

UNIVERSIDADE FEDERAL DO PARANÁ

FERNANDO DE PAULA OLIVEIRA

OTIMIZAÇÃO DO PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS
COM USO DA HEURÍSTICA DE CLARKE E WRIGHT E ALGORITMOS
GENÉTICOS

CURITIBA

2022

FERNANDO DE PAULA OLIVEIRA

OTIMIZAÇÃO DO PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS
COM USO DA HEURÍSTICA DE CLARKE E WRIGHT E ALGORITMOS
GENÉTICOS

TCC apresentado ao curso de Graduação de Engenharia de Produção, Setor de Tecnologia, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Bacharel em Engenharia de Produção.

Orientador(a): Prof(a). Dr(a). Mariana Kleina

CURITIBA

2022

AGRADECIMENTOS

Agradeço primeiramente e principalmente aos meus pais, que sempre colocaram a educação dos filhos como prioridade. Ser o primeiro da família a se formar em uma universidade não é uma conquista de uma pessoa só. Estendo os agradecimentos ao meu irmão, meu maior incentivador e uma pessoa que sei que posso contar em todos os momentos.

Agradeço também todos os professores que passaram pelo meu caminho e tornaram essa conquista possível, mas principalmente à Prof. Dr. Mariana Kleina por sempre estar disposta a ajudar desde os primeiros meses de curso e chegando até o final da graduação agora como minha orientadora, muitíssimo obrigado.

Por fim agradeço aos meus colegas e a todos que pude compartilhar experiências durante a faculdade.

RESUMO

A logística é uma das áreas que está no centro das atenções atualmente, uma boa operação logística pode ser considerada uma grande vantagem competitiva. Em se tratando de transporte e otimização de rotas, o Problema de Roteamento de Veículos (PRV) representa a melhor forma de atender vários pontos e o Problema de Roteamento de Veículos Capacitados (PRVC) é uma forma de PRV onde diferentes veículos suprem a demanda de clientes, respeitando suas capacidades de cargas. Existem softwares capazes de encontrar a solução ótima, mas tanto o custo quanto o tempo para se obter a solução acabam se tornando barreiras para a utilização. Uma diferente forma de encontrar uma solução viável são os métodos heurísticos, que são algoritmos criados fornecem uma solução um tempo consideravelmente melhor. O objetivo desse trabalho foi implementar a heurística de Clarke e Wright (C&W) e, depois de se obter a solução, implantar o Algoritmo Genético (AG), utilizado neste trabalho como método de melhoria. E então, testar diferentes problemas e comparar tanto as soluções obtidas quanto o tempo computacional quando utilizado um software de otimização como o Gurobi. Os resultados foram comparados entre si levando em consideração os aspectos acima. Nota-se a grande vantagem do uso de heurísticas, tanto em relação a qualidade da solução quanto em tempo computacional, em comparação ao uso de softwares de otimização para resolução de grandes problemas.

Palavras-chave: Otimização. Heurísticas. PRVC. Logística. Roteamento de Veículos.

ABSTRACT

Logistics is one of the areas that is currently in the spotlight, a good logistics operation can be considered a great competitive advantage. When it comes to transport and route optimization, the Vehicle Routing Problem (VRP) represents the best way to serve several points and the Capacitated Vehicle Routing Problem (CVRP) is a form of VRP where different vehicles supply the demand of customers, respecting their load capacities. There are software capable of finding the optimal solution, but both the cost and the time to obtain the solution end up becoming barriers to use. A different way of finding a viable solution is heuristic methods, which are algorithms created that provide a solution in a considerably better time. The objective of this work was to implement the Clarke and Wright (C&W) heuristic and, after obtaining the solution, to implement the Genetic Algorithm (GA), used in this work as an improvement method. And then, test different problems and compare both the solutions obtained and the computational time when using an optimization software such as Gurobi. The results were compared with each other considering the above aspects. There is a great advantage in the use of heuristics, both in relation to the quality of the solution and in computational time, compared to the use of optimization software for solving large problems.

