

NICHOLAS MAMORU HANEDA GOTO

**FERRAMENTA PARA OTIMIZAÇÃO DA GESTÃO DE FLUXOS DE LOGÍSTICA
REVERSA EM UMA PLATAFORMA MODULAR**

São Paulo

2020

NICHOLAS MAMORU HANEDA GOTO

**FERRAMENTA PARA OTIMIZAÇÃO DA GESTÃO DE FLUXOS DE LOGÍSTICA
REVERSA EM UMA PLATAFORMA MODULAR**

**Trabalho de Formatura apresentado à Escola
Politécnica da Universidade São Paulo para
obtenção do Diploma de Engenheiro de
Produção**

Orientador: Prof. Dr. Leonardo Junqueira

**São Paulo
2020**

À minha família e amigos

AGRADECIMENTOS

Antes do desenvolvimento deste trabalho, tenho que agradecer a todas as pessoas que contribuíram diretamente e indiretamente ao mesmo, estas que me incentivaram e me auxiliaram de alguma maneira ao longo deste período.

Primeiramente, gostaria de agradecer ao meu orientador, Prof. Dr. Leonardo Junqueira, por me instruir e me ajudar durante todo esse trabalho. Em seguida, gostaria de agradecer a Prof. Dra. Wahiba Ramdane Cherif Khettaf, Diretora de conferências na Ecole des Mines de Nancy, por suas aulas expositivas e estudos de caso que proporcionaram os meios para aprofundar meus conhecimentos para esse trabalho. Ambos compartilharam conhecimentos de seus domínios de *expertise*, o que me permitiu avançar e superar rapidamente os problemas complexos e delicados que surgiram ao longo do trabalho. Além disso, esses se mostravam sempre presentes em caso de alguma dúvida ou problema.

Gostaria também de agradecer a toda minha família, pais e irmãos, minha namorada e meus amigos que me apoiaram e me encorajaram durante todo o desenvolvimento do trabalho, me fornecendo forças quando eu mais precisava.

Assim, eu expresso meus sentimentos de satisfação e agradecimento.

*“Every great dream begins with a dreamer.
Always remember, you have within you the
strength, the patience, and the passion to
reach for the stars to change the world.”*

Harriet Tubman

RESUMO

O desenvolvimento sustentável é uma das grandes temáticas da atualidade e está ganhando cada vez mais visibilidade na mídia e no cotidiano das pessoas. Com o crescimento acelerado da população e a necessidade do aperfeiçoamento da infraestrutura local através de obras públicas e privadas, o setor da construção civil demonstra ser de grande relevância para essa questão, principalmente pela excessiva geração de resíduos e seus descartes irregulares em grande parte das cidades brasileiras. Dessa forma, plataformas modulares de fácil montagem estão sendo desenvolvidas para diminuir os grandes custos logísticos tanto econômicos quanto ambientais, através de novas áreas de estudos sustentáveis como a logística reversa. Neste trabalho, uma ferramenta de apoio à decisão baseada em métodos heurísticos para gestão de fluxos de canteiros de obras é desenvolvida. A ferramenta trata um problema de roteamento de veículos que visa otimizar o fluxo de materiais e de descartes entre os canteiros de obras e uma plataforma modular, respeitando estritamente as restrições de um problema com entregas e coletas divisíveis e janelas de tempo, tais como o número de visitas, a partida dos veículos de um único ponto, a continuidade dos roteiros, a consistência temporal, a capacidade de carga dos veículos, as janelas de tempo de atendimento e a demanda dos canteiros de obras. Assim, ela prioriza a logística reversa e a diminuição de custos do problema, resultando na otimização do tempo e na facilidade de decisão dos roteiros. Posteriormente, diversos cenários gerados aleatoriamente foram reproduzidos para a realização de testes da ferramenta, uma vez que as plataformas devem ser frequentemente instaladas e reinstaladas para diferentes conjuntos de canteiros de obras. Uma análise mais aprofundada dos resultados é então realizada, avaliando o custo das soluções obtidas em função de diferentes parâmetros, como o número de canteiros de obras existentes e a distância máxima entre os locais, observando quais são os casos em que o método demonstra maior e menor eficácia. Em seguida, uma comparação dos resultados obtidos com a ferramenta proposta é feita com as soluções encontradas por um aplicativo de otimização exato de propósito geral, avaliando as vantagens e desvantagens para cada caso. Por fim, uma análise de aperfeiçoamento da ferramenta é efetuada, a fim de compreender e aprimorar os critérios de inserção.

Palavras-chave: logística reversa, otimização, métodos heurísticos.

ABSTRACT

The sustainable development is one of the trending themes of the present and it is gaining more visibility each time on the medias and in the everyday life. With the rapid population growth, and the improvement requirements in the local infrastructure through public and private works, the construction sector seems to be quite relevant for this issue, especially due to the excessive waste generation and its irregular disposal in all major Brazilian cities. Thus, modular platforms with simple assembly have been developed in order to diminish the great logistics costs, both economics and environmental, through new sustainable areas of study such as reverse logistics. In this study, a decision tool based on heuristic methods for flow management in working sites is developed. It solves a vehicle routing problem with the aim of optimizing the flow of materials and waste between working sites and a modular platform, respecting strictly the constraints of a problem with divisible deliveries and pickups and time windows such as the number of visits, the unique point of depart, the continuity of routing, the temporal consistency, the vehicle's load capacity, the time windows service and the working sites' demand. In order to prioritize the reverse logistic and the cost reduction of the problem, resulting in the improvement of the time and the easiness of a decision in the vehicle itineraries. Then, several random scenarios were performed to test the tool, since the platforms must be often installed and reinstalled for plenty of working sites sets. A deep analysis of the results is then performed, evaluating the solution costs in function of different parameters, such as the quantity of available working sites and the maximum distance between them, remarking which are the cases that the algorithm demonstrates more and less effectiveness. In sequence, a comparison of the results obtained with the proposed tool is carried out with the solution from a general-purpose optimization software based on exact methods, assessing the advantages and disadvantages of the tool for each case. Lastly, an improvement tool analysis is performed, in order to grasp and enhance the insertion criteria.

Keywords: reverse logistics, optimization, heuristic methods.

LISTA DE FIGURAS

Figura 1 – Exemplo de plataforma de distribuição modular para logística reversa do setor da construção civil	27
Figura 2 – Ciclo de descartes dentro da plataforma modular do setor da construção civil	27
Figura 3 – Atividades típicas do processo de logística reversa	31
Figura 4 – Grafo completo com possíveis roteiros calculados pelo problema do caixeiro viajante.....	32
Figura 5 – Exemplo de resolução de um problema de roteamento de veículos (PRV): 3 veículos e 9 nós	33
Figura 6 – Ilustração do número de operações em função das complexidades mais comuns.....	40
Figura 7 – Esquema modelo da distribuição do setor da construção civil	45
Figura 8 – Esquema do processo geral da plataforma modular	45
Figura 9 – Esquema da logística à montante com seus respectivos roteiros	46
Figura 10 – Esquema da logística à jusante	47
Figura 11 – Padronização dos tipos de produtos e materiais.....	48
Figura 12 – Classificação de descartes.....	49
Figura 13 – Exemplo de descartes aceitáveis (papelão) utilizados no setor da construção civil.....	50
Figura 14 – Separação do problema em dois subproblemas	51
Figura 15 – Caminhão de entrega/coleta do tipo VUC	52
Figura 16 – Exemplo de solução para o problema com quatro canteiros de obras e dois veículos.....	54
Figura 17 – Fluxograma do método heurístico de inserção modificado	62
Figura 18 – Pseudocódigo da função de seleção de nós sem demanda	63
Figura 19 – Pseudocódigo da função de retirada de influência dos nós sem demanda	63
Figura 20 – Pseudocódigo da função de viabilidade dos nós.....	64
Figura 21 – Pseudocódigo da função de classificação de nós	64
Figura 22 – Pseudocódigo da função de impossibilidade de roteiros relacionada a um nó i	65
Figura 23 – Pseudocódigo da função de criação de roteiros iniciais.....	66
Figura 24 – Exemplo de um roteiro do método heurístico	67
Figura 25 – Pseudocódigo da função de atualização dos valores de G_i , W_i e Y_i	69
Figura 26 – Pseudocódigo da função de cálculo da capacidade de carga dos arcos	70
Figura 27 – Ilustração da função $\text{calcula_capacidade}(n, \text{routes}, q_i)$	70
Figura 28 – Pseudocódigo da função de verificação da capacidade dos arcos	71
Figura 29 – Pseudocódigo da função de verificação de inserção pela restrição de janelas de tempo	71
Figura 30 – Algoritmo do cálculo do custo C_1 da heurística de inserção modificada	72
Figura 31 – Algoritmo do cálculo de custo C_2 da heurística de inserção modificada	73
Figura 32 – Algoritmo do cálculo do custo C_3 da heurística de inserção modificada	73
Figura 33 – Pseudocódigo da função do custo geral de inserção	74

Figura 34 – Pseudocódigo da função do cálculo da distância total percorrida por todos os roteiros.....	75
Figura 35 – Pseudocódigo da função da metodologia heurística de inserção modificada.....	76
Figura 36 – Pseudocódigo da função da distância dos arcos de um roteiro	77
Figura 37 – Remoção dos arcos dos roteiros 1 e 2 pelo operador 2-opt inter-roteiros	77
Figura 38 – Criação de dois novos roteiros 1' e 2' pelo operador 2-opt inter-roteiros	78
Figura 39 – Pseudocódigo do procedimento de busca local do operador 2-opt inter-roteiros	79
Figura 40 – Solução obtida pela busca inter-roteiros	80
Figura 41 – Exemplo da busca local 2-opt intra-roteiro do roteiro 3	80
Figura 42 – Pseudocódigo do procedimento de busca local do operador 2-opt intra-roteiro	81
Figura 43 – Gráfico do desvio em função do número de canteiros	89
Figura 44 – Gráfico do desvio em função das distâncias máximas.....	89
Figura 45 – Gráfico do tempo de execução médio em função do número de nós	90
Figura 46 – Gráfico do <i>Dif</i> médio em função do número de canteiros de obras	94
Figura 48 – Gráfico do Gap médio em função do número de canteiros de obras	94
Figura 49 – Gráfico do <i>Dif</i> mínimo em função do número de canteiros de obras.....	96
Figura 50 – Gráfico do tempo médio em função do número de canteiros de obra....	98
Figura 51 – Gráfico do <i>Dif</i> médio em função do “coef_roteiro_inicial_max” e do número de canteiros.....	101
Figura 52 – Gráfico do <i>Dif</i> médio em função do “coef_roteiro_inicial_max” e da distância máxima.....	102
Figura 53 – Gráfico do tempo médio em função do “coef_roteiro_inicial_max”	103

LISTA DE TABELAS

Tabela 1 – Parâmetros para a modelagem matemática.....	56
Tabela 2 – Conjuntos para a modelagem matemática.	56
Tabela 3 – Variáveis de decisão para a modelagem matemática.	57
Tabela 4 – Variáveis do método heurístico.	67
Tabela 5 – Avaliação da abordagem heurística de inserção modificada e da busca local 2-opt.....	87
Tabela 6 – Difs, desvio padrões, gaps e proporções médios sem a distinção do “coef_roteiro_inicial_max”.	93
Tabela 7 – Difs e desvio padrões médios com a melhor variação do “coef_roteiro_inicial_max”.	96
Tabela 8 – Difs e tempos médios com a variação do “coef_roteiro_inicial_max”.	97
Tabela 9 – Difs e tempos médios com a variação do “coef_roteiro_inicial_max”.	99
Tabela 10 – Difs médios da variação do custo de inserção “c3”.	105

SUMÁRIO

1. INTRODUÇÃO	21
1.1. Objetivos do trabalho.....	22
1.2. Motivações	22
1.3. Estrutura do trabalho	23
2. REVISÃO BIBLIOGRÁFICA	25
2.1. O setor da construção civil.....	25
2.1.1. Um novo tipo de estrutura: a construção modular	26
2.1.2. Plataformas modulares	26
2.1.3. O impacto ambiental do setor da construção civil	28
2.1.4. A classificação de descartes gerados no setor da construção civil	28
2.2. A otimização da cadeia de suprimentos	29
2.2.1. O conceito de <i>Green Supply Chain Management</i>	29
2.3. Problemas de roteamento de veículos	31
2.3.1. Problema do caixeiro viajante (PCV)	31
2.3.2. Problema de roteamento de veículos (PRV)	32
2.3.3. Problema de roteamento de veículos com entregas e coletas (PRVEC)	33
2.4. Métodos de solução para problemas de roteamento de veículos	39
2.4.1. Métodos exatos	39
2.4.2. Métodos heurísticos.....	40
3. DESCRIÇÃO DO PROBLEMA	45
3.1. Apresentação geral do problema.....	45
3.1.1. Logística à montante	46
3.1.2. Plataforma modular	47
3.1.3. Logística à jusante.....	47
3.2. Separação do problema em dois subproblemas	50
3.2.1. Veículos basculantes para o transporte de descartes inaceitáveis.....	51
3.2.2. Veículos VUC para o transporte de produtos e descartes aceitáveis	51
3.3. Definição do problema	52
4. MODELAGEM MATEMÁTICA	55
4.1. Parâmetros e conjuntos.....	55
4.2. Variáveis de decisão	56
4.3. Função objetivo	57

4.4. Restrições	57
5. MÉTODOS DE SOLUÇÃO	61
5.1. A metodologia heurística da ferramenta desenvolvida	61
5.1.1. Heurística de construção por inserção modificada	61
5.1.2. Método de aprimoramento pela busca local 2-opt	76
6. RESULTADOS E ANÁLISES	82
6.1. Características dos métodos de solução empregados	82
6.1.1. Algoritmo exato	82
6.1.2. Algoritmo heurístico	82
6.2. Circunstâncias dos testes	83
6.3. Detalhamento da análise do método heurístico	85
6.3.1. Instâncias e cenários	85
6.3.2. Resultados	86
6.4. Detalhamento da análise de comparação entre os métodos	90
6.4.1. Instâncias e cenários	90
6.4.2. Resultados	92
6.5. Detalhamento da análise inicial de aperfeiçoamento da ferramenta através da variação dos pesos dos custos de inserção	103
6.5.1. Instâncias e cenários	103
6.5.2. Resultados	104
6.6. Análises e considerações finais	105
7. CONCLUSÕES E PERSPECTIVAS FUTURAS	107
REFERÊNCIAS BIBLIOGRÁFICAS	109

1. INTRODUÇÃO

O aumento da visibilidade das grandes temáticas ambientais, atrelado ao crescimento populacional e à necessidade de novas obras para o setor da construção civil, incitaram o surgimento de um novo mercado mais consciente e alinhado às práticas ambientais, impulsionando uma mudança no sistema de distribuição de materiais e descartes neste setor. Tais fatores são impulsionados pela enorme representatividade da construção civil na economia global, sendo considerado o maior consumidor de recursos naturais, entre 20% e 50% dos recursos naturais do planeta (John, 2000). Este, se encontra na lista dos três setores chaves que necessitam melhorias segundo um planejamento da comissão europeia para melhorar a eficiência dos recursos globais. Pois, estima-se que um melhor uso desses recursos apenas na União Europeia, influenciaria 42% do consumo final de energia, aproximadamente 35% da emissão de gases do efeito estufa e mais de 50% dos recursos naturais extraídos mundialmente. Além de auxiliar na redução de até 30% do consumo de água (European Commission, 2011).

Ademais, este mesmo setor expressa uma exacerbada geração de descartes ao longo de seu processo de construção e demolição, impactando de maneira relevante o ambiente. Através do descarte irregular de resíduos gerados por esses processos, este volume de entulho é considerado até duas vezes maior que o volume de lixo sólido urbano (Barreto, 2005). Todo ano são gerados mais de 650 milhões de toneladas de descartes provenientes deste setor no mundo, e estima-se que até o ano de 2025 este número praticamente dobrará, alcançando 1,1 bilhões de toneladas ao ano, apenas o setor da construção civil é responsável por 50% dos descartes gerados mundialmente (Hoornweg & Bhada-Tata, 2012).

Dessa forma, a fim de manter a vantagem competitiva no setor, houve o aparecimento de novas metodologias e concepções mais sustentáveis para tratar o sistema, buscando reduzir não só o custo financeiro, mas o enorme impacto ambiental causado.

Portanto, surgiram diversos conceitos inovadores como a logística reversa (CSCMP, 2005) (Murphy e Poist, 1989), a economia circular, 4Rs, construções modulares (Smadja, 2014), entre outros, que necessitam ser inseridos no processo de distribuição, e adaptados aos atuais métodos de otimização (Neto, 2011). Estes que fornecem uma solução mais efetiva para o processo, respeitando a sua

viabilidade e buscando a sua redução de custos logísticos (financeiros e ambientais).

Desse modo, neste trabalho serão abordados dois principais temas recorrentes e necessários para a adaptação do setor em sua atual conjuntura: as construções modulares e, principalmente, a logística reversa. Estes serão utilizados para traçar uma nova solução logística mais eficaz para o mesmo.

1.1. Objetivos do trabalho

Os objetivos deste trabalho são:

- a) Desenvolver uma ferramenta que auxilie na decisão dos roteiros de caminhões para um problema de roteamento de veículos com entregas e coletas divisíveis e janelas de tempo no setor da construção civil. Assim, a ferramenta utilizará o conceito inovador de logística reversa no problema de roteamento de veículos, a fim de otimizar tanto a entrega de materiais quanto a coleta dos descartes dos canteiros de obras;
- b) Realizar análises e aprimoramentos com a ferramenta desenvolvida, visando alcançar uma solução que forneça roteiros com um custo satisfatório, se comparado com a solução ótima, bem como um tempo de resolução viável para a resolução do problema de entrega e coleta.

1.2. Motivações

A motivação para a realização deste trabalho foi o surgimento de uma nova vertente de pensamento, a qual aborda tanto o impacto financeiro quanto o ambiental, almejando um negócio sustentável. Dessa forma, a aspiração foi buscar impactar uma das grandes áreas da economia, a construção civil, que é responsável pelo consumo exacerbado de recursos naturais do planeta, bem como pela enorme geração de resíduos com descarte impróprio.

Portanto, o principal intuito foi impulsionar a utilização das novas técnicas sustentáveis no setor da construção civil, salientando sua importância e demonstrando que suas implementações são intrínsecas para o planeta. Este estímulo será incitado através do desenvolvimento de uma ferramenta para a otimização da gestão de fluxos em uma plataforma modular, que busca utilizar conceitos sustentáveis no setor, como a logística reversa.

1.3. Estrutura do trabalho

Este trabalho de formatura está subdividido em sete Capítulos:

- a) Primeiro Capítulo – Introdução: o foco do Capítulo foi introduzir o leitor à motivação e à temática, aos objetivos principais do trabalho, além de descrever a estrutura do trabalho;
- b) Segundo Capítulo – Revisão bibliográfica: apresenta uma revisão da literatura sobre os conceitos e teorias utilizados ao longo do trabalho;
- c) Terceiro Capítulo – Descrição do problema: especifica o contexto geral do trabalho, definindo os três processos envolvidos para o funcionamento da plataforma geral. Em seguida, são descritos com mais detalhes o problema a ser abordado repartindo-o em subproblemas mais específicos. Por fim, o problema a ser tratado é definido, problema de roteamento de veículos com entregas e coletas divisíveis e janelas de tempo;
- c) Quarto Capítulo – Modelagem matemática: o enfoque deste Capítulo é a concepção de um modelo matemático que será utilizado para estruturar os métodos de solução do problema, evidenciando a função objetivo, os conjuntos e parâmetros, as variáveis de decisão, bem como as restrições envolvidas no mesmo;
- d) Quinto Capítulo – Métodos de solução: caracteriza e descreve os algoritmos utilizados para a resolução do problema, esclarecendo o

desenvolvimento das abordagens através de pseudocódigos e modelos para as duas abordagens, o método heurístico (inserção modificada e buscas locais), e o método exato (modelo matemático aplicado no *software* AMPL com o *solver* CPLEX utilizando combinações dos algoritmos de *Branch-and-Bound* e Simplex);

- e) Sexto Capítulo – Resultados e análises: aborda os resultados obtidos pelos métodos implementados, realizando uma avaliação minuciosa da abordagem heurística desenvolvida, isto é, avaliando a eficácia do método de aprimoramento do resultado, uma avaliação entre abordagens, ou seja, comparando os métodos heurísticos desenvolvidos com o *solver* CPLEX, detectando os pontos fortes e fracos de cada aplicação, e finalmente uma análise inicial de aperfeiçoamento da ferramenta, avaliando o balanceamento da ponderação dos pesos dos critérios de inserção. Ademais, ao longo dessas análises, os parâmetros mais importantes são examinados para fins de análise de sensibilidade.
- f) Sétimo Capítulo – Conclusões e perspectivas futuras: sintetiza todo o desenvolvimento do trabalho, desde a apresentação e detalhamento do problema, elaboração e implementação das abordagens, análise dos resultados obtidos, bem como recomendações futuras para o aprimoramento dos mesmos. Além disso, este Capítulo apresenta as contribuições pessoais e pedagógicas que esse trabalho propiciou.

2. REVISÃO BIBLIOGRÁFICA

A fim de construir uma base teórica sólida para o desenvolvimento da ferramenta e auxiliar o leitor em sua compreensão, serão apresentadas sucintamente as principais definições dos termos e conceitos utilizando referências da literatura que sustentam o trabalho.

2.1. O setor da construção civil

O setor da construção civil é um dos setores mais importantes para a economia brasileira, sendo o crescimento dele diretamente relacionado com o desenvolvimento e a capacidade de produção do país. Entre 2004 e 2013, mesmo considerando os efeitos da crise internacional, ele passou por um significativo processo de expansão, devido, principalmente, ao crescimento do Produto Interno Bruto (PIB) do setor (Sistema Firjan, 2014).

Além disso, nos últimos anos, esse setor tem se tornado cada vez mais dinâmico e competitivo, demonstrando grandes tendências positivas para os próximos anos. De acordo com a pesquisa “Sondagem da construção”, realizada pela FGV/IBRE em agosto de 2019 (FGV/IBRE, 2019), é mencionada uma marca histórica de 87,6 pontos do Índice de Confiança da Construção (ICST), maior nível desde dezembro de 2014, demonstrando um reaquecimento do mercado imobiliário. Ademais, este fato é evidenciado no artigo de Nakamura (2019), que cita os grandes fatores desse resultado como a redução dos estoques, o PIB da construção, as novas políticas públicas, e as novas tecnologias.

Dessa forma, para sustentar esse dinamismo e a vantagem competitiva das empresas do setor, uma nova estratégia imobiliária ganhou forças, esta baseada em flexibilidade e agilidade de adaptação às adversidades. Dito isso, fica clara a grande busca por novas tecnologias e inovações que sustentem essa nova tática, por exemplo as construções modulares, se mostrando bastante promissor devido a um tamanho de mercado global avaliado em 112,3 bilhões de dólares e uma expectativa de crescimento médio anual (Compound Annual Growth Rate) de 6.5%, segundo a análise de mercado realizada pela Grand View Research (GVR, 2019). Além disso, segundo pesquisa da Turner & Townsend (Turner & Townsend, 2019), a expectativa é que esse tipo de construção aumente globalmente 6% até 2022.

2.1.1. Um novo tipo de estrutura: a construção modular

A construção modular é uma solução industrializada, praticamente toda moldada na fábrica, criando perfis ou vigas que são conectados entre si de maneira rápida e flexível (Potenza, 2017). Ela busca otimizar operações, aumentar o padrão de segurança e agregar mais qualidade aos projetos, através da sua versatilidade (instalação rápida em diversos lugares e fácil retirada após o seu uso), e da sua transportabilidade (fácil transporte para diversas localidades e sem grandes custos) (Mobuss Construção, 2019). Assim, este tipo de estrutura resulta em uma ótima solução econômica em relação à construção de estruturas definitivas, pois garantem uma instalação de três a cinco vezes mais rápida de acordo com o diretor geral da Spaciotempo na revista *Supply Chain* (Smadja, 2014).

2.1.2. Plataformas modulares

Plataformas de construção modulares e genéricas (cf. Figura 1) consistem em centros de distribuição que possuem a estrutura de uma construção modular, visando o conceito de versatilidade para adaptar-se aos novos desafios e estratégias do setor, isto é, ser facilmente e rapidamente instaláveis/desmontáveis, e ser transportáveis/reutilizáveis para diversas localizações.

Assim, uma plataforma tem duas funções fundamentais: funcionar como um *hub* de estocagem de produtos expedidos pelos fornecedores e que serão posteriormente distribuídos para os canteiros de obras; funcionar como um local de triagem, reciclagem e tratamento de descartes gerados pelos canteiros de obras (cf. Figura 2). Estas finalidades visam expandi-la no âmbito do desenvolvimento sustentável no setor da construção civil através do *Green Supply Chain Management* (Srivastava, 2007) e a solução de logística reversa (Murphy e Poist, 1989), adaptando a plataforma modular aos programas ecológicos existentes de tamanhos diversos no mundo, com o serviço pós-coleta e concretizando uma economia circular, visto que um dos grandes impactos ambientais na economia mundial é proveniente do descarte irregular e impróprio de resíduos desse setor (Neto, 2011).

Atualmente, existem alguns projetos que visam a otimização logística desse tipo de plataforma modular, um bom exemplo é o PILC (Projet Innovation Logistique

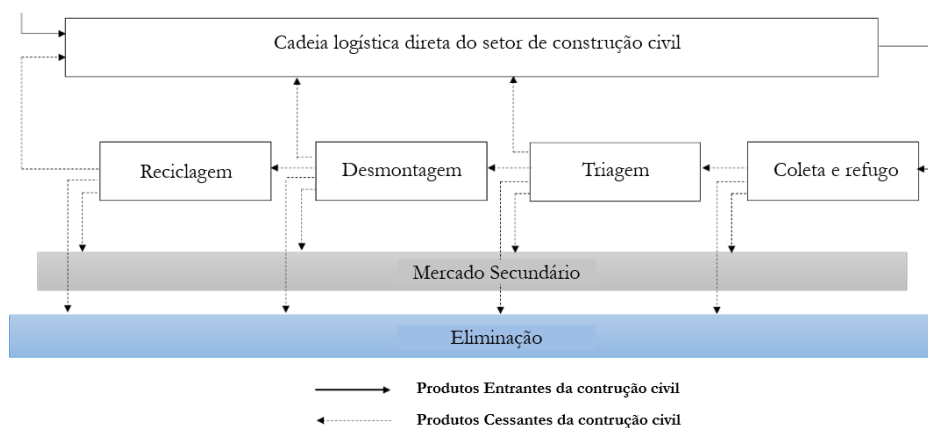
Chantiers). Esse projeto auxilia as empresas do setor da construção civil a implementar técnicas ecológicas através da utilização de plataformas modulares desmontáveis e transportáveis, do fornecimento de ferramentas logísticas de gestão e planificação, e da realização de serviços destinados a ajudar pontualmente essas empresas. Alguns atores envolvidos nesse projeto são empresas como a ADEME (Agence de l'Environnement, et de la Maîtrise de l'Energie) e FLORANGEe2i (Lerouvillois, P et al., 2019).

Figura 1 – Exemplo de plataforma de distribuição modular para logística reversa do setor da construção civil



Fonte: Medium, 2017

Figura 2 – Ciclo de descartes dentro da plataforma modular do setor da construção civil



Fonte: Adaptado, Fall, 2016

2.1.3. O impacto ambiental do setor da construção civil

O setor da construção civil, além de ser um dos maiores da economia, produz os bens de maiores dimensões físicas do planeta, sendo consequentemente o maior consumidor de recursos naturais de qualquer economia, isto é, entre 20% e 50% dos recursos naturais do planeta (madeira, cobre, etc.) (John, 2000). A comissão Europeia traçou um planejamento de melhoria da utilização dos recursos naturais na economia global e listou o setor da construção civil como um dos três setores alvos de reestruturação. De acordo com o documento, a melhor utilização de recursos neste setor apenas na União Europeia, influenciaria diretamente 42% do consumo final de energia, cerca de 35% da emissão de gases estufas na atmosfera, e mais de 50% da extração mundial de recursos naturais. Além de afetar diretamente 30% o consumo de água global (European Commission, 2011).

Ademais, a excessiva geração de descartes provenientes das perdas ocorridas durante o processo de construção ou de demolição, e do seu descarte irregular, são responsáveis por aumentar ainda mais o impacto ambiental do setor, causando grandes desastres como a obstrução e contaminação dos leitos dos rios e canais, comprometimento do tráfego em vias públicas, degradação da paisagem urbana e rural, poluição do ar devido aos gases e partículas contaminantes (poeira, CO₂, etc.) expelidos pelos veículos e maquinários em geral, entre outros. É constatado que o volume de entulho gerado é até duas vezes maior que o volume do lixo sólido urbano (Barreto, 2005). Atualmente, estima-se que anualmente são gerados mais de 1,3 bilhões de toneladas de descartes no mundo, e espera-se que até o ano de 2025 este número praticamente dobrará, alcançando 2,2 bilhões de toneladas ao ano. Destes números, o setor da construção civil é responsável por 50%, ou seja, 650 milhões de toneladas de descartes são gerados todo ano (Hoornweg & Bhada-Tata, 2012).

2.1.4. A classificação de descartes gerados no setor da construção civil

Segundo a classificação da resolução do CONAMA nº 307 Art. 3º, existem quatro classes de descartes A, B, C e D (Conselho Nacional do Meio Ambiente – CONAMA, 2002):

I - Classe A - são os resíduos reutilizáveis ou recicláveis como agregados, tais como:

a) de construção, demolição, reformas e reparos de pavimentação e de outras obras de infraestrutura, inclusive solos provenientes de terraplanagem;

b) de construção, demolição, reformas e reparos de edificações: componentes cerâmicos (tijolos, blocos, telhas, placas de revestimento, etc.), argamassa e concreto;

c) de processo de fabricação e/ou demolição de peças pré-moldadas em concreto (blocos, tubos, meios-fios, etc.) produzidas nos canteiros de obras;

II - Classe B - são os resíduos recicláveis para outras destinações, tais como: plásticos, papel, papelão, metais, vidros, madeiras e gesso (Redação dada ao inciso pela Resolução CONAMA nº 431, de 24.05.2011, DOU 25.05.2011).

