

# Sistema para Resolver o Problema de Roteamento de Estoque Baseado em Técnicas de Monte Carlo

## Alternative Title: System to Solve the Inventory Routing Problem based on Monte Carlo Techniques

Raucer Curdulino  
Universidade de São Paulo  
raucer@usp.br

Pedro Yuri Araujo Lima Alves  
Universidade de São Paulo  
pedro.yuri.alves @usp.br

Karina Valdivia Delgado  
Universidade de São Paulo  
kvd@usp.br

### RESUMO

O Problema de Roteamento de Estoque com Demanda Estocástica (SIRP–*Stochastic Inventory Routing Problem*) é uma combinação dos problemas de controle de inventários com demandas estocásticas por mercadorias em centros comerciais e do roteamento de veículos utilizados no abastecimento desses centros a partir de um único centro de distribuição. Este trabalho apresenta uma variante do algoritmo proposto por [8] para o SIRP utilizando técnicas de Monte Carlo. O novo algoritmo foi implementado e comparado ao algoritmo original considerando diversas políticas, tendo demonstrado resultados semelhantes em alguns casos e melhores em outros em termos de eficiência de tempo e custo total da solução. A análise, comparação e avaliação dos algoritmos foram feitas com base em *benchmarks* de problemas existentes na literatura.

### Palavras-Chave

Roteamento de veículos, IRP (*Inventory Routing Problem* – Problema de Roteamento de Estoque), SIRP (*Stochastic Inventory Routing Problem*), Algoritmos heurísticos, Técnicas de Monte Carlo.

### ABSTRACT

The Stochastic Inventory Routing Problem (SIRP) is a combination of the inventory control problem with stochastic demands for goods in commercial centers and vehicle routing problem used in their supply from a single distribution center. This paper presents an alternative to the algorithm proposed by [8] for SIRP using Monte Carlo techniques. The new algorithm was implemented and compared to the original one considering several policies, showing similar results in some cases and better results in others in terms of time efficiency and quality solution. The analysis, comparison and assessment of both algorithms were based on benchmark problems from the literature.

### Categories and Subject Descriptors

H.4 [Information Systems Applications]: Miscellaneous;

H.4.2 [Types of Systems]: [Logistics]

**General Terms** Algorithms, Theory.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SBSI 2016, May 17–20, 2016, Florianópolis, Santa Catarina, Brazil.  
Copyright SBC 2016.

### Keywords

Vehicle routing, Inventory Routing Problem, Stochastic Inventory Routing Problem, Heuristic Algorithms, Monte Carlo Techniques.

### 1. INTRODUÇÃO

Os problemas de transporte e de gestão da demanda de mercadorias têm sido estudados nos últimos anos pela comunidade científica e diversas técnicas têm sido propostas para resolvê-los ao longo dos últimos trinta anos [4]. Esses problemas são conhecidos como Problema de Roteamento de Estoque (IRP – *Inventory Routing Problem*).

Tradicionalmente, algumas abordagens tratam esses dois problemas (o problema de roteamento de veículos e de controle de inventários) de forma não integrada, gerando políticas de entrega e transporte de mercadorias de forma sequencial. No entanto, outras abordagens têm tratado o problema de maneira integrada, a fim de obter redução dos custos globais por meio da melhor política para o sistema inteiro [2].

Ao considerar demandas estocásticas em vez de determinísticas, o problema torna-se mais realístico, pois as demandas dos clientes por produtos são estocásticas por natureza [8]. Essa classe de problemas é o que se denomina Problema de Roteamento de Estoque com Demanda Estocástica (SIRP – *Stochastic Inventory Routing Problem*).

Um dos primeiros trabalhos sobre SIRP foi proposto por [5]. Esse trabalho foi uma extensão da solução heurística proposta por [6] para o Problema de Roteamento de Veículos. Sendo adicionado ao modelo original os custos de escassez, de transporte, de armazenamento e a demanda estocástica. Os autores ainda propõem possíveis aplicações da solução desenvolvida, como entregas de combustível a postos de gasolina e fornecimento de produtos perecíveis.

Outro aspecto considerado nesse problema é a quantidade de períodos de decisão. Se o problema tem mais de um período, ele é chamado de multi-período, caso contrário é chamado de um único período. O tratamento de problemas com um único período ainda carece de mais pesquisas por parte da comunidade científica [8]. Ao adotar esse tratamento, considera-se que as demandas sejam aleatórias e com alta variabilidade, aproximando-se mais da realidade e tornando a definição de políticas de abastecimento mais eficiente [8]. Em geral, é mais difícil prever os níveis de estoques futuros para tentar resolver o problema multi-período do que resolver o problema com informação atualizada no final de cada período [8], i.e., tentar resolver um problema de um único período por vez.

A utilização de um horizonte de tempo de curto prazo no SIRP é defendida também por [7]. De maneira que a taxa de consumo dos

clientes apresenta dificuldade em ser antecipada com veracidade, podendo ser representada no melhor dos casos por uma variável estocástica com distribuição de probabilidade conhecida. Sendo assim, planejar antecipadamente para um tempo longo, por exemplo, para todo um ano de distribuição, poderia não ser confiável e sujeito a adaptações devido às variações da demanda.

Neste trabalho será proposta uma variante do algoritmo apresentado em [8] para resolver o SIRP de um único período.

## 2. O IRP COM DEMANDAS ESTOCÁSTICAS – SIRP

O SIRP considera a existência de  $n$  centros comerciais (CCs) com seus próprios estoques e um centro de distribuição (CD), que gerencia as demandas e a quantidade de mercadorias entregues aos CCs.

Como em [2], é considerada a existência de um único período e que, ao final dele, existe informação atualizada sobre os níveis vigentes de estoque de cada CC. Ao se considerar a existência de um único período, deseja-se minimizar os custos totais esperados (distribuição mais custos relacionados ao estoque) em cada cenário de período único.

Para cada CC, o nível de estoque no final de um período depende do nível inicial de estoque e também das demandas dos clientes durante esse período, estocásticas por natureza. Assume-se também que o CD possui um estoque suficiente para suprir as demandas de cada CC.