Keywords: Optimization. Heuristics. CVRP. Logistics. Vehicle Routing.

LISTA DE FIGURAS

FIGURA 1 - Ilustração do jogo de Hamilton	14
FIGURA 2 - Representação da combinação de clientes em um roteiro	19
FIGURA 3 - Fluxograma da representação dos Algoritmos Genéticos	22
FIGURA 4 - Representação de um cromossomo	23
FIGURA 5 - Representação de uma população	24
FIGURA 6 - Cruzamento com diferentes pontos de corte	25
FIGURA 7 - Fluxograma da Metodologia	29
FIGURA 8 - Rotas CVRPLIB - Problema A-n32-k5	33
FIGURA 9 - Rotas C&W - Problema A-n32-k5	34
FIGURA 10 - Rotas Gurobi - Problema A-n32-k5	34
FIGURA 11 - Rotas CVRPLIB – Problema P-n16-k8	36
FIGURA 12 - Rotas C&W - Problema P-n16-k8	36
FIGURA 13 - Rotas Gurobi - Problema P-n-16-k5	37
FIGURA 14 - Rotas CVRPLIB - Problema A-n45-k7	38
FIGURA 15 - Rotas C&W - Problema A-n45-k7	39
FIGURA 16 - Rotas Gurobi – Problema A-n45-k7	39

LISTA DE TABELAS

Tabela 1 - Resultados problema A-n32-k5	31
Tabela 2 - Resultados problema P-n16-k8	31
Tabela 3 - Resultados problema A-n45-k7	31
TABELA 4 - Rotas CVRPLIB – Problema A-n32-k5	32
TABELA 5 - Rotas C&W - Problema A-n32-k5	33
TABELA 6 - Rotas Gurobi - Problema A-n32-k5	33
TABELA 7 - Rotas CVRPLIB – Problema P-n16-k8	35
TABELA 8 - Rotas C&W – Problema P-n16-k8	35
TABELA 9 - Rotas Gurobi - Problema P-n16-k8	35
TABELA 10 - Rotas CVRPLIB – Problema A-n45-k7	37
TABELA 11 - Rotas C&W – Problema A-n45-k7	38
TABELA 12 - Rotas Gurobi – Problema A-n45-k7	38

LISTA DE QUADROS

QUADRO 1 - Analogia de termos usados na biologia e no AG.....	23
---	----

SUMÁRIO

1 INTRODUÇÃO	10
2 REVISÃO DE LITERATURA	13
2.1 CAIXEIRO VIAJANTE	13
2.2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS.....	16
2.3 CLARKE E WRIGHT	18
2.4 ALGORITMO GENÉTICO	20
3 METODOLOGIA	27
4 RESULTADOS.....	30
4.1 RESULTADOS COMPUTACIONAIS	30
4.2 ROTAS ENCONTRADAS.....	32
4.2.1 Problema A-n32-k5	32
4.2.2 Problema P-n16-k8	35
4.2.3 Problema A-n45-k7	37
5 CONSIDERAÇÕES FINAIS	41
REFERÊNCIAS.....	42

1 INTRODUÇÃO

Devido ao crescimento de consumo nos últimos meses, a logística está em foco. De acordo com o relatório da Mastercard *SpendingPulse* (2021), um indicador de vendas no varejo, que inclui pagamentos de todos os tipos em uma série de mercados globais, o e-commerce, por exemplo, cresceu 75% durante o ano de 2020, sendo a logística parte essencial do processo de vendas online. A operação logística acaba sendo considerada uma vantagem competitiva, já que uma entrega não realizada no prazo esperado pode definir se o cliente vai ou não realizar uma compra nessa companhia novamente. Como absorver esse aumento no número de vendas na operação logística acaba por ser o principal desafio, algumas decisões acabam passando pelas companhias, como a expansão de centros de distribuição e o atendimento a áreas antes não atendidas.

Segundo Bowersox e Closs (2001), o objetivo da logística é tornar disponíveis os produtos e serviços nos locais onde são necessários, quando são desejados. Segundo os autores, a logística envolve a integração de informação, transporte, estoque, armazenamento, manuseio de matérias e embalagem. Sendo “estar no local requisitado com o menor custo possível” o principal objetivo, já que o custo com a logística pode chegar em até 35% do valor das vendas.