III - Classe C - são os resíduos para os quais não foram desenvolvidas tecnologias ou aplicações economicamente viáveis que permitam a sua reciclagem ou recuperação, são resíduos perigosos oriundos do processo de construção, tais como tintas, solventes, óleos e outros; (NR) (Redação dada ao inciso pela Resolução CONAMA nº 431, de 24.05.2011, DOU 25.05.2011).

IV - Classe D: são resíduos perigosos oriundos do processo de construção, tais como tintas, solventes, óleos e outros, ou aqueles contaminados ou prejudiciais à saúde oriundos de demolições, reformas e reparos de clínicas radiológicas, instalações industriais e outros, bem como telhas e demais objetos e materiais que contenham amianto ou outros produtos nocivos à saúde (Redação dada ao inciso pela Resolução CONAMA nº 348, de 16.08.2004, DOU 17.08.2004).

Nesse documento serão abordados apenas as coletas dos resíduos de classe A e B, uma vez que os resíduos de classe C e D não são recicláveis, e não sofrem o mesmo tipo de coleta e descarte que serão utilizados na plataforma (triagem, reciclagem e descarte). Normalmente essas duas últimas classes são coletadas por entidades especiais, especializadas em descarte de materiais nocivos e peculiares.

2.2. A otimização da cadeia de suprimentos

2.2.1. O conceito de *Green Supply Chain Management*

A *Green Supply Chain Management* representa uma nova vertente na economia mundial alinhada aos princípios da produção mais limpa, unificando os conceitos da gestão da cadeia de suprimentos e da gestão ambiental (Srivastava, 2007). Assim, ela busca sanar os impactos negativos ambientais promovidos pela

produção industrial através de soluções inovadoras atreladas à noção de sustentabilidade, como a integração vertical da cadeia de suprimentos, a logística reversa, os parques industriais ecológicos, etc. (Neto, 2011).

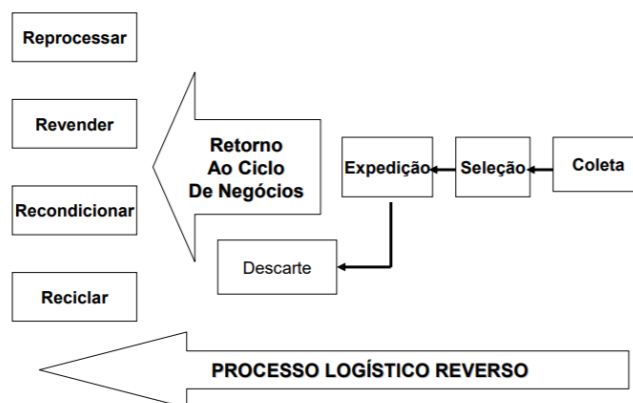
Portanto, ansiando por um mundo mais sustentável, equilibrando os três pilares (econômico, ambiental e social), este novo conceito de cadeia de suprimentos leva em consideração as necessidades das empresas do setor da construção civil e as necessidades ambientais do meio, para otimizar a logística do fluxo de transporte, um elemento bastante importante para a tomada de decisão nas grandes empresas desse setor.

Além disso, a partir deste novo conceito em cadeias de suprimentos, a *Green Supply Chain Management*, houve o desenvolvimento de uma solução mais sustentável, a logística reversa (Murphy e Poist, 1989). Esta é tratada como um “segmento especializado da logística que foca no movimento e gerenciamento de produtos e materiais após a venda e após a entrega ao consumidor, incluindo produtos retornados para reparo e/ou reembolso financeiro” (CSCMP, 2005).

Assim, a partir do elo final da cadeia produtiva, existe um fluxo reverso dos materiais após a utilização do consumidor final, aplicando os princípios dos 4R's (reprocessar, recondicionar, reciclar e revender), otimizando a recuperação máxima dos componentes, peças, materiais utilizados na construção, e minimizando, portanto, o volume do descarte gerado (Neto, 2011), como ilustrado na Figura 3.

Assim, a logística reversa vem se tornando cada vez mais importante nas últimas duas décadas, principalmente pela tendência sustentável e o aparecimento de novas leis de reciclagem e descarte, como o da regulação na união europeia de resíduos eletrônicos (*Waste Electrical and Electronic Equipment - WEEE*) (EUR LEX, 2019).

Figura 3 – Atividades típicas do processo de logística reversa



Fonte: Adaptado, Lacerda, 2003 p.478

Este conceito é evocado no estudo de Soysal (2016), no qual expõe o problema de logística reversa e os princípios de sustentabilidade com o planejamento dos roteiros de veículos.

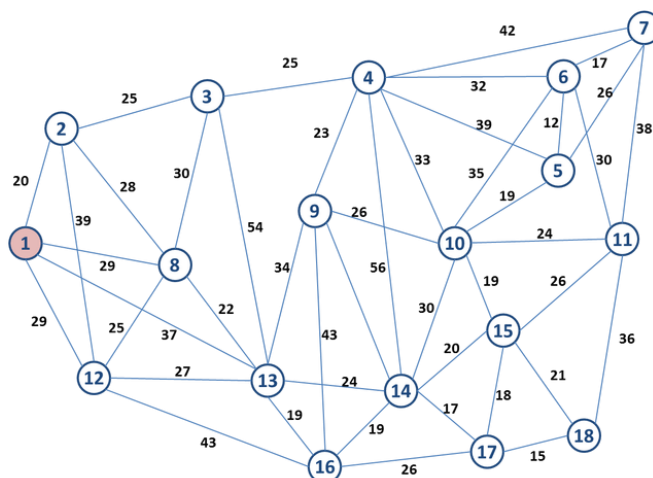
2.3. Problemas de roteamento de veículos

2.3.1. Problema do caixeiro viajante (PCV)

O problema do caixeiro viajante (PCV) é um dos problemas clássicos na literatura de Pesquisa Operacional e que é bastante utilizado em estudos de logística (Dantzig et al., 1954) (Rego et al., 2011). Ele serve como base para diversos tipos de problemas como a resolução dos problemas de roteamento de veículos e suas variações como evidenciado no estudo de Desrochers et al. (1988).

Na literatura, o PCV é definido como um problema NP-Difícil que pode ser modelado em grafos completos (conjunto de vértices e arestas), em termos de ciclos Hamiltonianos (caminho que passa por todos os vértices de um grafo sem repeti-los). Assim, esse problema é ilustrado como um caixeiro viajante que parte de uma cidade e deseja visitar todo um conjunto de cidades uma única vez, com o objetivo de obter um menor custo (Lopes, S. A.; 2009). A Figura 4 apresenta bem esse problema, expondo um grafo completo com os possíveis arcos e os seus devidos custos calculados, que são utilizados para escolher o roteiro menos oneroso partindo e retornando do nó 1 (ciclo hamiltoniano).

Figura 4 – Grafo completo com possíveis roteiros calculados pelo problema do caixeiro viajante



Fonte: Souza, 2018

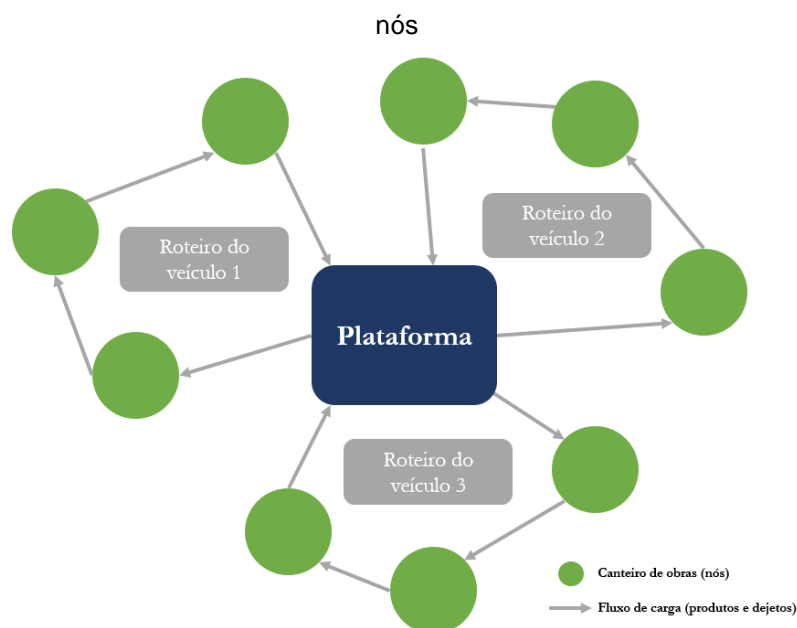
2.3.2. Problema de roteamento de veículos (PRV)

O problema de roteamento de veículos (PRV) é bem conhecido no domínio da Pesquisa Operacional ligado à logística e foi introduzido formalmente em 1959 por Dantzig, G. B. e Ramser, J. H. (1959), no qual é considerado como um problema generalizado do PCV, isto é, o PRV pode ser visto como múltiplos PCV's como na obra de Desrochers et al. (1988).

Esse problema clássico de roteamento é composto por clientes com demandas conhecidas que são supridas por um depósito. Este visa encontrar um conjunto de roteiros que satisfazem todos requisitos e restrições, como a partida de e o retorno para um único ponto (depósito), a capacidade de carga dos veículos e a demanda dos clientes, bem como objetiva fornecer um custo mínimo associado aos mesmos (Liong et al., 2008).

No setor da construção civil, esse tipo de problema se apresentaria como uma plataforma (depósito) no centro, do qual veículos de carga carregados de materiais partem, a fim de entregá-los aos canteiros de obras designados, e, em seguida, retornar a mesma (cf. Figura 5).

Figura 5 – Exemplo de resolução de um problema de roteamento de veículos (PRV): 3 veículos e 9



Fonte: elaboração do autor

O PRV vem ganhando bastante importância nessas últimas décadas, principalmente, pelo seu papel vital no planejamento de sistemas de distribuição e logística em muitos setores, como a coleta de descartes, entrega de produtos, sequenciamento de tarefas etc.

Sendo assim, ele pode ser dividido em muitos tipos e variações, como o PRV com janelas de tempo (Simchi-Levi et al., 2005) (Gan et al., 2012), com entrega e coleta (Berbeglia et al., 2007) (Parragh et al., 2008a e 2008b), entre outros. Estas diversas variações são bem descritas no Capítulo de Irnich et al. (2014) do livro de Toth, V.; Vigo, D. (2014), bem como definida e demonstrada com aplicações práticas na obra de Ismail et al. (2011).

2.3.3. Problema de roteamento de veículos com entregas e coletas (PRVEC)

Os problemas de roteamento de veículos com entregas e coletas (PRVEC) são também uma das importantes extensões do PRV clássico, ou seja, eles possuem todos os atributos e características do mesmo, e mais algumas outras particularidades.

O PRVEC pode ser visto na obra de Savelsbergh e Sol (1995), onde estes discutem as características de um problema clássico de roteamento de veículos com

um problema integrado de entregas e coletas, ressaltando suas definições, características e diferenças conceituais. É interessante perceber que Savelsbergh e Sol especificam casos especiais do PRVEC e os comparam com um PRV ordinário, desenvolvendo um modelo geral (*framework*) para lidar com situações complexas deste problema na realidade, como especificações de janelas de tempo, problemas com respostas estáticas (demandas pré-estabelecidas) ou dinâmicas (demandas de serviços em tempo real).

No estudo de Berbeglia et al. (2007), há uma discussão mais específica das classes do PRVEC, definindo e caracterizando os diversos subgrupos que esse problema possui. Segundo Berbeglia et al., existem três elementos de classificação: estrutura, visitas e veículos.

Assim, o primeiro elemento de classificação é definido pela estrutura do grafo, onde os vértices do grafo representam as origens ou destinos para as diferentes entidades a serem transportadas.

Existem três categorias principais segundo a obra de Battarra, M. (2014) do livro de Toth, V. e Vigo, D. (2014):

1. *Many-to-many* (M-M):

- a. Problemas que possuem múltiplas origens e destinos sem um papel definido;

2. *One-to-many to one* (1-M-1):

- a. Problemas que possuem um único depósito que serve como origem dos produtos para os múltiplos clientes e destino das coletas dos mesmos;

3. *One-to-one* (1-1):

- a. Problemas que possuem um único ponto de origem e um único ponto de destino.

O segundo elemento fornece a informação de como as entregas e as coletas serão executadas nos clientes.

1. Entrega e coleta (EC):

- a. Cada cliente é visitado uma única vez, combinando os serviços de entrega e coleta;

2. Entrega e/ou coleta (E-C):

- a. Ambos os serviços podem ser executados. juntos ou separados, exigindo no máximo duas visitas por cliente;

3. Entrega ou coleta (E/C):

- a. Cada cliente só pode demandar um único serviço, entrega ou coleta, assim é visitado uma única vez.

O terceiro elemento é o número de veículos utilizado no problema, ou seja, com um único veículo ou múltiplos veículos (frota).

No estudo de Parragh et al. (2008a, 2008b), é possível constatar uma revisão detalhada das extensões dos PRVEC, definindo duas classes para a mesma. A primeira classe é caracterizada pela entrega de produtos do depósito aos clientes através de uma rede direta (*linehaul*), e pela coleta de produtos dos clientes ao depósito através de uma rede de retorno (*backhaul*). Nesta primeira categoria, Parragh et al. (2008a) considera quatro subgrupos:

1. Problema de roteamento de veículos com coletas agrupadas (PRVCA):

- a. Extensão do PCV com agrupamentos, no qual os clientes só podem possuir um tipo de serviço (entrega ou coleta), e, assim, o roteamento de veículos irá considerar primeiro os clientes de entrega e depois os de coleta;

2. Problema de roteamento de veículos com sequenciamento misto de entrega e coleta (PRVSMEC):

- a. Parte do mesmo princípio que o problema anterior, onde os clientes só podem possuir um único tipo de serviço. Porém, com uma relaxação maior, pois a visita aos clientes de entrega e coleta podem se alternar;

3. Problema de roteamento de veículos com entregas e coletas divisíveis (PRVECD):

- a. Diferentemente dos dois problemas anteriores, os clientes podem possuir dois tipos de serviço (entrega e/ou coleta), assim cada cliente poderá ser visitado até duas vezes no período. A grande peculiaridade é que cada demanda é tratada de maneira distinta,

onde a entrega e a coleta podem ser executadas por diferentes veículos;

4. Problema de roteamento de veículos com entregas e coletas simultâneas (PRVECS):
 - a. Bastante parecido com o terceiro subgrupo, no qual o cliente pode demandar os dois tipos de serviço. Entretanto, neste problema as demandas devem ser atendidas em uma única visita. Dessa forma, a entrega e a coleta devem ser realizadas simultaneamente.

A segunda classe refere-se aos produtos transportados entre os pontos de entrega e coleta, ou seja, a entrega e a coleta ocorrem apenas dentro dos roteiros, assim, os veículos saem vazios do depósito e voltam vazios para o mesmo (Parragh et al., 2008b). Nesta segunda categoria, três subgrupos são descritos:

1. Problema clássico com pontos de coleta e entrega pareados:
 - a. Há um relacionamento entre os pontos de coleta e entrega, dessa maneira para cada ponto de coleta há um ponto de entrega específico, garantindo um sequenciamento de visitas entre pontos de coleta e de entrega;
2. Problema de roteamento de veículos com pontos de coleta e entrega não pareados:
 - a. Distintamente do problema anterior, esse problema não possui paridade entre os pontos de coleta e entrega, assim os veículos podem visitar diversos pontos de coleta antes do ponto de entrega, pois os produtos de um ponto de coleta podem ser utilizados para todos os pontos de entrega e não apenas um ponto de entrega em específico;
3. Problema *Dial - A- Ride*:
 - a. Este último problema é bastante similar ao primeiro, devido a paridade dos pontos de coleta e entrega, contudo ele é utilizado para o transporte de pessoas e não de produtos, o que leva a demandar restrições diferentes.

Para manter a objetividade, este documento será focado em problemas da primeira classe, mais especificamente no problema de roteamento de veículos com entregas e coletas divisíveis (PRVECD), no qual possui a estrutura *One-to-many to one* (1-M-1).

2.3.3.1. Problema de roteamento de veículos com entregas e coletas divisíveis

Assim como referenciado nas obras de Berbeglia et al. (2007) e Parragh et al. (2008a), o problema de roteamento de veículos com entregas e coletas divisíveis (PRVECD) é uma extensão do PRVEC, onde é possível assumir que todos os produtos são fornecidos pelo depósito e todas as coletas são levadas a ele. Neste tipo de problema, os caminhões carregam simultaneamente os produtos de entrega e coleta ao longo dos nós, porém ao final retornam ao depósito apenas com os produtos coletados. Em cada nó, a capacidade do veículo pode aumentar ou diminuir, resultando em uma flutuação de carga. Consequentemente, a checagem da viabilidade de carga deve ser realizada em cada serviço, resultando em uma das principais dificuldades para resolver este tipo de problema (Nagy et al., 2013).

Ademais, o PRVECD é visto como um importante e desafiador problema de logística com amplas aplicações no mercado sustentável (logística reversa), pois existem novas restrições complexas para garantir a sincronização das entregas e das coletas, que não estão presentes em um PRV clássico (Wassan, A. N.; e Nagy, G.; 2014).

2.3.3.2. Problema de roteamento de veículos com entregas e coletas divisíveis e janelas de tempo

O problema de roteamento de veículos com entregas e coletas divisíveis e janelas de tempo (PRVECDJT) é uma variação do PRVEC de primeira classe (Parragh et al., 2008a), assim todas as restrições e requisitos do problema de roteamento de veículos com entregas e coletas divisíveis são utilizadas, juntamente, com a adição das restrições de janelas de tempo (Furtado et al., 2017).

As restrições de janelas de tempos são impostas pelos clientes, e são, principalmente, caracterizadas por um intervalo de tempo em que as entregas e/ou as coletas devem ocorrer, ou seja, qualquer serviço feito no cliente fica inviável fora

desse período. Esse tipo de restrição pode ser constatado com mais facilidade em uma variação mais básica do problema, o problema de roteamento de veículos com janelas de tempo (PRVJT). Neste, não existe a complexidade de flutuação de carga durante o percurso (entregas e coletas) como visto em um PRVECD, e assim, acaba facilitando a compreensão das restrições de janelas de tempo. Tal característica é, por exemplo, vista na obra de Simchi-Levi et al. (2005), na qual eles desenvolvem um modelo genérico com as restrições necessárias de janelas de tempo para um problema clássico de roteamento de veículos.

Esse tipo de problema se encontra cada vez mais presente na literatura, como na pesquisa de Gan et al. (2012), que descreve um problema PRVJT com veículos de capacidade limitada, onde estes só podem realizar entregas uma única vez por período. Outro exemplo é averiguado na obra de Desrochers et al. (1988), onde ele utiliza como base um PCV clássico para desenvolver a variação do problema de roteamento de veículos com janelas de tempo, resultando em um problema de múltiplos caixeiros viajantes com janela de tempo (PMCVJT). Em seguida, eles utilizam dessa mesma variação para integrar o modelo de coletas e entregas ao problema.

Além disso, é possível estudar o problema com entregas e coletas e janelas de tempo no estudo de Iassinovskaia et al. (2017), que descreve um problema de roteamento de veículos com entregas e coletas de um fluxo de materiais retornáveis. Na obra, eles apresentam um modelo matemático, utilizam um método exato de programação linear inteira para a resolução do caso em menor escala, e, em seguida, eles propõem uma meta-heurística para problemas reais em larga escala. A grande diferença é que neste artigo eles utilizam um modelo mais complexo, onde os custos de estocagem dos objetos retornáveis são considerados. Porém, se as restrições de inventário forem eliminadas através da monoperiodicidade (Campbell, A. et al., 1998), o problema tratado por Iassinovskaia et al. (2017) se torna um problema similar ao PRVECJT.

É interessante constatar que na literatura existem diversos métodos de resolução para o problema de roteamento de veículos com entregas e coletas e janelas de tempo, envolvendo métodos heurísticos e exatos como nas obras de Lau et al. (2002), Lu e Dessouky (2004; 2006), Campbell e Savelsbergh (2004), Sassi et al. (2015), bem como na de Dridi et al. (2011).

2.4. Métodos de solução para problemas de roteamento de veículos

Há duas vertentes bastante conhecidas para a resolução de problemas de roteamento de veículos, uma conhecida como metodologia exata e outra como metodologia heurística. Estas duas abordagens serão tratadas com mais detalhes neste documento, esclarecendo suas vantagens e desvantagens, bem como os fatores que devem ser levados em consideração dependendo do problema abordado, de suas restrições, e de suas variáveis e complexidades.

2.4.1. Métodos exatos

Os métodos exatos são algoritmos que buscam exaustivamente a melhor solução para um problema estudado, ou seja, eles calculam todas as possibilidades relevantes, assegurando-se que a solução escolhida é a ótima. Geralmente, eles possuem complexidade fatorial ou exponencial dependendo do algoritmo utilizado, e, dessa forma, são utilizados para problemas de pequena ordem e com menos variáveis. Como estes calculam exaustivamente as combinações relevantes possíveis para o problema, se este for muito complexo e com múltiplos números a serem processados, o tempo de compilação pode se tornar inviável, como demonstrado no estudo de Ahuja et al. (2007).

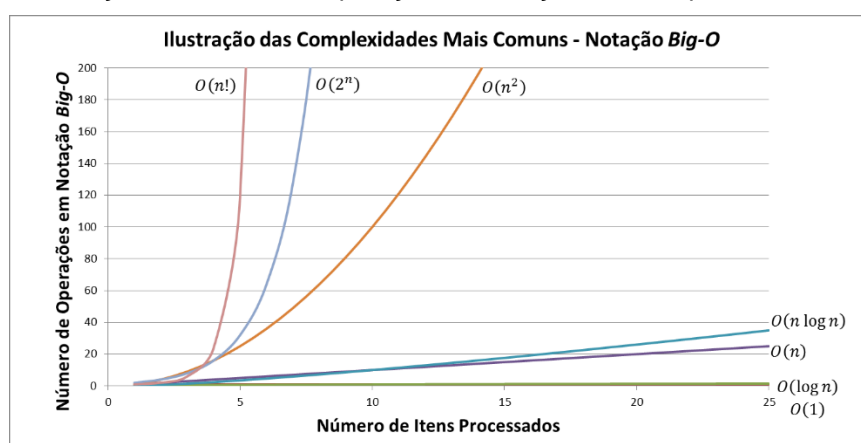
É possível constatar a grande complexidade de resolver problemas de roteamento de veículos através de métodos exatos na obra de Laporte e Nobert (1987), onde comparam a resolução de problemas gerais de PRV e PCV, enfatizando a grande dificuldade de implementar algoritmos exatos para os PRVs. Se analisarmos ainda mais precisamente, os PRVEC são ainda mais difíceis que os PRVs clássicos, pois na obra de Nagy et al. (2013), os autores reforçam que os PRVEC são conceitualmente mais difíceis que um PRV, devido às restrições de viabilidade de roteiro e sua flutuação carga durante o percurso.

Para a resolução dos métodos exatos, há diferentes algoritmos conhecidos, como o Simplex, *Branch-and-Bound*, programação dinâmica etc., que, apesar de possuírem procedimentos e complexidades diferentes, fazem a busca exaustiva das possibilidades relevantes para a solução. Na obra de Lu e Dessouky (2004) é desenvolvida uma nova metodologia de métodos exatos para o problema de

roteamento de veículos com entregas e coletas e janelas de tempo através do algoritmo *Branch-and-Cut*.

Na Figura 6, é possível comparar a complexidade (*Big “O”*) dos algoritmos exatos, geralmente com complexidade fatorial ou exponencial, em relação às outras complexidades mais comuns existentes, por exemplo, a complexidade polinomial ($O(n)$). Constata-se um aumento expressivo do tempo de execução em função do aumento do número de variáveis nas complexidades fatorial e exponencial se compararmos com a complexidade polinomial ou logarítmica.

Figura 6 – Ilustração do número de operações em função das complexidades mais comuns



Fonte: BIG-O cheat sheet, 2013

2.4.2. Métodos heurísticos

Os métodos heurísticos são algoritmos exploratórios utilizados, principalmente, para resolver problemas robustos, ou seja, problemas extensos e difíceis, que não podem ser resolvidos pelos métodos exatos devido à sua complexidade intratável (Ahuja et al., 2007). O principal objetivo desses métodos é propor uma solução satisfatória para o problema, na qual esta, pode ser ou não a melhor solução possível. Portanto, através dos métodos heurísticos, busca-se uma resolução mais simples do problema, porém com respostas viáveis, ainda que não necessariamente ótimas (Savelsbergh e Sol, 1995) (Lu e Dessouky, 2006).

Na obra de Colin (2007), há uma analogia para simular o funcionamento de um método heurístico genérico, onde ele reproduz um procedimento para desvendar, empiricamente, o ponto mais alto da Terra. Assim, inicialmente, partir-se-ia de um ponto viável da superfície terrestre com o objetivo de localizar as montanhas mais altas. Neste processo, várias montanhas seriam escaladas e suas

alturas comparadas, aumentando, progressivamente, o ponto mais alto de acordo com as novas descobertas (os chamados ótimos locais). Até que em um determinado momento as buscas se dessem por encerradas por algum motivo (p.e.; a não descoberta de pontos mais altos por um longo período, restrições de tempo ou financeiras, etc.) e o ponto mais alto fosse definido, mesmo que outro ponto mais alto possa ainda estar oculto. Analogamente ao problema apresentado, os métodos heurísticos são buscas contínuas e empíricas, com vários ótimos locais, cujo resultado é o melhor que se pode encontrar sob determinadas condições.

De acordo com Laporte e Semet (2002), os métodos heurísticos destinados aos problemas de roteamento de veículos podem ser classificados em duas classes principais, que são divididas em categorias menores:

1. Heurística clássica:

a. Heurística construtiva:

- i. Constrói gradativamente uma solução viável, baseando-se no custo da solução. Existem dois principais algoritmos utilizados: algoritmos de inserção e algoritmos de Clarke e Wright;

b. Heurística de duas fases:

- i. Possui as mesmas características de uma heurística construtiva, porém esta possui duas fases específicas, a fase de agrupamento, no qual consiste em agrupar clientes em subgrupos menores, e a fase de roteamento, onde cria-se roteiros para cada subgrupo. O principal algoritmo utilizado para essa categoria é o algoritmo de agrupamentos.

c. Métodos de aprimoramentos:

- i. Executa operações em uma determinada solução inicial, visando o aprimoramento da mesma. Um exemplo, seria a heurística de busca local.

2. Meta-heurística:

- a. São estratégias genéricas para resolução de problemas se embasando em técnicas tradicionais de pesquisa e métodos heurísticos. Os métodos meta-heurísticos não são desenvolvidos

para um tipo específico de problema, eles propõem modelos genéricos (*frameworks*) de alta aplicabilidade a uma ampla gama de problemas. Normalmente, eles são baseados em fenômenos naturais e físicos, como por exemplo os algoritmos genéticos, redes neurais, comportamento de insetos, etc. (Nesmachnow, 2014).

Diversos estudos na literatura utilizam algoritmos heurísticos para solucionar problemas de roteamento de veículos. Essas resoluções contribuíram para o discernimento de diferentes possíveis abordagens de resolução, evidenciaram testes pertinentes para aferir a solução, e, assim, elevaram a compreensão dos fatores intrínsecos de funcionamento.

Nos estudos de Lu e Dessouky (2006) foi desenvolvido um novo algoritmo através de um método heurístico de construção por inserção para um problema de PRVEC de segunda classe com pontos de coleta e entrega pareados e múltiplos veículos, na obra de Dridi et al. (2011) foi desenvolvido um algoritmo genético para a resolução de um PRVEC de segunda classe com múltiplos veículos, e nos estudos de Desrochers et al. (1988) e de Lau et al. (2002), foi apresentado um método heurístico de construção por inserção, juntamente com um método de aprimoramento, a diferença é que no primeiro estudo foi utilizado um método heurístico de aprimoramento por busca local e no segundo um método meta-heurístico de aprimoramento por busca tabu.

Neste trabalho de formatura destinaremos o enfoque para os métodos heurísticos de construção, mais precisamente para os algoritmos heurísticos de construção por inserção, e para os métodos heurísticos de aprimoramento, mais especificamente para os algoritmos heurísticos de aprimoramento por busca local.

2.4.2.1. Heurística de inserção

A heurística de inserção é um método heurístico clássico de construção (Laporte e Semet, 2002), que consiste em gerar um circuito viável de nós partindo de um conjunto inicial e, em seguida, modificá-lo a cada iteração através da adição de mais nós ao sistema. Esta inserção segue critérios de escolha decididos previamente pelos desenvolvedores, de acordo os objetivos do problema de estudo (Gomes et al., 2013). Assim, a seleção deles são essenciais para desenvolver um

método eficaz, um exemplo, seria a redução de custo pela distância mais próxima, um dos critérios de escolha mais comuns na literatura (Campbell e Savelsbergh, 2004). Esse critério é evidenciado no algoritmo de inserção da obra de Nascimento et al. (2004).

A grande importância desses algoritmos em um PRVEC com janelas de tempo é constatada na obra de Lu e Dessouky (2006), onde mostra que os algoritmos de inserção são bastante utilizados na construção de soluções iniciais/finais desse tipo de problema, devido à geração de soluções com rapidez e qualidade.

