PROJETO DE REDE DE UMA CADEIA DE SUPRIMENTOS MULTIPRODUTO, CAPACITADA E EM CIRCUITO FECHADO

Jéssica Diogo de Vasconcelos[1], **Elisangela Martins de Sá**[1], **Sérgio Ricardo de Souza**[1]

[1] Centro Federal de Educação Tecnológica de Minas Gerais
Av. Amazonas, 7675, CEP: 30510-000, Belo Horizonte, MG, Brasil
jess.dvasconcelos@gmail.com, elisangelamartins@cefetmg.br, sergio@cefetmg.br

RESUMO

O problema de projeto de rede de cadeias de suprimentos em circuito fechado capacitada consiste em definir a localização das instalações que compõem o canal reverso, como os centros de coleta e centros de reparo, e rotear os produtos ao longo de todas as instalações pertencentes à cadeia de suprimentos, considerando a estrutura já existente do canal direto, composto por fábricas, centros de distribuição e pontos de varejo. Para resolver instâncias do problema, são propostos algoritmos heurísticos baseados nas meta-heurísticas ILS e VND, considerando duas estratégias de busca: busca Primeiro de Melhora e busca Melhor Vizinho. Os resultados dos experimentos computacionais mostram que, apesar da qualidade das soluções obtidas pela estratégia de busca Melhor Vizinho ser, em média, melhor que a qualidade das soluções obtidos pela estratégia de busca Primeiro de Melhora, ela demanda maior tempo computacional.

PALAVRAS CHAVE. Cadeia de Suprimentos em Circuito Fechado, *Iterated Local Search* (ILS), *Variable Neighborhood Descent* (VND).

Metaheurísticas. Logística e Transportes.

ABSTRACT

The capacitated closed-loop supply chain network design problem consists of defining the location of the facilities that composes the reverse channel, the collection centers, and repair centers, and routing product throughout all facilities belonging to the supply chain network, considering the existing structure of the direct channel, composed of factories, distribution centers, and retail locals. To solve instances of this problem, it is proposed heuristic algorithms, based on the meta-heuristics ILS and VND, considering two search strategies: Best improvement and First improvement. The computational experiments results show that, although the quality of the solutions achieved by the Best Improvement strategy is, on average, better than the quality of the solutions obtained by the First Improvement strategy, it demands more computational time.

KEYWORDS. Closed Loop Supply Chain. Iterated Local Search (ILS). Variable Neighborhood Descent (VND).

Metaheuristics. Logistics and Transport.

1. Introdução

As cadeias de suprimentos em circuito fechado (*Closed Loop Supply Chain* – CLSC) estendem a definição tradicional de cadeias de suprimentos, explorando explicitamente a sinergia entre dois fluxos, o fluxo direto e o fluxo reverso. Assim, as CLSCs envolvem questões associadas à transformação e fluxo de produtos novos e de produtos remanufaturados, criando um desafio adicional para o problema de projeto e planejamento da rede. O projeto de uma rede CLSC é uma estratégia que pode ser adotada no meio empresarial por, além de trazer benefícios sociais e ambientais, ser altamente lucrativa [Srivastava, 2007].

O fluxo de produtos em uma rede CLSC pode ser caracterizado pelos movimentos direto e reverso, que estão inter-relacionados. O fluxo direto consiste no fluxo de produtos de uma rede tradicional, em que produtos são enviados de fabricantes ou fornecedores para varejistas através de centros de distribuição (CDs) [Geoffrion e Graves, 1974]. Os CDs atuam como locais de armazenagem e de consolidação e desconsolidação de carga. A Figura 1(a) ilustra uma cadeia de suprimentos tradicional.



(a) Fluxo de produtos em uma cadeia de suprimento tradicional.

(b) Fluxo de produtos em uma rede CLSC.

Figura 1: Fluxo de produtos em uma cadeia de suprimento tradicional e em uma rede CLSC.

A logística reversa se refere ao processo de movimentação de bens do ponto onde está localizado o consumidor final, com o propósito de garantir o descarte adequado, ou com o objetivo de capturar valor, através de reciclagem, reuso ou remanufatura [Rogers e Tibben-Lembke, 1999]. O fluxo reverso em uma rede CLSC é o fluxo de material no canal de coleta, que funciona da seguinte forma. Os subprodutos devolvidos pelos consumidores aos locais de varejo são enviados para centros de reparo (CRs) para a sua recuperação (remanufatura), através de centros intermediários de coleta (CCs). Os CCs atuam como locais de classificação e armazenamento desses subprodutos. Em alguns casos, devido à garantia fornecida pelos fabricantes, clientes não distinguem produtos remanufaturados e novos. Como consequência, os produtos remanufaturados podem ser usados para satisfazer à demanda do consumidor no canal direto, levando, assim, a uma inter-relação entre os fluxos direto e reverso. Então, para uma implementação sistemática de uma estratégia de recuperação de produtos, como a remanufatura, deve-se considerar simultaneamente os fluxos diretos e fluxos reversos [Sahyouni et al., 2007; Üster et al., 2007].

Ao se considerar os fluxos direto e reverso em uma cadeia de suprimentos, o resultado é uma CLSC. Com o propósito de projetar uma rede CLSC, Easwaran e Uster [2009] propõem uma mudança em uma cadeia de produtos existente, responsável pelo fluxo direto, que consiste no projeto da rede de canais reversos e na integração do fluxo direto e reverso de produtos. Especificamente, eles consideram um problema de projeto de rede capacitada para multiprodutos com

instalações de manufatura, distribuição, coleta e remanufatura que atendem um conjunto de varejistas. A Figura 1(b) ilustra uma rede CLSC genérica.

