 4 5]	$\rightarrow$ factível	S' : [1 4 3 5 2]	$\rightarrow$ infactível
q : [5 4 4 2 7]		q : [5 2 4 7 4]	

Figura 5 – Ilustração de soluções factíveis e infactíveis

O grafo auxiliar $\mathcal{H}$ para o HVRP é o mesmo que para o VFMP, com um arco para cada viagem $(C_i, C_{i+1}, \dots, C_j)$, tal que $W_{ij} \leq Q_t$. A única diferença é que não sabemos de antemão, devido a disponibilidades limitadas, se um veículo do tipo mais barato estará disponível. Portanto, o comprimento de viagem L_{ij} é usado como peso provisório para o arco em $\mathcal{H}$: o custo exato dependerá do tipo de veículo alocado para este arco.

Uma divisão ótima de rota correspondente ao caminho de menor custo do nó 0 para n em $\mathcal{H}$, com não mais do que a_k arcos para cada tipo de veículo k . Nesse Problema do Caminho Mínimo com Restrições de Recursos, cada tipo de veículo pode ser visto como um recurso disponível em unidades a_k e cada arco requer uma unidade de um tipo de veículo compatível. Em geral, tais problemas são NP-hard, mas podem ser resolvidos rapidamente na prática, usando métodos de programação dinâmica. Seja ψ o número máximo de vetores distintos de consumo x , levando em conta o consumo nulo, como em (5).

$$\psi = \prod_{k=1}^t (a_k + 1) \quad (5)$$

Os cálculos podem ser organizados usando uma matriz Y , $(n + 1) * \psi$. Cada linha j de Y é indexada de 0 a $\psi - 1$. Na prática, um rótulo Y_{jh} para j representa um caminho mais curto do nó 0 para o nó j no grafo auxiliar. Y contém informações como o custo $Y_{jh} \cdot Z$, e o padrão de utilização parcial de frota para o caminho $Y_{j,h} \cdot p$. Ao contrário do PRV e do VFMP, vários padrões podem ser usados para o mesmo nó, com diferentes consumos de recursos.

Um rótulo pode ser acessado em $O(1)$, definindo-se uma chave $h(x)$ para converter cada vetor possível x . Essa chave pode ser calculada pelo algoritmo da Figura 6 conforme sugere Prins

(2009).

```

01.  h: = 0
02.  for k:=1 to t do h: = h(ak + 1) + xk

```

Figura 6 – Algoritmo para calcular a chave referente a um vetor x

Na prática, o algoritmo pode ser usado conforme o exemplo da Figura 7, assumindo-se que $t = 4$, e $a_k = (4,2,3,1)$, tem-se $\psi = 5 * 3 * 4 * 2 = 120$. A estrutura de dados $\mathcal{H}_{p,k}$, utilizada posteriormente, também é apresentada.

Para calcular a chave de $x = (4,2,3,1)$					
$h: = h(a_k + 1) + x_k$					
$h(0,0,0) = 0;$					
$h: = 0 * (4 + 1) + 4 = 4;$					
$h: = 4 * (2 + 1) + 2 = 14;$					
$h: = 14 * (3 + 1) + 3 = 59;$					
$h: = 59 * (1 + 1) + 1 = 119;$					
Para calcular a chave de $x = (1,0,0,0)$					
$h: = h(a_k + 1) + x_k$					
$h(0,0,0) = 0;$					
$h: = 0 * (4 + 1) + 1 = 1;$					
$h: = 1 * (2 + 1) + 0 = 3;$					
$h: = 3 * (3 + 1) + 0 = 12;$					
$h: = 12 * (1 + 1) + 0 = 24;$					

	k				h(x)
	1	2	3	4	
x	0	0	0	0	0
	0	0	0	1	1
	0	0	1	0	2
	0	0	1	1	3
	0	0	2	0	4
	0	0	2	1	5
	0	0	3	0	6
	0	0	3	1	7
	:	:	:	:	:
	1	0	0	0	24
	:	:	:	:	:
	4	2	3	1	$\psi - 1 = 119$