Na abordagem proposta por [8], é assumida a possibilidade de utilizar os dados históricos de cada CC para modelar as demandas dos clientes finais por meio de uma distribuição de probabilidades empírica ou teórica. No final de um período, podem haver alguns custos associados com o excedente ou com a falta de estoque, que podem ser incorporados ao processo de tomada de decisão e adicionados aos custos de roteamento (baseados em distância ou em número de viagens).

No final de cada período, os níveis de estoque são registrados pelo centro de distribuição e uma estratégia de roteamento é definida para o novo período levando em consideração esses níveis.

### Definição Formal de SIRP

O Problema de Roteamento de Estoque com Demandas Estocásticas é definido em [8] como um grafo completo e não direcionado  $G = (V, A)$ . Em que  $V = V^* \cup \{0\}$ , o centro de distribuição é o vértice 0, os  $n$  centros comerciais são dados por  $V^* = (1, 2, \dots, n)$  e  $A = \{(i, j) | i \in V, j \in V\}$  é o conjunto de arestas (caminhos) que conectam os centros.

O centro de distribuição é o local de onde partem os veículos transportando as mercadorias que serão entregues a cada centro comercial. Para cada centro comercial  $i$ , são conhecidos previamente o nível vigente de estoque  $L_i \geq 0$  e o nível máximo de estoque permitido,  $\hat{L}_i > 0$ . Cada centro comercial  $i \in V^*$  atende diversos clientes, cujas demandas somadas resultam na demanda agregada  $D_i \geq 0$ , que é uma variável aleatória. Essa variável segue uma distribuição de probabilidade com  $E[D_i] = d_i > 0$ .

A frota utilizada no transporte das mercadorias é constituída de  $k$  veículos homogêneos, isto é, com mesma capacidade de carga  $Q > 0$ , com  $Q \geq D_i$  para cada  $i \in V^*$ . Para cada  $(i, j) \in A$ , é conhecido o custo  $c_{ij} > 0$  de transporte do nó  $i$  ao nó  $j$  e é

assumida a existência de uma matriz de custo simétrica, em que  $c_{ij} = c_{ji}$ .

A solução procurada deve fornecer, para um determinado período e com o melhor custo global, um conjunto de rotas que iniciem no centro de distribuição, que atendam cada centro comercial uma única vez e não excedam a capacidade  $Q$  de cada veículo.

A fim de minimizar os custos totais esperados (custos de roteamento + custos de estoque esperados), a quantidade  $q_i$  de mercadorias que cada centro comercial receberá e o plano de roteamento são definidos globalmente. Assim, o custo de estoque esperado para cada centro comercial é determinado pela *função estoque-custo*, definida por:

$$f(q_i) = \begin{cases} \lambda s_i, & \text{se } s_i \geq 0 \\ 2c_{oi}, & \text{se } s_i < 0 \end{cases} \quad (1)$$

Em que  $\lambda \geq 0$  representa o custo de manter uma unidade de mercadoria em estoque no final do período considerado e  $s_i$  representa o excedente no final desse período, ou seja,  $s_i = L_i + q_i - D_i$ , para todo  $i \in V^*$ . No caso de esgotamento de estoque, essa função considera o custo  $2c_{oi}$ , i.e., o custo de ida e volta de um novo veículo do depósito ao centro comercial.

## 3. ALGORITMO SIMHEURISTIC PARA RESOLUÇÃO DE SIRP

Um dos algoritmos propostos para resolver o problema SIRP é o proposto em [8], denominado *Simheuristic*. Esse algoritmo é utilizado para estimar os custos de estoque associados às políticas de cada centro comercial, na determinação dos custos totais (estoque e roteamento) relacionados a várias políticas de reposição e, finalmente, na seleção da estratégia de roteamento de veículos com menor custo global fazendo uma busca.

Uma **política de reposição** é uma regra que define o percentual máximo de reposição de mercadorias em relação ao nível máximo de estoque permitido  $\hat{L}_i$  de um centro comercial  $i$ . Nesse algoritmo, para *um centro comercial  $i$* , são consideradas cinco *políticas de reposição* de mercadorias:

- Nenhuma reposição de estoque;
- Reposição a um quarto da capacidade de estoque do centro comercial;
- Reposição até a metade da capacidade de estoque do centro comercial;
- Reposição até 3/4 da capacidade de estoque do centro comercial;
- Reposição total.

Um termo importante usado nesse algoritmo é *estratégia de reposição básica (ERB)*, que considera a mesma política de reposição para *todos os centros comerciais*.

A Figura 1 apresenta o fluxograma correspondente ao algoritmo *Simheuristic*. Nela, estão indicados por letras os quatro passos desse algoritmo, que são:

- A: Estimativa dos custos esperados de estoque associados a cada combinação de centro comercial e política de reposição (CC-Políticas);
- B: Cálculo dos custos de roteamento associados a cada ERB;
- C: Identificação da ERB com o menor custo total (custo de estoque + custo de roteamento);
- D: Processo *Multistart*.

A seguir, são descritos esses quatro passos do algoritmo *Simheuristic*.

### Passo A: estimativa dos custos esperados de estoque associados a cada combinação CC-Políticas

O algoritmo *Simheuristic* emprega Simulação de Monte Carlo (MCS – *Monte Carlo Simulation*) para gerar as demandas randômicas e os custos de estoque esperados associados a cada combinação entre centro comercial e política (*CC-Política*). MCS é empregada para obter estimativas dos custos de inventário associados a ela, incluindo os custos de falta e de excedente de estoque. Por se tratar de uma abordagem de aproximação, este método não produz uma solução ótima, mas pode produzir soluções aproximadamente ótimas em um tempo razoavelmente pequeno, da ordem de segundos [8].

Inicialmente, para cada centro comercial, é gerada uma quantidade  $m$  de demandas aleatórias  $D_i$ , calcula-se o custo de estoque correspondente a cada demanda usando a Equação (1) (considerando possíveis excedentes e esgotamentos de estoque) e finalmente determina-se o custo médio sobre essas  $m$  demandas.