Este trabalho aborda o problema de projeto de rede CLSC capacitada (PPRCLSCC) proposto por Easwaran e Uster [2009]. Este problema consiste em definir a localização das instalações que compõem o canal reverso de uma rede CLSC, ou seja, os CCs e os CRs, considerando uma estrutura de canal direto já existente. Além disso, deve-se determinar o roteamento do fluxo de produtos ao longo de todas as instalações pertencentes à cadeia de suprimentos, de forma a minimizar o custo total e considerando restrições de capacidade. Devido à complexidade do PPRCLSCC, para resolver instâncias do problema são propostos dois algoritmos que integram as metaheurísticas *Iterated local search* (ILS) e *Variable Neighboorhoood Decent* (VND). A diferença entre os dois algoritmos se refere à estratégia de busca no VND, que é feita considerando a estratégia Primeiro de melhora e a estratégia Melhor Vizinho.

Este trabalho está organizado da seguinte forma. A Seção 2 apresenta uma caracterização do problema. A Seção 3 apresenta uma revisão da literatura dos trabalhos relacionados. A Seção 4 descreve os algoritmos propostos. A Seção 5 apresenta os resultados dos experimentos computacionais, enquanto a Seção 6 apresenta as conclusões deste trabalho e propostas de trabalhos futuros.

2. Descrição do Problema

O presente trabalho foi baseado no problema apresentado por Üster et al. [2007], em que fabricantes de equipamentos originais produzem e distribuem produtos que são caracterizados pela sua alta durabilidade, ciclos de vida longos e alto valor de recuperação. Esses produtos utilizam um canal direto, que é constituído pelas plantas (*New Product Plants* – NPPs) e por centros de distribuição (CDs), para satisfazer as demandas nos locais de varejo. Motivados por incentivos econômicos e obrigações ambientais, são adotadas práticas de recuperação de produtos por meio da implementação de um canal reverso, que, ao ser integrado à rede da cadeia de produtos existente, gera uma rede CLSC.

Easwaran e Uster [2009] propõem duas mudanças em relação ao cenário apresentado por Üster et al. [2007]. Primeiramente, propuseram a inclusão de restrições de capacidade, generalizando o problema de projeto de rede CLSC, para que se possa considerar as questões associadas às capacidades de processamento finito nas atividades de produção, distribuição e coleta de produtos. Na prática, tem-se limitações de capacidade, devido à limitações de espaço de armazenamento, recursos finitos para processamento e manuseio, limitações de mão-de-obra ou uma combinação desses fatores. A segunda alteração é a possibilidade de fontes múltiplas, em que um varejista pode receber produtos de diferentes CDs e enviar os produtos retornados para diferentes CCs.

De acordo com Easwaran e Uster [2009], a rede CLSC pode ser descrita como um sistema de três níveis, que envolve componentes dos canais direto e reverso. O primeiro nível consiste nos varejistas, que executam duas funções nas atividades dos canais direto e reverso. Eles satisfazem a demanda do mercado, por meio do canal direto, e recebem os produtos devolvidos, que serão remanufaturados no sistema. Cada varejista satisfaz dois tipos de demanda, que podemos classificar como demanda nova e demanda induzida. Na segunda camada da rede, tem-se dois tipos de instalações: CDs, no canal direto, e CCs, no canal reverso. No canal direto, cada varejista recebe produtos para atender sua demanda total, nova e induzida, vindos de um conjunto de CDs que operam sob restrições de capacidade. No canal reverso, temos um conjunto de CCs com capacidade finita, para os quais os produtos devolvidos pelos consumidores são enviados pelos varejistas. As quantidades de remessas nesses CCs dependem da fração de retorno de cada produto a um determinado varejista. O terceiro nível da rede é composto por NPPs, que abastecem o canal direto com produtos novos, e CRs, que conectam os canais reverso e direto. Assume-se que cada produto no sistema

pode ser produzido em várias NPPs com capacidade finita, mas apenas um CR remanufatura cada tipo de produto a uma taxa determinada pela fração de recuperação. Considera-se que a saída de produtos de um CR é totalmente injetada de volta no sistema, através do canal direto (por intermédio dos CDs). Produtos novos e produtos remanufaturados não podem ser distinguidos, ou seja, não é possível diferenciar um produto vindo de um CR de um produto vindo de uma NPP. Além disso, assume-se que os NPPs e os CRs atendem a demanda total (nova e induzida) dos varejistas.

O PPRCLSCC abordado neste trabalho busca determinar (i) as melhores localizações dos CRs e dos CCs (dentro dos seus respectivos conjuntos de locais candidatos), assumindo que as NPPs, os CDs e os varejistas estão dispostos em locais conhecidos, e (ii) o fluxo ótimo de produtos nos canais direto e reverso, de modo que o custo total de localização, processamento (NPPs, CRs, CCs e CDs) e transporte seja minimizado. Assume-se que as capacidades nos CDs e CCs representam capacidades agregadas que podem ser compartilhadas por todos os produtos. Assim, com o propósito de incorporar a não uniformidade no uso da capacidade, utilizou-se um coeficiente específico para cada produto.

2.1. Definições e notação

Neste trabalho $\mathcal{P}$, $\mathcal{R}$, $\mathcal{K}$ e $\mathcal{D}$ representam, respectivamente, o conjunto de produtos, o conjunto de varejistas, o conjunto de candidatos a CCs e o conjunto de CDs. Além disso, para cada produto $p \in \mathcal{P}$, define-se $\mathcal{S}_p$ como o conjunto de candidatos a CR para p e $\mathcal{T}_p$ como o conjunto de NPPs que fabricam p.