Esse trabalho consiste em explorar o Problema de Veículos Capacitados, aplicando técnicas a bases de dados, onde existem um centro de distribuição e uma rede de clientes que precisam receber uma entrega apenas uma vez, com restrições com relação a capacidade dos veículos e outras possíveis restrições, com o objetivo de minimizar as distâncias percorridas pelos veículos. E então, obter a solução ótima com o uso de softwares já conhecidos para que em seguida seja avaliada a eficiência da heurística de Clarke e Wright para obtenção de soluções aproximadas e a atuação do Algoritmo Genético para melhoria de soluções intermediárias, evitando possíveis problemas com ótimos locais.

Existem alguns softwares conhecidos que são capazes de resolver o mesmo tipo de problema trazendo a solução ótima, como por exemplo Gurobi e o Lingo, mas os principais ofensores para utilização acabam por ser o tempo de resposta (que pode ser relativamente alto), a necessidade de uma grande memória computacional e a dificuldade em conseguir fornecer as informações referentes aos problemas dentro de tais softwares, já que não existe uma interface definida onde simplesmente os dados

são adicionados. Os problemas precisam ser imputados por arquivo de texto, com um preenchimento padrão, o que acaba por exigir em alguns casos o conhecimento em mais do que uma linguagem de programação.

O objetivo desse trabalho é desenvolver um programa na linguagem VBA (*Visual Basic Application*) na ferramenta Microsoft Excel capaz de resolver Problemas de Veículos Capacitados pela heurística de Clarke e Wright e depois de se obter a solução implementar o Algoritmo Genético como método de melhoria e testar sua eficiência quando comparado com softwares já conhecidos, avaliando tempo computacional e a solução atingida. Espera-se que a diferença entre a solução ótima e a solução encontrada seja suficientemente pequena para que não haja prejuízo em considerar o uso da heurística proposta. Também é esperado que a ferramenta desenvolvida tenha fácil usabilidade pelos usuários.

A pesquisa e o desenvolvimento de uma ferramenta para obter a solução de um Problema de Veículos Capacitados em um programa já conhecido pelos estudantes (Microsoft Excel) pode ter um impacto positivo na rotina dos acadêmicos, já que por muitas vezes é encontrada uma dificuldade no domínio dos softwares capazes de encontrar a solução ótima de tais problemas, sendo que os softwares não são gratuitos e demandam grande esforço e tempo computacional.

A metodologia consistirá na construção da ferramenta capaz de resolver os problemas pelo método de Clarke e Wright e Algoritmo Genético para melhoria da solução. Na criação da ferramenta se espera a construção de um programa da forma mais otimizada possível para que o tempo computacional seja reduzido, além disso, a implantação do Algoritmo Genético após a solução obtida pela heurística será realizada esperando que haja uma melhoria na resposta. E por fim, será realizada uma comparação com resultados e tempos computacionais já obtidos, justificando o uso da nova ferramenta.

Esse trabalho tem cinco capítulos, sendo o primeiro deles a introdução, onde é apresentado o tema e é mostrado de forma breve o objetivo do trabalho. O segundo capítulo é a revisão bibliográfica, onde será apresentado o Problema do Caixeiro Viajante, o Problema de Roteamento de Veículos, o Problema de Roteamento de Veículos Capacitados, a heurística de Clarke e Wright e por fim explicado como funciona o Algoritmo Genético e como ele foi usado no trabalho. No terceiro capítulo apresentará a metodologia do trabalho, onde será mostrado o método adotado para chegar aos resultados. O quarto capítulo tem o objetivo de apresentar os resultados,

do PRVC aplicado a três problemas da literatura, comparando a solução ótima com a solução encontrada pela heurística. No quinto e último capítulo apresentará as considerações finais do autor e sugestões para os próximos trabalhos.