	$\mathcal{H}_{p,k}$	k			
		1	2	3	4
p	0	0	0	0	0
	1	0	0	0	1
	2	0	0	1	0
	3	0	0	1	1
	4	0	0	2	0
	5	0	0	2	1
	6	0	0	3	0
	7	0	0	3	1
	:	:	:	:	:
	24	1	0	0	0
	:	:	:	:	:
	$\psi - 1 = 119$	4	2	3	1

Figura 7 – Exemplo numérico para o cálculo da chave referente a um vetor x e da estrutura $\mathcal{H}_{p,k}$

O algoritmo apresentado na Figura 8 fornece uma implementação possível do método, utilizando uma matriz Y e um grafo auxiliar $\mathcal{H} = (X, A, Z)$, embora, $\mathcal{H}$ não seja gerado explicitamente. X contém $n + 1$ nós indexados de 0 a n . A contém um arco (i, j) , $i < j$, se a sequência de clientes C_{i+1} para C_j apresenta-se dentro de limites aceitáveis em termos de carga e duração, verificado na linha 4. Basicamente, para cada nó $j = 1, 2, \dots, n$, armazena-se os custos do caminho mínimo do nó 0 para o nó j em $\mathcal{H}$, e o predecessor de j neste caminho.

Todos os rótulos de Y são inicializados na linha 1: os custos na linha 0 são definidos como 0 e os outros custos como infinito. Isso pode ser feito em $O(n\psi t)$. No final, a solução é dada pelo rótulo mais barato na linha n de Y . Se todos os rótulos da linha n ainda tiverem um custo infinito, o problema é infactível (vide Figura 11).

O algoritmo original de Prins (2009), permite que apenas soluções factíveis sejam aceitas. No entanto, estudos realizados desde o trabalho de Glover (1986) afirmam que uma exploração controlada de soluções infactíveis pode melhorar a busca. O que permite transitar mais facilmente entre soluções viáveis estruturalmente diferentes. Além disso, o uso de soluções infactíveis pode contribuir para melhorar a robustez do método, que é menos dependente da disponibilidade de uma solução inicial viável (encontrar uma solução viável é muitas vezes um problema NP-hard). Para a avaliação de uma solução o custo penalizado para cada rota Z_{ij} está definido em (6).

$$\forall (i, j) \in A: Z_{ij} = f_k + v_k * L_{ij} + \max(0, Q_k - W_{ij}) * \omega_Q + \max(0, L - L_{ij}) * \omega_D \quad (6)$$

Este custo consiste na soma da distância e duração total de todas as rotas considerando suas penalizações de excessos relacionados ao carregamento, onde ω_Q e ω_D representam os coeficientes de penalização associados. No artigo de Vidal (2014), as penalizações são ajustadas dinamicamente durante a execução do algoritmo, o que favorece a geração de indivíduos factíveis.

```

01. Inicializar matriz Y
02. for i:=1 to n do
03.      $W_{ij} := 0$ ;  $L_{ij} := 0$ ;  $j := i$ 
04.     while ( $j \leq n$  &&  $W_{ij} < Q_t * \delta_Q$ ) do
05.          $W_{ij} := W_{ij} + q_{C_j}$  //cálculo da carga da rota
06.         if  $W_{ij} < Q_t * \delta_Q$  then
07.             if  $i = j$  then  $L_{ij} := d_{0,C_j} + t_{C_j} + d_{C_j,0}$  //cálculo da duração da rota
08.             else  $L_{ij} := L_{ij} - d_{C_{j-1},0} + d_{C_{j-1},C_j} + t_{C_j} + d_{C_j,0}$ 
09.             if  $L_{ij} < L * \delta_D$  then //  $C_i \dots C_j$  correspondem ao arco  $(i-1, j)$  em  $\mathcal{H}$ 
10.                 for p:=0 to  $\psi - 1$  with  $Y_{ip}.Z = \infty$  do
11.                      $k = t$  //a avaliação dos veículos começa pelo tipo
                                     de veículo de maior carga
12.                     capacity_exceeded:=false
13.                     while  $k > 0$  && capacity_exceeded==false do
14.                         if  $h_{p,k} + 1 \leq a_k$  && ( $W_{ij} < Q_k$ ) then //verificação da disponibilidade
                                                         de veículos do tipo  $k$  e carga
15.                              $Z_{ij} = f_k + v_k * L_{ij} + \max(0, Q_k - W_{ij}) * \omega_Q + \max(0, L - L_{ij}) * \omega_D$ 
                                     //cálculo do custo rota considerando penalizações
16.                              $h = 0$ , with  $i = 1$ 
                                      $h = p$ 
17.                              $h_{new} = h + \pi_k$  //cálculo da nova chave para o consumo x
18.                             if  $Z_{ij} + Y_{i-1,p}.Z < Y_{j,h_{new}}.Z$  then
19.                                  $Y_{j,h_{new}}.Z = Z + Y_{i-1,p}.Z$ 
20.                                  $Y_{j,h_{new}}.p = p$ 
21.                                  $Y_{j,h_{new}}.i = i - 1$ 
22.                                  $Y_{j,h_{new}}.k = k$ 
23.                             endif
24.                             else capacity_exceeded=true;
25.                                  $k=k-1$ 
26.                             endwhile
27.                         endfor
28.                     endif
29.                      $j=j+1$ ;
30.                 endwhile
31.             endfor