Os custos fixos de instalação são representados pelo parâmetro $F_k^{'}$, que denota o custo fixo de se abrir um CC na localização $k \in \mathcal{K}$, e pelo parâmetro F_{ps}, que representa o custo fixo de se abrir um CR para o produto $p \in \mathcal{P}$ na localização $s \in \mathcal{S}_p$. O custo unitário de transporte da localização i até a localização j para $i, j \in \mathcal{R} \cup \mathcal{D} \cup \mathcal{K} \cup \mathcal{S}_p \cup \mathcal{T}_p$ é representado pelo parâmetro G_{ij}. Além dos custos fixos e de transporte, é atribuído um custo de processamento a cada unidade da rede no segundo e terceiro nível. O parâmetro η_{pd} denota o custo unitário de processamento do produto $p \in \mathcal{P}$ no CD $d \in \mathcal{D}$; κ_{pk} representa o custo unitário de processamento do produto $p \in \mathcal{P}$ no CC $k \in \mathcal{K}$; ν_{pt} representa o custo unitário de fabricação do produto $p \in \mathcal{P}$ no NPP $t \in \mathcal{T}_p$; e ρ_{ps} representa o custo unitário de remanufatura do produto $p \in \mathcal{P}$ enviado para o CR $s \in \mathcal{S}_p$. Os parâmetros D_{rp} e $D_{rp}^{'}$ representam a demanda induzida e a demanda nova, respectivamente, do varejista $r \in \mathcal{R}$ pelo produto $p \in \mathcal{P}$. O parâmetro δ_{pk} denota a fração de retorno do produto $p \in \mathcal{P}$ ao varejista $r \in \mathcal{R}$ e o parâmetro α_{ps} representa a fração recuperada do produto $p \in \mathcal{P}$ no RPP $s \in \mathcal{S}_p$.

Seja C_d a capacidade de processamento/armazenamento no CD $d \in \mathcal{D}$; B_k a capacidade de processamento/armazenamento no CC $k \in \mathcal{K}$; e Q_{pt} a capacidade de produção/processamento de $p \in \mathcal{P}$ no NPP $t \in \mathcal{T}_p$. Além destes parâmetros de capacidade, o parâmetro γ_p denota o coeficiente de capacidade de processamento/armazenamento associado à distribuição de produtos $p \in \mathcal{P}$, enquanto o parâmetro β_p representa o coeficiente de capacidade de processamento/armazenamento associado à coleta de produtos $p \in \mathcal{P}$. Por fim, temos os seguintes parâmetros usados na modelagem das restrições: M_{prk}, cujos valores são $\min\{\delta_{rp}D_{rp},\ B_k/\beta_p\}$ para $p \in \mathcal{P}$, $r \in \mathcal{R}$, $k \in \mathcal{K}$; e $M_{pk}^{'}$, cujos valores são $\min\{\sum_{r\in\mathcal{R}} \delta_{rp}D_{rp}, B_k/\beta_p\}$ para $p \in \mathcal{P}$, $k \in \mathcal{K}$.

A formulação matemática para o PPRCLSCC utiliza os seguintes conjuntos de variáveis. A variável binária v_{ps} indica se o produto $p \in \mathcal{P}$ é reparado no CR $s \in \mathcal{S}p$ ($v_{ps} = 1$) ou não ($v_{ps} = 0$). A variável binária c_k indica se o CC $k \in \mathcal{K}$ será aberto ($c_k = 1$) ou não ($c_k = 0$). A variável de fluxo u_{prk} representa a quantidade do produto $p \in \mathcal{P}$ que é enviado do varejista $r \in \mathcal{R}$ para o CC $k \in \mathcal{K}$. A variável de fluxo w_{pdr} representa a quantidade do produto $p \in \mathcal{P}$ que é enviado do CD $d \in \mathcal{D}$ para o varejista $r \in \mathcal{R}$. A variável de fluxo x_{pks} representa a quantidade do produto

$p \in \mathcal{P}$ que é enviado do CC $k \in \mathcal{K}$ para o CR $s \in \mathcal{S}_p$. A variável de fluxo y_{psd} representa a quantidade do produto $p \in \mathcal{P}$ enviada do CR $s \in \mathcal{S}_p$ para o CD $d \in \mathcal{D}$. A variável de fluxo z_{ptd} representa a quantidade do produto $p \in \mathcal{P}$ enviada do NPP $t \in \mathcal{T}_p$ para o CD $d \in \mathcal{D}$.

2.2. Formulação matemática

A formulação matemática para o PPRCLSCC [Easwaran e Uster, 2009] é dada por:

$$\min \sum_{p\in\mathcal{P}}\sum_{s\in\mathcal{S}_p} F_{ps}v_{ps} + \sum_{k\in\mathcal{K}} F_k^{'}c_k + \sum_{p\in\mathcal{P}}\Bigg[\sum_{r\in\mathcal{R}}\sum_{k\in\mathcal{K}}(G_{rk}+\kappa_{pk})u_{prk} + \sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_p}(G_{ks}x_{pks}) + \sum_{s\in\mathcal{S}_p}\sum_{d\in\mathcal{D}}(G_{sd}+\rho_{ps})y_{psd} + \sum_{t\in\mathcal{T}_p}\sum_{d\in\mathcal{D}}(G_{td}+\nu_{pt})z_{ptd} + \sum_{d\in\mathcal{D}}\sum_{r\in\mathcal{R}}(G_{dr}+\eta_{pd})w_{pdr}\Bigg] \quad (1)$$