2 REVISÃO DE LITERATURA

Esse capítulo apresenta a fundamentação teórica para melhor entendimento do trabalho e estrutura foi realizada da seguinte forma: primeiramente é apresentado o Problema do Caixeiro Viajante (PCV) e então o Problema de Roteamento de Veículos Capacitados (PRVC). Por fim são abordadas técnicas empregadas para a resolução do PRVC, sendo elas a heurística de Clarke e Wright e o Algoritmo Genético.

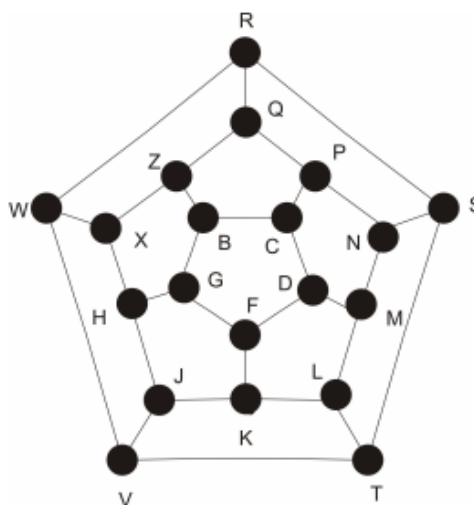
2.1 CAIXEIRO VIAJANTE

O Problema do Caixeiro Viajante (PCV) é um problema clássico da pesquisa operacional e consiste em encontrar a menor rota a ser percorrida a fim de visitar todas as cidades (pontos) apenas uma única vez com o menor custo possível. Segundo Belfiore (2006) a origem do PCV vem de Willian Rowan Hamilton que em 1859 criou um jogo que tinha como objetivo encontrar uma rota passando pelos vértices de um dodecaedro de forma que o início e o fim fossem no mesmo vértice, sem nunca repetir uma visita. Por isso, o caminho de sair de um vértice i passar por todos os outros vértices e retornar ao vértice i denominado ciclo hamiltoniano.

A distância entre as cidades i e j pode ser denominada como c_{ij} e o problema é denominado simétrico quando $c_{ij} = c_{ji}$. Segundo Arenales et al. (2011) o problema do caixeiro viajante é um dos problemas combinatórios mais conhecidos e pesquisados devido à sua aplicação em diversas áreas, tais como manufatura de circuitos, programação da produção, telecomunicações e sequenciamento de DNA.

A FIGURA 1 representa em um grafo como foi o jogo proposto por Hamilton.

FIGURA 1 - Ilustração do jogo de Hamilton



Fonte: Carretero e Rangel (2007)

A quantidade de rotas possíveis em um problema do caixeiro viajante é relacionada de forma fatorial ao número de pontos n . Considerando um PCV simétrico, o número de rotas é $k = \frac{(n-1)!}{2}$. Segundo Oliveira, Domingos e Junior (2001) os problemas de otimização combinatória podem ser classificados, segundo sua complexidade em P, NP, NP-difícil e NP-completo. Os problemas pertencentes à classe P são ditos tratáveis computacionalmente, já os demais são considerados intratáveis, mesmo considerando que a capacidade dos computadores tenha crescido consideravelmente nos últimos anos. O PCV pertence à classe de problemas NP-difíceis, ou seja, o tempo gasto para resolvê-lo é exponencial em relação ao tamanho da instância. Isso faz com que a resolução do problema por meio de métodos heurísticos ganhe maior importância, principalmente tratando-se de grandes instâncias.

Apesar da evolução computacional que se presencia ano após ano é fácil imaginar que, em alguns casos, listar todas as rotas possíveis seja bem demorado, e nesse cenário que surgem as heurísticas, que são métodos capazes de encontrar um resultado próximo do ótimo, mas em um tempo muito menor do que a solução ótima.

Segundo Arroyo (2002) existem três tipos de heurísticas, a primeira delas são as chamadas construtivas, que constroem uma solução pela adição de componentes por meio de regras específicas com a estrutura do problema. Já a segunda classe é a

chamada de busca local ou busca de vizinhança. E por fim, as metaheurísticas, que são métodos mais flexíveis, pois possuem uma estrutura de componentes genéticos que são adaptados ao problema que se busca resolver.