```

Figura 8 – Algoritmo para o problema do caminho mínimo com restrição de recursos para HVRP

Na linha 3, inicializa-se valores de carga e distâncias. A partir de $j = i$, todos os clientes C_j são avaliados para compor a rota desde que a carga total da rota não tenha ultrapassado a carga do maior veículo (Q_t), conforme linha 4. Em seguida calcula-se a nova carga da rota considerando-se o novo cliente j a ser adicionado, e avalia-se se a permissibilidade de carga (linhas 5 e 6). Se essa carga for menor que a permitida ($Q * \delta_Q$), calcula-se a duração da rota L_{ij} (linhas 7 e 8). Se a duração dessa rota estiver dentro do limite permitido ($L * \delta_D$), tem-se um arco $(i - 1, j)$ em $\mathcal{H}$, sendo que este arco possui os clientes $C_i \dots C_j$. Assim, para cada arco, são inspecionados cada rótulo $\psi - 1$ (linha 10). Também se verifica para cada tipo de veículo a ser utilizado a sua disponibilidade (a_k), e se excede ou não a carga do veículo k . Para isso, inicia-se a avaliação por $k = t$, ou seja, pelo veículo de maior carga. Assim, a partir do primeiro veículo com carga superior à carga necessária para a rota, os veículos com cargas ainda menores nem chegam a ser testados.

Na linha 15, o custo da rota é calculado considerando-se as penalizações, conforme mencionado. Em seguida, realiza-se o cálculo da nova chave para o consumo x . De fato, a linha 17 pode ser computada em $O(1)$, se apenas adicionarmos a contribuição π_k em cada chave associada ao acréscimo do veículo k adicionado a x_k . Essa contribuição pode ser calculada como em (7).

$$\pi_k = \prod_{i=k+1}^t (a_i + 1), \quad k \neq t \quad (7)$$

Para exemplificar, apresenta-se a Figura 9, na qual supõe-se que 4 tipos de veículos ($t=4$) estão disponíveis. Partindo da chave inicial $h = 0$, com $x = (0,0,0,0)$, e supondo os seguintes acréscimos de um veículo de cada tipo k , com $a_k = (4,2,3,1)$.