Segundo a obra de Ropke (2006), os métodos heurísticos de inserção podem ser caracterizados em dois grupos:

1. Heurística de inserção sequencial;
 - a. Algoritmos de inserção que constroem um roteiro por vez;
2. Heurística de inserção em paralelo;
 - a. Algoritmos de inserção que constroem mais de um roteiro em paralelo;

Neste documento o enfoque será apenas nos métodos heurísticos de inserção em paralelo. Esta heurística é observada em diversos documentos de estudo realizados por Lu e Dessouky (2006), Lau et al. (2002), Benoist (2010) e Desrochers et al. (1988) que foram utilizados métodos heurísticos de inserção para solucionar um PRVEC com janelas de tempo.

2.4.2.2. Heurística de Busca local

O algoritmo de busca local é uma metodologia de aprimoramento pertencente à heurística clássica, ela visa melhorar os resultados viáveis dos métodos de construção, como a heurística de inserção, através do aumento da variabilidade entre os arcos dos roteiros construídos. Assim, o algoritmo aplica modificações locais para alcançar novas soluções viáveis (soluções candidatas) até encontrar uma solução ótima local ou ser suspensa devido a uma restrição aplicada, por exemplo, o número de iterações (Hoos e Stutzle, 2005).

Os algoritmos de busca local são amplamente difundidos e aplicados para problemas computacionais difíceis, principalmente nos campos de inteligência

artificial e pesquisa operacional. Alguns exemplos desses algoritmos são o recozimento simulado, o operador 2-opt, busca tabu etc. (Pirlot, 1996).

2.4.2.2.1. Busca local 2-opt

Este operador pertence à família heurística dos métodos de aprimoramento de buscas locais e é um dos algoritmos mais conhecidos e promissores da literatura de roteamento de veículos. Este possui como objetivo melhorar a solução atual através da variação iterativa de uma vizinhança pré-definida. Seu princípio funciona através da remoção de dois arcos de um determinado roteiro, e, em seguida, comutá-los, obtendo-se novas combinações (Mersmann, 2012). Se essa modificação for melhor que a precedente ela é mantida, senão a antiga permanece. Assim, esse processo se repete até a escassez de permutações ou até que um fator de parada seja atingido (Venhuys, 2012) (Hashimoto et al., 2008).

As vantagens desse método se demonstram bem estabelecidos tanto na teoria quanto na prática, primeiramente por ser pouco oneroso à complexidade da solução desenvolvida, ou seja, não há um aumento significativo dos tempos de execução da metodologia, secundamente por propiciar uma melhoria considerável das soluções em relação às iniciais (Mersmann, 2012).

Esse tipo de algoritmo é bastante utilizado no roteamento de veículos, principalmente sob forma de permutações intra-roteiro e inter-roteiros. Assim, quando o algoritmo realiza permutações dentro de um mesmo roteiro, é denominado intra-roteiro, efetuando uma modificação da ordem de visitas no mesmo. E quando ele realiza uma permutação entre os diferentes roteiros, é denominado inter-roteiros (Tavares et al., 2008).

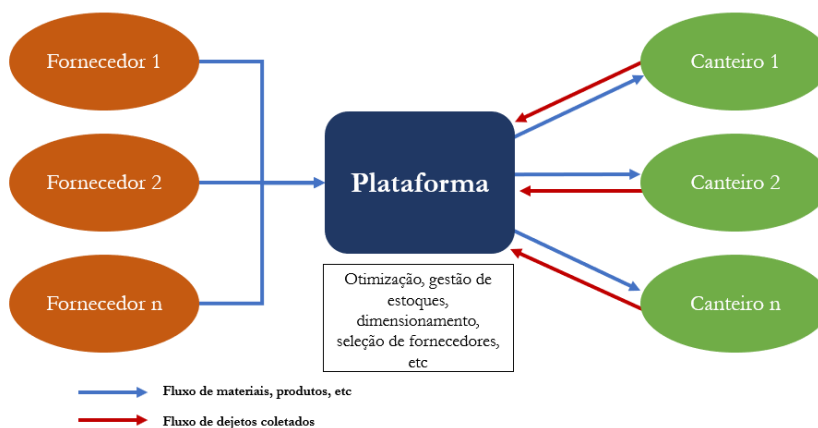
Dessa maneira, esse algoritmo é observado nos artigos de Hashimoto et al. (2008) que propõem um algoritmo de busca local através do operador 2-opt para um PRVJT; nas obras de Benoist (2010) e Tavares et al. (2008) que evocam a criação de um roteiro inicial por meio de um algoritmo de inserção e, em seguida, efetuam a busca local através de permutações intra e inter roteiros; e no artigo de Sassi et al. (2015) que utiliza a busca local 2-opt para o aprimoramento de um problema de roteamento de veículos com entregas e coletas de produtos.

3. DESCRIÇÃO DO PROBLEMA

3.1. Apresentação geral do problema

O propósito geral é o de trabalhar com as plataformas modulares por meio da concepção e do desenvolvimento de métodos eficazes para a otimização de uma cadeia de fluxo sustentável e durável no setor da construção civil, desde o provisionamento dos produtos à gestão e valorização dos descartes dos canteiros de obras, enfatizando suas especificidades e restrições. A figura 7 ilustra a dinâmica da distribuição de maneira geral e simples, salientando os fluxos dos produtos e dos descartes, enquanto a Seção 2.1.2. caracteriza em detalhes o funcionamento das plataformas modulares.

Figura 7 – Esquema modelo da distribuição do setor da construção civil



Fonte: elaboração do autor

O funcionamento da plataforma é baseado em um processo extenso, que faz intervir três domínios relacionados entre si (cf. Figura 8). Cada domínio será especificado com mais detalhes para se obter uma visão ampla e clara do funcionamento global da plataforma.

Figura 8 – Esquema do processo geral da plataforma modular

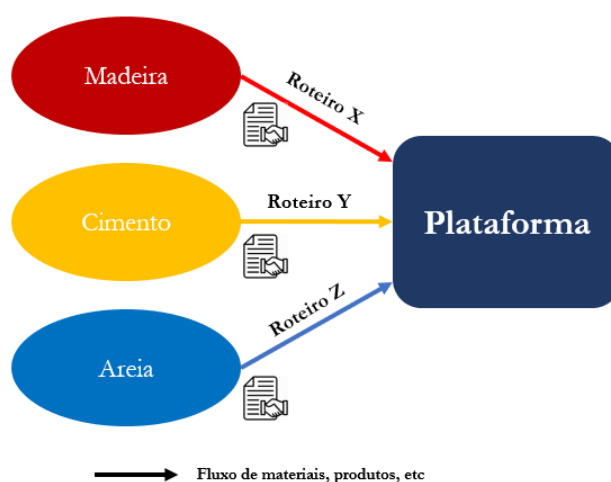


Fonte: elaboração do autor

3.1.1. Logística à montante

A partir do contrato de parceria entre fornecedor e cliente, a plataforma efetua seus aprovisionamentos com um certo número de fornecedores, por exemplo, o fornecedor de cimento, de madeira, de tinta, entre outros materiais. Desta forma, é basicamente uma relação onde o cliente é a plataforma e a empresa o fornecedor como indicado no esquema da Figura 9.

Figura 9 – Esquema da logística à montante com seus respectivos roteiros



Fonte: elaboração do autor

Esta parceria entre os dois atores é bastante decisiva do ponto de vista tanto da plataforma quanto dos fornecedores, estabelecendo entre si um negócio colaborativo. Além disto, essa relação possui um impacto intimamente ligado aos objetivos e metas da plataforma, e ao seu crescimento, uma vez que os materiais fornecidos são intrínsecos para o funcionamento básico dela.

Entretanto, como a plataforma é o cliente de múltiplos fornecedores, torna-se muito mais difícil a interferência no planejamento logístico dos mesmos, uma vez que cada fornecedor é responsável por gerar e desenvolver sua própria gestão de suprimentos, levando em consideração sua carteira de clientes e parceiros para a criação de seus roteiros e sua solução logística “ótima”. Assim, não há necessidade de um estudo mais aprofundado desse domínio.

3.1.2. Plataforma modular

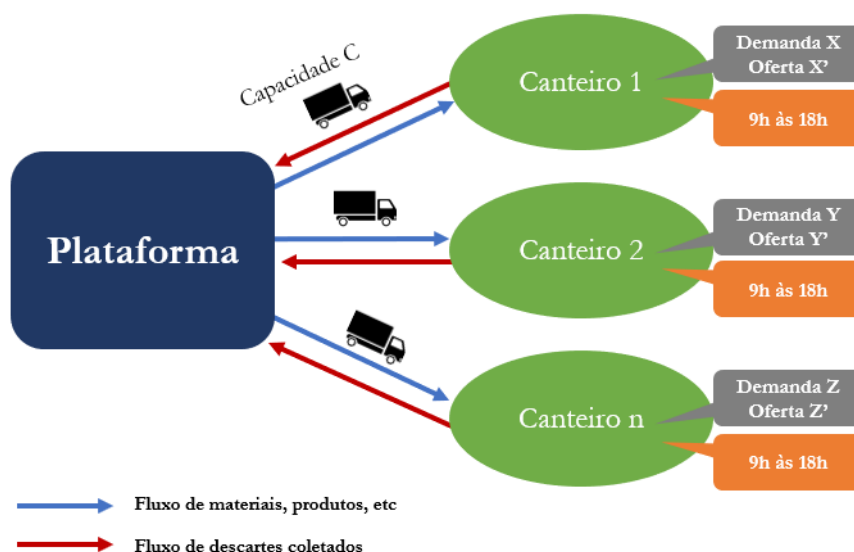
Como explicitado anteriormente na Seção 2.1.2.; a plataforma opera como um centro de distribuição versátil e modular para os produtos, bem como um local de triagem, reciclagem e descarte para os resíduos coletados pela logística reversa.

Contudo, como a logística interna da plataforma e a gestão de descartes não estariam no escopo de estudo, estes não serão aprofundados com mais detalhes, porém é de suma importância ter em mente que esse é um procedimento intrínseco para operações que visam utilizar uma plataforma modular para realizar a logística reversa.

3.1.3. Logística à jusante

Essa última etapa tem como finalidade responder às necessidades dos canteiros de obras através do transporte dos materiais demandados, bem como da coleta de seus descartes gerados, exigindo um trabalho de coordenação e sincronismo entre ambas as partes (cf. Figura 10). Dessa maneira, é indiscutível desenvolver/achar uma boa combinação de exploração de recursos, como roteiros de serviço otimizados. Este terceiro processo será o enfoque desse trabalho, que será detalhado nos próximos tópicos.

Figura 10 – Esquema da logística à jusante



Fonte: elaboração do autor

3.1.3.1. Transporte de produtos e descartes

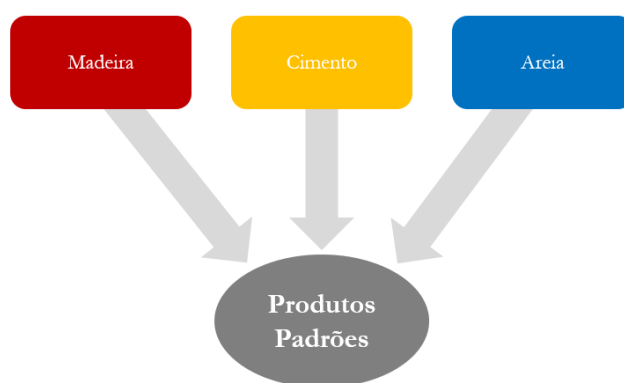
Como dito anteriormente, a logística à jusante consiste no transporte do fluxo direto (transporte de produtos) e reverso (transporte de descartes), entre os canteiros de obras e a plataforma. Assim, ambos serão especificados e caracterizados a seguir.

1. Fluxo direto

Neste tipo de fluxo, os caminhões entregam diversos tipos de produtos aos canteiros de obras de acordo com suas respectivas demandas, como cimento, madeira, entre outros materiais. Como são produtos vedados e embalados corretamente pelos fornecedores, não existem problemas ou quaisquer dificuldades no transporte e gestão dos mesmos. Em outras palavras, é possível carregar diferentes quantidades e tipos de produtos em um mesmo caminhão sem nenhuma complicação de misturá-los ou afetá-los de alguma maneira.

Dessa forma, estes produtos e materiais serão considerados como um único tipo de produto (cf. Figura 11), visto que não existem diferenças em seu tipo de transporte, além do que, essa padronização dos materiais de entrega busca simplificar a complexidade do problema em questão.

Figura 11 – Padronização dos tipos de produtos e materiais



Fonte: elaboração do autor

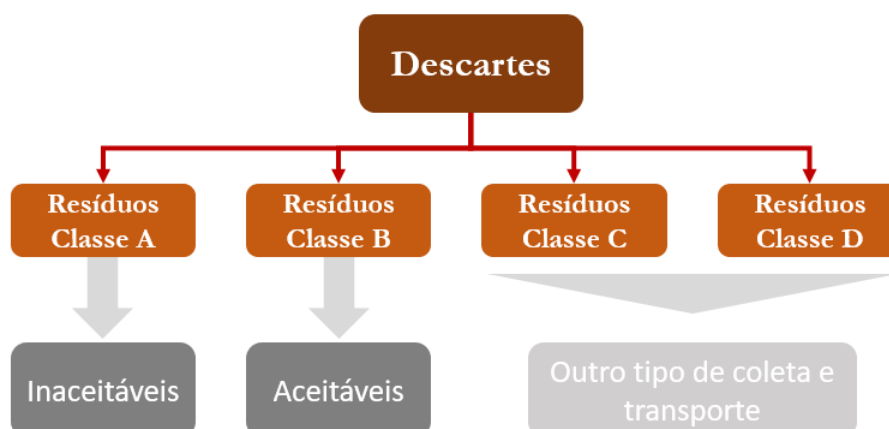
2. Fluxo reverso

Nesta parte da distribuição, os veículos coletam uma gama de descartes gerados pelos canteiros de obras, segundo as “ofertas” de cada um. Infelizmente,

diferentemente do que acontece no fluxo direto, existem descartes que não podem ser misturados com outros tipos de carga (produto ou descarte) durante o transporte deles. Dessa forma, eles devem receber uma coleta especial por um caminhão particular para essa situação.

A partir da classificação de descartes realizada na Seção 2.1.4.; é possível considerar dois tipos de descartes para este trabalho, os aceitáveis e os inaceitáveis (cf. Figura 12). Pois, existem diferenças substanciais em suas logísticas, uma vez que, no primeiro caso, os descartes podem ser misturados com os produtos do fluxo direto e, logo, utilizar o mesmo tipo de veículo de entrega, enquanto que, no segundo caso, eles não são misturáveis, e, portanto, existe a necessidade de utilizar um veículo diferente para a coleta dos mesmos. Os dois tipos de descartes são especificados a seguir.

Figura 12 – Classificação de descartes



Fonte: elaboração do autor

- **Descartes aceitáveis**

São todos os descartes que podem ser misturados com os produtos do fluxo direto. Ou seja, são os descartes recicláveis gerados nos canteiros de obras, no qual é possível, colocá-los em um mesmo espaço que os produtos demandados, sem qualquer risco de danificação para os bens ou o transporte em si. Geralmente, os descartes aceitáveis são papelões, vidros, papéis, madeira, gesso e alguns tipos de metais, isto é, os resíduos sólidos de classe B (cf. Figura 13). Para o transporte desses descartes, são utilizados caminhões de carga $\frac{3}{4}$ ou VUC (Veículo Urbano de Carga), que realizam tanto a operação de entrega, quanto de coleta.

Figura 13 – Exemplo de descartes aceitáveis (papelão) utilizados no setor da construção civil



Fonte: CBS, 2019

- Descartes inaceitáveis

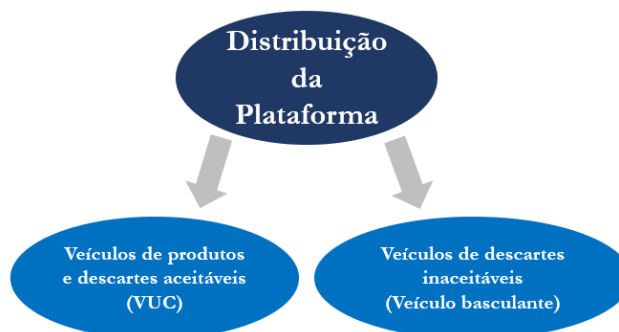
São os descartes recicláveis que podem causar danos às partes funcionais dos produtos e, dessa forma, necessitam de um outro tipo de coleta e transporte para a plataforma modular. Esses descartes são geralmente compostos por tijolos, telhas, blocos, placas de revestimento, argamassa, concreto etc., ou seja, são os resíduos sólidos recicláveis e reutilizáveis de classe A.

Assim, especialmente para esse tipo de descarte, o transporte é realizado através de caminhões basculantes, um tipo de veículo que faz a coleta direta desses resíduos nos canteiros de obras. Normalmente, esse tipo de caminhão possui apenas a capacidade de uma caçamba basculante por vez, necessitando retornar à plataforma após a coleta. Dessa maneira, para diminuir a dependência dos canteiros de obras em relação às coletas dos descartes inaceitáveis, e facilitar a triagem e a reciclagem dos mesmos na plataforma, são, normalmente, instalados cinco caçambas basculantes por canteiro de obras, a fim de aumentar a capacidade de armazenagem e incentivar uma separação prévia de alguns tipos de materiais.

3.2. Separação do problema em dois subproblemas

Dado que o problema possui um enfoque na logística à jusante, onde a distribuição é efetuada por dois tipos de veículos bem precisos e distintos, uma repartição em dois subproblemas independentes foi realizada (cf. Figura 14), a fim de diminuir a complexidade do problema.

Figura 14 – Separação do problema em dois subproblemas



Fonte: elaboração do autor

3.2.1. Veículos basculantes para o transporte de descartes inaceitáveis

Este subproblema requer a utilização de veículos basculantes para o transporte dos descartes inaceitáveis dos canteiros de obras à plataforma modular. Dada a capacidade de uma única caçamba basculante por veículo, há a necessidade do retorno do mesmo à plataforma após a coleta, criando-se roteiros diretos (plataforma-canteiro e canteiro-plataforma). A necessidade de coleta desses descartes se baseia pelo nível do estoque em cada canteiro de obras, ou seja, a caçamba de um determinado canteiro será apenas coletada quando o nível de descartes inaceitáveis exceder um limite imposto. É interessante constatar que o número de caçambas varia de canteiro para canteiro, bem como o preenchimento de cada caçamba. Sendo esses fatores, baseados no modelo de gestão dos canteiros de obras (p.e. pré-separação de descartes) e no nível de “oferta” de cada canteiro.

Apesar do transporte dos descartes inaceitáveis ser uma tarefa intrínseca para o papel da plataforma no desenvolvimento sustentável, este subproblema não será o enfoque deste documento. Assim, o estudo se concentrará totalmente no problema de logística reversa de entrega de produtos e coleta de descartes aceitáveis (descartes de classe B) que será especificado a seguir.

3.2.2. Veículos VUC para o transporte de produtos e descartes aceitáveis

Este subproblema engloba o transporte entre a plataforma modular e os diversos canteiros de obras, cuja distribuição é realizada por uma frota de veículos homogêneos do tipo VUC (cf. Figura 15), ou seja, caminhões semi-leves destinados

à entrega urbana, que possuem as mesmas características entre si (custos, capacidades, etc.). Cada canteiro de obras pode demandar dois tipos de operações não exclusivas:

1. Entrega de produtos;
2. Coleta de descartes aceitáveis.

Como os produtos e os descartes não são dependentes, e não existe uma ordem prioritária de entregas e coletas, os canteiros de obras que demandarem tanto a entrega quanto a coleta, não serão, necessariamente, servidos de ambas as operações ao mesmo tempo. Ademais, como no problema não ocorrem entregas e coletas fracionadas, cada canteiro de obras só é visitado no máximo duas vezes.

Por fim, durante as operações, cada veículo deve respeitar, obrigatoriamente, os horários de abertura e encerramento estipulados por cada canteiro de obras, assim não é permitida a realização de qualquer serviço fora desse espaço de tempo.

Figura 15 – Caminhão de entrega/coleta do tipo VUC



Fonte: La Plateforme du batiment, 2019

3.3. Definição do problema

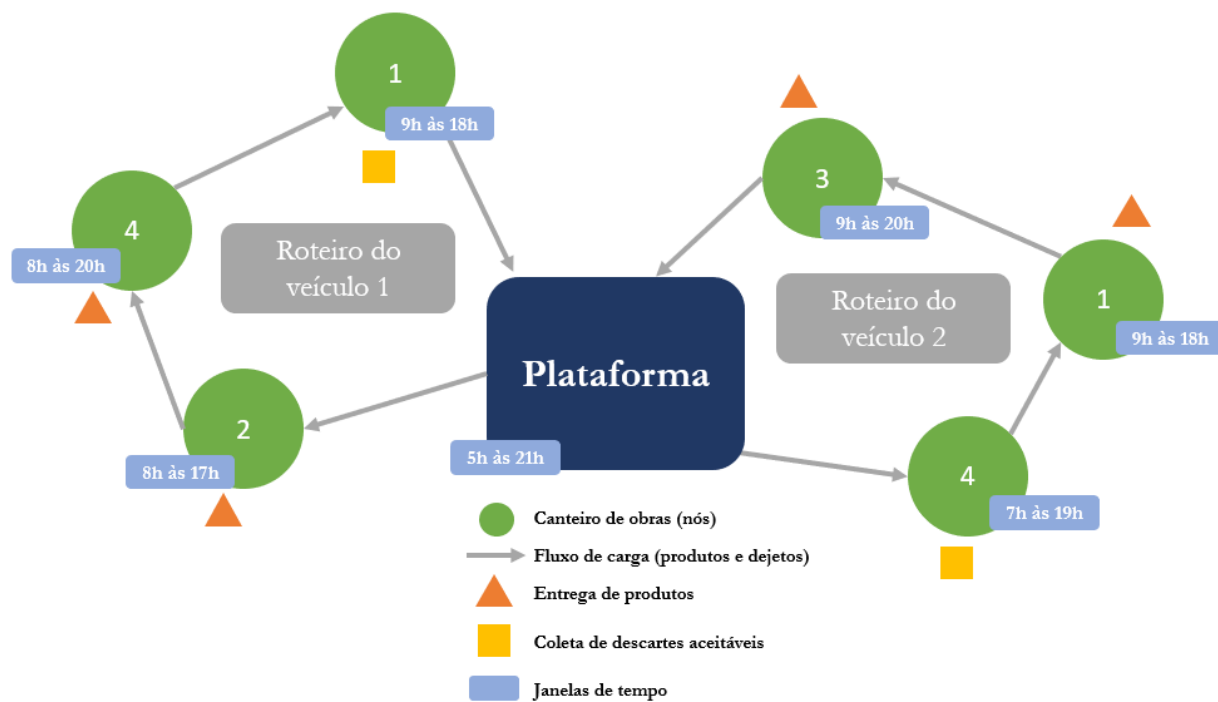
De acordo com a revisão bibliográfica no Capítulo 2 e com o contexto em que está inserido o problema, é possível constatar que ele é uma das variações do problema de roteamento de veículos clássico (PRV). Assim, existem alguns fatores peculiares, que o tornam um pouco mais complexo que um PRV.

Primeiramente, seria a caracterização da estrutura do grafo do problema como um tipo *One-to-many-to-one* (1-M-1) (Battarra, 2014), pois este trata apenas de uma única plataforma modular de onde os veículos partem para os canteiros e, em seguida, retornam. O segundo fator seria o tipo de visita, pois o problema possui a particularidade da integração de um parâmetro ecológico na otimização de distribuição, a logística reversa (Murphy e Poist, 1989), originando um segundo fluxo de transporte. Além disso, cada canteiro de obras pode demandar dois tipos de operações não exclusivas e independentes, visto que os produtos e os descartes não são substituíveis, sendo os produtos fornecidos apenas pela plataforma modular, e os descartes entregues apenas para a mesma. Esse tipo de problema é bem conhecido na literatura como um PRVECD (Parragh et al., 2008a). O terceiro fator seria a inserção das janelas de tempo junto ao problema, pois cada canteiro de obras possui seu próprio horário de funcionamento, que deve ser respeitado (Simchi-Levi et al., 2005). Por fim, por questões de complexidade, o horizonte de tempo será tratado como monoperiódico, possuindo uma quantidade restrita de veículos homogêneos com capacidade de carga limitada.

Portanto, de acordo com a classificação de taxonomia de um PRV nas obras de Berbeglia et al. (2007) e Parragh et al. (2008a), o problema poderia se formalizar como um problema de roteamento de veículos monoperiódico com uma estrutura de grafo do tipo *One-to-many-to-one* (1-M-1), com um modelo de visitas segundo um problema de roteamento de veículos com entregas e coletas divisíveis e janelas de tempo (PRVECDJT), e com múltiplos veículos homogêneos de quantidade e capacidade limitada.

Assim, a solução consistiria na proposição de roteiros para os veículos, de maneira a satisfazer todos os canteiros de obras, servindo-os no máximo duas vezes. A Figura 16 apresenta um exemplo de solução para quatro canteiros de obras utilizando dois veículos, onde os canteiros “1” e “4” possuem as operações de entrega e coleta, e os canteiros “2” e “3” possuem apenas um tipo de operação. Assim, nos canteiros “3” e “4” são realizadas duas visitas por veículos distintos, enquanto nos canteiros “2” e “3” é realizada apenas uma.

Figura 16 – Exemplo de solução para o problema com quatro canteiros de obras e dois veículos



Fonte: elaboração do autor

4. MODELAGEM MATEMÁTICA

A modelagem matemática é utilizada para sintetizar problemas reais através de relações matemáticas, ou seja, eles são traduções de leis que conduzem fenômenos, processos ou sistemas, possíveis de serem modelados por equações (Arenales et al., 2007).

Para o presente problema, PRVECDJT do tipo (1-M-1), monoperiódico e com veículos homogêneos limitados, o método heurístico é o mais recomendável. Principalmente devido a dois aspectos, a eficiência computacional com a habilidade de operar em tempo real na planificação de roteiros (Palmer et al., 2004) e a concentração na viabilidade ao invés da otimalidade para problemas de larga escala (Desrochers et al., 1988).

Contudo, para avaliação posterior dos resultados da ferramenta desenvolvida, a implementação de uma solução por um aplicativo de otimização exato de propósito geral também foi empregada. Dessa maneira, ocorreram simplificações de alguns fatores para possibilitar a viabilidade de execução de ambos os métodos, como a unificação de tipos de produtos e descartes (Seção 3.1.3.1), e a monoperiodicidade. Pois, a grande quantidade de variáveis impossibilitaria a execução da abordagem por métodos exatos e, conseqüentemente, a comparação dos resultados com a heurística desenvolvida.

Assim, foi preparado e organizado um modelo de declaração de variáveis e parâmetros para a implementação do modelo no programa AMPL. Este modelo foi baseado nas obras literárias de roteamento de veículos clássicos com janelas de tempo como a de Desaulniers (2011) e Ismail et al. (2011), bem como em obras de roteamento de veículos com entregas e coletas como a de Nagy et al. (2013) e a de Iassinovskaia et al. (2017), sem a inclusão dos custos de inventários.

4.1. Parâmetros e conjuntos

Os parâmetros (cf. Tabela 1) e conjuntos (cf. Tabela 2) são extremamente importantes para a estruturação e a modelagem desse problema, uma vez que eles serão utilizados como base para obter a solução ótima.

Como a situação envolve um problema de roteamento de veículos com entregas e coletas divisíveis, ao invés de tratarmos como um único canteiro de obras

com duas demandas distintas (entrega e coleta), foi realizada a criação de nós fictícios dos canteiros de obras, assim, cada canteiro foi desagregado em dois nós, nó de entrega pertencente ao conjunto N_e e nó de coleta pertencente ao conjunto N_c , onde a distância entres eles permanece nula.

Tabela 1 – Parâmetros para a modelagem matemática.

Parâmetros	Descrição	Notação
n	O número de canteiros de obras	$\{n \in \mathbb{R}_+^*\}$
C	A capacidade de carga dos veículos (m^3)	$\{C \in \mathbb{R}_+^*\}$
vel	A velocidade média de cada veículo (km/h)	$\{vel \in \mathbb{R}_+^*\}$
$D_{i,j}$	As distâncias entre os nós (km)	$\{(D_{0,0} \dots D_{2n,2n}) \mid D_{i,j} \in \mathbb{R}_+\}$
q_i	As demandas de cada nó (m^3) <i>Entrega, se $i \leq n$;</i> <i>Coleta, se $i > n$;</i>	$\{(q_1 \dots q_{2n}) \mid q_i \in \mathbb{R}_+\}$
e_i	O horário de abertura de cada nó (h)	$\{(e_0 \dots e_{2n}) \mid 0 < e_i \leq 24\}$
l_i	O horário de encerramento de cada nó (h)	$\{(l_0 \dots l_{2n}) \mid 0 < l_i \leq 24\}$
TS_i	Tempo médio de serviço (h)	$\{TS_i \in \mathbb{R}_+^*\}$

Fonte: elaboração do autor

Tabela 2 – Conjuntos para a modelagem matemática.

Conjuntos	Descrição	Notação
N_e	Nós de entrega	$\{N_1, \dots, N_n\}$
N_c	Nós de coleta	$\{N_{n+1}, \dots, N_{2n}\}$
N	Nós dos canteiros de obras	$\{N_e \cup N_c\}$
N_0	A plataforma modular	$\{N_0\}$
S	Nós dos canteiros de obras e a plataforma	$\{N_0 \cup N\}$
V	A frota de veículos	$\{v_1, \dots, v_k \mid k \in \mathbb{R}_+^*\}$

Fonte: elaboração do autor

4.2. Variáveis de decisão

As variáveis de decisão do problema possuem valores mutáveis de acordo com as especificidades e momentos do problema, diferentemente dos parâmetros que possuem valores fixos pré-estabelecidos. Para este problema, dois tipos de variáveis de decisão são detalhados na Tabela 3:

1. Variáveis binárias:
 - a. São representadas somente por dois valores, “0” ou “1”;
2. Variáveis Reais:

a. São representadas por valores reais.

