Um Algoritmo Híbrido para o Problema de Roteamento de Veículos com Frota Heterogênea Robusto

Carlos Neves[1], **Anand Subramanian**[1], **Pedro Munari**[2]

[1]Centro de Informática – Universidade Federal da Paraíba (UFPB)
Rua dos Escoteiros, s/n – Mangabeira, João Pessoa – PB, 58055-000

[2]Universidade Federal de São Carlos – Departamento de Engenharia de Produção
Rod. Washington Luís km 235 – SP-310 – São Carlos – SP, 13565-905

carlosneves@eng.ci.ufpb.br, anand@ci.ufpb.br, munari@dep.ufscar.br

Abstract. *We address the Fleet Size and Mix Vehicle Routing Problem under demand uncertainty, which seeks an optimal fleet composition, as well as a set of routes with minimal cost. We treat the problem under the perspective of robust optimization by assuming that variations in customer demands belong to a well-defined uncertainty set. To solve the problem, we combine the Iterated Local Search algorithm with an exact Set Partitioning procedure. We also devise efficient data structures to perform local search efficiently. Computational experiments show that the proposed method is effective as we not only found the best known upper bound for most benchmark instances, but also improve existing upper bounds for some of them.*

Resumo. *Este trabalho aborda o Fleet Size and Mix Vehicle Routing Problem, que visa não apenas planejar rotas de custo mínimo, mas também determinar a melhor composição de veículos para realizar as entregas. Sob a perspectiva da otimização robusta, modelou-se as demandas como valores pertencentes a um conjunto de incertezas. Foram propostas uma combinação do algoritmo de Busca Local Iterada com um método exato de particionamento de conjuntos e estruturas de dados para realizar as buscas locais de maneira eficiente. Experimentos computacionais mostram a eficácia do método, que foi capaz de alcançar a maioria dos melhores limitantes superiores conhecidos da literatura, além de melhorar limitantes para algumas instâncias.*

1. Introdução

Este trabalho considera o Problema de Roteamento de Veículos (PRV) com frota heterogênea com demandas incertas. Mais especificamente, aborda-se o *Fleet Size and Mix Vehicle Routing Problem* (FSMVRP), e considera-se que as demandas são parâmetros incertos e, portanto, modelados como variáveis aleatórias. Isso é feito através do paradigma de Otimização Robusta (OR) [Ben-Tal and Nemirovski 1999, Bertsimas and Sim 2004], assumindo-se que as realizações das variáveis aleatórias referentes às demandas pertencem a um conjunto de incertezas bem definido, possibilitando uma análise de pior caso das soluções. Nessa abordagem, uma solução só pode ser ótima se permanecer factível para todas as possíveis realizações do conjunto, ficando, portanto, protegida contra incertezas.

2. Descrição do problema

Seja $G = (V, A)$ um grafo direcionado em que V e A são os conjuntos de vértices e arcos, respectivamente. Tem-se que $V = V' \cup \{0, n+1\}$, onde o subconjunto $V' = \{1, \ldots, n\}$ representa os clientes, enquanto, por conveniência, tanto 0 quanto $n+1$ representam o depósito, e $A = \{(i, j) \in V^2 \mid i \neq j\}$. Cada cliente $i \in V$ possui uma demanda q_i. O depósito, por sua vez, é munido de uma frota heterogênea ilimitada de veículos representada pelo conjunto $K = \{1, \ldots, m\}$, em que m é o número de tipos distintos de veículos. Cada tipo de veículo $k \in K$ está associado a uma capacidade Q_k. O custo de viajar por um arco $(i, j) \in A$ usando um veículo do tipo $k \in K$ é c_{ijk}, e obedece à desigualdade triangular. Além disso, o uso de cada unidade de veículo do tipo $k \in K$ incorre um custo fixo $f_k \in \mathbb{R}^+$. O problema consiste em determinar um conjunto de rotas de custo total mínimo tal que: (*i*) cada rota está associada a um veículo do tipo $k \in K$ e começa e termina no depósito, visitando um subconjunto de V' durante o processo; (*ii*) a capacidade do veículo associado a cada rota não pode ser excedida (i.e., a soma das demandas dos clientes visitados por um veículo deve ser menor ou igual à sua capacidade); (*iii*) cada cliente precisa ser visitado uma única vez, tendo sua demanda suprida.