s. a
$$\sum_{s\in\mathcal{S}_p} v_{ps} = 1 \qquad \forall p \in \mathcal{P} \quad (2)$$
$$\sum_{k\in\mathcal{K}} u_{prk} = \delta_{rp}D_{rp} \qquad \forall p \in \mathcal{P}, \forall r \in \mathcal{R}, \quad (3)$$
$$\sum_{s\in\mathcal{S}_p} x_{pks} - \sum_{r\in\mathcal{R}} u_{prk} = 0 \qquad \forall p \in \mathcal{P}, \forall k \in \mathcal{K}, \quad (4)$$
$$\sum_{k\in\mathcal{K}} x_{pks} - \sum_{r\in\mathcal{R}} \delta_{rp}D_{rp}v_{ps} = 0 \qquad \forall p \in \mathcal{P}, \forall s \in \mathcal{S}_p, \quad (5)$$
$$\sum_{d\in\mathcal{D}} y_{psd} - \alpha_{ps}\sum_{r\in\mathcal{R}} \delta_{rp}D_{rp}v_{ps} = 0 \qquad \forall p \in \mathcal{P}, \forall s \in \mathcal{S}_p, \quad (6)$$
$$\sum_{s\in\mathcal{S}_p} y_{psd} + \sum_{t\in\mathcal{T}_p} z_{ptd} - \sum_{r\in\mathcal{R}} w_{pdr} = 0 \qquad \forall p \in \mathcal{P}, \forall d \in \mathcal{D}, \quad (7)$$
$$\sum_{d\in\mathcal{D}} w_{pdr} = (D_{rp} + D_{rp}^{'}) \qquad \forall p \in \mathcal{P}, \forall r \in \mathcal{R}, \quad (8)$$
$$\sum_{p\in\mathcal{P}}\sum_{r\in\mathcal{R}} \beta_p u_{prk} \leq B_k c_k \qquad \forall k \in \mathcal{K}, \quad (9)$$
$$u_{prk} \leq M_{prk}c_k \qquad \forall p \in \mathcal{P}, \forall r \in \mathcal{R}, \forall k \in \mathcal{K}, \quad (10)$$
$$\sum_{s\in\mathcal{S}_p} x_{psk} \leq M_{pk}^{'}c_k \qquad \forall p \in \mathcal{P}, \forall k \in \mathcal{K}, \quad (11)$$
$$\sum_{d\in\mathcal{D}} z_{ptd} \leq Q_{pt} \qquad \forall p \in \mathcal{P}, \forall t \in \mathcal{T}_p, \quad (12)$$
$$\sum_{p\in\mathcal{P}}\sum_{r\in\mathcal{R}} \gamma_p w_{pdr} \leq c_d \qquad \forall d \in \mathcal{D}, \quad (13)$$
$$u_{prk}, w_{pdr}, x_{psk}, y_{psd}, z_{ptd} \geq 0 \qquad \forall p \in \mathcal{P}, \forall k \in \mathcal{K}, \forall d \in \mathcal{D}, \forall r \in \mathcal{R}, \forall t \in \mathcal{T}_p, \quad (14)$$
$$v_{ps}, c_k \in \{0,1\} \qquad \forall p \in \mathcal{P}, \forall s \in \mathcal{S}_p. \quad (15)$$

Os dois primeiros termos da função objetivo (1) representam os custos fixos associados à instalação dos CRs e dos CCs, respectivamente. O terceiro termo do modelo refere-se ao custo de transporte dos varejistas até os CCs, somado ao custo de processamento nos CCs. O quarto

termo refere-se ao custo de transporte dos CCs para os CRs. Os dois termos citados anteriormente compõem o custo do canal reverso. O quinto e o sexto termo representam o custo de transporte no canal direto, ou seja, dos CRs para os CDs e das NPPs para os CDs, juntamente com os custos de processamento nos CRs e NPPs, respectivamente. O último termo calcula o custo de transporte entre CDs e varejistas e o custo de processamento nos CDs. As restrições (2) garantem que apenas um CR será instalado para remanufaturar cada produto. As restrições (3) garantem que a quantidade de produto retornado para cada varejista seja enviada para os CCs. Os conjuntos de restrições (4)–(7) asseguram a conservação de fluxo para cada tipo de produto nos CCs (restrições (4)), nos CRs (restrições (5)–(6)) e nos CDs (restrições (7). O conjunto de restrições (5) também garante que, para um produto p, um CR seja estabelecido em um local s se houver fluxo desse produto de um CC k para o local s. As restrições (6) impõem que os produtos remanufaturados sejam totalmente devolvidos ao sistema. As restrições (8) garantem que a demanda de cada cliente seja satisfeita usando produtos novos ou remanufaturados, ou ambos. As restrições (9) garantem que cada CC será usado apenas se for instalado e também garante que a limitação de capacidade de processamento agregado seja respeitada. As restrições (10) e (11) são restrições redundantes que melhoram a qualidade da relaxação linear da formulação, como mostrado por Easwaran e Uster [2009]. Os conjuntos de restrições (12) e (13) reforçam as restrições de capacidade nas fábricas (NPP) e nos CDs, respectivamente. Os conjuntos de restrições (14) e (15) são as restrições de domínio das variáveis de decisão.