Tabela 3 – Variáveis de decisão para a modelagem matemática.

Variáveis	Descrição	Notação
$Z_{i,j,v}$	Se um veículo percorre ou não um arco	$\{ Z_{i,j,v} \in \{0, 1\} \mid (i,j) \in S, v \in V \}$
$Xp_{i,j,v}$	A quantidade de carga de produtos em um veículo em um determinado arco	$\{ Xp_{i,j,v} \in \mathbb{R}_+ \mid (i,j) \in S, v \in V \}$
$Xd_{i,j,v}$	A quantidade de carga de descartes aceitáveis em um veículo em um determinado arco	$\{ Xd_{i,j,v} \in \mathbb{R}_+ \mid (i,j) \in S, v \in V \}$
$A_{i,v}$	O instante de chegada do veículo a um nó	$\{ A_{i,v} \in \mathbb{R}_+^* \mid i \in S, v \in V \}$

Fonte: elaboração do autor

4.3. Função objetivo

O objetivo primário desse trabalho é minimizar os custos envolvidos nos trajetos de todos os veículos, ou seja, reduzir a distância total percorrida pela frota. Dessa forma, os custos variáveis são estritamente ligados ao somatório da distância percorrida por cada veículo.

$$\text{Minimizar} \left(\sum_{i,j \in S, v \in V} (Z_{i,j,v} * D_{i,j}) \right) \quad (4.1)$$

4.4. Restrições

As restrições de um problema servem para limitar o domínio de pertinência das soluções, em outras palavras, a solução deve respeitar as exigências do problema real para contribuir com a veracidade dele. Dessa forma, através de documentos clássicos de roteamento de veículos e suas variações como Ismail et al. (2011), Campbell et al. (1998 e 2002) e Iassinovskaia et al. (2017), as seguintes restrições foram desenvolvidas:

1. Número de visitas:

- a. A restrição (4.2) garante que cada nó seja visitado no máximo uma vez. Assim, um determinado arco (i,j) será utilizado por um único veículo se ambos os nós, i e j , apresentarem demanda, ou se i for a plataforma e j um nó com demanda, ou o inverso;

$$\sum_{v \in V, i \in S: i \neq j} Z_{i,j,v} \leq 1, \forall j \in N \quad (4.2)$$

2. Partida da plataforma modular “ N_0 ”:

- a. A restrição (4.3) garante que todos os veículos possuam um ponto de partida comum, a plataforma modular. Além disso, ela também garante que exista no máximo um arco que inicie na plataforma para cada veículo, limitando os veículos a um único roteiro;

$$\sum_{j \in N} Z_{N_0,j,v} \leq 1, \forall v \in V \quad (4.3)$$

3. Continuidade de roteamento:

- a. A restrição (4.4) garante a continuidade do trajeto de um veículo. Dessa forma, o somatório das entradas de um nó deve ser igual ao somatório das saídas do mesmo. Como todos os veículos sempre partem da plataforma, eles são obrigados a retornar à mesma ao final do roteiro;

$$\sum_{i \in S, i \neq s} Z_{i,s,v} - \sum_{j \in S, s \neq j} Z_{s,j,v} = 0, \forall v \in V, \forall s \in S \quad (4.4)$$

4. Capacidade de carga do caminhão:

- a. A restrição (4.5) garante que um veículo não transporte uma quantidade superior ao seu limite máximo de carga. Portanto, se um veículo utiliza um determinado arco (i,j) , o somatório de carga (produtos e descartes aceitáveis) desse arco deve ser inferior ou igual à capacidade do veículo;

$$Xp_{i,j,v} + Xd_{i,j,v} \leq C * Z_{i,j,v}, \forall i, j \in S, \forall v \in V \quad (4.5)$$

5. Consistência temporal:

- a. A restrição (4.6) garante a consistência temporal do roteiro. Assim, o instante de chegada a um determinado nó é calculado pelo instante de chegada do nó precedente somado aos tempos do percurso e do serviço neste último;

$$A_{i,v} + Z_{i,j,v} * \left(Ts_i + \frac{D_{i,j}}{vel} \right) - l_{N_0} * (1 - Z_{i,j,v}) \leq A_{j,v} \quad (4.6)$$

$$, \forall i \in S, \forall j \in N, \forall v \in V$$

6. Os horários de abertura e encerramento dos nós:

- a. A restrição (4.7) garante que o acesso a cada nó do problema é realizado, estritamente, entre o horário de abertura e de encerramento estabelecidos pelo mesmo;

$$e_i * \sum_{j \in S, j \neq i} Z_{j,i,v} \leq A_{i,v} \leq l_i * \sum_{j \in S, j \neq i} Z_{j,i,v}, \forall i \in S, \forall v \in V \quad (4.7)$$

7. Demanda de produtos e descartes aceitáveis:

- a. As restrições (4.8) e (4.9) garantem o descarregamento de carga de produtos no nó “i”, se este pertencer ao conjunto de entregas N_e ;

$$\sum_{v \in V, s \in S: s \neq i} (Xp_{s,i,v}) - \sum_{v \in V, s \in S: s \neq i} (Xp_{i,s,v}) = q_i, \forall i \in N_e \quad (4.8)$$

$$\sum_{v \in V, s \in S: s \neq i} (Xp_{s,i,v}) - \sum_{v \in V, s \in S: s \neq i} (Xp_{i,s,v}) = 0, \forall i \in N_c \quad (4.9)$$

- b. As restrições (4.10) e (4.11) certificam o carregamento de carga de descartes aceitáveis no nó “i”, se este pertencer ao conjunto de coletas N_c ;

$$\sum_{v \in V, s \in S: s \neq i} (Xd_{i,s,v}) - \sum_{v \in V, s \in S: s \neq i} (Xd_{s,i,v}) = q_i, \forall i \in N_c \quad (4.10)$$

$$\sum_{v \in V, s \in S: s \neq i} (Xd_{i,s,v}) - \sum_{v \in V, s \in S: s \neq i} (Xd_{s,i,v}) = 0, \forall i \in N_e \quad (4.11)$$

- c. A restrição (4.12) garante que todo o veículo parta da plataforma sem carga de descartes aceitáveis;

$$Xd_{i,j,v} = 0, \forall i \in N_0, \forall j \in S, \forall v \in V \quad (4.12)$$

- d. A restrição (4.13) garante que todo o veículo volte para plataforma sem carga de produtos;

$$Xp_{j,i,v} = 0, \forall i \in N_0, \forall j \in S, \forall v \in V \quad (4.13)$$

8. Domínio das variáveis de decisão:

- a. A restrição (4.14) garante que $Z_{i,j,v}$ seja binária, e as restrições (4.15), (4.16) e (4.17) garantem que as variáveis desse problema sejam não negativas.

$$Z_{i,j,v} \in \{0, 1\}, \forall i, j \in S, \forall v \in V \quad (4.14)$$

$$Xp_{i,j,v} \geq 0, \forall i, j \in S, \forall v \in V \quad (4.15)$$

$$Xd_{i,j,v} \geq 0, \forall i, j \in S, \forall v \in V \quad (4.16)$$

$$A_{i,v} \geq 0, \forall i \in S, \forall v \in V \quad (4.17)$$

5. MÉTODOS DE SOLUÇÃO

5.1. A metodologia heurística da ferramenta desenvolvida

A partir da revisão bibliográfica realizada no Capítulo 2, onde evidenciam-se problemas de roteamento de veículos similares ao tratado neste trabalho, foi possível desenvolver um algoritmo para a sua resolução.

O algoritmo desenvolvido foi baseado em uma heurística clássica de construção por inserção, inspirada no estudo realizado por Lu e Dessouky (2006). Todavia, o algoritmo foi adaptado para o PRVECDJT, visto que na obra deles, a heurística é destinada a um problema de roteamento de veículos com entregas e coletas de segunda classe e pontos de entrega e coleta pareados e janelas de tempo (Parragh et al. 2008b). Assim, durante a concepção do método, não foram efetuadas as restrições de pareamento e priorização dos pontos de entrega e coleta. Ademais, foi imposto que todos os produtos seriam fornecidos apenas pela plataforma e os descartes coletados seriam entregues apenas à mesma, resultando em um problema (1-M-1) com produtos e descartes independentes.

Juntamente com a heurística de construção, foi desenvolvido um método heurístico de aprimoramento por busca local utilizando o operador 2-opt a fim de melhorar as soluções encontradas pelo primeiro método. Esse algoritmo se baseou nas obras de Benoist (2010), de Sassi et al. (2015), e de Tavares et al. (2008).

As próximas seções explicarão com mais detalhes os algoritmos desenvolvidos, bem como suas funções indispensáveis para o desenvolvimento da ferramenta.

5.1.1. Heurística de construção por inserção modificada

O algoritmo de construção por inserção empregado nesse trabalho possui uma peculiaridade em relação aos métodos de inserção clássicos da literatura, o critério de inserção. Pois, o cálculo desse critério de inserção não leva em consideração apenas a distância, ele também privilegia a utilização dos nós de acordo com suas janelas de tempo, bem como a distância percorrida relativa ao número de nós inseridos (Lu e Dessouky, 2006).

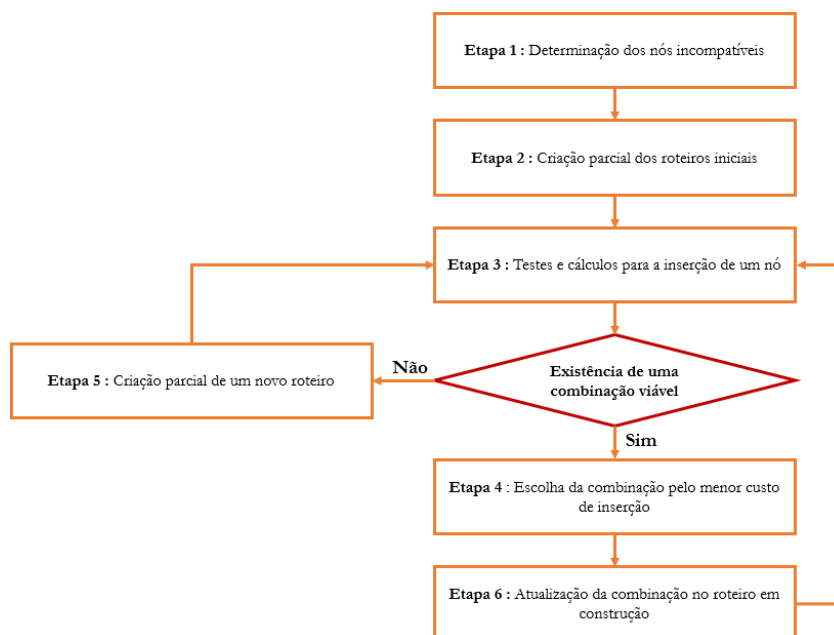
Esses fatores adicionais visam otimizar outros aspectos como a diminuição dos tempos de espera, dos tempos ociosos dos veículos empregados, e da utilização desnecessária de veículos extras sem necessidade.

Em suma, o critério desse método heurístico de construção por inserção se torna mais vantajoso devido a dois quesitos precisamente selecionados (Campbell e Savelsbergh 2004):

1. Inserção do nó que menos afeta as janelas de tempo dos nós já inseridos;
2. Inserção do nó que menos desvia do trajeto de origem.

Para a melhor compreensão do método, as etapas do algoritmo desenvolvido são apresentadas no fluxograma (cf. Figura 17), que será utilizado como base para a explicação dos procedimentos.

Figura 17 – Fluxograma do método heurístico de inserção modificado



Fonte: elaboração do autor

5.1.1.1. Etapa 1: Determinação dos nós incompatíveis

Na primeira etapa, existe uma classificação e determinação dos nós que não podem existir em um mesmo roteiro devido às restrições de capacidade (restrição (4.5)) e janelas de tempo de cada nó (restrição (4.7)). Assim, a primeira avaliação é

a de detecção de nós com demanda nula (cf. Figura 18), onde detectam-se os nós que não deverão ser servidos no roteamento calculado. Em seguida, para mitigar a influência desses nós no algoritmo, é utilizada uma função que anula as distâncias relacionadas a eles, anula os seus tempos de serviço, e relaxa as suas janelas de tempo, designando o menor valor possível ao horário de abertura (0) e um valor extremamente alto ao horário de encerramento (1000) (cf. Figura 19).

Figura 18 – Pseudocódigo da função de seleção de nós sem demanda

```

Função : verifica demanda zero ( $q_i$ )
{
    Criar lista com nós sem demanda = no_sem_demanda

    Verificar a demanda  $q_i$  para cada nó  $i$ 

        Se  $q_i = 0$ , adicionar o nó  $i$  à lista no_sem_demanda

    retornar : a lista de nós sem demanda
}

```

Fonte: elaboração do autor

Figura 19 – Pseudocódigo da função de retirada de influência dos nós sem demanda

```

Função : retira nó sem demanda ( $n, q_i, D_{i,j}, e_i, l_i, Ts_i$ )
{
    Pegar lista de nós sem demandas = verifica_demanda_zero ( $q_i$ )

    Para :  $i \rightarrow$  na lista de nó sem demandas

        Para :  $j \rightarrow 2n + 1$ 
             $D_{i,j} = 0$ 
             $D_{j,i} = 0$ 

         $e_i = 0$ 
         $l_i = 1000$ 
         $Ts_i = 0$ 

    retornar :  $D_{i,j}, e_i, l_i, Ts_i$  alterados
}

```

Fonte: elaboração do autor

Em sequência, há a avaliação de viabilidade de nós (“*avalia_viabilidade*”) (cf. Figura 20), que testa quais nós poderão coexistir em um mesmo roteiro e quais não, segundo as restrições de capacidade de carga (restrição (4.5)) e de janelas de tempo (restrição (4.7)). Ela retorna a matriz binária $check_{i,j}$ de tamanho $2n + 1$ por $2n + 1$, onde “0” indica a impossibilidade, e “1” a possibilidade de coexistência no mesmo roteiro. Por exemplo, $check_{i,j} = 0$ demonstra a impossibilidade de coexistência entre os nós n_i e n_j no mesmo roteiro. Se houver um nó que não possa existir com todos os outros nós, é criado um roteiro para o mesmo, porém este processo é executado em outra etapa discutida mais à frente.

Figura 20 – Pseudocódigo da função de viabilidade dos nós

```

    Função : avalia_viabilidade (n, Di,j, vel, C, ei, li, Tsi, qi)
{
    Para : i → 2n
        Para : j → 2n
            Se (i ≠ j) e (i e j ∈ Ne) ou (i e j ∈ Nc) ou (i ∈ Nc e j ∈ Ne) :
                Se qi + qj > C :
                    i e j não podem coexistir
                Se começando por i, a sequência não respeita a janela de tempo j :
                    i e j não podem coexistir
            retornar : lista de nós que não podem existir na mesma rota, checki,j
}

```

Fonte: elaboração do autor

Em seguida, há a função de classificação de nós (cf. Figura 21), que retorna uma lista de nós, denominada *soma*, organizada em ordem decrescente segundo o somatório de suas não coexistências determinadas pela matriz $check_{i,j}$. Assim, o primeiro nó da lista é sempre aquele que não pode coexistir com o maior número de nós.

Figura 21 – Pseudocódigo da função de classificação de nós

```

    Função : classifica_viabilidade (n, Di,j, vel, C, ei, li, Tsi, qi)
{
    Atribuir a matriz viabilidade de nós à lista check = avalia_viabilidade(n, Di,j, vel, C, ei, li, Tsi, qi)
    Criar lista soma de tamanho 2 x 2n + 1
        Para : i → 2n + 1
            Para : j → 2n + 1
                soma[i][1] = soma[i][1] + check[i][j]
                soma[i][0] = i
            Triagem da lista soma pela segunda componente
    retornar : a lista soma em ordem decrescente em relação a impossibilidade de coexistência relativa a cada nó
}

```

Fonte: elaboração do autor

A partir da classificação, é desenvolvida uma lista ($link_i$) para cada nó i , pela função de impossibilidade de roteiros (cf. Figura 22). Esta função agrupa em uma lista os nós que não podem coexistir em um mesmo roteiro, ou seja, a lista $link_j$ apresenta o conjunto de nós que não podem coexistir em um mesmo roteiro com o nó j . Em sequência, ordena-se a lista de maneira decrescente de acordo com o número de ligações impossibilitadas, privilegiando os nós que possuem o maior número de impossibilidades.

Figura 22 – Pseudocódigo da função de impossibilidade de roteiros relacionada a um nó i

```

Função : impossibilidade_roteiro ( $n, D_{i,j}, vel, C, e_i, l_i, Ts_i, q_i$ )
{
  Atribuir a matriz viabilidade de nós à lista check = avalia_viability( $n, D_{i,j}, vel, C, e_i, l_i, Ts_i, q_i$ )

  Classificar pela lista soma = classifica_viability( $n, D_{i,j}, vel, C, e_i, l_i, Ts_i, q_i$ )

  Criar lista link[ $i$ ]

  Para :  $i \rightarrow 2n + 1$ 
    Para :  $j \rightarrow 2n + 1$ 
      Se ( $soma[i][0] \neq j$ ) e ( $check[soma[i][0]][j] = 0$ ) :
        Anexação do elemento  $j$  na lista de link[ $soma[i][0]$ ]

  retornar : a lista link[ $i$ ]
}

```

Fonte: elaboração do autor

Dessa forma, ao fim da etapa 1, é possível saber quais são os nós incompatíveis entre si, e quais possuem maior grau de incompatibilidade.

5.1.1.2. Etapa 2: Criação parcial dos roteiros iniciais

Na segunda etapa, há a criação parcial dos roteiros iniciais de acordo com o grau de incompatibilidade de cada nó, e a quantidade máxima de roteiros iniciais adicionais.

Este último parâmetro se refere ao número máximo de roteiros adicionais que poderão ser criados antes do início da inserção, e é definido pela variável “num”. Esta variável é calculada pelo quociente inteiro da divisão do número de veículos disponíveis ($|V|$) pelo valor arbitrário do “coef_roteiro_inicial_max”. O valor de $coef_roteiro_inicial_max \in \mathbb{Z}_+$, e é estipulado pelo usuário, e, através da sua variação, visa avaliar como o número de roteiros iniciais afeta a construção da solução e, conseqüentemente, o seu custo. Portanto, se $\frac{|V|}{coef_roteiro_inicial_max} = 0$, não há roteiros iniciais extras, e o algoritmo inicia apenas com um único roteiro.

Dessa forma, foi criado o algoritmo para construção dos roteiros iniciais (cf. Figura 23), que insere os nós da lista *soma* da etapa 1 à lista *roteiros_iniciais*, de acordo com a lista de coexistência $link_i$ desenvolvida na etapa 1, e a quantidade máxima de roteiros iniciais adicionais estabelecida pelo quociente inteiro de

$$\frac{|V|}{coef_roteiro_inicial_max}.$$

O procedimento começa com a inserção do primeiro nó i da lista *soma* à lista *roteiros_iniciais*. Em seguida, esse processo se mantém até que não tenha mais nós não coexistentes, ou até que o tamanho desta lista coincida com o valor superior

limite. Este valor é determinado pelo valor mínimo entre a quantidade de roteiros iniciais limite ($1 + \text{"num"}$) e o número máximo de roteiros possíveis. Esta última limitação é estabelecida pelo número de veículos disponíveis ($|V|$), pois cada veículo realiza apenas um único roteiro.

Figura 23 – Pseudocódigo da função de criação de roteiros iniciais

```

Função : criacao roteiros iniciais (n, Di,j, vel, C, ei, li, Tsi, qi, V, coef roteiro inicial max)
{
  Atribuir a matriz viabilidade de nós à lista check = avalia_viabilidade (n, Di,j, vel, C, ei, li, Tsi, qi)
  Classificar pela lista soma = classifica_viabilidade (n, Di,j, vel, C, ei, li, Tsi, qi)
  Criar lista link[i] = impossibilidade_roteiro (n, Di,j, vel, C, ei, li, Tsi, qi)
  Remover nós da lista soma com demanda zero
  Criar lista roteiros_iniciais pela lista soma
  Enquanto : j < número de nós da lista soma
    Se (i ≤ len(soma) - 1) e (soma[i][0] in link[soma[i][0]]) e (soma[i][0] not in roteiros_iniciais) e (num ≤ rotas extras) :
      Anexar o valor da lista soma[i][0] na lista roteiros_iniciais
      Remover o valor da lista soma[i][0] da lista link[soma[j][0]]
      num = num + 1
      i = i + 1
    Se (i > número de nós da lista soma - 2):
      i = 1
      j = j + 1
      i = i + 1
  retornar : lista roteiros_iniciais e valor de num
}

```

Fonte: elaboração do autor

Dessa maneira, ao final da etapa 2, há o retorno da lista de nós que deverão compor cada roteiro. Assim, cada nó originará um roteiro com o início e o fim na plataforma, que será posteriormente construído a partir da inserção dos nós restantes.

5.1.1.3. Etapa 3: Testes e cálculos para a inserção de um nó

Na terceira etapa, são realizados testes e cálculos para a inserção de um nó i , não selecionado previamente, entre dois nós sucessivos de um determinado roteiro. Desta maneira, esta inserção é testada para todas as combinações viáveis, segundo as restrições de capacidade (restrição (4.5)) e janelas de tempo (restrição (4.7)), em todos os roteiros construídos. E, assim, para cada tentativa é calculado o seu custo de inserção.

Primeiramente, antes de iniciar os testes de inserção, é necessário apresentar algumas variáveis que serão utilizadas no método, e que não foram apresentadas ainda (cf. Tabela 4).

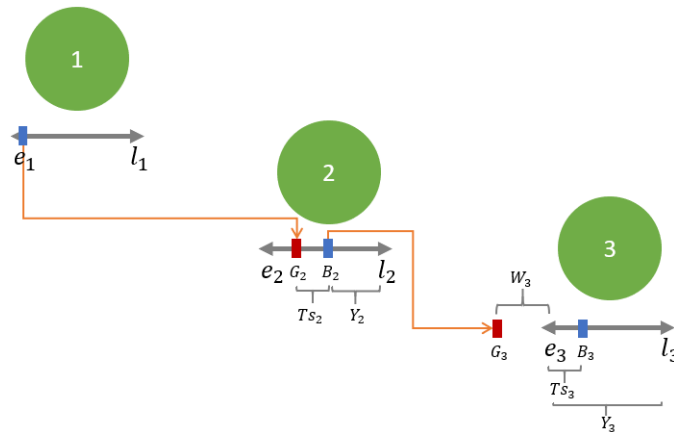
Tabela 4 – Variáveis do método heurístico.

Variáveis	Descrição	Notação
G_i	Tempo nominal mínimo necessário para a chegada de um veículo ao nó i	$\{ G_i \in \mathbb{R}_+^* \mid i \in S \}$
B_i	Tempo nominal mínimo necessário para que um veículo saia do nó i após realizar sua operação	$\{ B_i \in \mathbb{R}_+^* \mid i \in S \}$
W_i	Tempo de espera do veículo para a abertura do nó i	$\{ W_i \in \mathbb{R}_+^* \mid i \in S \}$
Y_i	Tempo disponível para o encerramento do nó i .	$\{ Y_i \in \mathbb{R}_+ \mid i \in S \}$
U_i	Tempo de atraso possível em cada nó i , $(W_i + Y_i)$	$\{ U_i \in \mathbb{R}_+ \mid i \in S \}$

Fonte: elaboração do autor

Para facilitar a compreensão das variáveis mencionadas, a Figura 24 ilustra a utilização delas através da apresentação do funcionamento de um roteiro, onde um veículo percorre três nós. A partir dela, é possível perceber a relação de cada variável com as janelas de tempo, e como elas são representadas fisicamente no problema.

Figura 24 – Exemplo de um roteiro do método heurístico



Fonte: elaboração do autor

Após a compreensão dos valores físicos das variáveis, é possível representá-las em um formato matemático. Assim, suponhamos que existe um roteiro que contenha n nós, (n_1, \dots, n_n) , e um nó n_k , ainda não selecionado, é escolhido para inserção entre os nós n_r e n_s . Assim, a equação (5.1) calcula os novos tempos de chegada ao nó inserido n_k , bem como atualiza os valores de G_i dos seus nós sucessores, n_s a n_n , através do somatório do novo instante de chegada do nó antecessor com o tempo do trajeto deste nó ao seu subsequente.

$$G_{n_k} = \max (G_{n_r}; e_{n_r}) + Ts_{n_r} + \frac{D_{n_r, n_k}}{vel} \quad (5.1)$$

A equação (5.2) calcula o tempo de espera no nó inserido n_k e atualiza os valores de W_i dos seus nós sucessores, n_s a n_n . O valor de W_{n_k} é calculado pela máxima variação não negativa entre o horário de abertura da janela de tempo do nó n_k e o tempo de chegada ao mesmo. Assim, se um determinado veículo chega após o horário de abertura do nó, o tempo de espera é nulo.

$$W_{n_k} = \max (0 ; e_{n_k} - G_{n_k}) \quad (5.2)$$

Por fim, as equações (5.3) e (5.4) calculam o tempo disponível para o encerramento do nó inserido n_k e atualizam os valores de Y_i dos seus nós sucessores, n_s a n_n .

O valor de Y_i de cada nó sucessor é calculado pela diferença entre o horário de encerramento da sua janela de tempo (l_i) e o tempo nominal mínimo necessário para chegada ao mesmo (G_i) se o tempo de atraso possível do seu respectivo nó subsequente for superior a essa diferença (U_{i+1}). Caso contrário, o valor de Y_i é determinado pela subtração entre o tempo de atraso possível do seu respectivo nó subsequente (U_{i+1}), e o tempo calculado do novo trajeto construído com a inserção de n_k adicionado ao seu respectivo tempo de serviço (Δ , na equação (5.4)).

O valor de U_i de cada nó subsequente é relevante na condição, pois é fisicamente impossível um nó antecessor possuir um tempo disponível de encerramento (Y_i) superior ao do nó seguinte se retirarmos os tempos de trajeto e de serviço.

$$Y_{n_k} = \min (l_{n_k} - \max (G_{n_k}, e_{n_k}) ; U_{n_s} - \Delta) \quad (5.3)$$

Tal que,

$$\Delta = \frac{(D[n_r][n_k] + D[n_k][n_s] - D[n_r][n_s])}{v} + Ts[n_k] + W_{n_k} \quad (5.4)$$

Como visto anteriormente, as variáveis apresentadas sofrem constantes atualizações. Dessa forma, todo teste de inserção ou mudança no roteiro acarreta na alteração dos valores de todos os nós subsequentes. Assim, a função “computa_variaveis” (cf. Figura 25) é desenvolvida para automatizar este procedimento, retornando o valor correspondente de cada variável após os cálculos empregados para um determinado roteiro da lista *routes*.

Figura 25 – Pseudocódigo da função de atualização dos valores de G_i , W_i e Y_i

```

    Função : computa_variaveis (routes,  $D_{i,j}$ ,  $G_i$ ,  $W_i$ ,  $Y_i$ ,  $Ts_i$ , vel)
    {
        Para :  $i \rightarrow$  número de roteiros da lista routes

            Se  $i = 1$  :
                 $G_i = e_0 + D_{0,i}/vel$ 

            Senão :
                 $G_i = \max(G_{i-1}, e_{i-1}) + Ts_{i-1} + D_{i-1,i}/vel$ 

                 $W_i = \max(l_i - G_i, 0)$ 

            Para :  $i \rightarrow$  iniciando do número de roteiros da lista routes - 2, decrescendo até - 1 ao passo de - 1

                 $\Delta = \frac{D_{i-1,i} + D_{i,i+1} - D_{i-1,i+1}}{vel} + Ts_i + W_i$ 

                Se  $i =$  número de roteiros da lista routes - 1
                     $Y_i = l_i - \max(G_i, e_i)$ 

                Senão :
                     $Y_i = \min(l_i - \max(G_i, e_i), U_{i+1} - \Delta)$ 

            retornar : a lista [ $G_i, W_i, Y_i$ ]
    }

```

Fonte: elaboração do autor

Após a execução dos cálculos dos tempos de G_i , W_i e Y_i para a inserção de um nó, é realizada a análise da viabilidade. Dessa maneira, para testar a inserção do nó n_k entre os nós n_r e n_s em um determinado roteiro da lista *routes*, é necessário satisfazer as restrições da capacidade de carga e das janelas de tempo.

Portanto, para verificar a restrição da capacidade de carga dos veículos, bem como facilitar a gestão de distribuição da plataforma, a quantidade de carga em cada arco de cada veículo é calculada pela função de cálculo da capacidade, “calcula_capacidade” (cf. Figura 26). Nesta função, utiliza-se a demanda (q_i) de cada nó de um roteiro da lista *routes*, juntamente com o seu respectivo tipo de nó (entrega ou coleta) para determinar a carga de cada trecho do roteiro, pois nos nós de coleta ($n_i > n$) há uma adição de carga e nos nós de entrega ($n_i \leq n$) há uma redução da mesma, lembrando que o veículo parte da plataforma com todos os produtos que serão entregues e retorna a mesma apenas com os descartes aceitáveis coletados.

Assim, a função determina a quantidade de carga de cada trajeto, através da variável $cap[n_k]$, que indica o respectivo valor entre o nó n_k e o seu subsequente. Na Figura 27, essa variável é ilustrada através de um roteiro com k nós, onde $cap[0]$ indica a carga entre a plataforma e o seu nó subsequente, $cap[1]$ indica a carga entre o nó n_1 e o seu nó subsequente, e assim por diante.