Assim como é feito na maioria dos trabalhos sobre OR da literatura, denota-se o *budget* de demanda como $\Gamma^q \in \mathbb{Z}^+$. Além disso, considera-se que $\bar{q}_i$ é a demanda nominal do vértice $i \in V'$, e que $\hat{d}_i$ é um desvio associado a esse valor. Portanto, a demanda do vértice $i \in V'$ no pior caso é $\bar{q}_i + \hat{q}_i$. Para definir o conjunto de incertezas relacionado à demanda dos clientes, considere γ_i^q variáveis aleatórias cujo suporte é $[0, 1]$, de modo que $q_i = \bar{q}_i + \hat{q}_i \gamma_i^q$. Assim, o conjunto de incertezas referente às demandas é dado por:

$$\mathcal{Q} = \left\{ q \in \mathbb{R}^{|\mathcal{V}'|} \;\middle|\; q_i = \bar{q}_i + \hat{q}_i \gamma_i^q, \sum_{i \in V} \gamma_i^q \leq \Gamma^q, 0 \leq \gamma_i^q \leq 1, i \in V \right\}.$$

Quando uma solução satisfaz os itens (*i*) a (*iii*) definidos no início da seção para todas as realizações de demanda associadas ao conjunto $\mathcal{Q}$, ela é dita robusta viável. Observe que fazendo-se $\Gamma^q = 0$, o problema se resume à sua versão determinística, pois o conjunto de incertezas reduz-se a um único ponto em $\mathbb{R}^n$. Note, ainda, que quanto maior o valor de Γ^q, maior é o grau de incerteza quanto à possibilidade de desvios nas demandas. Portanto, soluções associadas a valores mais altos de Γ^q são consideradas mais conservadoras.

Já que $\mathcal{Q}$ é um conjunto compacto e convexo e as restrições associadas às demandas q são todas lineares, é necessário se preocupar apenas com os pontos extremos de $\mathcal{Q}$, o seguinte lema se aplica.

Lema 1. *[Subramanyam et al. 2018, Subramanyam et al. 2020]. Uma rota é viável para todos os cenários de demanda em $\mathcal{Q}$ se e somente se ela for viável quando suas Γ^q maiores demandas atingem o pior caso, enquanto as demandas restantes assumem seus valores nominais.*

A propriedade apresentada no Lema 1 é explorada ao longo de todo algoritmo, e foi utilizada na elaboração de uma formulação compacta que visa facilitar a compreensão da definição do problema.

3. Algoritmo proposto

O método desenvolvido para resolver o problema une a meta-heurística *Iterated Local Search* (ILS) [Lourenço et al. 2019] com um método baseado em particionamento de conjuntos que visa, a partir de um conjunto de rotas obtidas no decorrer do algoritmo, construir uma boa solução de forma exata. O algoritmo consiste em uma adaptação do algoritmo proposto por [Penna et al. 2019]. Como em [Penna et al. 2019], inicia-se construindo-se uma solução inicial para o problema, que é então melhorada através de procedimentos de busca local. Quando a solução não pode mais ser melhorada, aplica-se uma perturbação aleatória à melhor solução obtida. O processo se repete até que um critério de parada seja atingido. Ao fim do procedimento, tenta-se, de maneira exata, construir uma solução utilizando-se rotas obtidas durante as buscas locais, que foram armazenadas em um *pool* de rotas. Isso é feito através de um resolvedor de problemas lineares inteiros.

O método construtivo, estruturas de vizinhança da busca local e procedimentos de perturbação são semelhantes aos de [Penna et al. 2019], mas tiveram que ser adaptados à versão robusta do FSMVRP. A principal diferença é que, neste trabalho, soluções inviáveis são permitidas. Para isso, o custo total de viagem das rotas é acrescido de suas violações de capacidade quando as demandas atingem o pior caso. O valor das violações é multiplicado por um fator de penalização grande o suficiente para que as buscas locais evitem soluções inviáveis.