As demandas são geradas randomicamente por meio de técnicas de Monte Carlo segundo uma distribuição log-normal, que gera valores positivos no conjunto dos números reais. Ao considerar essa quantidade  $m$  de demandas, obtêm-se estimativas mais precisas dos custos de estoque associados a cada combinação CC-Política [8].

Em seguida, são determinados a quantidade  $q_i$  de mercadorias a serem entregues por política de reposição e o custo de estoque médio para cada combinação CC-Política. Calcula-se também o custo de estoque médio por ERB, que será utilizado no Passo C para calcular o custo total.

### Passo B: cálculo dos custos de roteamento

O algoritmo *Simheuristic* emprega o algoritmo heurístico CWS - *Clarke & Wright Savings* [3] para resolver o Problema de Roteamento de Veículos Capacitados (CVRP – *Capacitated Vehicle Routing Problem*), por ser rápido e eficiente [8]. CWS [3] é um dos métodos heurísticos mais conhecidos e estudados na resolução de CVRPs e está baseado no princípio da economia. Por se tratar de um método heurístico, não há garantia de que a solução ótima será encontrada. No entanto, é possível obter uma solução aproximada com baixos custos de tempo e de recursos computacionais sem que o espaço de busca seja explorado profundamente [3,9,10].

Neste passo, primeiro é gerada a especificação do problema de roteamento de veículos. Essa especificação se baseia no número  $n$  de centros comerciais, na quantidade  $k$  de veículos, na capacidade  $Q$  de carga de cada veículo, na posição de cada centro comercial em relação ao centro de distribuição e nas quantidades de produtos ( $q_i$ ) a serem enviados a cada centro comercial para cada política de reposição (quantidades calculadas no Passo A).

Montada essa especificação, calculam-se os custos de roteamento associados a cada ERB usando o algoritmo CWS.

### Passo C: identificação da ERB com o menor custo total

Concluídos os passos A e B do algoritmo *Simheuristic*, o passo C calcula os custos totais esperados (custos esperados de estoque + custos de roteamento) referentes a cada ERB. A ERB com o menor custo total e atribuída à variável *bestSol*, que armazena a melhor solução durante a execução do algoritmo.

### Passo D: Processo *Multistart*

Nesse passo são executadas diversas iterações enquanto o tempo limite de execução não for excedido. A cada iteração desse processo, gera-se uma lista de centros comerciais ordenada de forma randômica para tornar aleatório o processo de busca local por uma solução com menor custo total. Em seguida, inicializa-se a variável *baseSol* com a ERB com o menor custo total.

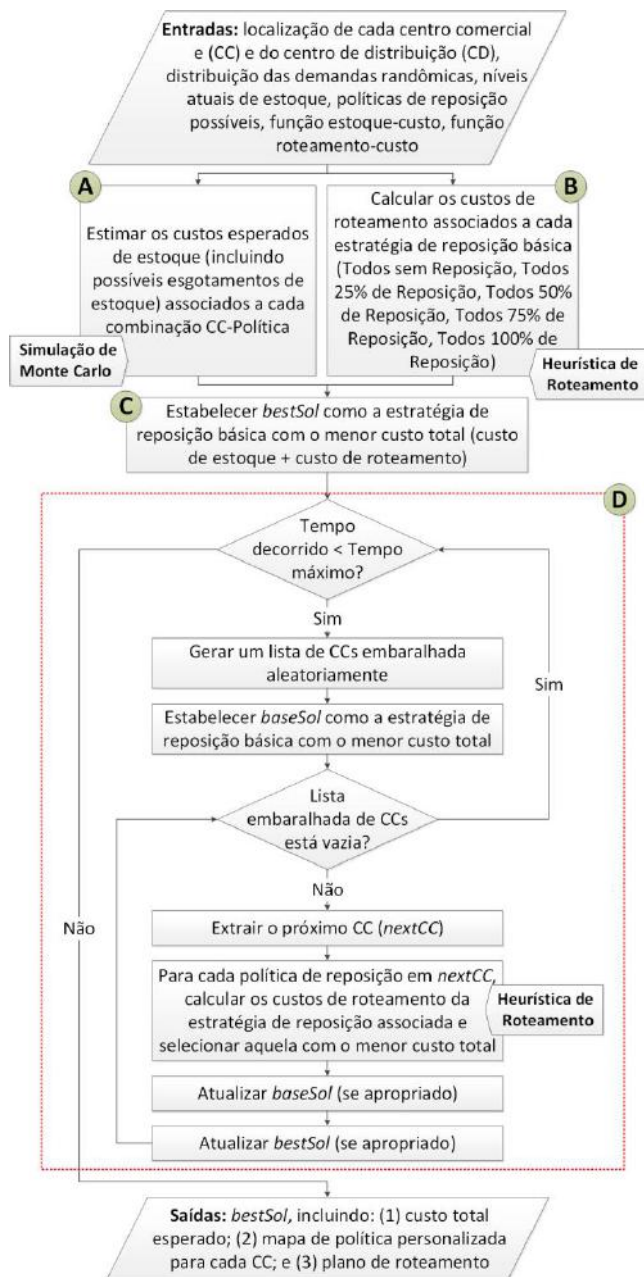


Figura 1. Fluxograma para o algoritmo *Simheuristic* [8]

Para cada centro comercial dessa lista ordenada, são calculados os custos de roteamento referentes a cada uma das políticas de reposição, mantendo a política global de melhor custo total para os demais centros comerciais. Toda vez que for encontrada uma estratégia básica com menor custo total, atualiza-se *baseSol* com a respectiva estratégia. Tanto

Caso a estratégia estabelecida em *baseSol* seja melhor que a registrada em *bestSol*, esta é atualizada com a de *baseSol*. Todos os centros comerciais dessa lista são percorridos seguindo o mesmo procedimento.

Ao terminar de percorrer os centros comerciais da lista ordenada e caso o tempo decorrido não tenha excedido o tempo máximo especificado, o processo *Multistart* é iniciado novamente e uma nova lista ordenada de centros comerciais é gerada.