Esse modelo pode ser resolvido por intermédio de um aplicativo de resolução de problemas de programação linear. Porém, devido ao fato do PPRCLSCC ser um problema NP-difícil, este trabalho foca na aplicação de metaheurísticas para resolução de instâncias do problema.

3. Trabalhos relacionados

Dentre os principais motivos para se considerar o projeto de uma rede CLSC, destaca-se que projetar as redes para fluxo direto e reverso separadamente pode não resultar na melhor configuração de rede final possível, pois os fluxos direto e reverso podem ser interdependentes. Os primeiros estudos na área de planejamento de redes logísticas concentraram-se apenas em fluxos em um sentido. Geoffrion e Graves [1974] e Keskin e Üster [2007] consideram uma rede capacitada para fluxo de múltiplos produtos, composta por NPPs, DCs e pontos de demanda, enquanto Barros et al. [1998] abordaram o fluxo reverso de materiais.

A integração dos fluxos direto e reverso foi abordada por Fleischmann et al. [2001]. Já Beamon e Fernandes [2004] consideram uma rede CLSC com um único tipo de produto, assumindo que as NPPs e os varejistas já estão dispostos. As instalações da rede CLSC são caracterizadas por restrições de capacidade finita, exceto os CDs, e também é adotada a estratégia de fonte múltipla na rede. Ao considerar que, em alguns casos práticos, os modelos de cadeia de suprimentos de múltiplos produtos são mais realistas que os de um único produto, Salema et al. [2006] estendem um modelo para fluxo de um produto único, gerando um modelo para projetar uma rede CLSC multiprodutos. Easwaran e Uster [2009] abordam o projeto de uma rede CLSC multiproduto e capacitada, tratada neste trabalho. Govindan et al. [2015] apresentam uma revisão abrangente da literatura.

O desenvolvimento de procedimentos para resolver o PPRCLSCC leva em consideração que o alto custo de implantação das decisões referentes ao projeto de uma rede CLSC reforça a importância de uma tomada de decisão otimizada. Alguns autores propuserem procedimentos exatos, como método *branch-and-cut*, método *branch-and-bound* e método de decomposição de Benders, para resolver instâncias do problema [Cordeau et al., 2006; Listeş, 2007; Salema et al., 2006]. Para resolver instâncias do PPRCLSCC, Easwaran e Uster [2009] aplicam o método de decomposição de

Benders, que não se mostrou eficiente para resolver instâncias de grandes dimensões do problema. Portanto, devido à dificuldade de se resolver instâncias de grande porte para esta classe de problemas via métodos exatos, foram propostas heurísticas aplicadas ao projeto de rede CLSC, como o algoritmo genético [Lieckens e Vandaele, 2007] e busca tabu [Easwaran e Uster, 2009].

4. Metodologia

Nesta seção, será apresentado um método para a construção de uma solução inicial, bem como como o método de refinamento baseado nas meta-heurísticas **ILS** e **VND** e na heurística de avaliação das soluções.

4.1. Representação da Solução

Uma solução pode ser caracterizada por um par de conjuntos $(\mathcal{K}^u, \mathcal{S}^u)$, no qual $\mathcal{K}^u$ representa o conjunto de CCs abertos e $\mathcal{S}^u$ representa o conjunto de CRs abertos. O conjunto $\mathcal{K}^u$ foi implementado como um vetor binário c de tamanho $|\mathcal{K}|$, em que, para todo $k \in \mathcal{K}$, c_k é igual a 1 se $k \in \mathcal{K}^u$, e é igual a 0, caso contrário. Como existe um único CR para cada tipo de produto, o conjunto $\mathcal{S}^u$ foi representado por um vetor de listas de tamanho $|\mathcal{P}|$, no qual o p-ésimo elemento deste vetor é uma lista binária cujos elementos representam cada local candidato à abertura de um CR para o produto $p \in \mathcal{P}$. Especificamente, o conjunto $\mathcal{S}^u$ foi representado pelo vetor de listas binárias v, em que v_{ps} é igual a 1 se $s \in \mathcal{S}^u$ e é igual a 0, caso contrário.

É necessário garantir que cada solução $(\mathcal{K}^u, \mathcal{S}^u)$ satisfaça à condição:

$$\sum_{k \in \mathcal{K}^u} B_k \geq \sum_{r \in \mathcal{R}} \sum_{p \in \mathcal{P}} \beta_p \delta_{rp} D_{rp} \tag{16}$$

que garante que uma solução viável deve implicar em uma configuração de rede em que a capacidade agregada de armazenamento/processamento nos CCs abertos seja suficiente para acomodar todo o fluxo reverso de produtos vindos dos varejistas. Além disso, deve-se garantir que apenas um CR será selecionado para cada produto p, como representado na expressão (16).

4.2. Algoritmo híbrido ILS-VND

De acordo com Lourenço et al. [2003], a meta-heurística *Iterated Local Search* (ILS) consiste em uma perturbação da solução corrente, geralmente um ótimo local, levando a alguma solução intermediária, que é usada como novo ponto de partida para um método de refinamento. O procedimento ILS usado neste trabalho está ilustrado no Algoritmo 1. A ideia do algoritmo é gerar uma solução por um procedimento guloso e, em seguida, realizar uma busca local até que um ótimo local seja obtido. Em seguida, para escapar deste ótimo local, uma perturbação de nível $level$ é realizada, resultando em uma nova solução, que será aprimorada utilizando uma busca local. Um critério de aceitação adicional decide quais das soluções obtidas são mantidas para continuar esse processo.