O Lema 1 possibilita que a violação das rotas seja calculada ordenando-se as demandas dos clientes. Contudo, fazer isso para cada movimento durante a busca local pode ser muito custoso. Por isso, este trabalho propõe métodos para calcular as violações das rotas de forma eficiente durante as buscas locais. O primeiro método consiste em um algoritmo capaz de avaliar a mudança na violação após troca de subconjuntos quaisquer de rotas distintas. O algoritmo possui complexidade proporcional o tamanho dos subconjuntos trocados, sendo ideal para estruturas de vizinhança que envolvem a troca de blocos de clientes com tamanho constante (e.g., Swap, Shift). O segundo método, por sua vez, baseia-se na ideia de concatenação de subsequências [Savelsbergh 1985, Vidal et al. 2014], i.e., movimentos de busca local podem ser compreendidos como a quebra das rotas em subsequências que são então concatenadas para formar novas rotas. O método possui complexidade proporcional ao *budget* Γ^q, e é apropriado para movimentos que envolvem a troca de muitos clientes (e.g., 2-opt*, K-Shift).

4. Resultados e discussão

O algoritmo foi implementado na linguagem C++ (g++ 9.4.0) e executado em um PC Intel® Core™ i7-2600 CPU 3,40GHz com 16 GB de memória RAM e sistema operacional Linux 64 bits. Os testes foram feitos no conjunto de instâncias originalmente concebido por [Golden et al. 1984] para a versão determinística do FSMVRP. Para gerar as instâncias do RFSMVRP, utilizou-se o método proposto por [Subramanyam et al. 2018]. O método trata as demandas originais das instâncias como os valores nominais. Os desvios de demanda são então calculados como $\hat{q}_i = \alpha \times \bar{q}_i$ para todo $i \in V'$. De forma similar, o *budget* é definido como $\Gamma^q = \lceil \beta n \rceil$. Então, os conjuntos de incerteza de cada instância foram parametrizados fazendo-se $\alpha = 0.1$ e $\beta = 0.2$. Para garantir a viabilidade da instância levando-se em conta os desvios propostos, as capacidades de todos os tipos de veículos foram aumentadas em 10%. As instâncias foram divididas em três grupos:

- **FSMF** – Cada veículo $k \in K$ possui um custo fixo $f_k > 0$ associado, mas os custos de viagem são os mesmos (i.e., $c_{ijk} = c_{ijk'}$ para todo $k, k' \in K, k \neq k'$);
- **FSMD** – Os custos de viagem dependem do tipo do veículo, e todos os custos fixos são nulos (i.e., $f_k = 0$ para todo $k \in K$);

Tabela 1. Resultados para o conjunto de instâncias proposto por [Subramanyam et al. 2018].

	FSMF			FSMD			FSMFD		
Instância	**Média**	**Melhor**	**Gap (%)**	**Média**	**Melhor**	**Gap (%)**	**Média**	**Melhor**	**Gap (%)**
3	951.61	951.61	0	623.22	623.22	0	1144.22	1144.22	0
4	6437.33	6437.33	0	380.71	380.71	0	6437.33	6437.33	0
5	988.63	988.63	0	742.85	742.85	0	1312.65	**1274.22**	-3.6
6	6516.47	6516.47	0	406.19	406.19	0	6516.47	6516.47	0
13	2408.77	2406.36	0	1491.86	1491.86	0	2964.64	2964.64	0
14	9119.03	9119.03	0	603.21	603.21	0	9126.90	9126.90	0
15	2586.37	2586.37	0	999.82	999.82	0	2634.96	2634.96	0
16	2724.92	2720.43	0	1131.00	1131.00	0	3168.91	3168.91	0
17	1746.94	1746.92	0.72	1040.93	1038.60	0	2005.07	2004.48	0
18	2372.22	2372.85	0.13	1805.61	1801.40	0.03	3149.30	**3148.99**	-0.1
19	8664.53	**8661.81**	-0.01	1107.88	1105.44	0	8664.60	8662.90	0
20	4046.66	**4040.52**	-0.49	1537.31	1533.71	0.03	4154.14	**4153.84**	-0.35

- **FSMFD** – Tanto os custos de viagem quanto os custos fixos são únicos, dependendo do tipo do veículo.