Existem algumas formas de modelar o Problema do Caixeiro Viajante, Arenales et al. (2011) formularam como um problema de programação binária:

Minimizar:

$$\sum_{i=1}^n \sum_{j>1}^n c_{ij} x_{ij} \quad (1)$$

Sujeito a:

$$\sum_{j<1}^n x_{ji} + \sum_{j>1}^n x_{ij} = 2, \quad i = 1, \dots, n \quad (2)$$

$$\sum_{i \in S} \sum_{\substack{j \notin S \\ j>i}}^n x_{ij} + \sum_{i \notin S} \sum_{\substack{j \in S \\ j>1}}^n x_{ij} \geq 2, \quad S \subset N, \quad 3 \leq |S| \leq \left\lceil \frac{n}{2} \right\rceil \quad (3)$$

$$x_{ij} \in \{0,1\} \quad (4)$$

Onde:

- x_{ij} : operador binário que define se o arco que liga as cidade i e j será utilizado
- c_{ij} : custo de percorrer o arco que liga a cidade i a cidade j
- S : subrotas
- N : conjunto de n cidades
- E : número de arestas entre as cidades

Considerando um grafo não orientado $G = (N, E)$ e supondo que c é um grafo completo, ou seja, para qualquer par de cidades existe uma aresta que as liguem. A função objetivo (1) expressa a minimização da rota, dado o custo c_{ij} para percorrer o arco, o operador binário x_{ij} vai assumir o valor 1 quando essa rota for escolhida e será multiplicado pelo custo da rota. Fazendo esse processo para todas as rotas, tem-se o valor total da operação. A primeira restrição (2) impõe que cada cidade tenha somente uma cidade sucessora imediata e uma cidade predecessora imediata. A

restrição (3) garante que não existam sub-rotas, a restrição garante que existe no mínimo duas arestas entre as cidades de S e as cidades não pertencentes a S . Por fim a restrição (4) garante que as variáveis são binárias, ou seja, só podem assumir o valor de 0 ou 1.

Segundo Cunha (2000) novas restrições vêm sendo incorporadas ao PCV para dar mais representatividade aos tipos de problema que são encontrados no mundo real, gerando variantes do PCV. Essas restrições envolvem: restrições de horário de atendimento (janelas de tempo); capacidade de veículos; frota compostas de veículos de diferentes tamanhos; duração máxima dos roteiros dos veículos (tempo ou distância) além de restrições do tipo de veículos para atender determinados clientes.

2.2 PROBLEMA DE ROTEAMENTO DE VEÍCULOS CAPACITADOS

O Problema de Roteamento de Veículos (PRV) consiste em encontrar rotas a serem percorridas por uma frota de veículos de forma que todos os pontos sejam visitados e suas demandas satisfeitas. E o objetivo é diminuir a distância ou diminuir o custo desse percurso como um todo.

Então, o Problema de Roteamento de Veículos pode ser descrito como um processo de definição das melhores rotas e itinerários, que por consequência define o caminho que chegará no resultado esperado ótimo pela função objetivo, sendo que para cada ligação entre um par de pontos há uma distância ou custo associado. A intenção é encontrar um conjunto de rotas que maximiza ou minimiza a função objetivo, respeitando as condições de restrição que dependem de cada situação (LIMA, 2015).

Uma das versões do Problema de Roteamento de Veículos é o Problema de Roteamento de Veículos Capacitados (PRVC) no qual clientes possuem uma demanda definida e veículos para atenderem a esta demanda. O problema pode ser com veículos homogêneos, onde todos os veículos têm a mesma capacidade ou veículos heterogêneos, onde as capacidades são diferentes. Nesse trabalho foi utilizado o modelo de PRVC com veículos homogêneos.