$k = 1: x = (1,0,0,0)$ $\pi_1 = \prod_{i=2}^{t=4}(a_i + 1) = (a_2 + 1) * (a_3 + 1) * (a_4 + 1) = 3 * 4 * 2 = 24;$ $h := 0 * (4 + 1) + 1 = 1;$ $h := 1 * (2 + 1) + 0 = 3;$ $h := 3 * (3 + 1) + 0 = 12;$ $h := 12 * (1 + 1) + 0 = 24;$ $h := 0 + 24 = 24;$	$k = 3: x = (0,0,1,0)$ $\pi_3 = \prod_{i=4}^{t=4}(a_i + 1) = 2;$ $h := 0 * (4 + 1) + 0 = 0;$ $h := 0 * (2 + 1) + 0 = 0;$ $h := 0 * (3 + 1) + 1 = 1;$ $h := 1 * (1 + 1) + 0 = 2;$ $h := 0 + 2 = 2;$	<table><tr><th>k</th><th>π_k</th></tr><tr><td>1</td><td>24</td></tr><tr><td>2</td><td>8</td></tr><tr><td>3</td><td>2</td></tr><tr><td>4</td><td>1</td></tr></table>	k	π_k	1	24	2	8	3	2	4	1
k	π_k											
1	24											
2	8											
3	2											
4	1											
$k = 2: x = (0,1,0,0)$ $\pi_2 = \prod_{i=3}^{t=4}(a_i + 1) = 8;$ $h := 0 * (4 + 1) + 0 = 0;$ $h := 0 * (2 + 1) + 1 = 1;$ $h := 1 * (3 + 1) + 0 = 4;$ $h := 4 * (1 + 1) + 0 = 8;$ $h := 0 + 8 = 8;$	$k = t = 4: x = (0,0,0,1):$ $h := h(a_k + 1) + x_k$ $h := 0 * (4 + 1) + 0 = 0;$ $h := 0 * (2 + 1) + 0 = 0;$ $h := 0 * (3 + 1) + 0 = 0;$ $h := 0 * (1 + 1) + 1 = 1;$ $h := 0 + 1 = 1;$	Assim, ao adicionarmos um veículo do tipo k=1, precisamos adicionar à chave 24 unidades.										

Figura 9 – Exemplo numérico do cálculo da contribuição π_k para a chave associada ao acréscimo do veículo k

Na linha 18, avalia-se o custo calculado Z_{ij} somado aos custos dos arcos anteriores a i em relação a $Y_{j,h_{new}}$. Os rótulos em Y são alterados em caso de melhoria no custo (linhas 19 a 22). A Figura 10 apresenta um algoritmo para extrair rotas de Y . O procedimento *enqueue* adiciona cada cliente no final da viagem.

```

01.  j = n
02.  for p:=0 to  $\psi - 1$  with  $Y_{np} \cdot Z \neq \infty$  do  $Y_{aux} := \min(Y_{np} \cdot Z)$  //Busca do rótulo mais barato na linha
                                         n de Y
03.  t:= 0
04.  repeat
05.    i =  $Y_{aux} \cdot i$ 
06.    p =  $Y_{aux} \cdot p$ 
07.    for c:=i to j-1 do enqueue(trip(t),  $C_c$ ) endfor
08.     $Y_{aux} = Y_{ip}$ 
09.    j := i
10.    t:=t+1
11.  until i=0

```

Figura 10 – Algoritmo para extrair as rotas

A Figura 11 apresenta um exemplo de um roteiro gigante e a extração de suas rotas usando o procedimento apresentado na Figura 10.

Roteiro gigante:

Seja $n = 50$:

i	C_i
0	5
1	21
2	8
$\vdots$	$\vdots$
47	4
48	10
49	2

Trip(0)

Iniciando $j = 50$ (linha 1)

Supondo que o menor custo na linha n de Y , seja referente ao rótulo $Y_{50,23}$:

$Y_{i,h}$	0	1	...	23	...	$\psi - 1$
0	0	0	...	0	...	0
1	∞	20	...	30	...	∞
2	∞	∞	...	40	...	∞
$\vdots$	$\vdots$	$\vdots$	...	$\vdots$	...	$\vdots$
50	∞	∞	...	1500	...	1555

Supondo que o rótulo $Y_{50,23}$ traga as seguintes informações:

$Y_{50,23} \cdot Z = 1500$; $\rightarrow$ custo acumulado de todas as rotas até o cliente $i = 50$

$Y_{50,23} \cdot i = 47$; $\rightarrow i = 47$ (linha 7)

$Y_{50,23} \cdot p = 21; \rightarrow p = 21$ (linha 8)

$Y_{50,23} \cdot k = 1;$

A rota será percorrida pelo veículo $k=1$, formada pelos clientes $i = 47$ a $j - 1 = 49$, ou seja, pelos clientes C_{47} a C_{49} .

$\therefore Trip(0) = \{4,10,2\}$

O novo rótulo será avaliado considerando-se $Y_{48,21}$. O valor de j passa a ser 48 (linha 12) e i o valor do novo rótulo $Y_{48,21} \cdot i$. E o custo da viagem $Trip(0)$ será o custo acumulado de todas as rotas até i ($Y_{50,23} \cdot Z = 1500$) menos o custo acumulado das rotas anteriores ($Y_{48,21} \cdot Z = 1400$), logo, o custo da rota formada pelos clientes 4,10 e 2 seria de 100. O procedimento é repetido até que i seja 0.