Figura 26 – Pseudocódigo da função de cálculo da capacidade de carga dos arcos

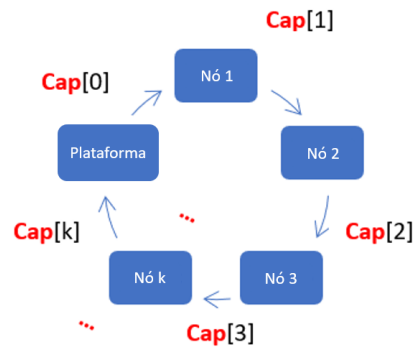
```

Função : calcula_capacidade (n, routes, qi)
{
  Criar lista cap[]
  Para : i → número de roteiros da lista routes
    Para : j → i + 1
      Se j > n :
        cap[i] = cap[i] + qj
    Para : k → (i + 1, número de roteiros da lista routes)
      Se k ≤ n :
        cap[i] = cap[i] + qk
  retornar : lista cap[]
}

```

Fonte: elaboração do autor

Figura 27 – Ilustração da função $calcula_capacidade(n, routes, q_i)$



Fonte: elaboração do autor

Em seguida, com a quantidade de carga calculada pela função “calcula_capacidade”, é possível executar a função “verifica_capacidade” (cf. Figura 28), que verifica se a restrição da capacidade de carga dos veículos é respeitada para todos os arcos do roteiro utilizando a variável m para iterar entre eles, retornando “1” se positivo, e “0” caso contrário.

Figura 28 – Pseudocódigo da função de verificação da capacidade dos arcos

```

Função : verifica_capacidade (cap)
{
  Para :  $m \rightarrow$  número de valores da lista cap

    Se  $cap[m] > C$  :
      viabilidade = 0

  retornar : viabilidade "0" ou "1"
}

```

Fonte: elaboração do autor

Em relação à restrição das janelas de tempo, foi desenvolvida a função “verifica_janelas_de_tempo” (cf. Figura 29), que avalia se é possível a inserção do nó em uma determinada posição de um roteiro da lista *routes* sem infringir a restrição de janelas de tempo. Assim, em um determinado nó n_k , o algoritmo assegura que o somatório do instante de chegada no nó ($G_{n_k} + W_{n_k}$) com o seu respectivo tempo de serviço empregado (Ts_{n_k}), é inferior ou igual ao seu horário de encerramento. Dessa forma, se a inserção do nó é avaliada como possível, o retorno é “1”, caso contrário “0”.

Figura 29 – Pseudocódigo da função de verificação de inserção pela restrição de janelas de tempo

```

Função : verifica_janelas_de_tempo (routes)
{
  Se a roteiro é vazio, :
    retornar : 1, inserção possível

  Definição dos instantes de chegada para um roteiro de routes, onde  $G_0 = e_0$ 

  Para :  $i \rightarrow$  número de roteiros da lista routes - 2

    Cálculo de  $G_i, W_i$ 

  Para :  $i \rightarrow$  número de roteiros da lista routes - 1

    Se  $G_i + W_i + Ts_i > l_i$ :
      retornar : 0, inserção impossível

  Senão :
    retornar : 1, inserção possível
}

```

Fonte: elaboração do autor

Por fim, se a viabilidade da inserção de n_k é positiva, o cálculo do custo de inserção é efetuado. Esse cálculo é o grande diferencial da metodologia, pois ele é desenvolvido para aumentar a produtividade dos veículos empregados nos roteiros construídos através da otimização das distâncias percorridas e da melhor utilização das janelas de tempo. O cálculo é composto por três elementos:

$$C_G = C_1 + C_2 + C_3 \quad (5.5)$$

1. Cálculo de C_1

O custo C_1 corresponde ao somatório da redução dos tempos disponíveis de atraso dos nós antecessores ao nó de inserção, ou seja, a variação total que a inserção causa a esses nós do roteiro. Ele é representado pela equação (5.6), onde “beta” é o somatório do tempo de atraso possível dos nós posteriores ao nó do cálculo, e n_r é o nó antecessor ao de inserção.

$$\sum_{n_i=n_1}^{n_r} \min(Y_{n_i} - \text{beta}; W_{n_{i+1}}) \quad (5.6)$$

A diminuição de C_1 , reduziria a “perda” do intervalo de tempo disponível dos nós antecessores, devido a um menor tempo disponível de encerramento do nó inserido, e assim acarretaria maiores chances de compatibilidade para a inserção de outros nós.

O algoritmo do cálculo do C_1 é apresentado na Figura 30, no qual é possível averiguar o procedimento de cálculo ao se inserir um nó n_k entre os nós n_r e n_s , sendo n_l o nó de iteração e p a posição do nó n_r em um roteiro da lista *routes*.

Figura 30 – Algoritmo do cálculo do custo C_1 da heurística de inserção modificada

```

Função : calc1(p,nk,routes,Wi,Yi)
{
  C1 = 0

  beta = Wnk + Ynk ; l = p; nl = nr

  Enquanto beta < Ynl e l > 1 :

    Se l = p:
      C1 = Ynl - beta
      beta = Wnl + beta
      p = p - 1
    Senão :
      nl = Nó l de um roteiro de routes
      nl+1 = Nó l + 1 de um roteiro de routes

      Se Wnl+1 > 0 :
        C1 = C1 + min ( Ynl - beta, Wnl+1 )

      beta = Wnl + beta
      l = l - 1
  retornar : C1
}

```

Fonte: elaboração do autor

2. Cálculo de C_2

O custo C_2 corresponde à redução do tempo disponível de atraso do próprio nó, n_k , após a sua inserção. Assim, ele é o cálculo complementar ao de C_1 , pois, enquanto C_1 calcula o efeito da inserção aos nós antecessores, C_2 avalia o efeito do nó sucessor (n_s) ao nó inserido (n_k).

O algoritmo é representado pela Figura 31, no qual apresenta a diferença entre o valor de Y_{n_k} sem o efeito do nó subsequente, ou seja, sem levar em consideração o termo $U_{n_s} - \Delta$ da equação (5.3) do cálculo de Y_i , e o valor ordinário de Y_{n_k} , calculando exatamente como na equação (5.3).

Assim, C_2 é sempre não negativo, e calcula a possível “perda” do tempo disponível para o encerramento do nó n_k , devido ao nó sucessor n_s , em outras palavras, ele visa a inserção de um nó que sofra o menor efeito possível do nó subsequente.

Figura 31 – Algoritmo do cálculo de custo C_2 da heurística de inserção modificada

```

Função : calc2( $e_i, l_i, n_k, Y_i, G_i$ )
{
     $C_2 = 0$ 
     $C_2 = l_{n_k} - \max(G_{n_k}, e_{n_k}) - Y_{n_k}$ 
    retornar :  $C_2$ 
}

```

Fonte: elaboração do autor

3. Cálculo de C_3

O custo C_3 corresponde ao efeito da inserção do nó n_k em um determinado arco (n_r, n_s) no fator distância, e, conseqüentemente, no fator tempo, devido à padronização da velocidade média.

Assim, ele visa a inserção de um nó que gerará o menor desvio da distância do arco (n_r, n_s) . Este cálculo é explicitado na Figura 32, a qual apresenta a criação dos dois novos arcos (n_r, n_k) e (n_k, n_s) e a retirada do arco antigo (n_r, n_s) .

Figura 32 – Algoritmo do cálculo do custo C_3 da heurística de inserção modificada

```

Função : calc3( $n_k, D_{i,j}, vel$ )
{
     $C_3 = 0$ 
     $C_3 = \frac{(D_{n_r, n_k} + D_{n_k, n_s} - D_{n_r, n_s})}{vel}$ 
    retornar :  $C_3$ 
}

```

Fonte: elaboração do autor

4. Cálculo de C_G

O cálculo do custo de inserção geral, equação (5.5), representa o somatório dos três custos explícitos anteriormente. O cálculo é indicado na Figura 33, onde integra os cálculos dos três custos em uma única função, retornando o custo geral

de inserção do nó n_k após o nó da posição p de um determinado roteiro da lista $routes$.

Figura 33 – Pseudocódigo da função do custo geral de inserção

```

Função : custoinsercao(p, nk, routes, ei, li, Gi, Wi, Yi, Di,j, vel)
{
  Cálculo de  $G_{n_k}$ ,  $W_{n_k}$  e  $Y_{n_k}$ 

   $C1 = calc1(p, n_k, routes, W_i, Y_i)$ 

   $C2 = calc2(e_i, l_i, n_k, Y_i, G_i)$ 

   $C3 = calc3(n_k, D_{i,j}, vel)$ 

   $CG = C_1 + C_2 + C_3$ 

  retornar : CG de inserção de  $n_k$  em um determinado roteiro de routes
}

```

Fonte: elaboração do autor

Desse modo, ao final da etapa 3, são retornados os custos de inserção de um determinado nó n_k para diferentes combinações viáveis dos roteiros construídos.

5.1.1.4. Etapa 4: Escolha da combinação pelo menor custo de inserção

Na quarta etapa, é realizada uma comparação de custos das inserções possíveis retornadas da etapa 3. Assim, é escolhida a combinação $(n_k, routes, n_r$ e $n_s)$ que possui o menor custo geral de inserção, onde n_k é um nó que ainda não foi selecionado, $routes$ é um roteiro já construído, e n_r e n_s são dois nós sucessivos.

5.1.1.5. Etapa 5: Criação parcial de um novo roteiro

A quinta etapa é executada se a inserção de um nó n_k não for possível em nenhum roteiro já construído, ou seja, se não existir o retorno de uma combinação viável pela etapa 3.

Inicialmente, antes de prosseguir com a criação parcial de um novo roteiro, o número de roteiros criados é avaliado, pois se este número ultrapassar o número de veículos disponíveis, o problema é considerado infactível. Assim, se o método heurístico for considerado inviável, ele é parado e não retorna solução, caso contrário a quinta etapa continua.

Com prosseguimento da etapa 5, é realizada a criação parcial de um novo roteiro, ao qual é constituído inicialmente por um nó, ainda não inserido, com a menor distância em relação à plataforma. Como no problema todos os veículos devem partir e retornar a plataforma, e cada veículo é responsável por apenas um roteiro, o primeiro e o último nó desse novo roteiro são designados à plataforma.

Visto isso, a etapa 3 é executada novamente, onde os testes de inserção são realizados para os nós restantes.

5.1.1.6. Etapa 6: Atualização da combinação no roteiro em construção

A sexta etapa realiza a inserção do nó n_k segundo a melhor combinação $(n_k, routes, n_r$ e $n_s)$ escolhida na etapa 4. Desta forma, os valores das variáveis G_i, W_i e Y_i são atualizados para todos os nós i de um determinado roteiro da lista $routes$, e o nó n_k ficará impossibilitado de ser inserido novamente.

5.1.1.7. Etapa final: Iteração e unificação do algoritmo

Durante esta última etapa é realizada a repetição das etapas 3 a 6 até que todos os nós com demanda sejam selecionados, e, em seguida, é efetuado o cálculo do custo da solução, ou seja, a soma das distâncias totais percorridas em todos os roteiros.

Dessa forma, para calcular esse custo foi desenvolvida a função “calcula_custo_total” (cf. Figura 34), que soma as distâncias de cada arco de cada roteiro construído, obtendo a distância total da solução, onde “custo_total” indica a distância total do roteamento de veículos.

Figura 34 – Pseudocódigo da função do cálculo da distância total percorrida por todos os roteiros

```

Função : calcula custo total (routes)
{
  Para :  $i \rightarrow$  número de roteiros da lista routes
    Para :  $j \rightarrow$  número de roteiros da lista routes - 1
       $custo\_total = custo\_total + Distância\ do\ arco\ \{j \rightarrow j + 1\}$  do roteiro  $i$ 

  retornar : distância total do roteamento de veículos =  $custo\_total$ 
}

```

Fonte: elaboração do autor

Para unificar a heurística de construção por inserção em uma única função, foi desenvolvida a função global “*heuristicainsercao*” (cf. Figura 35), que utiliza o fluxograma (cf. Figura 17) como base para estruturar o algoritmo. Essa função foi desenvolvida para retornar valores essenciais para a utilização dos métodos de aprimoramento por busca local, apresentando os roteiros definidos e as listas com os valores de G_i , W_i e Y_i para cada nó i .

Figura 35 – Pseudocódigo da função da metodologia heurística de inserção modificada

```

Função : heuristicainsercao ( $n, D_{i,j}, vel, C, e_i, l_i, G_i, W_i, Y_i, Ts_i, q_i, V, coef\_roteiro\ inicial\ max$ )
{
    Faz apelo a todas as etapas citadas segundo o fluxograma do algoritmo

    retornar : lista de roteiros,  $G_i$ ,  $W_i$  e  $Y_i$ 
}

```

Fonte: elaboração do autor

5.1.2. Método de aprimoramento pela busca local 2-opt

Para complementar a metodologia heurística de construção por inserção, foi desenvolvido um método de aprimoramento por busca local 2-opt com variação inter-roteiros e intra-roteiro como especificado na Seção 2.4.2.2.1. Essas metodologias visam encontrar melhores combinações para a solução inicial construída pela metodologia da Seção 5.1.1., através do desenvolvimento de novos roteiros pela permutação dos arcos.

Como o método de inserção leva em consideração as janelas de tempo, ou seja, ele visa inserir os nós em uma sequência que otimize a viabilidade dessa restrição, adicionando nós com janelas de tempo limitadas na frente e nós com janelas de tempo mais relaxadas atrás. É possível inferir que a ordem dos roteiros após a metodologia de inserção, possuirá mais chances de obter sucesso de viabilidade segundo as janelas de tempo e, conseqüentemente, um maior êxito nas permutações entre roteiros. Assim, foi decidido implementar primeiro a busca 2-opt inter-roteiros e depois a busca 2-opt intra-roteiro.

Inicialmente, para a comparação dos custos entre os roteiros desenvolvidos pelo método de aprimoramento, foi desenvolvida a função “*calcula_custo_do_roteiro*” (cf. Figura 36), que soma as distâncias entre os nós de um determinado roteiro *route*, e retorna o custo total dele. Assim, ela será útil para selecionar o melhor roteiro viável desenvolvido após a realização das trocas.

Figura 36 – Pseudocódigo da função da distância dos arcos de um roteiro

```

Função : calcula custo do roteiro (route)
{
  Para :  $i \rightarrow \text{número de roteiros da lista routes} - 1$ 
    custo_roteiro = custo_roteiro + Distância do arco  $\{i \rightarrow i + 1\}$ 

  retornar : Distância total de um roteiro = custo_roteiro
}

```

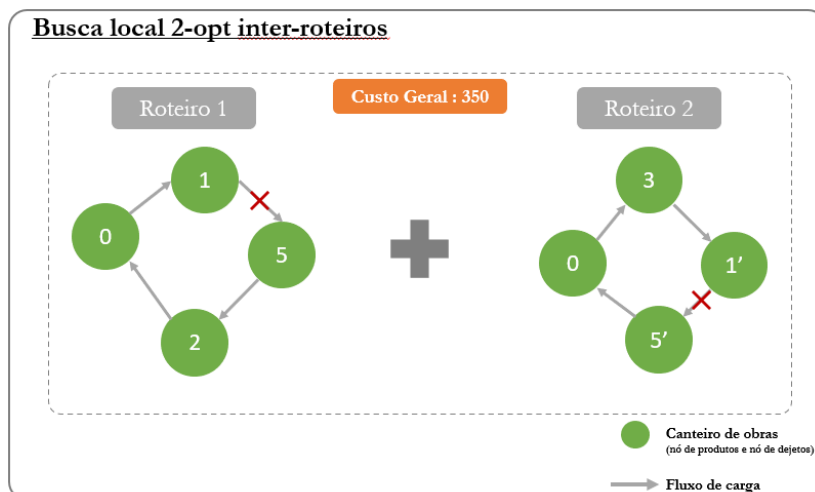
Fonte: elaboração do autor

5.1.2.1. Busca inter-roteiros

A busca local 2-opt inter-roteiros realiza a remoção de dois arcos de dois roteiros distintos, em seguida ela executa a troca dos mesmos para a formação de dois novos roteiros. Se a troca é considerada eficaz, ou seja, se otimiza o custo geral da solução e mantém a sua viabilidade segundo as restrições de capacidade de carga (restrição (4.5)) e janelas de tempo (restrição (4.7)), esses dois roteiros são mantidos.

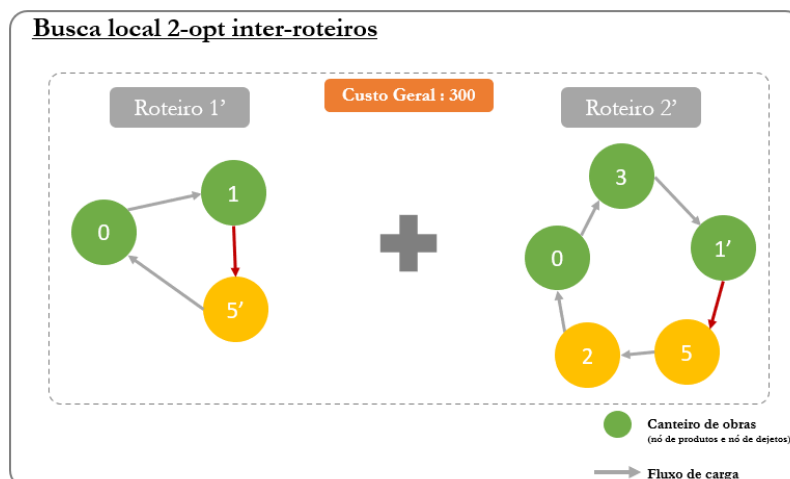
As Figuras 37 e 38 ilustram o procedimento com os roteiros 1 e 2, no qual os arcos do $(n_1 \text{ e } n_5)$ e $(n_{1'} \text{ e } n_{5'})$ são removidos e trocados para a construção de dois novos roteiros, onde o final do roteiro 1 se torna o final do roteiro 2, e vice-versa.

Figura 37 – Remoção dos arcos dos roteiros 1 e 2 pelo operador 2-opt inter-roteiros



Fonte: elaboração do autor

Figura 38 – Criação de dois novos roteiros 1' e 2' pelo operador 2-opt inter-roteiros



Fonte: elaboração do autor

Como resultado, é observada uma redução do custo geral, de 350 para 300. Dessa forma, se essa troca for factível, os roteiros criados passarão a ser os novos roteiros e os roteiros de origem serão removidos.

A fim de se obter a melhor solução possível através da troca, o operador realiza todas as combinações possíveis entre todos os roteiros. Preservando apenas as combinações que otimizem o custo geral e sejam executáveis pelas restrições impostas do problema. Todavia, a busca inter-roteiros envolve dois roteiros distintos, tornando esse procedimento complexo.

Pois, supondo que haja três roteiros e que as permutações sigam uma ordem de análise crescente, ou seja, analisa-se os roteiros 1 e 2, em seguida os roteiros 1 e 3, e por fim os roteiros 2 e 3. Se for identificada uma melhor combinação entre as permutações dos roteiros 2 e 3, e essa troca for efetuada, os roteiros 2 e 3 não serão mais os mesmos que foram analisados com o roteiro 1, necessitando uma nova análise. Assim, para reiteração contínua dos roteiros, é intrínseco um critério de parada, nesse problema foi adotado o critério de não melhoria dos custos, em outras palavras, se após uma iteração completa dos roteiros não houver otimização dos custos, a iteração é interrompida.

Portanto, através das especificações da busca acima, a função “inter_rotatorios” (cf. Figura 39) é executada utilizando como entrada a solução obtida pelo método heurístico por inserção. Cada iteração se inicia com dois roteiros selecionados, “roteiro x” e “roteiro y”, onde os seus respectivos custos iniciais são calculados. Em seguida, ocorre a remoção e a execução da troca de dois arcos (i, j) e (r, s) , um de

cada roteiro, para originar dois novos roteiros, e, assim, efetuar o cálculo de custo da nova solução. Após a troca, os custos da solução, inicial e final, são comparados pela função “calcula_custo_do_roteiro”, conservando a solução factível de menor custo segundo as funções de avaliação da viabilidade, “calcula_capacidade”, “verifica_capacidade” e “verifica_janelas_de_tempo”. Esse procedimento é efetuado em todas as combinações de dois arcos possíveis entre roteiros, conservando a solução viável com o menor custo calculado. Por fim, se houver um aprimoramento dos custos da solução, os roteiros são permutados novamente, caso contrário finaliza-se a busca local 2-opt inter-roteiros.

Figura 39 – Pseudocódigo do procedimento de busca local do operador 2-opt inter-roteiros

```

Função : inter roteiros (n, Di,j, vel, C, ei, li, Gi, Wi, Yi, Tsi, qi, V, coef_roteiro_inicial_max)
{
  Solução da heurística de inserção = heuristicainsercao(n, Di,j, vel, C, ei, li, Gi, Wi, Yi, Tsi, qi, V, coef_roteiro_inicial_max)
  Custo = Solução da heurística de inserção
  Continuar permutando enquanto houver aprimoramento do Custo

  Para cada 2 roteiros da solução :

    Custo = Custo do roteiro x + Custo do roteiro y

    Para cada par de arcos (i, j), (r, s), pertencendo, respectivamente ao roteiro x, e ao roteiro y:

      roteiro x' = Início do roteiro x até o arco (i, j), e a partir do arco (r, s) até o final do roteiro y
      roteiro y' = Início do roteiro y até o arco (r, s), e a partir do arco (i, j) até o final do roteiro x

      Se custo do roteiro x' + roteiro y' < Custo :

        Se solução for viável pela capacidade de carga e janelas de tempo :

          Permanecer com a solução permutada
          Custo = Custo da solução permutada

  retornar : lista com a solução original e lista com a solução inter – roteiros
}

```

Fonte: elaboração do autor

5.1.2.2. Busca intra-roteiro

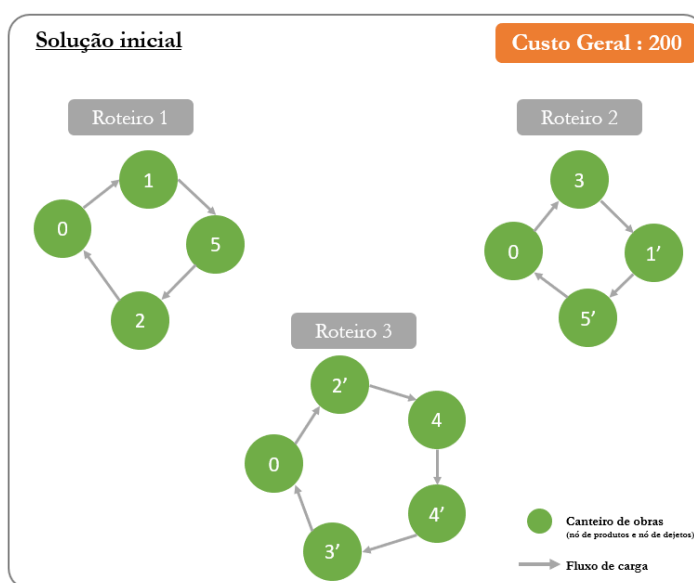
Após a aplicação da busca dos roteiros pela operação inter-roteiros, o método de aprimoramento por busca local 2-opt intra-roteiro é executado. Similarmente à primeira busca, a operação intra-roteiro também remove dois arcos e realiza a troca entre eles. Contudo, desta vez, os arcos não concernem à dois roteiros distintos, mas à apenas um único roteiro.

Como exemplo, a Figura 40 apresenta uma solução obtida pela busca inter-roteiros, onde há três roteiros (1, 2 e 3), e o seu custo geral. Em seguida, a busca

local 2-opt intra-roteiro é aplicada ao roteiro 3 (cf. Figura 41), no qual constata-se a troca entre os arcos $(n_2 \text{ e } n_4)$ e $(n_4 \text{ e } n_{3'})$, gerando o roteiro 3'. Como resultado, houve a inversão das posições do nó n_4 com o $n_{4'}$, acarretando a redução do custo geral. Se a viabilidade do novo roteiro 3' for comprovada, este substituirá o roteiro 3.

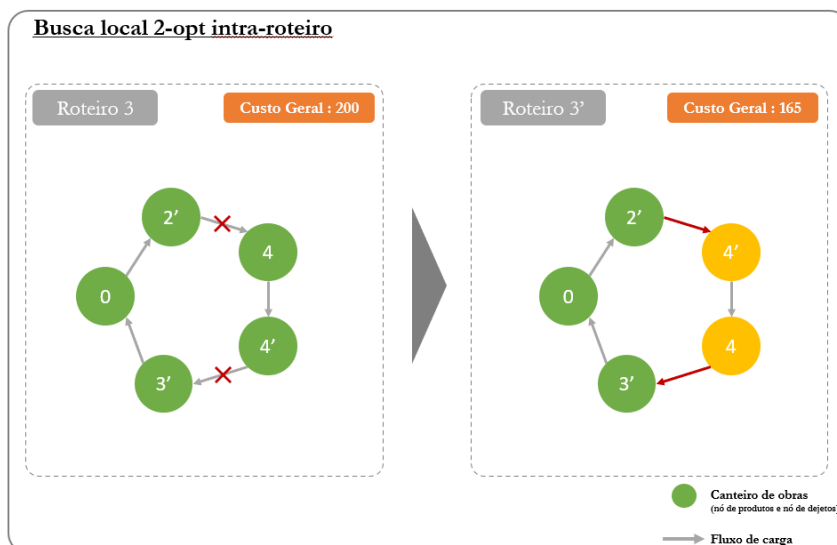
Identicamente ao procedimento do operador 2-opt inter-roteiros, a operação intra-roteiro realiza todos os movimentos de arcos possíveis em cada um dos roteiros. Assim, a melhor sequência de cada roteiro será escolhida.

Figura 40 – Solução obtida pela busca inter-roteiros



Fonte: elaboração do autor

Figura 41 – Exemplo da busca local 2-opt intra-roteiro do roteiro 3



Fonte: elaboração do autor

Dessa maneira, foi criada a função “intra_roteiro” (cf. Figura 42) para simular a busca intra-roteiro especificada acima. Ela realiza a troca de dois arcos não adjacentes em um mesmo roteiro, testando todas as combinações possíveis dentro dele, e, em seguida, conserva a solução factível de menor custo. Esse algoritmo é repetido para todas as combinações em todos os roteiros construídos.

O algoritmo utiliza a saída da função “inter_roteiros” como entrada, utiliza as funções “calcula_capacidade”, “verifica_capacidade” e “verifica_janelas_de_tempo” para avaliar a viabilidade, e utiliza a função “calcula_custo_do_roteiro” para comparação de roteiros.

Figura 42 – Pseudocódigo do procedimento de busca local do operador 2-opt intra-roteiro

```

Função : intra_roteiro (n, Di,j, vel, C, ei, li, Gi, Wi, Yi, Tsi, qi, V, coef_roteiro_inicial_max)
{
  Solução do operador inter-roteiros = inter_roteiros (n, Di,j, vel, C, ei, li, Gi, Wi, Yi, Tsi, qi, V, coef_roteiro_inicial_max)

  Custo = Custo da solução do operador inter-roteiros

  Para cada roteiro construído

    Para cada par de arcos dentro de um determinado roteiro construído

      Se custo da troca < Custo :

        Se solução for viável pela capacidade e janelas de tempo :

          Permanecer com a solução permutada
          Custo = Custo da solução permutada

  retornar : lista com a solução original, lista com a solução inter – roteiros e lista com a solução otimizada final
}

```

Fonte: elaboração do autor

6. RESULTADOS E ANÁLISES

6.1. Características dos métodos de solução empregados

6.1.1. Algoritmo exato

No método de solução exato do modelo matemático da Seção 4.1., foi utilizado o *software* AMPL (*A Mathematical Programming Language*) com o *solver* CPLEX 12.7.1.0. Este foi configurado com um critério de tempo de parada de 1800 segundos (30 minutos), assim, se o tempo de compilação do método ultrapassa esse valor, o algoritmo exato para a compilação e apresenta a melhor solução factível inteira calculada (Fourer et al., 2013). Esse critério de parada foi implementado, pois como a ferramenta é monoperiódica, a planificação é realizada diariamente, dessa forma, é necessário impor alguns limites de tempo para a compilação do método.

Intimamente ligado com o critério de parada, há o *gap* de otimalidade, este é calculado quando o método exato atinge o critério de parada, mas não alcança a solução ótima. Assim, ele é determinado pela diferença absoluta entre a melhor solução não inteira e a melhor solução inteira divididos por um mais o valor absoluto do melhor nó (equação 6.1) (Fourer et al., 2013). Portanto, se há uma solução factível inteira exata com um *gap* de 0.2, foi comprovado que essa solução possui 20% de otimalidade. O melhor *gap* é o 0, indicando uma solução ótima.

$$gap = \frac{|melhor\ solução\ não\ inteira - melhor\ solução\ inteira|}{(1 + |melhor\ solução\ inteira|)} \quad (6.1)$$

Por ser um método exato, o programa realiza uma enumeração implícita de subconjuntos para alcançar a solução ótima. Como foi utilizado o *solver* CPLEX 12.7.1.0 em um problema linear inteiro misto, utiliza-se por *default* a combinação dos algoritmos *Branch-and-Bound* e Simplex, que possui, em seu pior caso, uma complexidade exponencial, $O(2^n)$, onde “n” é o número de variáveis empregadas no problema (Klee e Minty, 1972) (Dasgupta et al., 2006).

6.1.2. Algoritmo heurístico

A linguagem utilizada para a compilação da solução heurística desenvolvida foi a Python (v. 3.6), com o VisualStudioCode como ambiente de desenvolvimento.

Após o desenvolvimento do algoritmo heurístico, uma análise de complexidade foi realizada, uma vez que a complexidade da metodologia foi um dos principais fatores para o desenvolvimento da metodologia heurística. No pior caso, o algoritmo permanece sempre com uma complexidade polinomial, e vale exatamente $O((2n + 1)^4) = O(n^4)$.