A Tabela 1 ilustra os resultados obtidos após 10 execuções, como em [Subramanyam et al. 2018]. As colunas Média e Melhor mostram, respectivamente, o custo médio e o custo da melhor solução obtida para cada instância. As colunas Gap mostram a diferença relativa (*gap*) entre a melhor solução alcançada pelo método proposto e as melhores soluções obtidas por [Subramanyam et al. 2018]. Instâncias para as quais os limitantes da literatura foram superados se encontram em negrito.

No geral, o método foi eficaz, encontrando a maioria dos melhores limitantes superiores conhecidos e melhorando três deles. Na maioria dos casos, obteve-se valores de *gap* nulos ou competitivos. Além disso, melhores limitantes superiores foram obtidos para cinco das instâncias. No entanto, a qualidade das soluções ficou mais comprometida nas instâncias do grupo FSMF. Isso provavelmente ocorre porque o método não foi capaz de determinar a composição de frota mais adequada para acomodar o pior caso do conjunto de demandas em questão, o que, por consequência, impacta negativamente na soma de custos fixos.

5. Considerações finais

Neste trabalho, propôs-se uma abordagem híbrida na forma de *matheuristic* para resolver a versão robusta do FSMVRP. Rotinas baseadas em buscas locais e métodos exatos de particionamento de conjuntos, que já tiveram sua eficácia mostrada em diversas variantes do PRV, foram adaptadas para lidar com incertezas. Para isso, nas buscas locais, empregou-se estruturas de dados e rotinas eficientes durante a checagem de viabilidade das soluções no pior caso durante. Os resultados mostram que o método é promissor, tendo sido capaz de encontrar a maioria dos melhores limitantes superiores da literatura. Foi possível, ainda, melhorar os resultados para algumas instâncias em aberto.

Sugere-se, para trabalhos futuros, a adaptação do método a variantes do PRV com Frota Heterogênea que envolvam frotas limitadas. Além disso, propõe-se a adição de mais elementos ao problema, como restrições de janelas de tempo sujeitas a incertezas quanto aos tempos de viagem.

Referências

Ben-Tal, A. and Nemirovski, A. (1999). Robust solutions of uncertain linear programming. *Operations Research Letters*, 25:1–13.

Bertsimas, D. and Sim, M. (2004). The price of robustness. *Operations Research*, 52(1):35–53.

Golden, B., Assad, A., Levy, L., and Gheysens, F. (1984). The fleet size and mix vehicle routing problem. *Computers & Operations Research*, 11(1):49–66.

Lourenço, H. R., Martin, O. C., and Stützle, T. (2019). *Iterated Local Search: Framework and Applications*, pages 129–168. Springer International Publishing, Cham.

Penna, P. H. V., Subramanian, A., Ochi, L. S., Vidal, T., and Prins, C. (2019). A hybrid heuristic for a broad class of vehicle routing problems with heterogeneous fleet. *Annals of Operations Research*, 273(1):5–74.

Savelsbergh, M. W. P. (1985). Local search in routing problems with time windows. *Annals of Operations Research*, 4(1):285–305.

Subramanyam, A., Repoussis, P. P., and Gounaris, C. E. (2018). Robust optimization of a broad class of heterogeneous vehicle routing problems under demand uncertainty. arXiv:1810.04348 [cs, math].

Subramanyam, A., Repoussis, P. P., and Gounaris, C. E. (2020). Robust optimization of a broad class of heterogeneous vehicle routing problems under demand uncertainty. *INFORMS Journal on Computing*, 32(3):661–681.

Vidal, T., Crainic, T. G., Gendreau, M., and Prins, C. (2014). A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research*, 234(3):658–673.