Figura 11 – Exemplo numérico referente ao procedimento de extração de rotas a partir de Y

A Figura 12(a) ilustra uma sequência $S = (1,2,3,4,5)$ com as respectivas demandas em parênteses. Com $L = \infty$, e quatro veículos disponíveis, sendo que um possui carga de $Q_1 = 10$ e três possuem carga de $Q_2 = 5$. A solução do problema é apresentada em Figura 12(b) com três viagens e utilização de dois dos veículos menores, e um veículo maior.

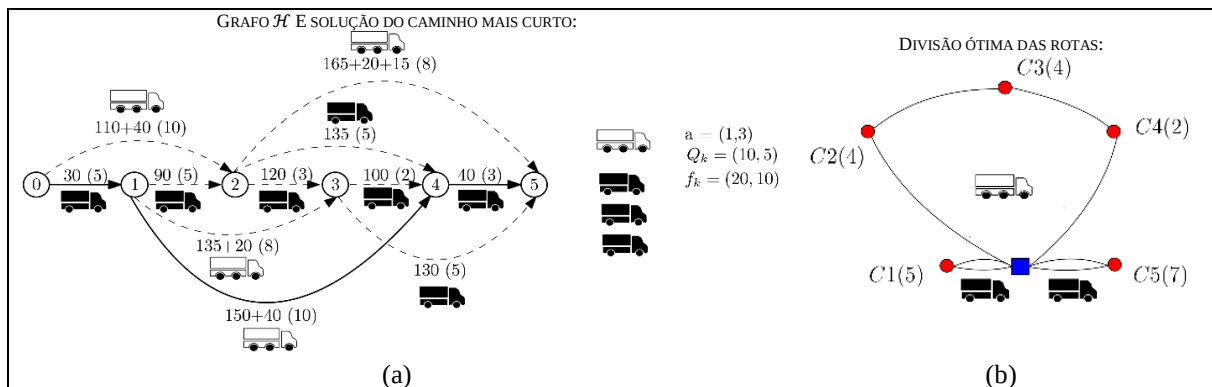


Figura 12 – Algoritmo para o procedimento *Split* com carga heterogênea

5. Conclusões

O presente artigo aborda uma possível implementação de um algoritmo para o problema do caminho mínimo com restrições de recursos. Esse método de particionamento do *roteiro gigante* garante que, para a sequência de clientes de determinada solução, seja obtida uma rota ótima. O método foi apresentado para o PRV com frota heterogênea.

Algumas adaptações foram realizadas em relação aos artigos originais de Prins (2004) e Prins (2009). Vale mencionar que uma característica relevante do espaço de busca vem do uso potencial de soluções infactíveis. Por conseguinte, foram permitidas soluções infactíveis incluindo-se penalizações para as mesmas.

A apresentação do método buscou ser didática, através de exemplos numéricos, para facilitar a compreensão. Essa técnica é utilizada em algoritmos de alta performance que buscam produzir métodos competitivos e escalonáveis. Seu estudo é, portanto, imprescindível para que sejam utilizadas e/ou compreendidas por aqueles interessados no tema. Afinal, mesmo uma pequena melhoria nas soluções pode ter enormes consequências econômicas e ambientais na prática.

Referências

ARAÚJO, C. E. G. *Algoritmos genéticos híbridos sem delimitadores de rotas para problemas de roteirização de veículos*. Tese de Doutorado. Universidade de São Paulo, 2008.

SCHMITT, R. *Metaphernanalyse und die Repräsentation biographischer Konstrukte*. *Journal für Psychologie*, n. 4/1995 1/1996, p. 47-61, 1995.

BEASLEY, J. E. *Route-first cluster-second methods for vehicle routing*. *Omega*, 11:403-8, 1983.

BOUDIA, M., PRINS, C., REGHIOU, M. "An effective memetic algorithm with population management for the Split delivery vehicle routing problem." *International Workshop on Hybrid Metaheuristics*. Springer, Berlin, Heidelberg, 2007.