As iterações terminam quando o tempo decorrido tiver ultrapassado o tempo máximo especificado e é finalmente gerada a saída com a melhor solução encontrada (*bestSol*), composta por:

- Custo total esperado;
- Mapa de política personalizada para cada centro comercial;
- Plano de roteamento.

#### 4. VARIANTE PROPOSTA PARA O ALGORITMO SIMHEURISTIC: GRANULARSIMHEURISTIC

Segundo [8], diferentes quantidades de políticas de reposição podem ser consideradas para produzir melhores soluções ao custo do aumento do esforço computacional [8]. Essa afirmação foi testada com experimentos realizados na Seção 6.2 com diferentes quantidades de políticas de reposição, entre elas 5, 7, 9, 11 e 13. Chamaremos essas *políticas de globais*. A partir dos experimentos realizados, não foram detectadas melhorias significativas e por esse motivo, apresentamos uma variante do algoritmo proposto por [8], denominado *GranularSimheuristic*.

*GranularSimheuristic* prevê algumas etapas adicionais para determinação de novas políticas chamadas de *granulares* que serão utilizadas no processo *MultiStart*. O fluxograma correspondente é apresentado na Figura 2.

O funcionamento desta variante é semelhante ao que foi proposto por [8]. Ela considera as políticas globais até o Passo C, que atribui a *bestSol* a melhor ERB. Em seguida são definidas um número *g* de políticas granulares abaixo e acima de *bestSol*.

A diferença entre os valores das políticas *dif* é um parâmetro do algoritmo e é sempre menor que a diferença entre duas políticas de reposição imediatamente consecutivas. A quantidade de políticas granulares também é um parâmetro. Por exemplo, considerando cinco políticas de reposição (nenhuma, 25%, 50%, 75% e total),  $dif=0,015$  e  $g=4$ , se a melhor estratégia de reposição básica obtida no passo C foi de 75%, as 4 políticas granulares assumirão os seguintes valores: 72,750%, 73,875%, 76,125% e 77,250% .

Também é feita uma estimativa da quantidade de mercadorias a serem entregues de acordo com cada política granular obtida e são calculados os custos esperados de estoque por meio de Simulação de Monte Carlo. É então montada a especificação do problema para o cálculo dos custos de roteamento considerando o novo conjunto de políticas granulares. Os custos de roteamento associados a cada nova estratégia de políticas também são calculados por meio da heurística CWS.

Finalmente, no processo *Multistart* temos duas opções: (i) ou apenas considerar as políticas granulares criadas e a melhor estratégia de reposição básica; ou (ii) considerar as políticas granulares e todas as políticas globais iniciais.

Assim, temos duas versões do algoritmo *GranularSimheuristic de acordo com* a classe de políticas usadas no processo *MultiStart*:

- *GranularSimheuristic com Políticas Granulares*
- *GranularSimheuristic com Políticas Granulares + Políticas Globais*

#### 5. METODOLOGIA

Como não existem *benchmarks* de instâncias de problemas disponíveis para o SIRP, foram feitas adaptações a partir de

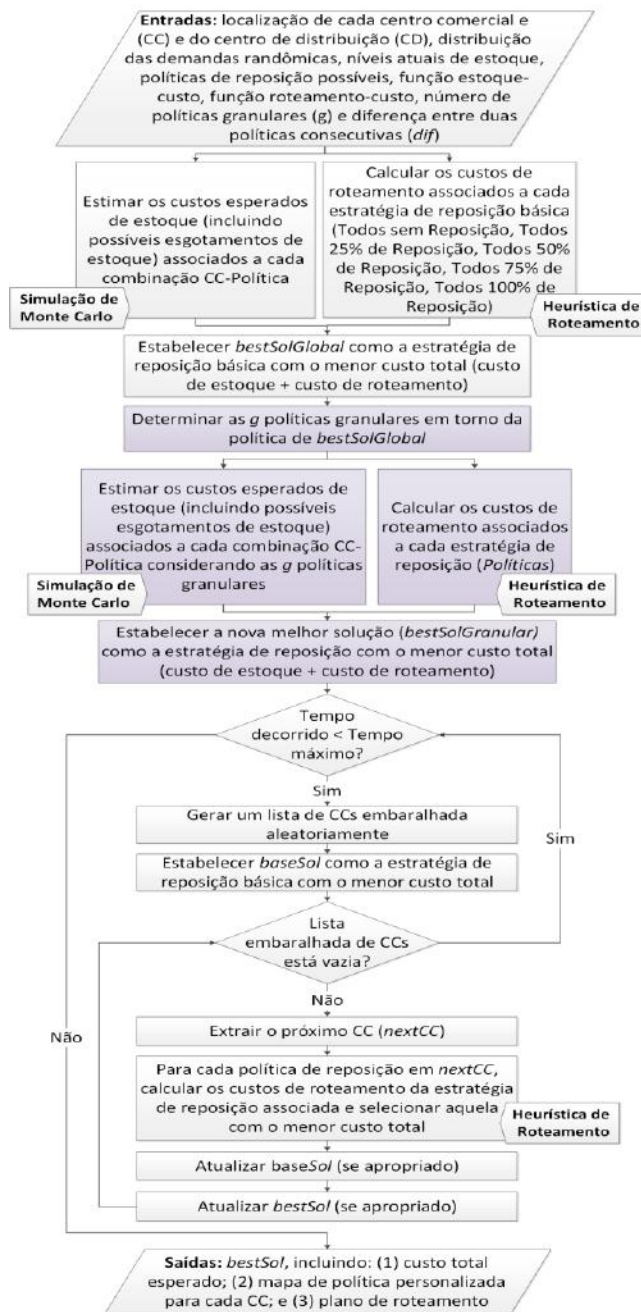


Figura 2. Fluxograma para *GranularSimheuristic*

*benchmarks* existentes com 27 instâncias de problemas definidos por Augerat [1] para CVRPs, com número de nós (centros comerciais) variando de 32 a 80. Essas instâncias também foram utilizadas e adaptadas em [8].