Neste trabalho, a busca local é realizada usando a meta-heurística VND. Trata-se de um procedimento de busca local, que explora iterativamente várias estruturas de vizinhança. Seu sucesso é baseado no fato de que diferentes estruturas de vizinhança geralmente não possuem os mesmos ótimos locais. Assim, o problema de ficar preso em ótimos locais, que não são ótimos globais, pode ser amenizado pela mudança de vizinhança.

As estruturas de vizinhanças usadas neste trabalho, bem como o algoritmo VND e o procedimento de perturbação propostos, serão apresentados nas Seções 4.3 e 4.4.

Algorithm 1: ILS($iter_{max}$, $level$)

```
/* Refinamento de uma solução X=(K^u , S^u)                    */
X_0 ← Solução Inicial gulosa
X ← Busca local(X_0)
iter ← 0 while iter < iter_max do
    iter ← iter + 1
    X' ←Perturbação(X, level)
    X'' ← Busca local(X')
    if Avaliação da solução(X'') < Avaliação da solução(X) then
        X ← X''
    end
end
Retorne X
```

4.3. Vizinhanças

O espaço de soluções foi explorado utilizando três tipos de movimentos: Troca CC, Altera CC e Troca CR. O movimento Troca CC consiste em, para cada CC aberto no conjunto $\mathcal{K}^u$, escolher um CC do conjunto $\mathcal{K} \setminus \mathcal{K}^u$ para formar os pares de troca. Se uma troca implica uma violação da restrição de capacidade agregada (16), ela é ignorada. O movimento Altera CC corresponde a abrir um CC fechado ou fechar um CC aberto. Deve-se ignorar um movimento que implica em uma solução que viola a restrição (16). O movimento Troca CR consiste em, para cada produto p, trocar o CR correspondente a este produto.

4.4. Construção e refinamento de uma Solução

Construir uma solução viável consiste em encontrar um conjunto de locais para a abertura de CCs e um conjunto de locais para a abertura de CRs, um por produto p. Para essa finalidade, foi utilizado o procedimento proposto por Easwaran e Uster [2009], em que uma solução inicial ($\mathcal{K}^u, \mathcal{S}^u$) é gerada de maneira gulosa. Este procedimento de geração da solução inicial considera a abertura de CCs e CRs em locais cujos custos fixos de abertura ($F_k^{'}$ e F_{ps}) são os menores possíveis e que atenda à restrição (16).

Após a construção da solução inicial, esta solução será refinada pelo Algoritmo 1, usando os seguintes procedimentos de perturbação e de busca local. As perturbações são realizadas usando o movimento Altera CC, que representa a vizinhança N_3, como ilustrado no Algoritmo 2. A busca local via VND é realizada utilizando as outras duas estruturas de vizinhança, ou seja, Troca CC e Troca CR, como apresentado no Algoritmo 3, em que o movimento Troca CC representa a vizinhança N_1 e o movimento Troca CR representa a vizinhança N_2. Neste trabalho, são consideradas duas estratégias de seleção de solução aprimorada (linha 4 do Algoritmo 3). A primeira estratégia é a busca Melhor Vizinho, que avalia todas as soluções em uma determinada vizinhança e seleciona a melhor delas. A segunda estratégia considerada é a estratégia de busca Primeiro de Melhora, que seleciona aleatoriamente soluções em uma certa vizinhança até que uma solução melhor que a solução corrente seja encontrada.

4.5. Função de Avaliação

Dada uma solução viável ($\mathcal{K}^u, \mathcal{S}^u$), o fluxo de produtos na rede CLSC (u, w, x, y, z | $\mathcal{K}^u$, $\mathcal{S}^u$) foi resolvido por um método composto por cinco estágios, baseado no método proposto por Üster et al. [2007] e [Easwaran e Uster, 2009].

O primeiro estágio determina o fluxo de produtos dos varejistas aos CDs de uma forma gulosa, com base nos custos unitários $G_{prk} + G_{pks}$ e na capacidade dos CCs abertos (atribuindo valores às variáveis de fluxo u_{prk}). O segundo estágio determina o envio de fluxo dos CCs para

Algorithm 2: Perturbação($X, level$)

```
X' ← X
cont ← 1
while cont ≤ level do
    X' ← Solução aleatoriamente selecionada na vizinhança N_3(X')
    cont ← cont + 1
end
retorne X'
```

Algorithm 3: Busca Local(X_0,n_{max})

```
X ← X_0
n ← 1
while n ≤ n_max do
    X' ← Solução aprimorada na vizinhança N_n(X).
    if Avaliação da solução(X') < Avaliação da solução(X) then
        X ← X'
        n ← 1
    end
    else
        n ← n + 1
    end
end
Retorne X
```

os CRs (variáveis de fluxo x_{pks}). Este cálculo é simples, pois, para cada produto p, os CCs devem enviar a quantidade total de subprodutos coletados para o CR que foi aberto para remanufaturar p. O terceiro estágio determina o fluxo de produtos entre os CDs e os varejistas (variáveis de fluxo w_{pdr}). A determinação do fluxo dos produtos é feita de forma gulosa, com base no custo unitário G_{pdr} e nas capacidades dos CDs.