6.2. Circunstâncias dos testes

Ambos os métodos de solução foram executados utilizando um computador pessoal com processador Intel® Core™ i5- 7300 HQ CPU e 2.5 GHZ, RAM: 8,00Go.

Como a plataforma modular é, eventualmente, instalada em diversos lugares ao longo do tempo, resultando em dados distintos de caso a caso, o objetivo da análise é testar a eficácia da ferramenta para diferentes conjuntos de dados e cenários, que se baseiem em casos reais.

Assim, os dados para a análise da ferramenta foram estudados através do *Benchmarking* de projetos implicados no assunto de logística reversa no setor da construção civil como o PILC (Projet Innovation Logistique Chantiers) (Lerouvillois, P et al., 2019).

Vale ressaltar que o intuito desse trabalho é a criação de uma ferramenta de apoio à decisão para uma plataforma modular, à qual deve ser utilizada para diferentes tipos de cenários e instâncias. Portanto, o objetivo não foi se basear em um único caso específico, mas observar como a ferramenta se comporta para diferentes situações factíveis.

Dessa maneira, a fim de estudar melhor a ferramenta desenvolvida, realizou-se três grandes análises nesse trabalho que serão detalhadas nas próximas seções:

6.2.1. Análise do método heurístico (sem e com busca local)

Busca-se compreender melhor o comportamento do método heurístico desenvolvido, e quais são os impactos da busca local para a solução. Dessa forma,

utiliza-se diversas variações da quantidade de canteiros de obras (9 variações) e da distância máxima (8 variações) para testar a capacidade da ferramenta. Para cada combinação de cenário (72 combinações de cenários) são geradas 100 instâncias diferentes, a fim de compreender o comportamento médio do método. Assim, um total de 7200 execuções do algoritmo são realizadas, analisando-se o custo da solução antes e depois da busca local.

6.2.2. Análise de comparação entre os métodos (heurístico vs. exato)

Visa-se comparar o método heurístico desenvolvido com o método exato, buscando-se compreender como a diferença dos custos das duas soluções se comporta, e se por meio das variações do parâmetro “coef_roteiro_inicial_max” do método heurístico, é possível reduzir a diferença. Dessa maneira, utiliza-se variações da quantidade de canteiros de obras (8 variações) e da distância máxima (4 variações), onde em cada combinação de cenário (32 combinações de cenários) são geradas 10 instâncias diferentes. Com o intuito de analisar o aprimoramento dessa diferença de custos através do parâmetro “coef_roteiro_inicial_max”, cada instância do método heurístico (240 instâncias) foi replicada 5 vezes com diferentes valores de “coef_roteiro_inicial_max”. Assim, um total de 1920 execuções (1600 do método heurístico e 320 do método exato) são realizadas, analisando-se a diferença dos custos entre a solução heurística e exata, bem como os seus respectivos e tempos de execuções. Especialmente para o método exato, foram calculados o *gap* de otimalidade e a proporção de soluções factíveis da combinação do cenário.

6.2.3. Análise inicial de aperfeiçoamento da ferramenta através da variação dos pesos dos custos de inserção

Procura-se aprimorar a diferença dos custos da solução heurística e exata através da variação dos pesos dos custos de inserção. Assim, utiliza-se as mesmas instâncias com as variações do parâmetro “coef_roteiro_inicial_max” da segunda análise para replicar três vezes o método heurístico em cada combinação de cenário (1600 combinações) segundo a variação dos pesos. Dessa forma, um total de 4800 execuções do algoritmo são realizadas, analisando-se a diferença entre os custos das novas soluções heurísticas e das exatas já calculadas.

6.3. Detalhamento da análise do método heurístico

6.3.1. Instâncias e cenários

A primeira análise mensura como o método de aprimoramento por busca local, afeta o resultado da solução criada pela heurística de construção por inserção. Assim, o objetivo é comparar os custos da solução do método de inserção modificado antes e depois da realização da busca local.

Portanto, baseando-se nas características coletadas dos *Benchmarkings* realizados, foram decididos os valores dos intervalos dos conjuntos e parâmetros que serão utilizados na análise. Dessa forma, com a ajuda da biblioteca “random” do Python, foram gerados aleatoriamente os valores dentro desses intervalos para as distâncias entre os nós, as janelas de tempo, os tempos de serviço em cada canteiro de obras, e as demandas e ofertas. Eles seguem as condições expressas pelas características a seguir.

Primeiramente, foram utilizadas nove variações da quantidade de canteiros:

- | | | | |
|----|------------------------|----|------------------------|
| 1. | 3 canteiros de obras; | 6. | 20 canteiros de obras; |
| 2. | 5 canteiros de obras; | 7. | 30 canteiros de obras; |
| 3. | 8 canteiros de obras; | 8. | 50 canteiros de obras; |
| 4. | 10 canteiros de obras; | 9. | 80 canteiros de obras; |
| 5. | 15 canteiros de obras; | | |

Para cada variação anterior, foram estimados oito cenários em cada qual a distância entre os nós varia de 1 km à distância máxima, como visto a seguir:

1. Distância máxima de 30 km entre os nós;
2. Distância máxima de 50 km entre os nós;
3. Distância máxima de 80 km entre os nós;
4. Distância máxima de 100 km entre os nós;
5. Distância máxima de 120 km entre os nós;
6. Distância máxima de 150 km entre os nós;
7. Distância máxima de 180 km entre os nós;
8. Distância máxima de 200 km entre os nós.

Cada combinação de cenário foi executada 100 vezes pelo algoritmo para diferentes instâncias, assim utiliza-se dados distintos para cada execução através da aleatoriedade dos dados descrita anteriormente, mas que respeitem os requisitos de cada cenário de estudo (número de canteiros e distância máxima). Portanto, foram executadas 7200 instâncias (9 quantidades de canteiros de obras, 8 tipos de distância máxima e 100 execuções do algoritmo para cada combinação de cenário).

Para manter a viabilidade das soluções e a realidade das situações, alguns parâmetros aumentam à medida que as instâncias se tornam maiores, como o caso da velocidade média (*vel*) que tende ser maior com o aumento da distância entre os nós, pois percebe-se que em distâncias maiores os veículos utilizam vias com um limite de velocidade máxima superior; da capacidade de carga (*C*) que simula a utilização de veículos maiores com o aumento do número de canteiros servidos e a demanda de cada um; e do número de veículos disponíveis ($|V|$), devido à necessidade de um aumento da frota em função do número de canteiros de obras servidos. Apenas o valor arbitrário de “coef_roteiro_inicial_max”, que auxilia na definição da quantidade máxima de roteiros iniciais adicionais (Seção 5.1.1.2), foi mantido constante durante todas as execuções.

Assim temos as seguintes funções para os parâmetros:

$$vel = \text{distância máxima} \quad (6.2)$$

$$C = \max \left(\frac{\sum_{i=1}^{2n} q_i}{\frac{3}{4} * n}, \max(q_i) \right) \quad (6.3)$$

$$|V| = n \quad (6.4)$$

$$\text{coef_roteiro_inicial_max} = 5 \quad (6.5)$$

6.3.2. Resultados

Os resultados obtidos são apresentados na Tabela 5, expressando a média dos custos da solução das 100 instâncias de cada combinação de cenário antes da execução da busca local 2-opt inter-roteiros e intra-roteiro (1), essa mesma média

após a execução da mesma (2), o aprimoramento do resultado dos valores anteriores pelo cálculo do desvio entre eles ($\frac{(1-2)}{1}$), e a média do tempo de execução total do método das 100 instâncias geradas para cada combinação de cenário.

Tabela 5 – Avaliação da abordagem heurística de inserção modificada e da busca local 2-opt.

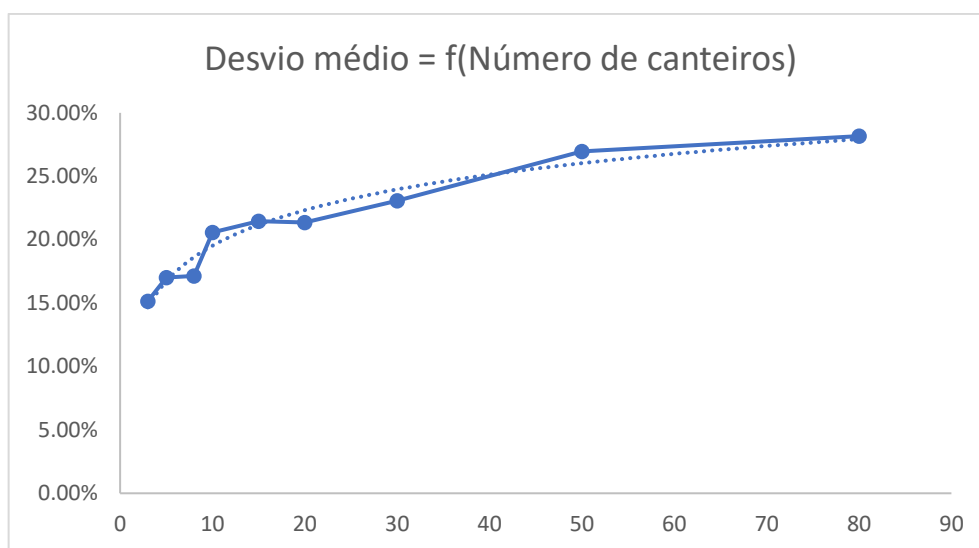
Canteiros [n]	Distância máxima [km]	Custo Inserção [km] (1)	Custo 2-opt [km] (2)	Desvio [%]	Tempo de execução [s]
3	30	113.99	100.43	11.89	0.01
3	50	184.07	159.69	13.24	0.01
3	80	276.09	234.77	14.97	0.01
3	100	362.75	313.58	13.55	0.02
3	120	439.66	364.51	17.09	0.01
3	150	534.80	444.50	16.88	0.01
3	180	652.05	557.07	14.57	0.01
3	200	743.39	601.85	19.04	0.01
5	30	171.29	141.70	17.28	0.06
5	50	282.39	234.30	17.03	0.06
5	80	461.61	381.03	17.46	0.06
5	100	584.26	482.80	17.37	0.06
5	120	657.53	550.49	16.28	0.05
5	150	859.53	723.79	15.79	0.05
5	180	1,036.04	843.17	18.62	0.05
5	200	1,113.38	932.23	16.27	0.06
8	30	319.63	268.79	15.90	0.14
8	50	526.00	433.13	17.66	0.12
8	80	851.84	715.48	16.01	0.13
8	100	1,104.68	895.34	18.95	0.13
8	120	1,257.57	1,043.86	16.99	0.14
8	150	1,600.56	1,335.36	16.57	0.15
8	180	1,945.12	1,593.04	18.10	0.13
8	200	2,170.88	1,801.67	17.01	0.15
10	30	380.43	300.93	20.90	0.28
10	50	647.87	515.05	20.50	0.28
10	80	1,018.96	808.33	20.67	0.28
10	100	1,246.90	986.12	20.91	0.29
10	120	1,533.25	1,214.69	20.78	0.29
10	150	1,851.07	1,455.79	21.35	0.29
10	180	2,244.95	1,789.53	20.29	0.52
10	200	2,526.82	2,040.52	19.25	0.54
15	30	619.49	473.60	23.55	1.45
15	50	995.24	777.75	21.85	1.60
15	80	1,567.77	1,223.02	21.99	1.53
15	100	1,997.41	1,571.13	21.34	1.45
15	120	2,344.83	1,826.04	22.12	1.54
15	150	2,927.23	2,359.42	19.40	1.53
15	180	3,593.53	2,856.56	20.51	1.77

Canteiros [n]	Distância máxima [km]	Custo Inserção [km] (1)	Custo 2-opt [km] (2)	Desvio [%]	Tempo de execução [s]
15	200	3,915.82	3,096.27	20.93	1.80
20	30	831.56	638.39	23.23	3.39
20	50	1,355.79	1,046.21	22.83	3.17
20	80	2,199.43	1,766.50	19.68	3.76
20	100	2,622.88	2,124.71	18.99	3.83
20	120	3,284.44	2,626.13	20.04	3.11
20	150	4,005.00	3,137.65	21.66	3.49
20	180	4,949.53	3,679.87	25.65	3.03
20	200	5,166.79	4,200.36	18.70	4.04
30	30	1,256.75	930.38	25.97	8.55
30	50	2,023.00	1,531.57	24.29	11.55
30	80	3,319.38	2,464.63	25.75	11.57
30	100	3,894.40	2,938.80	24.54	10.20
30	120	4,924.20	3,895.90	20.88	15.55
30	150	5,846.00	4,804.63	17.81	11.09
30	180	7,127.44	5,576.11	21.77	10.43
30	200	7,986.75	6,099.13	23.63	12.63
50	30	2,058.38	1,458.00	29.17	39.81
50	50	3,450.00	2,439.30	29.30	38.57
50	80	5,587.00	3,945.50	29.38	49.70
50	100	6,695.43	5,053.86	24.52	43.22
50	120	8,118.30	6,152.50	24.21	43.62
50	150	10,252.29	7,643.29	25.45	51.84
50	180	11,627.75	8,185.75	29.60	51.94
50	200	13,654.00	10,375.29	24.01	45.71
80	30	3,361.75	2,489.25	25.95	79.16
80	50 Km	5,670.80	4,065.00	28.32	78.77
80	80 Km	9,079.50	6,455.00	28.91	75.53
80	100 Km	11,086.80	7,686.60	30.67	85.73
80	120 Km	13,379.20	9,744.80	27.16	101.40
80	150 Km	17,327.00	12,529.33	27.69	74.85
80	180 Km	20,857.00	14,541.20	30.28	90.14
80	200 Km	22,015.00	16,219.40	26.33	70.02

Fonte: elaboração do autor

Primeiramente, foi avaliada a eficácia de aprimoramento da abordagem de busca local, onde foi observada a comparação dos custos médios através do desvio. Pelo gráfico do desvio em função do número de canteiros (cf. Figura 43), é possível perceber que o desvio aumenta em função do número de nós com uma tendência logarítmica (linha de tendência pontilhada). Assim, o método de aprimoramento se torna mais relevante com o aumento do número de canteiros do problema.

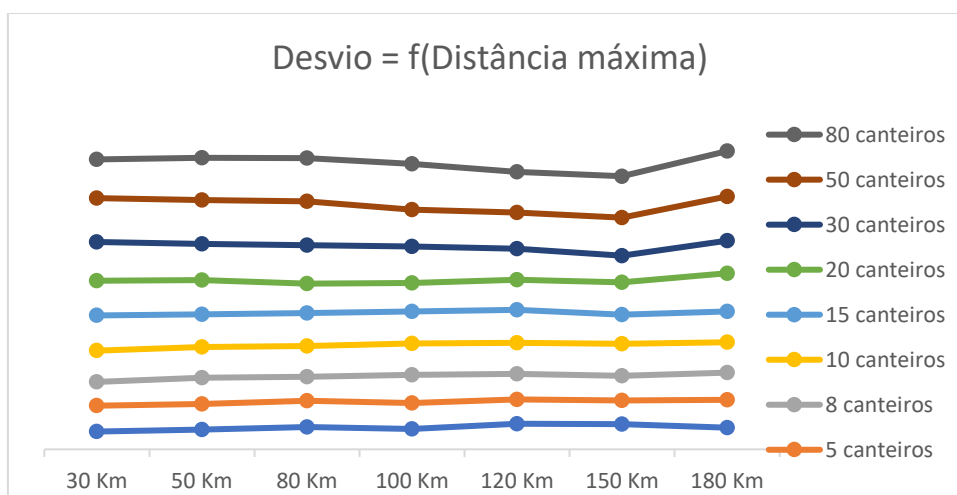
Figura 43 – Gráfico do desvio em função do número de canteiros



Fonte: elaboração do autor

Em seguida, foi analisado o desvio em função da variação das distâncias máximas (cf. Figura 44), comprovando que elas não alteram a relevância do método de aprimoramento, pois as curvas são praticamente paralelas e horizontais.

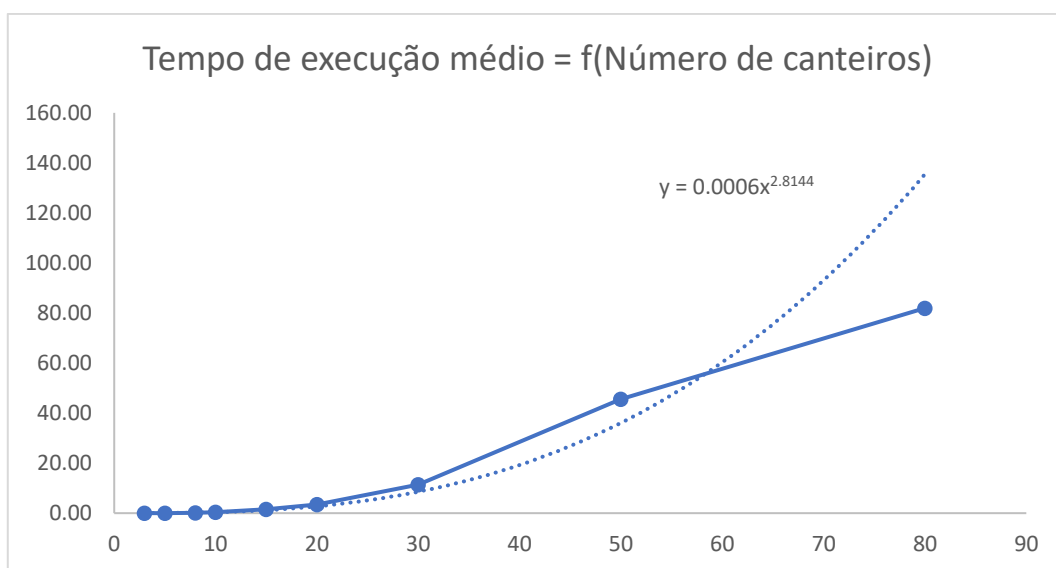
Figura 44 – Gráfico do desvio em função das distâncias máximas



Fonte: elaboração do autor

Em sequência, foi traçado o gráfico do tempo de execução médio em função do número de canteiros (cf. Figura 45), a fim de analisar a complexidade da metodologia e como o tempo de execução aumenta com o aumento do número de canteiros.

Figura 45 – Gráfico do tempo de execução médio em função do número de nós



Fonte: elaboração do autor

Através do gráfico é observado que o tempo de execução médio possui uma tendência de crescimento bem inferior ao de seu pior cenário, complexidade polinomial de grau quatro (Seção 6.1.1.), pois a curva de ajuste se identifica com uma função polinomial de grau 3.

6.4. Detalhamento da análise de comparação entre os métodos

6.4.1. Instâncias e cenários

Para a avaliação de comparação, foi construída uma tabela que compara os resultados do método heurístico desenvolvido com as soluções do método exato com critério de parada de 30 minutos empregado no *software* AMPL. Essa análise busca verificar a qualidade da ferramenta desenvolvida, e quais fatores e condições influenciam a sua eficácia. Assim, foram levantados diversos cenários para o estudo.

Inicialmente, foram realizadas oito variações da quantidade de canteiros:

- | | |
|--------------------------|---------------------------|
| 1. 3 canteiros de obras; | 5. 7 canteiros de obras; |
| 2. 4 canteiros de obras; | 6. 8 canteiros de obras; |
| 3. 5 canteiros de obras; | 7. 12 canteiros de obras; |
| 4. 6 canteiros de obras; | 8. 15 canteiros de obras. |

Para cada variação, foram executados quatro cenários em cada qual a distância entre os nós varia de 1 Km à distância máxima, como visto a seguir:

1. Distância máxima de 30 km entre os nós;
2. Distância máxima de 50 km entre os nós;
3. Distância máxima de 100 km entre os nós;
4. Distância máxima de 150 km entre os nós;

Para cada combinação de cenário foram realizadas 10 instâncias diferentes, onde as distâncias entre os canteiros, as janelas de tempo de cada nó e a lista de demanda e oferta foram geradas aleatoriamente respeitando os intervalos estudados pelo *Benchmarking*, como na análise da Seção 6.3. Dessa forma, são obtidas 320 instâncias (8 quantidades de canteiros de obras, 4 tipos de distância máxima e 10 execuções do algoritmo para cada combinação de cenário).

Para mensurar o impacto na diferença de custos entre os dois métodos de solução pela variação do parâmetro “coef_roteiro_inicial_max” do método heurístico, foram executadas cinco replicações das instâncias acima, no qual há uma variação da quantidade de roteiros iniciais devido ao valor arbitrário de “coef_roteiro_inicial_max”. Essas variações são expressas a seguir:

- | | |
|----------------------------------|----------------------------------|
| 1. coef_roteiro_inicial_max = 5; | 4. coef_roteiro_inicial_max = 2; |
| 2. coef_roteiro_inicial_max = 4; | 5. coef_roteiro_inicial_max = 1. |
| 3. coef_roteiro_inicial_max = 3; | |

Portanto, foram realizadas 1600 execuções do algoritmo para o método heurístico (8 quantidades de canteiros de obras, 4 tipos de distância máxima, 10 execuções do algoritmo para cada combinação de cenário, e 5 replicações pela variação do parâmetro “coef_roteiro_inicial_max”), e 320 execuções do AMPL para o método exato (8 quantidades de canteiros de obras, 4 tipos de distância máxima e 10 execuções do algoritmo para cada combinação de cenário). O tempo de serviço foi padronizado em 0,6 hora, e os parâmetros da velocidade média, da capacidade de carga e do número de veículos disponíveis seguiram, respectivamente, as funções das equações (6.2), (6.3) e (6.4).

Para a implementação da comparação de resultados, foi programado um *script* no Python que armazena os conjuntos e parâmetros utilizados na execução do método heurístico em um documento com o formato de entrada do *software* AMPL, assim foi possível comparar ambos os métodos com as mesmas instâncias.

Ademais, foi desenvolvida uma função para automatizar a execução de cenários no *software* AMPL, onde executa-se múltiplos cenários simultaneamente, armazenando os resultados (custo da solução, tempo de execução e *gap* de otimalidade) em arquivos paralelos.

6.4.2. Resultados

Para análise dos resultados, foram comparados os custos da solução heurística com os custos da solução exata para cada cenário através da coluna *Dif* ($\frac{\text{Custo Heurístico} - \text{Custo Exato}}{\text{Custo Heurístico}}$), bem como o armazenamento dos seus respectivos tempos de execução. Exclusivamente para o método exato, foram calculados os *gaps* de otimalidade (*gap*) e a proporção de soluções factíveis de cada cenário.

Com auxílio de uma planilha do “Microsoft Excel”, os milhares de dados foram manipulados, e alguns gráficos foram traçados para melhor compreendê-los.

Assim, foram analisados três aspectos:

1. A análise comparativa das duas metodologias através do *Dif*;
2. A análise comparativa dos tempos de execução em função da complexidade;
3. A análise do fator arbitrário “coef_roteiro_inicial_max” para o aprimoramento da eficácia do método.

Primeiramente, supondo que não haja distinção dos valores de “coef_roteiro_inicial_max” em cada execução, foi realizado o cálculo do *Dif* médio das 50 execuções do algoritmo heurístico (10 execuções para cada cenário com 5 replicações) que possuem a mesma combinação de cenário (número de canteiros e distância máxima), bem como os seus respectivos desvios padrões médios. Além disso, foram calculados os *gaps* médios para as 10 execuções de cada cenário do

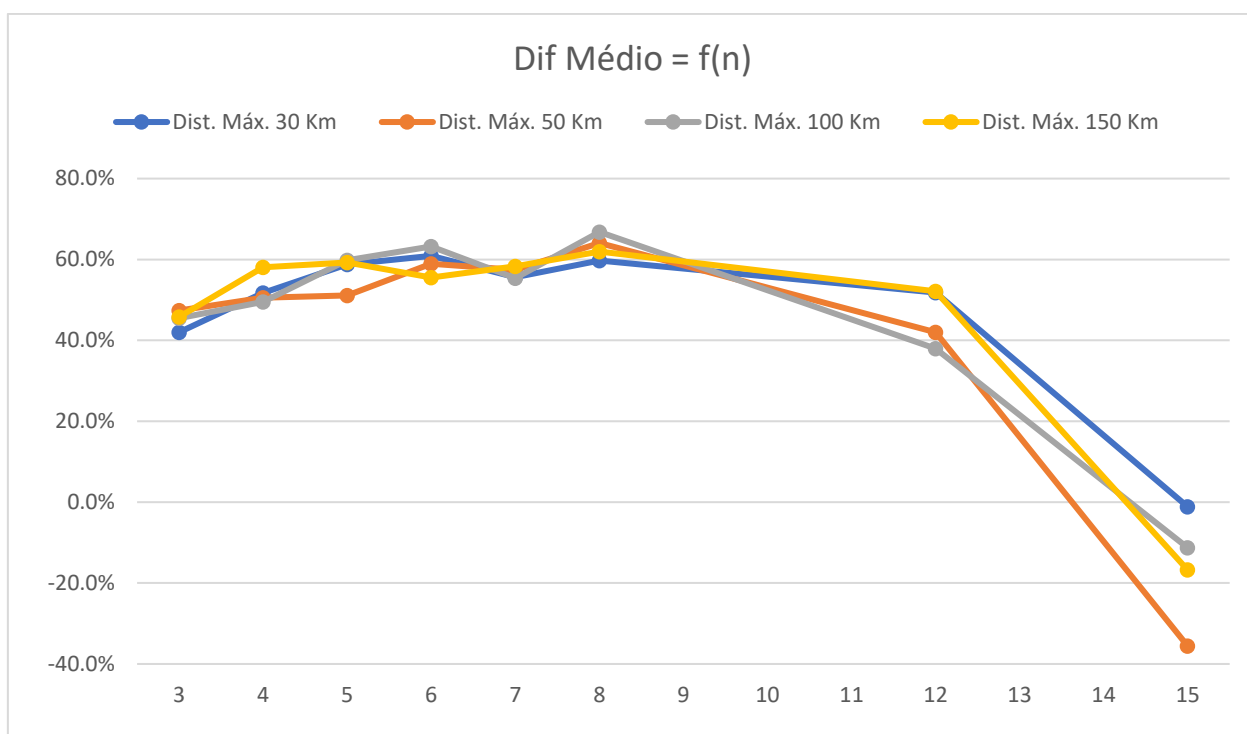
método exato, bem como suas respectivas proporções médias de soluções factíveis (cf. Tabela 6).

Tabela 6 – Difs, desvio padrões, *gaps* e proporções médios sem a distinção do “coef_roteiro_inicial_max”.

Canteiros [n]	Distância máxima [km]	Dif Médio [%]	Desvio Padrão do Dif Médio [%]	Gap Médio [%]	Proporção de soluções factíveis [%]
3	30	42.0	17.2	0.0	100.0
3	50	47.4	21.3	0.0	100.0
3	100	45.5	11.0	0.0	100.0
3	150	45.8	20.1	0.0	100.0
4	30	51.8	13.1	0.0	100.0
4	50	50.6	16.6	0.0	100.0
4	100	49.5	16.1	0.0	100.0
4	150	58.0	13.8	0.0	100.0
5	30	58.8	12.5	0.0	100.0
5	50	51.1	12.8	0.0	100.0
5	100	59.8	8.0	0.0	100.0
5	150	59.2	15.9	0.0	100.0
6	30	60.9	8.3	1.7	100.0
6	50	59.0	12.1	1.4	100.0
6	100	63.2	7.9	0.0	100.0
6	150	55.5	9.6	0.5	100.0
7	30	55.6	9.6	7.1	100.0
7	50	57.4	10.7	9.7	100.0
7	100	55.3	5.6	9.1	100.0
7	150	58.3	9.1	8.2	100.0
8	30	59.7	7.6	16.8	100.0
8	50	64.1	6.4	13.5	100.0
8	100	66.8	7.4	13.2	100.0
8	150	62.0	10.1	12.7	100.0
12	30	51.9	5.8	39.4	50.0
12	50	42.0	27.8	45.0	80.0
12	100	38.0	24.7	52.1	70.0
12	150	52.1	13.2	40.5	70.0
15	30	-1.1	9.3	58.5	30.0
15	50	-35.6	26.1	67.5	40.0
15	100	-11.3	12.8	65.4	40.0
15	150	-16.8	35.9	69.1	70.0

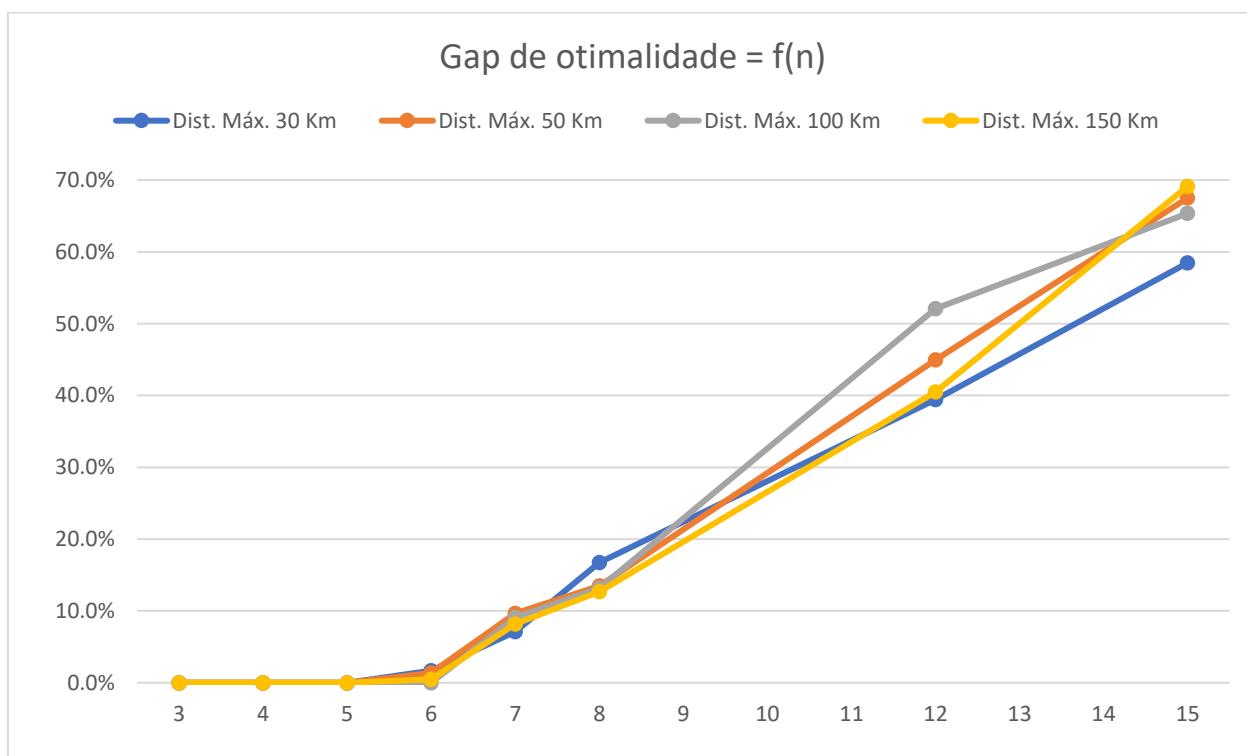
Fonte: elaboração do autor

Assim, foi possível traçar um gráfico do *Dif* médio e do *gap* de otimalidade em função do número de canteiros. A fim de obter uma melhor compreensão do parâmetro da distância máxima, em ambos os gráficos foram construídas quatro curvas de acordo com o mesmo (cf. Figura 46 e Figura 47).