As instâncias para o CVRP identificam a quantidade de centros comerciais ( $n$ ) e um centro de distribuição, a quantidade de veículos ( $k$ ) e a capacidade ( $Q$ ) de cada um, as localizações dos centros comerciais e a demanda ( $d$ ) determinística de cada centro comercial. Para adaptar o problema para SIRP foram incluídos outros dados, entre eles: a esperança e a variância da demanda, a capacidade máxima de estoque ( $\hat{L}$ ), o nível atual de estoque ( $L$ ) e o custo por unidade de mercadoria em estoque. Foi considerada uma distribuição log-normal para as demandas com  $E[D_i] = d_i$  e com variância  $Var[D_i] = kd_i$ , em que  $k$  representa o nível de variância do cenário de teste:  $k = 0,25$  para baixa variância,  $k = 0,5$  para uma variância intermediária e  $k = 0,75$  para alta variância [8].

Seguindo a mesma estratégia adotada para os testes em [8], para cada centro comercial  $i$ , foi assumido como nível máximo de estoque  $\hat{L}_i = 2d_i$ . Os níveis iniciais de estoque ( $L_i$ ) foram definidos de acordo com a seguinte expressão:

- 0 se  $i$  é ímpar e é múltiplo de 3;
- $\frac{d_i}{2}$  se  $i$  é ímpar e não é múltiplo de 3;
- $d_i$  se  $i$  é par e é múltiplo de 4;
- $\frac{3d_i}{2}$  se  $i$  é par e não é múltiplo de 4.

Foram avaliados os seguintes algoritmos:

- Algoritmo *Simheuristic* com 5 políticas globais. É importante notar que em [8] foi avaliado esse algoritmo com apenas essa configuração;
- Algoritmo *Simheuristic* considerando diversas políticas globais (7, 9, 11 e 13);
- As duas versões do algoritmo *GranularSimheuristic*, propostas neste trabalho.

Os resultados foram avaliados para cada uma das 27 instâncias adaptadas do *benchmark*. Cada um dos algoritmos apresentados neste trabalho foi executado dez vezes para cada instância a fim de obter valores médios dos custos totais, de inventário e de roteamento e os respectivos desvios-padrão. Também foram medidos os tempos totais médios de cada execução por instância.

Nos testes realizados, foi adotado  $k = 0,75$ , pois com esse valor a variabilidade da demanda aleatória gerada será maior. Em todos os testes, é possível observar que a contribuição dos custos de inventário para o custo total é menor que a contribuição dos custos de roteamento. Isso é devido ao baixo valor adotado para o custo de armazenamento por unidade de mercadoria ( $\lambda = 0,05$ ).

Todos os testes foram executados em um computador com processador Intel Core I7 – 4510U @ 2.00GHz – 2.60GHz, 8 GB de RAM, em Windows 8 64 bits. Todos os algoritmos foram implementados na linguagem Python, incluindo o algoritmo proposto por [8].

## 6. RESULTADOS

Nesta seção, é feita uma análise do algoritmo *Simheuristic* considerando 5 políticas de reposição. Além disso, são feitas comparações considerando diferentes quantidades de políticas globais (5, 7, 9, 11 e 13). Finalmente, são comparados os resultados das duas versões do algoritmo proposto (*GranularSimheuristic*) e o algoritmo *Simheuristic*.

### 6.1 Avaliação do Algoritmo *Simheuristic* com 5 Políticas Globais

Os resultados obtidos pelo algoritmo *Simheuristic* com 5 políticas globais nas 27 instâncias estão apresentados na Tabela 1. As colunas nomeadas pela ERB (“Todos sem Reposição”, “Todos 25% de Reposição”, “Todos 50% de Reposição”, “Todos 75% de Reposição” e “Todos 100% de Reposição”) apresentam os resultados obtidos na simulação de Monte Carlo (colunas “Inventário”) e na Heurística de Roteamento (colunas “Roteamento”), que foram apresentados nos passos A e B da Figura 1.

Pode-se constatar que para qualquer uma das 5 ERBs e para a melhor solução obtida pelo algoritmo *Simheuristic*, os custos totais crescem à medida que a quantidade de centros comerciais se torna maior. Os custos totais para as ERBs “Todos sem Reposição” e “Todos 25% de Reposição” apresentaram valores altos e aproximados entre si, revelando que reposições de pequenas quantidades não alteram os custos de inventário significativamente. Com relação aos custos de inventário, as ERBs “Todos 75% de Reposição” e “Todos 100% de Reposição” apresentaram valores em geral menores, revelando que tais custos diminuem à medida que quantidades maiores de mercadorias são repostas nos estoques dos centros comerciais.

Comparativamente aos resultados obtidos por [8], os valores de todos os custos variaram devido à natureza randômica das demandas geradas, porém são similares. De acordo com os autores, apesar de usarem os mesmos parâmetros, a reprodução dos mesmos resultados certamente não é possível.

O tempo de execução do algoritmo completo se elevou à medida que a quantidade de centros comerciais aumentou, devido à complexidade do grafo envolvido em cenários em que há mais nós (centros comerciais).

### 6.2 Avaliação do Algoritmo *Simheuristic* para diferentes Políticas Globais

A Tabela 2 apresenta os resultados do Algoritmo *Simheuristic*, considerando quantidades diferentes de políticas globais. Nessa tabela estão destacados os menores valores de custo total em cada instância. Comparando os custos totais médios obtidos pelo algoritmo *Simheuristic*, observa-se que para 11 e 13 políticas são produzidos os menores valores, destacando-se em relação às demais.

A Figura 3 apresenta os tempos totais médios de execução do algoritmo *Simheuristic* para 5, 7, 9, 11 e 13 políticas, constata-se que, os custos totais médios são muito próximos entre si até a instâncias A-n46-k7 (46 centros comerciais e 7 veículos de transporte), passando a crescer significativamente nas outras instâncias em função da quantidade de políticas globais consideradas no algoritmo.