Nos dois últimos estágios, resolve-se o problema de transporte das NPPs e dos CRs para os CDs. Esse processo é feito de maneira separada para cada produto p. Para um determinado p, o CR aberto, juntamente com as NPPs, constituem o conjunto de locais de fornecimento. Este conjunto será representado por $T_p^{'}$. Inicialmente, o CR pode enviar todos os produtos remanufaturados para atender aos requisitos nos CDs. Denotamos a capacidade de fornecimento de um CR aberto $s^{[p]}$ por Q_{ps}. Associado ao índice $s^{[p]}$, também define-se as variáveis de fluxo $z_{ps^{[p]}d}$ e os custos unitários $G_{ps^{[p]}d}$. Os produtos vindos das instalações do conjunto $T_p^{'}$ são enviados aos CDs (variáveis de fluxo z_{ptd}) também de forma gulosa, com base nos custos unitários G_{ptd} e nas capacidades de fornecimento Q_{pt}, para $t \in T_p^{'}$. Ao final dessas atribuições, o CR pode ter alguns produtos remanufaturados não utilizados, no caso $Q_{ps^{[p]}} > 0$. Para que se garanta o uso de toda a quantidade de produtos remanufaturados para atender às demandas dos CDs, na quinta etapa aumenta-se o fluxo do CR, enquanto, simultaneamente, reduz-se os fluxos das NPPs por uma quantidade equivalente, de uma forma gulosa com base no custo unitário $G_{ps^{[p]}d} - G_{ptd}$, até que a restrição (6) seja satisfeita. Finalmente, os valores do fluxo das variáveis u, w, x, y, z juntamente com os valores das variáveis binárias c e v, são usados para estimar o valor da função objetivo $Z(\mathcal{K}^u, \mathcal{S}^u)$ de uma solução viável $(\mathcal{K}^u, \mathcal{S}^u)$ (função objetivo (1)).

5. Resultados

Para a realização dos experimentos computacionais foi utilizado um conjunto de problemas-teste gerado por Easwaran e Uster [2009]. Esse conjunto está dividido em 12 classes, sendo que

Tabela 1: Características das instâncias.

Classe	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
$\|\mathcal{P}\|$	5	5	5	5	5	5	10	10	10	10	10	10
$\|\mathcal{R}\|$	60	60	90	90	120	120	60	60	90	90	120	120
$\|\mathcal{K}\|$	25	35	25	35	25	35	25	35	25	35	25	35

Tabela 2: Valor médio da função objetivo e *gap* médio por classe de instâncias.

	Valor da Função Objetivo			*Gap (%)*	
Classe	CPLEX	Método 1	Método 2	Método 1	Método 2
C1	130097831,52	**131820133,75**	142291372,17	**1,27**	8,73
C2	98871836,69	**99021879,50**	115972580,04	**0,17**	13,01
C3	231200158,90	**232661961,92**	242200902,25	**0,57**	4,53
C4	219742050,99	**225631583,26**	229752794,34	**2,81**	4,47
C5	183907444,07	**186234769,80**	194008187,42	**1,41**	5,40
C6	175841637,40	**177988562,71**	189342380,75	**1,08**	7,07
C7	369426314,86	**373617018,17**	379427058,21	**1,14**	2,67
C8	348560604,33	**352121788,29**	358561347,78	**1,08**	2,85
C9	235620033,54	**236591957,05**	245620776,89	**0,42**	4,40
C10	350269025,15	**350291353,09**	363269768,50	**0,60**	3,70
C11	548262746,98	**550356403,41**	558263490,33	**0,35**	1,81
C12	524522796,92	**526979959,53**	539245540,30	**0,45**	2,69
			Média:	**0,95**	5,11

cada classe contém dez problemas. As características das classes do conjunto encontram-se explicitadas na Tabela 1.

Os algoritmos apresentados neste trabalho foram implementados usando a linguagem C++. Todos os experimentos computacionais foram executados em um computador *Core i7* com 16 GB de memória RAM e sistema operacional *Linux Ubuntu*. Os resultados obtidos através dos procedimentos propostos foram comparados com os resultados obtidos por meio da resolução da formulação (1)-(15) utilizando o *solver* CPLEX, versão 12.7, com configurações padrão, para resolver o problema de programação matemática. O objetivo de se usar o *solver* é verificar a qualidade das soluções obtidas por cada uma das duas heurísticas em relação às soluções ótimas obtidas através da resolução do problema de programação matemática, bem como comparar os tempos computacionais. Por este motivo, utilizou-se apenas instâncias pequenas para o problema.

As iterações no ILS são controladas pelo parâmetro $iter_{max}$, que, nos experimentos computacionais, foi definido como a quantidade de tipos de produtos ($|\mathcal{P}|$) em uma dada instância. Os experimentos foram realizados considerando as duas estratégias de busca local: Melhor Vizinho, denotada por Método 1, e Primeiro de Melhora, denotada por Método 2. A resolução do modelo matemático via *solver* será denotada por CPLEX. Para cada problema teste, foram realizadas 30 execuções dos algoritmos aqui propostos e os resultados apresentados consideram as médias dos resultados de cada execução.

Tabela 3: Tempo computacional médio por classe de instâncias.