Figura 46 – Gráfico do *Dif* médio em função do número de canteiros de obras

Fonte: elaboração do autor

Figura 47 – Gráfico do Gap médio em função do número de canteiros de obras



Fonte: elaboração do autor

A partir da Figura 46, é constatado um leve crescimento do *Dif* médio com o aumento do número de canteiros, demonstrando que à medida que o número de canteiros cresce, o resultado da ferramenta não sofre grandes impactos. Contudo, devido ao critério de parada imposto ao método exato (1800 segundos), a partir de 8 canteiros, o *Dif* médio começa a cair rapidamente e se torna negativo a partir de 14 canteiros. Isso ocorre devido ao crescimento rápido da complexidade do método exato, demandando cada vez mais tempo para encontrar a solução ótima. Porém, como há um limite de tempo de compilação, a dificuldade de obter soluções satisfatórias/ótimas aumenta com a complexidade. Essa correlação é constatada na Figura 47, onde apresenta um crescimento rápido do *gap* em função do número de canteiros.

Um outro fator importante é a proporção de soluções inteiras factíveis, que de acordo com a Tabela 6, diminui rapidamente com o aumento do número de canteiros. Isso comprova que com o aumento da complexidade do problema, a utilização do método exato se torna inviável.

Além disso, é observado que o aumento da variação das distâncias não afeta a variação do *Dif* médio e do *gap* médio, uma vez que as quatro séries estão quase sobrepostas em ambos os gráficos.

Ademais, através dos desvios padrões da Tabela 6, é constatado uma variação relevante do *Dif* médio entre as execuções com as mesmas condições, indicando casos em que o valor do desvio padrão relativo à média chega a ser quase metade do valor.

Posteriormente, supondo que a ferramenta seja executada com a melhor variação do valor de “coef_roteiro_inicial_max”, ou seja, onde o *Dif* é o menor das 5 replicações realizadas para analisar a variação de “coef_roteiro_inicial_max”, foi realizado o cálculo do *Dif* mínimo das replicações para cada cenário. Assim, cada cenário possuirá uma amostra de 10 execuções com os seus respectivos *Difs* mínimos.

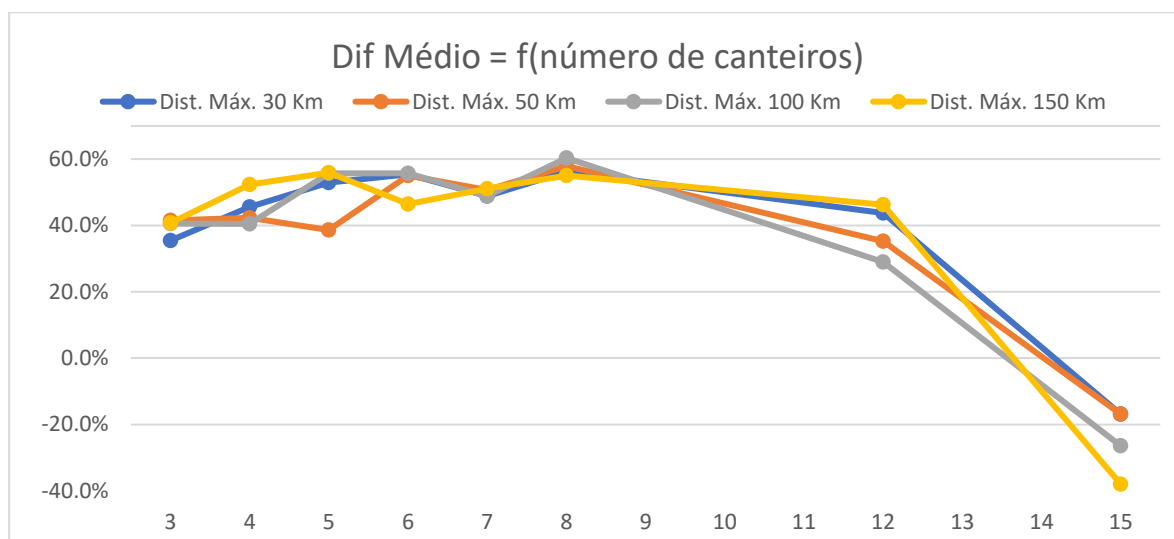
Em seguida, a partir dessas 10 execuções, foram calculados para cada cenário a média desses *Difs* e seus respectivos desvios padrões (cf. Tabela 7).

Tabela 7 – Difs e desvio padrões médios com a melhor variação do “coef_roteiro_inicial_max”.

Canteiros [n]	Distância máxima [km]	Dif Médio [%]	Desvio Padrão do Dif Médio [%]
3	30	35.4	20.3
3	50	41.5	23.2
3	100	40.6	12.1
3	150	40.8	18.4
4	30	45.5	23.5
4	50	42.3	23.5
4	100	40.4	20.8
4	150	52.4	13.8
5	30	52.9	26.6
5	50	38.6	16.4
5	100	55.7	20.1
5	150	55.9	23.2
6	30	55.5	8.3
6	50	55.0	13.5
6	100	55.7	10.1
6	150	46.4	11.4
7	30	48.9	12.2
7	50	50.7	13.1
7	100	48.9	6.8
7	150	51.1	11.4
8	30	56.2	9.1
8	50	57.9	7.2
8	100	60.3	6.3
8	150	55.0	11.4
12	30	43.7	7.6
12	50	35.2	28.9
12	100	29.0	29.8
12	150	46.3	14.7
15	30	-16.9	9.3
15	50	-52.2	21.8
15	100	-26.4	22.5
15	150	-38.0	54.2

Fonte: elaboração do autor

Como na análise anterior, foi traçado o gráfico do *Dif* médio em função do número de canteiros de obras para as diferentes distâncias máximas (cf. Figura 48).

Figura 48 – Gráfico do *Dif* mínimo em função do número de canteiros de obras

Fonte: elaboração do autor

Dessa forma, comparando os dois gráficos (cf. Figuras 46 e 48), é constatada uma redução média de 8% do *Dif* médio se utilizarmos um bom valor para a variável “coef_roteiro_inicial_max”. Ademais, a curva do *Dif* médio demonstra uma queda ainda mais acentuada chegando a valores médios mínimos de -52.2%.

Da mesma forma que a análise anterior, o parâmetro de distância máxima continua não correlacionado com o *Dif* médio, e o desvio padrão permanece alto em relação à média de acordo com a Tabela 7.

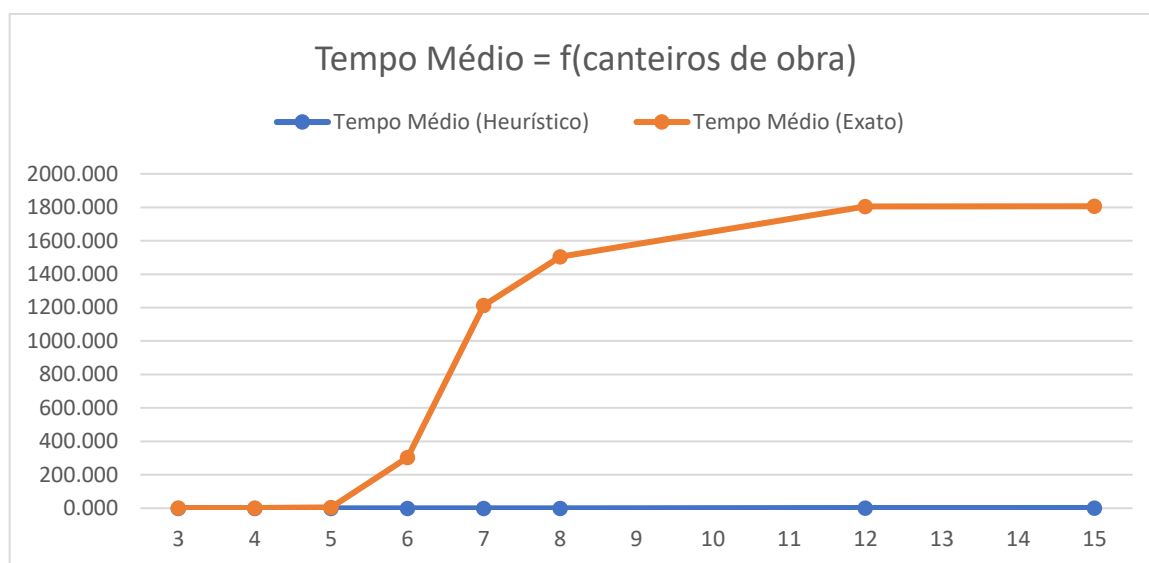
Em sequência, a partir da Tabela 8, foi traçado um gráfico para comparar os tempos de execução dos métodos heurístico e exato, buscando averiguar o tempo médio de cada metodologia para cada cenário (cf. Figura 49). A abordagem pela média das 10 execuções do algoritmo, foi utilizada para atenuar a variação do tempo de execução do método exato empregado, pois dependendo do conjunto de dados o *solver* CPLEX pode encontrar rapidamente o nó da solução ótima pelo algoritmo *Branch-and-Bound* e solucioná-lo utilizando o Simplex, independentemente da complexidade do problema.

Tabela 8 – *Difs* e tempos médios com a variação do “coef_roteiro_inicial_max”.

Canteiros [n]	Tempo Médio (Heurística) [s]	Tempo Médio (Exata) [s]
3	0.011	0.551
4	0.023	1.419
5	0.054	4.316
6	0.068	302.907
7	0.094	1213.449
8	0.115	1505.135
12	0.375	1805.400
15	1.396	1806.159

Fonte: elaboração do autor

Figura 49 – Gráfico do tempo médio em função do número de canteiros de obra



Fonte: elaboração do autor

Assim como observado em teoria, é constatado que o tempo de execução pela combinação dos algoritmos de *Branch-and-Bound* e Simplex segue um crescimento exponencial em função da complexidade do problema para os cenários com menores canteiros, onde a solução é ótima e o *gap* é próximo de zero. Porém, devido ao critério de parada imposto (1800 segundos), à medida que a complexidade aumenta o tempo de execução médio tende a esse valor. Já o tempo de execução da ferramenta realiza um crescimento polinomial que é bem inferior ao método exato.

Vale ressaltar, que apesar dos valores estarem em função do número de canteiros, o tempo de execução aumenta em função da complexidade do problema. Portanto, se adicionarmos qualquer variável adicional, como tipo de produtos, tipo de descartes, tipo de veículo (frota heterogênea), múltiplos horizontes, ou inclusive adicionar outros custos como o custo fixos por veículo ou custo de estocagem, o tempo de execução do método exato crescerá ainda mais rápido se comparado com o método heurístico desenvolvido.

Sucessivamente, visto que o valor ideal de “coef_roteiro_inicial_max” é importante para aprimorar o resultado da solução heurística, decidiu-se analisar melhor esse fator. Assim, foi calculado o *Dif* médio das 10 execuções do algoritmo para cada variação do “coef_roteiro_inicial_max” (5 variações), bem como os seus respectivos tempos de execução (cf. Tabela 9).

Tabela 9 – Difs e tempos médios com a variação do “coef_rotreiro_inicial_max”.

Canteiros [n]	Distância máxima [km]	“coef_rotreiro_inicial_max”	Dif Médio [%]	Tempo Médio (Heurística) [s]
3	30	5	42.4	0.009
3	30	4	51.9	0.009
3	30	3	48.1	0.010
3	30	2	47.2	0.009
3	30	1	38.8	0.010
3	50	5	44.7	0.010
3	50	4	45.2	0.010
3	50	3	44.4	0.009
3	50	2	38.8	0.011
3	50	1	44.7	0.011
3	100	5	45.2	0.010
3	100	4	44.4	0.010
3	100	3	45.0	0.014
3	100	2	47.8	0.013
3	100	1	44.4	0.010
3	150	5	46.5	0.014
3	150	4	45.0	0.014
3	150	3	47.8	0.012
3	150	2	44.4	0.009
3	150	1	46.5	0.015
4	30	5	53.5	0.017
4	30	4	48.5	0.017
4	30	3	53.4	0.014
4	30	2	60.7	0.017
4	30	1	52.2	0.020
4	50	5	44.7	0.019
4	50	4	51.1	0.018
4	50	3	60.1	0.019
4	50	2	46.8	0.028
4	50	1	51.0	0.045
4	100	5	43.9	0.025
4	100	4	55.5	0.024
4	100	3	46.8	0.025
4	100	2	51.0	0.046
4	100	1	43.9	0.030
4	150	5	55.5	0.024
4	150	4	59.4	0.014
4	150	3	57.6	0.019
4	150	2	55.3	0.021
4	150	1	58.5	0.016
5	30	5	62.4	0.021
5	30	4	53.6	0.036
5	30	3	64.9	0.025
5	30	2	59.4	0.040
5	30	1	53.2	0.033
5	50	5	45.4	0.055
5	50	4	57.6	0.039
5	50	3	58.7	0.054
5	50	2	59.4	0.069
5	50	1	52.1	0.056
5	100	5	58.9	0.062
5	100	4	59.4	0.076
5	100	3	59.4	0.066
5	100	2	52.1	0.053
5	100	1	58.9	0.058
5	150	5	59.4	0.069
5	150	4	59.4	0.094
5	150	3	52.1	0.055
5	150	2	58.9	0.056
5	150	1	59.4	0.069
6	30	5	64.5	0.038
6	30	4	64.4	0.038
6	30	3	64.6	0.035
6	30	2	58.7	0.034
6	30	1	57.5	0.057

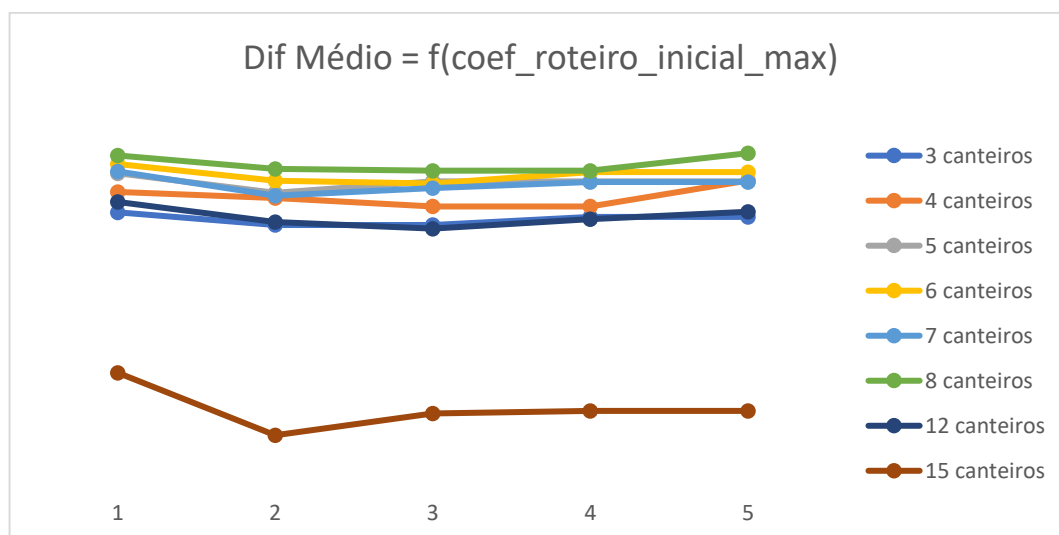
Canteiros [n]	Distância máxima [km]	"coef_roteiro_inicial_max"	Dif Médio [%]	Tempo Médio (Heurística) [s]
6	50	5	58.9	0.055
6	50	4	64.1	0.059
6	50	3	49.9	0.062
6	50	2	58.3	0.082
6	50	1	56.0	0.066
6	100	5	62.1	0.062
6	100	4	50.6	0.077
6	100	3	62.1	0.091
6	100	2	57.7	0.100
6	100	1	62.6	0.080
6	150	5	59.3	0.080
6	150	4	62.1	0.085
6	150	3	57.7	0.095
6	150	2	62.6	0.085
6	150	1	59.3	0.081
7	30	5	59.5	0.056
7	30	4	61.9	0.069
7	30	3	59.9	0.055
7	30	2	61.3	0.056
7	30	1	52.8	0.110
7	50	5	53.3	0.118
7	50	4	50.8	0.102
7	50	3	54.4	0.114
7	50	2	55.4	0.131
7	50	1	55.9	0.133
7	100	5	53.8	0.124
7	100	4	56.0	0.132
7	100	3	55.1	0.080
7	100	2	58.0	0.089
7	100	1	56.1	0.083
7	150	5	59.8	0.077
7	150	4	55.1	0.086
7	150	3	58.0	0.093
7	150	2	56.1	0.090
7	150	1	59.8	0.076
8	30	5	62.2	0.074
8	30	4	65.9	0.092
8	30	3	67.9	0.081
8	30	2	67.4	0.084
8	30	1	60.8	0.143
8	50	5	62.0	0.123
8	50	4	63.0	0.152
8	50	3	60.1	0.149
8	50	2	56.9	0.124
8	50	1	63.7	0.130
8	100	5	65.0	0.115
8	100	4	58.1	0.156
8	100	3	56.9	0.121
8	100	2	63.7	0.143
8	100	1	65.0	0.126
8	150	5	58.1	0.156
8	150	4	61.8	0.068
8	150	3	65.1	0.090
8	150	2	73.1	0.093
8	150	1	66.2	0.082
12	30	5	58.3	0.187
12	30	4	48.3	0.196
12	30	3	37.9	0.199
12	30	2	58.6	0.230
12	30	1	47.0	0.365
12	50	5	43.3	0.412
12	50	4	34.7	0.399
12	50	3	51.9	0.413
12	50	2	44.4	0.515
12	50	1	41.2	0.422
12	100	5	31.4	0.442
12	100	4	51.5	0.551

Canteiros [n]	Distância máxima [km]	"coef_roteiro_inicial_max"	Dif Médio [%]	Tempo Médio (Heurística) [s]
12	100	3	53.3	0.472
12	100	2	40.9	0.389
12	100	1	39.0	0.475
12	150	5	47.7	0.406
12	150	4	56.3	0.391
12	150	3	36.4	0.286
12	150	2	46.9	0.358
12	150	1	50.7	0.395
15	30	5	10.7	0.731
15	30	4	-27.4	0.802
15	30	3	1.4	0.712
15	30	2	-2.9	0.682
15	30	1	-0.4	1.658
15	50	5	-41.7	1.830
15	50	4	-22.3	1.804
15	50	3	-34.8	1.611
15	50	2	-16.9	1.588
15	50	1	-34.9	1.884
15	100	5	-7.1	1.734
15	100	4	-12.3	1.672
15	100	3	0.5	1.549
15	100	2	-36.9	1.146
15	100	1	-14.2	1.466
15	150	5	-16.9	1.339
15	150	4	0.5	1.555
15	150	3	-36.9	1.178
15	150	2	-14.2	1.499
15	150	1	-16.9	1.477

Fonte: elaboração do autor

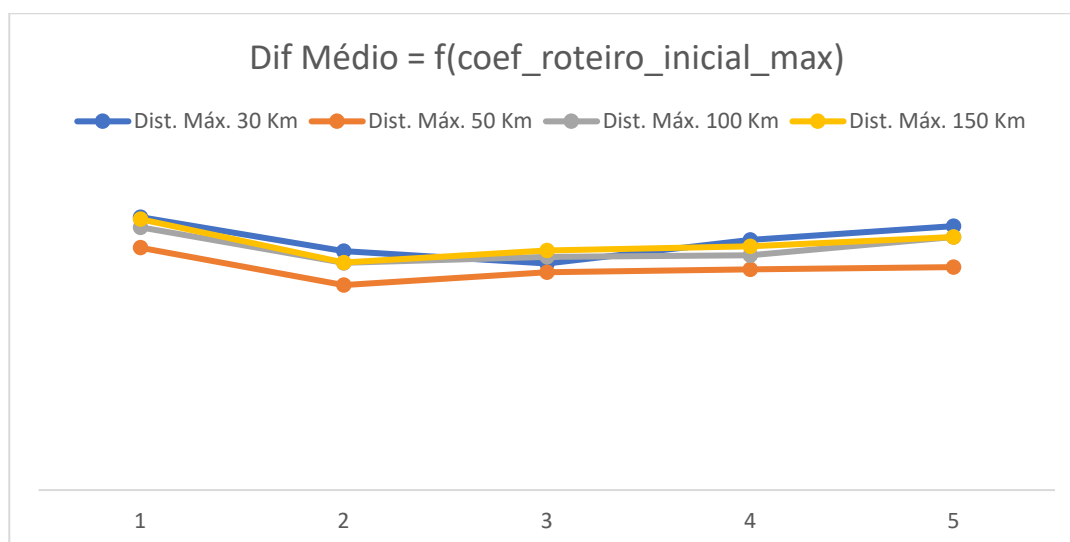
Em seguida, a fim de compreender os efeitos desse fator no resultado da ferramenta, bem como analisar sua relação com os outros parâmetros analisados, foram traçados dois gráficos do *Dif* médio em função do "coef_roteiro_inicial_max". O primeiro, é relacionado ao número de canteiros de obras utilizados (cf. Figura 50), e o segundo é vinculado à variação da distância máxima (cf. Figura 51).

Figura 50 – Gráfico do *Dif* médio em função do "coef_roteiro_inicial_max" e do número de canteiros



Fonte: elaboração do autor

Figura 51 – Gráfico do *Dif* médio em função do “coef_roteiro_inicial_max” e da distância máxima



Fonte: elaboração do autor

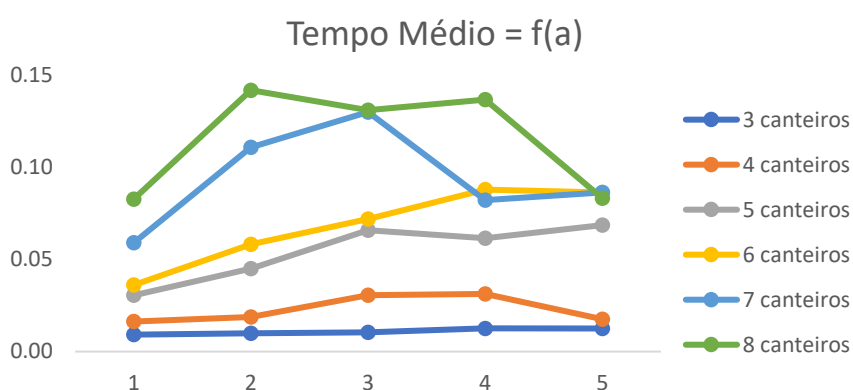
A partir da Figura 50, é observado que um valor unitário ou máximo do “coef_roteiro_inicial_max” não é preferível, e sim um valor intermediário, entre 2 e 4, dependendo do número de canteiros, pois é nesse intervalo que se encontra o valor mínimo do *Dif* médio. Contudo, as curvas não demonstram uma estrutura significativa evidente, possuindo vários valores mínimos distintos caso a caso.

A incógnita arbitrária “coef_roteiro_inicial_max” afeta diretamente o valor limite de roteiros iniciais adicionais, $\frac{|V|}{\text{coef_roteiro_inicial_max}}$. Assim, quando o valor de “coef_roteiro_inicial_max” é unitário, o algoritmo já se inicia com o número máximo de roteiros possíveis ($|V|$), resultando, muitas vezes, em um número superior ao número de roteiros necessários. Além disso, para o valor máximo de “coef_roteiro_inicial_max”, o algoritmo impede a criação de rotas iniciais adicionais, havendo a possibilidade de existir vários nós com alto grau de não coexistência em uma mesma rota e, conseqüentemente, impedindo a inserção de nós que poderiam gerar melhores soluções. Assim, os valores intermediários de “coef_roteiro_inicial_max” seriam os mais promissores, contudo o resultado do algoritmo varia bastante de acordo com os dados do problema e os graus de coexistências entre os nós, resultando em diferentes valores “ideais” de “coef_roteiro_inicial_max” para cada instância.

Da mesma forma, na Figura 51, a variação da distância máxima não parece demonstrar correlação direta com a variação de “coef_roteiro_inicial_max”.

Posteriormente, foi realizada a análise do tempo médio de execução em função do “coef_roteiro_inicial_max” (cf. Figura 52). No gráfico é observado que o tempo médio máximo é divergente para cada combinação de cenário, e não parece possuir uma estrutura correlacionada significativa com os valores de “coef_roteiro_inicial_max”, mas sim com o conjunto de dados do problema. Portanto, em cada caso há um valor de “coef_roteiro_inicial_max” que o torna mais complexo, demandando mais iterações do algoritmo.

Figura 52 – Gráfico do tempo médio em função do “coef_roteiro_inicial_max”



Fonte: elaboração do autor

Após realizar as análises do fator “coef_roteiro_inicial_max”, fica evidente que o seu valor é relevante para o resultado da solução. Infelizmente, não existe uma estrutura ou correlação direta desse fator com os parâmetros estudados, assim há a necessidade de testar as variações de “coef_roteiro_inicial_max” para encontrar o seu valor “ideal” no cenário estudado.

6.5. Detalhamento da análise inicial de aperfeiçoamento da ferramenta através da variação dos pesos dos custos de inserção

6.5.1. Instâncias e cenários

Para a análise inicial de aperfeiçoamento, foi construída uma tabela que compara a solução do método heurístico com os pesos do custo de inserção variados, com a solução do método exato.

Essa análise pretende compreender, especificamente, o funcionamento do custo de inserção “c3” referente à distância da equação (5.5), e como a variação dele pode aumentar a eficácia do método e, assim, diminuir a diferença entre os custos da solução heurística e exata.

A fim de comparar o aprimoramento dessa diferença com o já calculado pela variável *Dif* na Seção 6.4., utiliza-se as mesmas 1600 combinações de cenários para o método heurístico (8 quantidades de canteiros de obras, 4 tipos de distância máxima, 10 execuções do algoritmo para cada combinação de cenário, e 5 replicações pela variação do parâmetro “coef_roteiro_inicial_max”), e as mesmas 320 instâncias já calculadas pelo método exato (8 quantidades de canteiros de obras, 4 tipos de distância máxima e 10 execuções do algoritmo para cada combinação de cenário).

Em seguida, para cada combinação de cenário do método heurística, foram realizadas três execuções do algoritmo (Equação *Padrão* já foi calculada na Seção 6.4.), no qual há uma variação dos pesos do custo de inserção expressos a seguir:

- | | |
|--|--|
| 1. <i>Padrão</i> : $C_T = C_1 + C_2 + C_3$; | 3. <i>Tipo 2</i> : $C_T = C_1 + C_2 + 3 * C_3$; |
| 2. <i>Tipo 1</i> : $C_T = C_1 + C_2 + 2 * C_3$; | 4. <i>Tipo 3</i> : $C_T = C_1 + C_2 + 5 * C_3$. |

Dessa maneira, são realizadas 4800 execuções do algoritmo heurístico (1600 combinações de cenários e 3 tipos de variação de pesos), a fim de analisar melhor a sensibilidade do aperfeiçoamento referente à solução exata que já foi previamente calculada pelo *software* AMPL na Seção 6.4.

6.5.2. Resultados

A fim de analisar os aprimoramentos, foram comparados os custos da heurística do tipo padrão (calculada na Seção 6.4.), 1, 2 e 3, com os custos da solução exata (calculada na Seção 6.4.) para cada cenário estudado através da coluna *Dif* ($\frac{\text{Custo Heurístico} - \text{Custo da Solução Ótima}}{\text{Custo Heurístico}}$). Ademais, foi realizado o cálculo do melhor *Dif* (*Best Dif*) pela análise dos quatro tipos ao mesmo tempo, onde foi retido o melhor *Dif* de cada execução entre as quatro variações de peso, em seguida retirado a média das 10 execuções para cada cenário.

Como determinado na Seção 6.4., foi utilizado o valor ideal de “coef_roteiro_inicial_max” para cada cenário, ou seja, escolheu-se o menor custo entre as variações de “coef_roteiro_inicial_max”. Depois, foi calculado o *Dif* médio das 10 execuções do algoritmo para cada combinação de cenário (cf. Tabela. 10).

Tabela 10 – *Difs* médios da variação do custo de inserção “c3”.