Os tempos totais médios se tornam menores para 5 políticas globais e maiores para 13 políticas globais, revelando que a adoção de mais políticas no algoritmo proposto por [8] leva a um custo computacional maior no tempo de execução.

Associando essa análise sobre o tempo de execução à que foi feita anteriormente sobre os custos totais médios, constata-se que o aumento na quantidade de políticas globais com um tempo de execução maior (caso da variante de 13 políticas globais) levou à obtenção de resultados com singela melhoria em relação aos obtidos com 5 políticas globais.



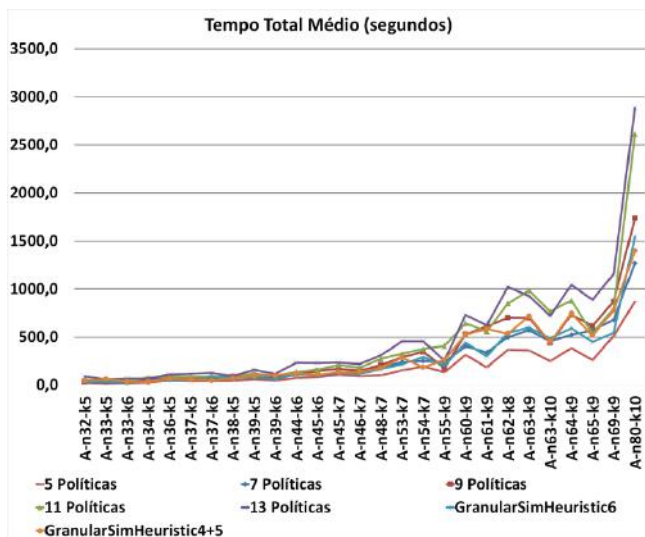


Figura 3. Comparação entre os tempos totais médios de execução

### 6.3 Avaliação das duas Versões do Algoritmo GranularSimheuristic

Nesta seção, comparamos o algoritmo *Simheuristic* com as duas versões do algoritmo proposto fixando a quantidade de políticas granulares e globais:

- *GranularSimheuristic* com 6 Políticas Granulares (*GranularSimheuristic6*): inclui 6 políticas granulares e a política da melhor ERB considerando 5 políticas de reposição;
- *GranularSimheuristic* com 4 Políticas Granulares + 5 Políticas Globais (*GranularSimheuristic4+5*): inclui 4 políticas granulares e todas as 5 políticas globais;

Os resultados obtidos para os custos totais médios para essas duas versões também cresceram segundo o aumento no número de centros comerciais presentes nas instâncias, conforme se pode observar na Tabela 3. Houve predominância de melhores resultados para *GranularSimheuristic4+5* e apenas em duas instâncias (A-n38-k5 e A-n39-k6), *GranularSimheuristic6* obteve custos totais médios menores.

Os tempos totais médios de execução dos algoritmos para cada uma das instâncias também aumentaram conforme o crescimento no número de centros comerciais. Para as primeiras instâncias, resultaram muito próximos, divergindo significativamente a partir da instância A-n53-k7 (53 centros comerciais). Na medição dos tempos totais médios de execução dos algoritmos, *GranularSimheuristic6* apresentou, de forma geral, valores menores que *GranularSimheuristic4+5*.

### 6.4 Comparação entre as Diversas Propostas

Os resultados demonstraram que houve concentração dos melhores resultados nos custos totais médios para *GranularSimheuristic4+5* na maioria das instâncias. Essa versão também apresentou o melhor custo total médio geral (soma de todos os custos totais médios de todas as instâncias), conforme se pode observar na Tabela 4.

Tabela 3. Resultados para GranularSimheuristic6 e GranularSimheuristic4+5

Instância	Melhor Solução (Média para 10 Execuções)							
	GranularSimheuristic6				GranularSimheuristic4+5			
	Custo		Tempo		Custo		Tempo	
Total	Desvio Padrão	Total (s)	Desvio Padrão	Total	Desvio Padrão	Total (s)	Desvio Padrão	
A-n32-k5	791,99	9,86	29,52	2,32	787,33	10,20	45,23	16,64
A-n33-k5	613,73	2,27	31,35	3,27	610,82	2,84	58,70	19,08
A-n33-k6	672,42	2,58	25,59	0,94	664,24	7,61	36,08	14,94
A-n34-k5	719,98	3,29	41,10	2,46	704,81	9,12	31,10	13,87
A-n36-k5	731,14	3,09	55,89	1,73	723,41	1,91	65,91	0,38
A-n37-k5	658,27	4,69	45,18	2,43	645,69	5,19	57,80	19,36
A-n37-k6	828,22	3,35	80,43	14,76	804,88	22,40	45,00	36,11
A-n38-k5	669,70	5,13	71,49	15,19	676,93	11,88	70,41	31,01
A-n39-k5	744,32	4,86	76,91	4,43	733,33	7,44	96,95	15,79
A-n39-k6	758,23	3,53	61,04	0,41	772,37	13,20	100,57	27,15
A-n44-k6	871,77	3,81	110,05	1,48	867,65	8,95	121,77	26,06
A-n45-k6	863,94	4,09	103,70	7,23	857,24	9,49	106,49	47,76
A-n45-k7	1076,56	4,32	132,47	0,21	1046,22	39,21	134,99	85,31
A-n46-k7	850,01	3,57	113,73	1,49	843,26	9,09	136,12	94,60
A-n48-k7	999,56	2,42	167,53	2,68	992,30	6,56	177,60	16,12
A-n53-k7	909,48	5,87	215,04	9,92	907,86	5,84	286,10	74,03
A-n54-k7	1016,49	5,17	290,90	9,98	995,28	13,30	185,72	141,11
A-n55-k9	1020,02	1,65	202,70	7,73	1014,62	10,14	261,82	162,83
A-n60-k9	1254,62	8,06	439,63	15,12	1249,92	9,89	527,24	218,69
A-n61-k9	897,21	3,13	301,08	1,00	896,44	4,05	584,93	194,64
A-n62-k8	1176,86	9,06	544,48	44,26	1148,82	31,99	535,46	277,50
A-n63-k9	1350,70	6,83	597,29	18,90	1342,61	12,23	715,42	289,97
A-n63-k10	1160,34	4,24	486,29	58,61	1145,04	11,05	449,74	256,19
A-n64-k9	1169,57	7,58	590,87	11,48	1168,26	11,48	746,18	84,86
A-n65-k9	1053,60	3,25	448,55	75,22	1044,35	12,54	515,81	319,37
A-n69-k9	1110,08	4,77	543,28	47,00	1106,50	10,57	774,53	373,92
A-n80-k10	1623,06	10,55	1542,85	65,63	1609,93	10,12	1396,94	736,23