Classe	Tempo (s)			Classe	Tempo (s)		
	CPLEX	Método 1	Método 2		CPLEX	Método 1	Método 2
C1	1711,16	1125,67	**6,42**	C7	2000,27	1187,79	**60,20**
C2	1868,85	1085,42	**14,14**	C8	3029,20	2036,01	**216,90**
C3	1379,30	1182,83	**21,97**	C9	1958,01	1207,35	**28,50**
C4	2437,64	1273,40	**35,73**	C10	12131,26	6732,77	**2555,44**
C5	2849,51	3210,19	**82,13**	C11	4996,45	1681,42	**102,33**
C6	2920,15	3204,26	**85,11**	C12	4793,93	2960,56	**857,95**

Os resultados dos experimentos computacionais são apresentados nas Tabelas 2 e 3. Na Tabela 2, a coluna Valor da Função Objetivo (VFO) apresenta o valor médio da função objetivo para cada classe e a coluna *gap* apresenta a média dos *gaps* obtidos em cada classe. A fórmula usada para cálculo do *gap* para o Método i ($i = 1, 2$) é

$$gap = 100 \times \frac{\text{(VFO do Método } i \text{ - VFO do CPLEX)}}{\text{VFO do Método } i}.$$

A Tabela 3 mostra a média de tempo computacional, em segundos, obtida em cada classe pelo Método 1, pelo Método 2 e pelo *solver* CPLEX. De acordo com estas tabelas, o Método 1 apresenta menores *gaps* médio que o método 2 (Tabela 2), mas demanda maior tempo computacional (Tabela 3). Uma justificativa para esta diferença de desempenho é o fato da busca Melhor Vizinho fazer uma busca extensa em cada vizinhança, requerendo mais esforço computacional na busca, mas encontrando, a cada iteração da busca local, melhores soluções. Comparando o Método 1 com o CPLEX, os VFOs do Método 1 e do CPLEX estão, em média, bem próximos, mas o Método 1 requer menos tempo computacional médio para resolver as instâncias. Comparando os resultados encontrados pelo Método 1 com os resultados obtidos pelo algoritmo heurístico proposto por Easwaran e Uster [2009], a média dos *gaps* obtida pelo Método 1, que foi igual a $0,95\%$, é menor que a média dos *gaps* obtida pelo algoritmo de Easwaran e Uster [2009], que foi igual a $1,49\%$.

6. Conclusão

Este trabalho abordou o projeto da rede de uma cadeia de suprimentos em circuito fechado capacitada. Baseados em três diferentes estruturas de vizinhança, são propostos dois algoritmos que integram as metaheurísticas ILS e VND. Os algoritmos partem de uma solução inicial gerada por um procedimento guloso que é refinada pela meta-heurística ILS com busca local feita pelo VND. Foram utilizadas duas estratégias de busca diferentes, Melhor vizinho (Método 1) e Primeiro de melhora (Método 2). De acordo, os resultados dos experimentos, o Método 1 obteve, em média, melhores soluções em relação ao Método 2. No entanto, devido ao maior esforço computacional aplicado na busca local do Método 1, o Método 2 se mostrou mais eficiente que o primeiro em relação ao tempo computacional. Apesar de ter um tempo computacional mais elevado em relação ao Método 2, o Método 1 se mostrou eficiente em relação ao tempo necessário para resolver o problema de programação matemática por meio de um *solver* com configuração padrão.

Como trabalho futuro, pretende-se analisar a influência do tamanho das vizinhanças a cada iteração do procedimento VND, o qual é responsável pela busca local dentro do algoritmo ILS, pois esse procedimento requer um tempo de processamento muito elevado para avaliar a vizinhança completa de uma solução.

Referências

Barros, A. I., Dekker, R., e Scholten, V. (1998). A two-level network for recycling sand: a case study. *European journal of operational research*, 110(2):199–214.

Beamon, B. M. e Fernandes, C. (2004). Supply-chain network configuration for product recovery. *Production Planning & Control*, 15(3):270–281.

Cordeau, J.-F., Pasin, F., e Solomon, M. M. (2006). An integrated model for logistics network design. *Annals of operations research*, 144(1):59–82.

Easwaran, G. e Uster, H. (2009). Tabu search and benders decomposition approaches for a capacitated closed-loop supply chain network design problem. *Transportation Science*, 43(3):301–320.

Fleischmann, M., Beullens, P., BLOEMHOF-RUWAARD, J. M., e Van Wassenhove, L. N. (2001). The impact of product recovery on logistics network design. *Production and operations management*, 10(2):156–173.

Geoffrion, A. M. e Graves, G. W. (1974). Multicommodity distribution system design by benders decomposition. *Management science*, 20(5):822–844.

Govindan, K., Soleimani, H., e Kannan, D. (2015). Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future. *European Journal of Operational Research*, 240 (3):603–626.

Keskin, B. B. e Üster, H. (2007). Meta-heuristic approaches with memory and evolution for a multi-product production/distribution system design problem. *European Journal of Operational Research*, 182(2):663–682.

Lieckens, K. e Vandaele, N. (2007). Reverse logistics network design with stochastic lead times. *Computers & Operations Research*, 34(2):395–416.

Listeş, O. (2007). A generic stochastic model for supply-and-return network design. *Computers & Operations Research*, 34(2):417–442.

Lourenço, H. R., Martin, O. C., e Stützle, T. (2003). Iterated local search. In *Handbook of metaheuristics*, p. 320–353. Springer.

Rogers, D. e Tibben-Lembke, R. (1999). *Going Backwards: Reverse Logistics Trends and Practices*. Reverse Logistics Executive Council.

Sahyouni, K., Savaskan, R. C., e Daskin, M. S. (2007). A facility location model for bidirectional flows. *Transportation Science*, 41(4):484–499.

Salema, M., Póvoa, A., e Novais, A. Q. (2006). A warehouse-based design model for reverse logistics. *Journal of the Operational Research Society*, 57(6):615–629.

Srivastava, S. K. (2007). Green supply-chain management: a state-of-the-art literature review. *International journal of management reviews*, 9(1):53–80.

Üster, H., Easwaran, G., Akçali, E., e Çetinkaya, S. (2007). Benders decomposition with alternative multiple cuts for a multi-product closed-loop supply chain network design model. *Naval Research Logistics (NRL)*, 54(8):890–907.