- CHRISTOFIDES, N.** *Combinatorial optimization*. A Wiley-Interscience Publication, Based on a series of lectures, given at the Summer School in Combinatorial Optimization, held in Sogesta, Italy, May 30th-June 11th, 1977, Chichester: Wiley, 1979, edited by Christofides, Nicos. 1979.
- COSTA, P. C., LONGO, H.** *O Problema do Caminho Elementar Mínimo com Restrições de Recursos e Subcaminhos Proibidos*. Anais/Resumos da 63ª Reunião Anual da SBPC, 2011.
- DUHAMEL, C., POTVIN, J-Y. AND ROUSSEAU, J-M.** "A multi-start evolutionary local search for the two-dimensional loading capacitated vehicle routing problem." *Computers and Operations Research* 38.3 (2011): 617-640.
- DIJKSTRA, E. W.** *A note on two problems in connexion with graphs*. *Numerische Mathematik*, 1(1):269–271, December 1959.
- FORD JR. AND LESTER RANDOLPH.** *Network flow theory*. Paper P-923, The RAND Corporation, Santa Monica, California, August 1956.
- GLOVER, F.** Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5) 533-549, 1986.
- GOLDEN, B. L., WASIL, E. A., KELLY, J. P., CHAO, I. M.** *The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results*. *Fleet management and logistics*. Springer, Boston, MA, 1998. 33-56, 1998.
- GOLDEN, B., ASSAD, A., LEVY, L., GHEYSENS, F.** *The fleet size and mix vehicle routing problem*, *Computers & Operations Research*, v. 11, n. 1, pp. 49– 66, 1984.
- LIU, S., HUANG, W. MA. H.** "An effective genetic algorithm for the fleet size and mix vehicle routing problems." *Transportation Research Part E: Logistics and Transportation Review* 45.3: 434-445, 2009.
- NGUEVEU, S. U., PRINS, C. CALVO, R. W.** "An effective memetic algorithm for the cumulative capacitated vehicle routing problem." *Computers and Operations Research* 37.11 (2010): 1877-1885.
- PRINS, C.** "A simple and effective evolutionary algorithm for the vehicle routing problem." *Computers and Operations Research* 31.12: 1985-2002, 2004.
- PRINS, C.** "Two memetic algorithms for heterogeneous fleet vehicle routing problems." *Engineering Applications of Artificial Intelligence*, 22.6 (2009): 916-928.
- BELLMAN, R.** On a routing problem. *Quarterly of Applied Mathematics*, 16(1):87–90, 1958.
- FLOYD, R. W.** Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345, June 1962.
- SANTOS, G.** *Gastos com logística consomem 12,37% do faturamento das empresas*. Disponível em: <https://veja.abril.com.br/economia/gastos-com-logistica-consomem-1237-do-faturamento-das-empresas/> Acesso: 14/08/2018.
- TAILLARD, É. D.** "A heuristic column generation method for the heterogeneous fleet VRP", *RAIRO-Operations Research*, v. 33, n. 1, pp. 1–14, 1999.
- ULUSOY, G.** *The fleet size and mix problem for capacitated arc routing*. *European Journal of Operational Research*, v. 22, n. 3, p. 329-337, 1985.
- VIDAL, T., CRAINIC, T. G., GENDREAU, M., & PRINS, C.** "A unified solution framework for multi-attribute vehicle routing problems." *European Journal of Operational Research* 234.3 (2014): 658-673.
- VIDAL, T., CRAINIC, T. G., GENDREAU, M., PRINS, C.** (2013). "A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows." *Computers and operations research*, 40.1 (2013): 475-489.
- VIDAL, T., CRAINIC, T. G., GENDREAU, M., LAHRICHI, N., REI, W.** (2012). *A Hybrid Genetic Algorithm for Multi-Depot and Periodic Vehicle Routing Problems*. *Operations Research*, 60(3), 611–624.
- VIEIRA, B. S.** *Uma Meta-Heurística Adaptativa Large Neighborhood Search Com Mecanismos De Paralelismo, Detecção De Estagnação E Perturbações Para O Problema De Roteamento De Veículos Com Frota Heterogênea, Periódico E Multi-Trips*. Dissertação de Mestrado do PPG em Engenharia de Transportes, COPPE, UFRJ, 2017.