Tabela 4. Comparação entre os custos totais médios

Instância	Custos Totais Médios (Roteamento + Estoque) com 10 Execuções por Instância						
	5 Políticas	7 Políticas	9 Políticas	11 Políticas	13 Políticas	GranularSim heuristic6	GranularSim heuristic4+5
A-n32-k5	786,75	799,95	781,96	789,08	805,54	791,99	787,33
A-n33-k5	629,76	631,23	633,27	617,26	621,84	613,73	610,82
A-n33-k6	686,19	685,99	687,32	666,58	688,39	672,42	664,24
A-n34-k5	711,02	732,63	710,50	718,60	708,30	719,98	704,81
A-n36-k5	728,88	760,81	724,63	747,13	724,61	731,14	723,41
A-n37-k5	649,07	701,52	649,30	680,94	652,12	658,27	645,69
A-n37-k6	801,72	865,20	797,57	858,23	830,41	828,22	804,88
A-n38-k5	681,27	673,99	697,36	663,59	666,35	669,70	676,93
A-n39-k5	753,79	751,20	746,41	744,06	750,91	744,32	733,33
A-n39-k6	780,80	773,40	774,57	759,23	766,36	758,23	772,37
A-n44-k6	861,91	864,35	867,47	857,14	861,75	871,77	867,65
A-n45-k6	859,40	854,30	855,97	855,83	855,23	863,94	857,24
A-n45-k7	1108,28	999,30	1028,41	1039,39	993,21	1076,56	1046,22
A-n46-k7	845,87	855,92	849,01	856,93	838,20	850,01	843,26
A-n48-k7	988,20	1051,15	994,29	1035,46	989,36	999,56	992,30
A-n53-k7	905,69	981,02	907,55	973,85	906,99	909,48	907,86
A-n54-k7	1008,07	1009,68	1003,04	1019,31	997,39	1016,49	995,28
A-n55-k9	1020,10	1028,73	1016,86	1005,79	1024,00	1020,02	1014,62
A-n60-k9	1244,01	1243,98	1244,07	1267,58	1232,97	1254,62	1249,92
A-n61-k9	894,74	902,38	897,43	910,86	894,51	897,21	896,44
A-n62-k8	1160,95	1172,78	1159,46	1204,89	1167,75	1176,86	1148,82
A-n63-k9	1334,51	1428,26	1338,17	1411,63	1334,46	1350,70	1342,61
A-n63-k10	1147,82	1184,14	1139,23	1202,02	1141,61	1160,34	1145,04
A-n64-k9	1157,49	1277,45	1164,42	1224,05	1161,02	1169,57	1168,26
A-n65-k9	1042,49	1117,99	1041,51	1108,43	1047,01	1053,60	1044,35
A-n69-k9	1114,39	1105,22	1113,75	1096,03	1109,39	1110,08	1106,50
A-n80-k10	1610,26	1708,09	1606,83	1707,56	1613,17	1623,06	1609,93
Total	23903,17	24452,55	23823,53	24313,89	23769,67	23968,78	23750,17

Também foram coletados os menores custos totais para cada instância na execução de cada algoritmo, apresentados na Tabela 5, em que é possível notar uma maior concentração de melhores resultados (menores custos) no algoritmo *Simheuristic* com 11 e 13 políticas globais. No entanto, *GranularSimheuristic4+5*

apresentou o menor valor total para a soma de todos os custos totais de todas as instâncias.

Conforme apresentado na Figura 3, os melhores tempos foram obtidos para o algoritmo *Simheuristic* com 5 políticas globais para todas as instâncias. Observando as demais variantes, nota-se que quanto maior a quantidade de políticas, maior é o tempo total de execução, que se torna ainda maior para as instâncias com mais centros comerciais.

Em suma, o que se pode notar é que os melhores resultados foram produzidos pelo algoritmo *GranularSimheuristic4+5*, mesmo que o tempo total médio de execução do algoritmo para cada instância tenha ficado acima da variante de 5 políticas globais original (mas abaixo das variantes de 9, 11 e 13 políticas globais).