Canteiros [n]	<i>Dif</i> Médio Padrão [%]	<i>Dif</i> Médio 1 [%]	<i>Dif</i> Médio 2 [%]	<i>Dif</i> Médio 3 [%]	Best <i>Dif</i> Médio [%]
3	39.58	38.58	38.43	40.39	38.08
4	45.43	46.28	45.35	45.35	43.66
5	50.73	51.49	52.72	52.95	49.71
6	53.16	53.75	52.56	52.50	50.60
7	49.88	48.70	48.53	49.07	46.35
8	57.36	57.52	58.13	56.90	54.80
12	38.04	37.07	35.29	35.05	32.63
15	-35.05	-36.64	-37.65	-33.03	-42.79

Fonte: elaboração do autor

A partir da Tabela 10, é constatado que a variação dos pesos dos custos de inserção também não possui um padrão fixo, e varia bastante de acordo com os dados da instância. Esse fato é constatado, porque em instâncias com janelas de tempo mais limitadas, os custos “c1” e “c2” devem possuir maior peso, da mesma forma, se as janelas de tempo forem mais esparsas, o custo “c3” deve ser mais valorizado.

Todavia, se utilizarmos a variação dos pesos para encontrar o equilíbrio ideal para cada cenário, é possível alcançar um aprimoramento significativo da ferramenta. Somente balanceando um pouco o custo de “c3”, foi possível alcançar uma redução média de 3% entre o *Best Dif* médio e o *Dif* Médio Padrão.

6.6. Análises e considerações finais

É evidente que ambas as metodologias possuem vantagens e desvantagens dependendo da situação. O método exato possui claramente uma vantagem em relação ao método heurístico em instâncias com poucos canteiros devido ao seu resultado ótimo, porém o seu tempo de execução é bastante grande e cresce em uma velocidade exponencial (cf. Figuras 49), tornando desvantajosa e muitas vezes inviável para a resolução do problema com mais variáveis.

Se pensado à longo prazo, a metodologia heurística desenvolvida se mostra mais interessante, primeiro, porque à medida que as variáveis do problema

aumentam, o *Dif* da metodologia não demonstra um crescimento expressivo. Segundo, pois, futuramente, com o intuito de aprimorar as especificidades do problema para uma melhor resolução, a sua complexidade aumentaria drasticamente, e tornaria limitada ou até inviável o uso dos métodos exatos. Ou seja, se adicionássemos outras variáveis como os custos de estocagem, a heterogeneidade dos veículos (custos fixos, velocidade, capacidade, etc.), a distinção por diferentes tipos de produtos e descartes, o aumento no horizonte de tempo de planificação, entre outras especificações, o método exato se tornaria inaplicável.

Ademais, assim como constatado nas análises anteriores, é possível aprimorar ainda mais o método heurístico pela utilização do valor arbitrário “ideal” de “coef_rotreiro_inicial_max” e pela calibragem da ponderação dos custos de inserção, desvendando os coeficientes multiplicadores “ideais” para as condições do problema.

Outra maneira de aprimorar a ferramenta, seria a inclusão de novos custos de inserção, buscando otimizar o cálculo do custo geral, ou, a mudança do algoritmo de busca local para outros mais elaborados, que permitam uma busca mais ampla do espaço de soluções, porém que aumentariam a complexidade da ferramenta.

Assim, existem diferentes formas de aprimorar a metodologia, reduzindo bastante o *Dif* entre os resultados heurísticos e exatos para menores instâncias, e mantendo a sua viabilidade de resolução.

7. CONCLUSÕES E PERSPECTIVAS FUTURAS

Neste trabalho, com o objetivo de desenvolver uma ferramenta para otimização da gestão de fluxos de logística reversa em uma plataforma modular, os processos gerais de funcionamento da mesma foram separados em processos menores, concentrando-se no seu fluxo à jusante, mais especificamente na logística de distribuição de produtos e coleta de descartes. Assim, para a simplificação do problema, este foi repartido em dois subproblemas totalmente independentes, “Veículos basculantes para o transporte de descartes inaceitáveis” e “Veículos VUC para o transporte de produtos e descartes aceitáveis”, no qual foi decidido dar um enfoque total para o último, aplicando métodos da Pesquisa Operacional para a otimização do roteamento de veículos.

Dessa forma, a modelagem matemática do problema foi efetuada, formalizando os conjuntos e parâmetros, as variáveis de decisão, a função objetivo, e as restrições que seriam intrínsecas para o desenvolvimento da ferramenta. A partir dessas especificações, o método heurístico de construção por inserção modificado com aprimoramento pela busca local 2-opt foi desenvolvido, implementado e executado. Em sequência, usando aplicativo de otimização AMPL, foi executado o método de solução exato para esse mesmo modelo matemático.

Posteriormente, foram realizadas três análises, a primeira visando analisar a eficácia da utilização do método de busca local na ferramenta para diferentes condições, a segunda comparando os resultados da heurística desenvolvida com a do método exato, e a terceira variando a ponderação dos custos de inserção para o aperfeiçoamento da ferramenta. Nas três análises, diversas variáveis foram avaliadas, buscando compreender as diferentes tendências de crescimento, a fim de otimizar a eficácia da metodologia.

Na análise do método heurístico, foi constatado que o método de aprimoramento por busca local ganha mais relevância com o aumento do número de canteiros, e que o seu aprimoramento não está diretamente correlacionado com a variância das distâncias entre nós. Ademais, foi observado que o tempo de execução em função do número de canteiros segue uma função polinomial de grau três, inferior à complexidade calculada do algoritmo para o pior caso, função polinomial de grau quatro.

Na análise comparativa entre os resultados do método heurístico e do método exato, o método exato possui vantagens com instâncias bem pequenas e sem muitas variáveis devido ao resultado ótimo da solução. Contudo, refletindo a longo prazo, para trazer mais realismo ao modelo, e visando uma solução mais aprofundada e específica, pode ser necessário um aprimoramento das especificações do problema. Assim, a utilização do método heurístico seria mais vantajosa. Pois, o aumento da complexidade tornaria a utilização do método exato limitado ou até mesmo inviável dependendo do critério de parada, enquanto a metodologia heurística desenvolvida permaneceria com soluções satisfatórias e realizáveis. Além disso, o custo da solução da ferramenta pode ser diminuído razoavelmente através da utilização do valor ideal de “coef_roteiro_inicial_max”, de melhores balanceamentos dos pesos do custo de inserção, do aprimoramento da busca local 2-opt para métodos mais elaborados. Apesar desses aprimoramentos aumentarem a complexidade e, conseqüentemente, o tempo de execução, a ferramenta continuaria sendo viável para a resolução do problema, pois a sua complexidade continuaria sendo polinomial. Portanto, a escolha do modelo heurístico desenvolvido se demonstra mais promissora para estudos futuros.

Do ponto de vista pedagógico, este trabalho permitiu colocar em prática diferentes conhecimentos teóricos e técnicos:

1. Redação de um relatório de especificações detalhado;
2. Formalização e modelagem de um problema de otimização;
3. Concepção de métodos heurísticos;
4. Implementação dos métodos de solução em Python e no AMPL;
5. Análise e comparação de resultados entre diferentes métodos.

De um ponto de vista pessoal, este trabalho permitiu o desenvolvimento de múltiplas competências de gestão e organização.

Como conclusão final, espera-se que a ferramenta possa ser utilizada para aplicações em logística reversa em diversas plataformas modulares, auxiliando na tomada de decisão dos roteiros. Ademais, deseja-se que, posteriormente, este método contribua para futuros estudos nesse domínio, aprimorando cada vez mais os resultados da solução e habilitando o setor a ser mais sustentável.

REFERÊNCIAS BIBLIOGRÁFICAS

- AHUJA, R. K.; KUMAR, A.; JHA, K. C.; ORLIN, J. B. Exact and Heuristic Algorithms for the Weapon-Target Assignment Problem. **Operations Research**. Vol. 55, num. 6, pp. 1136-1146, 2007.
- ÁLVAREZ, A.; MUNARI, P. Abordagens metaheurísticas para o problema de roteamento de veículos com janelas de tempo e múltiplos entregadores. *Gest. Prod.*; São Carlos. Vol. 23, n. 2, pp. 279-293, 2016 Disponível em: <<http://dx.doi.org/10.1590/0104-530X2359-15>>. Acesso em 19 de setembro de 2019.
- ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. **Pesquisa Operacional: para cursos de engenharia**. Elsevier Editora Ltda.; 2007.
- BATTARRA, M.; CORDEAU, J.; IORI, M. Pickup-and-Delivery Problems for Goods Transportation. In: TOTH, V.; VIGO, D. **Vehicle Routing Problems, Methods, and Applications Second Edition**. Chapter 6, pp. 161-181. The Society for industrial and Applied Mathematics and Mathematical Optimization Society, 2014.
- BARRETO, I. M. C. B. N. **Gestão de resíduos na construção civil**, pp. 28. Aracaju, Senai 2005.
- BENOIST, T.; GARDI, F.; JEANJEAN, A.; ESTELLON, B. **RECHERCHE LOCALE POUR UN PROBLÈME D'OPTIMISATION DE TOURNÉES DE VÉHICULES AVEC GESTION DES STOCKS**. 8^e Conférence Internationale de MODélisation et SIMulation - MOSIM'10. 10 ao 12 de maio de 2010, Hammamet, Tunísia.
- BERBEGLIA G.; CORDEAU, J.; GRIBKOVSKAIA, I.; LAPORTE, G. Static pickup and delivery problems: a classification scheme and survey. **TOP**. Vol. 15, pp. 1-31.; 2007.

BIG-O CHEAT SHEET. **Know Thy Complexities!** 2013 Disponível em: <<http://bigocheatsheet.com/>>. Acesso em: 22 de agosto de 2019.

CAMPBELL, A.; CLARKE, L.; KLEYWEGT, A.; SAVELSBERGH, M. The Inventory Routing Problem. In: CRAINIC, T.G.; LAPORTE G. (eds) **Fleet Management and Logistics**, pp 95-113. Centre for Research on Transportation. Springer, Boston, MA. Abril, 1998.

CAMPBELL, A.; CLARKE, L.; SAVELSBERGH, M. **Inventory routing in practice**. The Vehicle Routing Problem, pp. 309–330. SIAM Monographs on Discrete Mathematics and Applications 9. SIAM. 2002.

CAMPBELL, A.; SAVELSBERGH, M. Efficient insertion heuristics for vehicle routing and scheduling problems. **Transportation Science**. Vol. 38, pp. 369-378. 01 de Agosto de 2004.

CBS - Aparas de papel. **Reciclagem de papelão**, 2019. Disponível em: <<https://www.cbsaparasdepapel.com.br/reciclagem-papelao>>. Acesso em: 20 de agosto de 2019.

COLIN, E. C. **Pesquisa Operacional: 170 aplicações em estratégia, finanças, logística, produção, marketing e vendas**. Rio de Janeiro: LTC, 2007

COMERCIAL IGARA. **Contrato de parceria comercial: como fazê-lo com fornecedores**. Blog do comercia Igara, 24 de março de 2017. Disponível em: <<http://blog.comercialigara.com.br/contrato-de-parceria-comercial>>. Acesso em: 21 de agosto de 2019.

CONSELHO NACIONAL DO MEIO AMBIENTE - CONAMA. **Gestão de Resíduos Sólidos da Construção Civil (RSCC)**. 2002. Art. nº 3 – Lei Federal

CSCMP - COUNCIL OF SUPPLY CHAIN MANAGEMENT PROFESSIONALS. **Supply chain and logistics terms and glossary**, 2005.

Disponível em: <<http://www.cscmp.org/Terms/glossary03.htm>> Acesso em: 10 de setembro 2019.

DASGUPTA, S.; PAPADIMITRIOU, C. H.; VAZIRANI, U. V. **Algorithms**. Edição 1, McGraw-Hill Higher Education. 18 de julho de 2016.

DANTZIG, G. B.; FULKERSON, D. R.; JOHNSON, S. Solution of a Large-Scale Traveling-Salesman Problem. **Operations Research**. Vol. 2, pp. 365-462, 1954.

DANTZIG, G. B.; RAMSER, J. H. THE TRUCK DISPATCHING PROBLEM. **Management Science**. Vol. 6, n.1, 1 de outubro de 1959.

DESAULNIERS, G.; SPOORENDONK, S.; DESROSIERS, J. **The Vehicle Routing Problem with Time Windows: State-of-the-Art Exact Solution Methods**. Wiley Encyclopedia of Operations Research and Management Science. 15 de fevereiro de 2011.

DESROCHERS, M.; LENSTRA, J. K.; SAVELSBERGH, M. W. P.; SOUMIS, F. Vehicle Routing with Time Windows: Optimization and Approximation. In: GOLDEN, B. L.; ASSAD, A. A. **Vehicle Routing: Methods and studies**. Studies in management science and systems. Vol. 16, pp. 65-85, 1988.

DRIDI, I. H.; KAMMARTI R.; BORNE, P.; KSOURI M. Multi-objective Optimization For The Dynamic Multi-Pickup and Delivery Problem with Time Windows. **International Journal of Computers, Communications & Control (IJCCC)**, VI (2):246-257. Julho de 2011. Disponível em: <<https://arxiv.org/ftp/arxiv/papers/1101/1101.3396.pdf>>. Acesso em 15 de setembro de 2019.

EUR LEX. **Commission Implementing Regulation (EU) 2019/290 of 19 February 2019 establishing the format for registration and reporting of producers of electrical and electronic equipment to the register (Text with EEA relevance).** 19 de fevereiro de 2019. Disponível

em: <http://data.europa.eu/eli/reg_impl/2019/290/oj>. Acesso em 20 de outubro de 2019.

EUROPEAN COMISSION. **Roadmap to a Resource Efficient Europe. Bruxelas, 2011.** Communication from the commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions. Disponível em: <<https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:52011DC0571&from=EN>>. Acesso em: 28 de março de 2020.

FALL A. **Planification des activités en logistique inverse : modélisation et optimisation des performances par une approche stochastique en programmation linéaire.** Political Science, France, 2016.

FGV/IBRE. **Sondagem da Construção.** Instituto Brasileiro de Economia, agosto de 2019. Disponível em: <https://portalibre.fgv.br/data/files/59/70/A9/29/9D2DC6102B1DEBC68904CBA8/Sondagem%20da%20Constru_o%20FGV_press%20release_Ago19.pdf>. Acesso em: 11 de novembro de 2019.

FOURER, R.; GAY, D. M.; KERNIGHAN, B. W. **AMPL: A Modeling Language for Mathematical Programming**, Version 2003.

FURTADO, M. S.; MUNARI, P.; MORABITO, R. Pickup and delivery problem with time windows: a new compact two-index formulation. **Operations Research Letters**. Vol. 45, pp. 334-341, 2017.

GAN, X.; WANG, Y.; LI, S.; NIU, B. Vehicle Routing Problem with Time Windows and Simultaneous Delivery and Pick-Up Service Based on MCPSO. **Mathematical Problems in Engineering**, Hindawi Publishing Corporation, 2012.

LAPORTE, G.; NOBERT, Y. Exact Algorithms for the Vehicle Routing Problem. **North Holland Mathematics Studies**. Vol. 132, 1987. Pp. 147 – 184.

GOMES, D.; PERREIRA, F.; FILGUEIRA, B. **Heurística de Inserção em Grafos na resolução do Problema do Caixeiro Viajante**. GRAFOS, 2013. Disponível em: <<https://prezi.com/poelhh97vjxd/heuristica-de-insercao-em-grafos-na-resolucao-do-problema-do/>>. Acesso em 19 de setembro de 2019.

GVR - Grand View Research. **Modular Construction Market Size, Share & Trends Analysis Report By Type (Permanent, Relocatable), By Application (Residential, Commercial, Industrial, Healthcare), By Region, And Segment Forecasts, 2019 – 2025**. Market Research Report, setembro de 2019.

HASHIMOTO, H.; YAGIURA, M.; IBARAKI, T. An iterated local search algorithm for the time-dependent vehicle routing problem with time Windows. **Discrete Optimization**. Vol. 5, pp. 434-456, 2008.

HOMY QUÍMICA. **Canteiro de obras, você sabe como organizar?** 21 de janeiro de 2019. Disponível em: <<http://blog.homyquimica.com.br/canteiro-de-obras-voce-sabe-como-organizar/>>. Acesso em: 19 de agosto de 2019.

HOORNWEG, D.; BHADA-TATA, P. **What a waste: a global review of solid waste management**. World Bank, Washington, DC, 2012.

HOOS, H. H.; STUTZLE, T. **Stochastic Local Search: Foundations and Applications**. Morgan Kaufmann, 2005.

IASSINOVSKAIA, G.; LIMBOURG, S.; RIANE, F. The inventory-routing problem of returnable transport items with time windows and simultaneous pickup and delivery in closed-loop supply chains. **International Journal of Productions Economics**. Vol.183, 2017.

IRNICH, S.; TOTH, P.; VIGO, D. The Family of Vehicle Routing Problems. In: TOTH, V.; VIGO, D. **Vehicle Routing Problems, Methods, and Applications**

Second Edition. Chapter 6, pp. 01-23. The Society for industrial and Applied Mathematics and Mathematical Optimization Society, 2014.

ISMAIL, S. B.; LEGRAS, F, COPPIN, G. **Synthèse du problème de routage de véhicules.** Collection des rapports de recherche de Télécom Bretagne, 2011. Disponível em : https://portail.telecom-bretagne.eu/publi/public/fic_download.jsp?id=5745>. Acesso em: 5 de setembro de 2019.

JOHN, V.M. **Reciclagem de resíduos na construção civil – contribuição à metodologia de pesquisa e desenvolvimento.** São Paulo, 2000. pp. 102 Tese (livre docência) – Escola Politécnica, Universidade de São Paulo.

KLEE, V.; MINTY, G. J. How Good is the Simplex Algorithm? In: SHISHA, O.; Inequalities III. **Academic Press**, pp. 159-175, New York, 1972.

LACERDA, Leonardo. **Logística Reversa: Uma visão sobre os conceitos básicos e práticas operacionais LGC.** Disponível em: <https://docplayer.com.br/4272537-Logistica-reversa-uma-visao-sobre-os-conceitos-basicos-e-as-praticas-operacionais.html> >. Acesso em: 19 de agosto de 2019.

LAPORTE, G. SEMET, F. Classical heuristics for the capacitated VRP. In: TOTH, V.; VIGO, D. **Vehicle Routing Problems.** Vol. 9, Chapter 5, pp. 109-128. The Society for industrial and Applied Mathematics and Mathematical Optimization Society, Philadelphia, 2002.

LA PLATEFORME DU BATIMENT. **Gagner du temps sur vos chantiers.** Disponível em: <https://www.laplateforme.com/cms/c/342/>>. Acesso em: 19 de agosto de 2019.

- LAU, H. C.; LIANG, Z. **Pickup and delivery problem with time windows: Algorithms and test case generation.** *International Journal on Artificial Intelligence Tools*. 11, (3), 455-472. Research Collection School of Information Systems. Setembro de 2002. Disponível em: <https://ink.library.smu.edu.sg/sis_research/163>. Acesso em: 6 de setembro de 2019.
- LEROUVILLOIS, P.; CIREDEM, I.; HARGOUS, C; RAMDANE, W; DECK, O.; VIAN, J.; BISAGA, H. **PROJET INNOVATIONS LOGISTIQUES CHANTIERS.** ADEME (Agence de l'Environnement, et de la Maîtrise de l'Energie), pp. 13-22, fevereiro de 2019. Disponível em: <<https://www.ademe.fr/projet-innovations-logistiques-chantiers>>. Acesso em: 14 de agosto de 2019.
- LIONG, C. Y.; WAN, I.; OMAR, K. **VEHICLE ROUTING PROBLEM: MODELS AND SOLUTIONS.** *Journal of Quality Measurements and Analysis*. Vol. 4, pp. 205-218. Janeiro de 2008.
- LOPES, S. A. **Métodos Finitos em Matemática.** Mestrado em matemática para professores, cap. 3, pp. 72-115. Faculdade de ciências do Porto. Porto, Portugal, 2009.
- LU, Q.; DESSOUKY, M. M. A new insertion-based construction heuristic for solving the pickup and delivery problem with time windows. **European Journal of Operational Research**. Vol. 175, Issue 2, 1 dezembro de 2006, pp. 672-687.
- LU, Q.; DESSOUKY, M. M. An exact algorithm for the multiple vehicle pickup and delivery problem. **Transportation Science**. Vol. 38, pp. 503-514. 01 de novembro de 2004.
- MALAPERT, A. **Optimisation de tournées de véhicules pour l'exploitation de Réseau Telecom.** Fonction Groupe Technologie & Innovation – France Télécom R&D, setembro de 2006. Disponível em: <<http://www.i3s.unice.fr/~malapert/publications/malapert-06-FT.pdf>>. Acesso em: 17 de setembro de 2019.

MEDIUM. **Les plateformes d'économie circulaire dans le BTP**. 21 de setembro de 2017. Disponível em : <<https://medium.com/makesense-stories/les-plateformes-d%C3%A9conomie-circulaire-dans-le-btp-82a19789d251>>.

Acesso em: 19 de agosto de 2019.

MERSMANN, O.; BISCHL, B.; BOSSEK, J.; TRAUTMANN, H.; WAGNER, M.; NEUMANN, F. **Local Search and the Traveling Salesman Problem: A Feature-Based Characterization of Problem Hardness**. Learning and Intelligent Optimization - 6th International Conference, LION 6, Paris, France, pp. 115-129. Janeiro de 2012.

MÉTODOS EXATOS – Cursos avançados para análise de dados. **O que são Métodos Exatos**. 11 de junho de 2019. Disponível em: <<https://www.metodosexatos.com/o-que-sao-metodos-exatos/>>. Acesso em: 4 de setembro de 2019.

MOBUSS CONSTRUÇÃO. **Construção modular: quais os benefícios dessa tendência para o setor?** Blog Mobuss construção, 7 de fevereiro de 2019. Disponível em: <<https://www.mobussconstrucao.com.br/blog/construcao-modular/>>. Acesso em: 12 de novembro de 2019.

MURPHY, P.; POIST, R. Management of logistical retromovements: an empirical analysis of literature suggestions. **Transportation Research Forum**. Vol. 29, pp. 177-184, 1989.

NAGY, G.; WASSAN, N. A.; SPERANZA, M. G.; ARCHETTI, C. The Vehicle Routing Problem with Divisible Deliveries and Pickups. **Transportation Science**, 2013. Vol. 49, num. 2. Pp. 271-294.

NAKAMURA, J. **TENDÊNCIAS DA CONSTRUÇÃO CIVIL**. Buildin – Construção & Informação, 07 de outubro de 2019. Disponível em: <<https://www.buildin.com.br/tendencias-da-construcao-civil-2/>>. Acesso em 11 de novembro de 2019.

NASCIMENTO, D.; FIGUEIREDO, J.; CASALI, R.; MAYERLE, S. **Análise Comparativa de Algoritmos Heurísticos para Resolução do Problema do Caixeiro-Viajante em Grafos Não Clusterizados**. XXIV Encontro Nac. de Eng. de Produção - Florianópolis, SC, Brasil, 03 a 05 de nov. de 2004. Disponível em: http://abepro.org.br/biblioteca/ENEGEP2004_Enegep0601_0567.pdf.

Acesso em 19 de setembro de 2019.

NESMACHNOW, S. An overview of metaheuristics: Accurate and efficient methods for optimization. **International Journal of Metaheuristics**. Vol. 3, num. 4, 2014.

NETO, J. A. **Sustentabilidade & Produção: Teoria e prática para uma gestão sustentável**. 2011 - Cap. 6: Green Supply Chain Management: Princípios e aplicações, São Paulo.

PALMER, K.; DESSOUKY, M. M.; ABDELMAGUID, T. Impacts of management practices and advanced technologies on demand responsive transit systems. **Transportation Research Part A: Policy and Practice**. Vol. 38, pp. 495-509. Agosto de 2004.

PARRAGH, S. N.; DOERNER, K. F.; HARTL, R. F. A Survey on Pickup and Delivery Problems. Part I: Transportation between Customers and Depot. **Journal für Betriebswirtschaft**. Vol. 58, pp. 21-51, 2008a.

PARRAGH, S. N.; DOERNER, K. F.; HARTL, R. F. A Survey on Pickup and Delivery Problems. Part II: Transportation between Pick-up and Delivery Locations. **Journal für Betriebswirtschaft**. Vol. 58, pp. 81-117, 2008b.

PIRLOT, M. General local search methods. **European Journal of Operational Research**. Vol. 92, pp. 493-511, 1996.

PORTER, M. Competitive Advantage creating and sustaining superior performance with a new introduction. **Free Press**. Vol. 1, 1 de junho de 1998.

POTENZA, F. **O que é Construção Modular e como ela viabiliza seu negócio**. Blog HT construção civil, 2 de junho de 2017. Disponível em: <http://htconstrucao.com.br/construtora/construtora/oque-e-construcao-modular/>>. Acesso em: 12 de novembro de 2019.

PRINS, C. **Méthodes "route-first, cluster-second" pour les tournées de véhicules**. Laboratoire d'Optimisation des Systèmes Industriels (LOSI). Março de 2012. Disponível em : <http://www.lamsade.dauphine.fr/~jfro/JourneesPrecedentes/anciennesJournees/docs/jfro26/expose-1.pdf>>. Acesso em: 18 de setembro de 2019.

REGO, C.; GAMBOA, D.; GLOVER, F.; OSTERMAN, C. Traveling salesman problem heuristics: Leading methods, implementations and latest advances. **European Journal of Operations Research**. Vol. 211, pp. 427-441, 2011.

ROPKE, S. **Heuristic and exact algorithms for vehicle routing problems**. DTU Orbit, Chapter 3, pp. 25-35, 2006.

SALES, V. **PESQUISA OPERACIONAL APLICADA A LOGÍSTICA**. Faculdade Integrada Tiradentes, cap. Fundamentos da pesquisa operacional, pp. 4.; 2016. Disponível em: http://www.valdick.com/files/PO_aplicada_a_logistica.pdf>. Acesso em: 18 de agosto de 2019.

SANTIAGO, J. R. **Otimizar recursos! A busca pela eficiência contra a eficácia**. Fundação Vanzolini blog, 12 de julho de 2017. Disponível em: <https://vanzolini.org.br/weblog/2017/07/12/otimizar-recursos-busca-pela-eficiencia-contra-eficacia/>>. Acesso em: 18 de agosto de 2019.

SASSI, O.; CHERIF- KHETTAF, W. R.; OULAMARA, A. **A local search approach for the Two-Echelon Multi-products Location-Routing problem with**

Pickup and Delivery. IFAC-PapersOnLine, Elsevier, 2015, 15th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2015, 48 (3), pp.193-199. Disponível em: <10.1016/j.ifacol.2015.06.080>. <hal-01306993>.

SAVELSBERGH, M. W. P.; SOL, M. The General Pickup and Delivery Problem. **Transportations Science**. Vol. 29, n. 8, pp. 17-29. Fevereiro de 1995.

SIMCHI-LEVI, D.; CHEN, X.; BRAMEL, J. The PRV with Time Window Constraints. In: The Logic of Logistics. Springer Series in **Operations Research**, edition 2, pp. 257-273. Springer, New York, NY, 2005.

SISTEMA FIRJAN. **CONSTRUÇÃO CIVIL: DESAFIOS 2020**. Julho de 2014. Disponível em: <<https://www.firjan.com.br/construcao-civil/desafios.htm>>. Acesso em: 12 de novembro de 2019.

SMADJA, E. Immobilier “alternatif”. Des entrepôts démontables et relocalisables ! **Supply Chain Magazine**, n. 86, pp. 54, Julho - Agosto de 2014. Disponível em: <<http://supplychainmagazine.fr/TOUTE-INFO/Archives/SCM086/DOSSIER-86.pdf>>. Acesso em: 11 de novembro de 2019.

SOUZA, F. A. **O Caixeiro Viajante que pode ganhar um milhão de dólares**. Deviante, 22 de maio de 2018. Disponível em: <<https://www.deviante.com.br/noticias/ciencia/o-caixeiro-viajante-que-pode-ganhar-um-milhao-de-dolares/>>. Acesso em 22 de agosto de 2019.

SOYSAL, M. Closed-loop Inventory Routing Problem for returnable transport items. Transportation Research Part D. **Operations Management**. Vol.48, pp. 31-45, Ankara, Turkey, 2016.

SRIVASTAVA, S. Green supply-chain management: a state-of-the-art literature review. **International Journal of Management Reviews** 2007. Vol. 9, pp. 53-80, 2007.

TAVARES, L.; LOPES, H.; LIMA, C. **Construction and Improvement Heuristics applied to the Capacitated Vehicle Routing Problem**. World Congress on Nature & Biologically Inspired Computing, NaBIC 2009, 9-11, 2009.

TURNER & TOWNSEND. **International construction market survey 2019**. 2019. Disponível em : <http://www.turnerandtownsend.com/en/perspectives/international-construction-market-survey-2019/>>. Acesso em: 12 de novembro de 2019.

VENHUIS, M. **The basics of search algorithms explained with intuitive visualisations**. Towards Data Science, fevereiro de 2012. Disponível em: <https://towardsdatascience.com/around-the-world-in-90-414-kilometers-ce84c03b8552>>. Acesso em: 12 de novembro de 2019.

WASSAN, N. A.; NAGY, G. Vehicle Routing Problem with Deliveries and Pickups: Modelling Issues and Meta-heuristics Solution Approaches. **International Journal of Transportation**. Vol.2, No.1, pp.95-110, 2014.

ZHAO, X. **Une méthode génétique pour la résolution du problème dynamique de routage de véhicules avec temps de parcours variables**. Université d'Artois - Laboratoire de Génie Informatique et d'Automatique, dezembro de 2008. Disponível em: https://www.lgi2a.univ-artois.fr/spip/IMG/pdf/these_zhao_xin.pdf>. Acesso em: 18 de setembro de 2019.