Segundo Oliveira e Delgado (2015) o Problema de Roteamento de Veículos Capacitados pode ser formalizado como um grafo não direcionado $G = (V, E)$, onde V é o conjunto de vértices e E é o conjunto de arestas. Existe um número de veículos K que serão usados para atender as demandas dos clientes e esses veículos devem ter uma capacidade máxima, que será uma restrição no problema. Cada vértice é um

ponto a ser visitado, então ele terá uma demanda associada. Também, cada uma das arestas tem um custo correspondente c_{ij} que é o custo de ir do vértice i ao vértice j . A solução ótima é encontrada quando todos os pontos são visitados pelo menos uma vez sem que as rotas excedam a sua capacidade máxima e com o menor custo possível.

Segundo Vieira (2013) a formulação matemática do PRVC pode ser apresentada da seguinte forma:

Minimizar:

$$\sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n \sum_{k=1}^K c_{ij} x_{ijk} \quad (5)$$

Sujeito a:

$$\sum_{k=1}^K \sum_{j=1}^n x_{0ij} = K \quad (6)$$

$$\sum_{j=1}^n x_{0jk} = \sum_{j=1}^n x_{0jk} = 1, \quad k = 1, \dots, K \quad (7)$$

$$\sum_{k=1}^K \sum_{j=0}^n x_{ijk} = 1, \quad i = 1, \dots, n \quad (8)$$

$$\sum_{j=0}^n x_{ijk} - \sum_{j=0}^n x_{jik} = 0, \quad k = 1, \dots, K, \quad i = 1, \dots, n \quad (9)$$

$$x_{ij} + v_{li} - v_{lj} \leq 1, \quad i, j, l = 2, \dots, n \quad (10)$$

$$x_{ij} - v_{ij} \leq 0, \quad i, j = 2, \dots, n \quad (11)$$

$$x_{ij} + v_{ji} \leq 1, \quad i, j = 2, \dots, n \quad (12)$$

$$\sum_{i=0}^n m_i \sum_{\substack{j=0 \\ j \neq i}}^n x_{jik} \leq C, \quad k = 1, \dots, K \quad (13)$$

$$x_{ijk} \in \{0,1\}, \quad i, j = 1, \dots, n, \quad k = 1, \dots, K \quad (14)$$

No qual:

- x_{ijk} : percurso do vértice i ao vértice j com o veículo k
- c_{ij} : custo de se percorrer o arco que liga a cidade i a cidade j
- K : total de veículo
- k : total de veículo
- S : total de clientes
- $v(S)$: número mínimo de veículos para atender S
- C : capacidade total do veículo
- m_i : demanda do cliente i

A restrição (6) garante que o total de veículos que saem do depósito ($i = 0$ representa o depósito) seja igual a K que é o total do número de veículos. A restrição (7) garante que cada rota tenha início e fim no depósito. As restrições (8) e (9) garantem que cada cliente seja visitado exatamente uma vez e que um veículo que chega em um cliente também saia dele. Enquanto as restrições (10), (11) e (12) garantem que não seja formado um subciclo que exclua algum dos depósitos, elas são uma extensão da formulação MTZ e foram apresentadas por Oncan, Altinel e Laporte (2007) após ser proposta inicialmente por Gouveia e Pires (1999). A variável adicional v_{ij} indica se o vértice i está no caminho do vértice 1 ao vértice j .

Para garantir que o número de veículos usados não seja inferior ao mínimo necessário, a restrição (13) estabelece de forma explícita que essa capacidade não seja excedida. E por fim, a restrição (14) garante que a variável x_{ijk} apenas assumam os valores 0 ou 1.

2.3 CLARKE E WRIGHT

A heurística de Clarke e Wright (C&W) surgiu em 1964 no campo da logística para buscar a otimização de rotas de transporte. No início do método só existem o depósito e outro vértice, de forma que em cada etapa, ocorre a combinação de duas rotas com base no maior ganho que pode ser gerado (LAPORTE, 1992). O algoritmo de C&W, que é denominado uma heurística construtiva, além da rapidez em termos de processamento, também traz uma capacidade de resolver problemas com diferentes restrições, sem afetar a qualidade das soluções no tempo de processamento (TEIXEIRA, 2002).