Tabela 5. Comparação entre os menores custos totais

Instância	Menores Custos Totais (Roteamento + Estoque) dentre 10 Execuções por Instância						GranularSimheuristic4+5
	5 Políticas	7 Políticas	9 Políticas	11 Políticas	13 Políticas	GranularSimheuristic6	
A-n32-k5	769,47	783,23	767,93	775,82	765,43	777,53	773,74
A-n33-k5	603,28	608,69	604,06	582,58	601,60	608,75	603,22
A-n33-k6	679,53	672,67	679,47	655,48	680,53	667,44	647,54
A-n34-k5	693,89	711,16	690,76	701,93	691,18	716,06	689,79
A-n36-k5	718,52	745,97	716,40	735,43	715,22	726,49	719,42
A-n37-k5	633,77	650,20	638,42	640,95	638,55	650,08	634,30
A-n37-k6	768,13	844,08	762,32	823,21	812,84	822,61	771,58
A-n38-k5	652,13	657,70	683,74	646,56	651,68	662,37	654,51
A-n39-k5	736,11	740,46	731,59	723,03	734,51	738,79	723,61
A-n39-k6	762,76	762,25	756,45	744,63	752,03	753,55	754,83
A-n44-k6	850,88	849,63	851,32	844,94	854,54	867,09	851,82
A-n45-k6	829,64	833,42	826,13	842,66	831,70	859,25	835,37
A-n45-k7	1090,91	988,82	1014,45	1003,73	968,56	1069,38	986,54
A-n46-k7	828,93	810,56	837,89	833,58	804,94	847,06	828,08
A-n48-k7	961,95	1012,96	961,38	993,25	960,89	995,78	979,29
A-n53-k7	891,89	953,92	899,10	946,56	899,53	901,12	899,61
A-n54-k7	985,89	992,48	973,72	988,31	961,88	1008,07	969,45
A-n55-k9	1003,09	987,52	983,54	958,17	1008,76	1017,64	989,97
A-n60-k9	1220,16	1205,12	1214,08	1229,01	1162,11	1231,60	1230,52
A-n61-k9	888,54	881,90	890,03	888,23	887,84	893,89	889,90
A-n62-k8	1106,98	1144,29	1116,01	1180,60	1156,27	1162,87	1086,51
A-n63-k9	1324,97	1399,47	1322,14	1395,58	1316,54	1336,80	1330,06
A-n63-k10	1125,97	1138,88	1106,74	1187,95	1122,20	1153,55	1122,38
A-n64-k9	1140,26	1241,31	1148,71	1204,43	1140,30	1157,74	1142,57
A-n65-k9	1026,82	1107,34	1024,69	1095,45	1029,21	1048,18	1023,58
A-n69-k9	1102,33	1090,47	1091,04	1067,86	1095,10	1102,22	1088,00
A-n80-k10	1582,89	1650,95	1583,70	1668,72	1593,71	1607,24	1596,55
Total	24979,69	25465,46	24875,81	25358,65	24837,64	25383,15	24822,71

## 6.5 DISCUSSÃO

Os resultados obtidos demonstraram que os custos totais médios obtidos pelo algoritmo *Simheuristic* com 7, 9, 11 e 13 políticas globais não foram significativamente melhores que o de 5 políticas globais proposto e avaliado por [8].

O algoritmo *GranularSimheuristic4+5* apresentou resultados relativamente melhores que o *Simheuristic* original, que considera apenas políticas globais. A soma dos custos totais médios e dos menores custos para cada instância foi o menor de todas os algoritmos considerados. Semelhante avaliação cabe sobre os tempos totais médios de execução: o algoritmo *GranularSimheuristic4+5* foi executado em menor tempo que o algoritmo *Simheuristic* com 9, 11 e 13 políticas globais.

Nota-se, portanto, que o aumento da quantidade de políticas globais consideradas leva a um aumento do custo no tempo de execução do algoritmo proposto por [8] e ainda produz resultados piores que o algoritmo *GranularSimheuristic4+5*.

## 7. CONCLUSÃO

Neste trabalho, foi proposta uma variante do algoritmo *Simheuristic* [8]. Inicialmente, esperava-se que com o aumento da quantidade de políticas no algoritmo *Simheuristic*, fosse possível obter resultados melhores. Porém, ficou evidenciado neste trabalho que o aumento no número de políticas globais no *Simheuristic* não produz ganhos significativos. A decisão de usar apenas políticas granulares também não trouxe resultados satisfatórios.

Por outro lado, a adoção de políticas granulares combinadas com políticas globais levou a resultados relativamente melhores e, no geral, produziu uma redução no somatório dos menores custos totais em relação ao *Simheuristic* com 5 políticas globais. Esse resultado sugere que vale mais a pena investir numa combinação de políticas granulares com políticas globais em vez de investir em variantes que adotem somente políticas globais ou somente políticas granulares.

## 8. REFERÊNCIAS

- [1] Augerat et al, 1995. *Capacitated VRP Instances*. Networking and Emerging Optimization Research Group. Universidad de Málaga. Disponível em: <http://neo.lcc.uma.es/vrp/vrp-instances/capacitated-vrp-instances/>. Acesso em: 10 de maio 2015.
- [2] Cáceres-Cruz, J., Juan, A. A., Bektas, T., Grasman, S. E., Faulin, J. 2012. *Combining Monte Carlo Simulation with Heuristics for Solving the Inventory Routing Problem with Stochastic Demands*. Proceedings of the 2012 Winter Simulation Conference, article n. 274, p. 1–9, 2012.
- [3] Clarke, G., Wright, J.W. 1964. *Scheduling of vehicles from a central depot to a number of delivery points*. Oper. Res. v. 12, p. 568–581, 1964.
- [4] Coelho, L. C., Cordeau, J. F., Laporte, G. *Thirty Years of Inventory Routing*. Transportation Science, v 48(1), p. 1-19, 2013.
- [5] Federgruen, A., Zipkin P. H. 1984. *A combined vehicle-routing and inventory allocation problem*. Operations Research, v 32(5), p. 1019-1037, 1984.
- [6] Fisher, M. L., Jaikumar, R. 1981. *A generalized assignment heuristic for vehicle routing*. Networks, v. 11(2), p. 109-124, 1981.
- [7] Jaillet, P., Bard, J. F., Huang, L., & Dror, M. 2002. *Delivery cost approximations for inventory routing problems in a rolling horizon framework*. Transportation Science, v. 36(3), p. 292-300, 2002.
- [8] Juan, A. A., Grasman, S. E., Cáceres-Cruz, J., Bektas, T. 2014. *A Simheuristic Algorithm for the Single-Period Stochastic Inventory-Routing Problem with Stock-Outs*. Simulation Modeling Practice and Theory. Elsevier, v. 46, p. 40–52, 2014.
- [9] Oliveira, R. A. C., Delgado, K. V. *Sistema para roteamento de veículos capacitados aplicando Métodos de Monte Carlo*. SBSI - XI Simpósio Brasileiro de Sistemas de Informação, p. 1-8, 2015.
- [10] Oliveira, R. A. C., Delgado, K. V., Moreira D. A. M. *Sistema para roteamento de veículos capacitados aplicando Métodos de Monte Carlo*. iSys: Revista Brasileira de Sistemas de Informação, v. 8, p. 42-63, 2015.