O objetivo do método é minimizar a distância percorrida o que de forma indireta diminui o número de veículos necessários para que as demandas de todos os clientes sejam satisfeitas. O método se baseia em busca de cenários de rotas que diminua o custo. Gama (2011) descreve que o primeiro teste deve ser o pior cenário, no qual o veículo visita um cliente de cada vez e depois retorna ao centro de distribuição (CD), o segundo teste é o veículo visitando um par de clientes para então retornar ao centro de distribuição.

A equação (15) representa o cálculo de economias e pode ser escrita como:

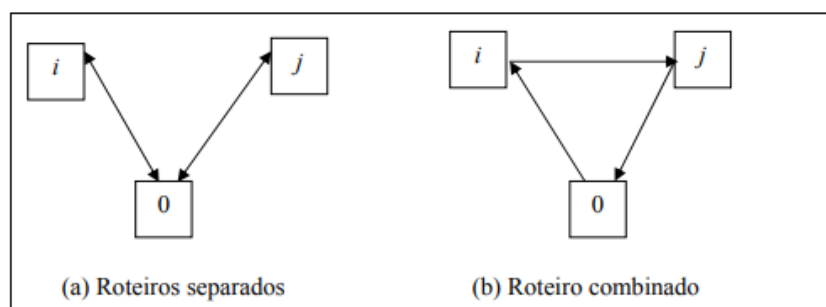
$$e_{ij} = d_{CD,i} + d_{CD,j} - d_{ij} \quad (15)$$

Onde:

- e_{ij} : *ganho da equação*
- $d_{CD,i}$: *distância do CD para o ponto i*
- $d_{CD,j}$: *distância do CD para o ponto j*
- d_{ij} : *distância entre os pontos i e j*

A FIGURA 2 representa a economia na combinação de clientes em uma única rota.

FIGURA 2 - Representação da combinação de clientes em um roteiro



Fonte: Belfiore (2006)

E segundo Gama (2011), o algoritmo pode ser dividido em 5 passos:

- **Passo 1:** Fazer a combinação dois a dois de todos os pontos e calcular a economia para cada combinação.

- **Passo 2:** Ordenar os ganhos em ordem decrescente.
- **Passo 3:** Iniciar a análise de pontos pelo par de maior ganho, e realizar de todos os pontos seguintes.

A partir desse momento, o algoritmo pode seguir um dos dois caminhos: A versão paralela, que efetua a melhor união factível ou a versão sequencial, que realiza a extensão máxima de uma rota (BELFIORE, 2006).

Versão paralela:

- **Passo 4:** Se a ligação dos pares i e j for uma rota viável e de acordo com a restrição, fazer a conexão, caso contrário, eliminar.
- **Passo 5:** Se ainda houver economia, tentar a união com o próximo par de nós da lista e voltar ao passo 4, caso contrário, o algoritmo termina.

Versão sequencial:

- **Passo 4:** Se a ligação dos pares i e j for uma rota viável de acordo com a restrição, fazer a conexão e ir para o passo 6, caso contrário, ir para o passo 5.
- **Passo 5:** Se a rota não puder ser estendida, terminar a rota e iniciar uma nova rota com o par de nós do passo 4.
- **Passo 6:** Repetir os passos 4 e 5 enquanto houver alguma economia na lista.

2.4 ALGORITMO GENÉTICO

Os Algoritmos Genéticos (AGs) são um método de busca que tem como base no processo de evolução das espécies de Charles Darwin (BJARNODÓTTIR, 2004). Os AGs são inspirados no modelo de seleção natural, provado pelo naturalista britânico Charles Darwin. O cientista foi primeiro a comprovar e explicar o mecanismo de seleção natural, quando em 1831 viajou ao mundo para coletar fósseis e plantas como bases para sua pesquisa, e com auxílio do estudo de Thomas Robert Malthus provou que quem sobrevive na natureza é quem consegue se adaptar melhor as mudanças de condições. Quem consegue se adaptar (e consequentemente,

sobreviver) se reproduz e “espalha” para seus descendentes os códigos genéticos que o fizeram sobreviver.

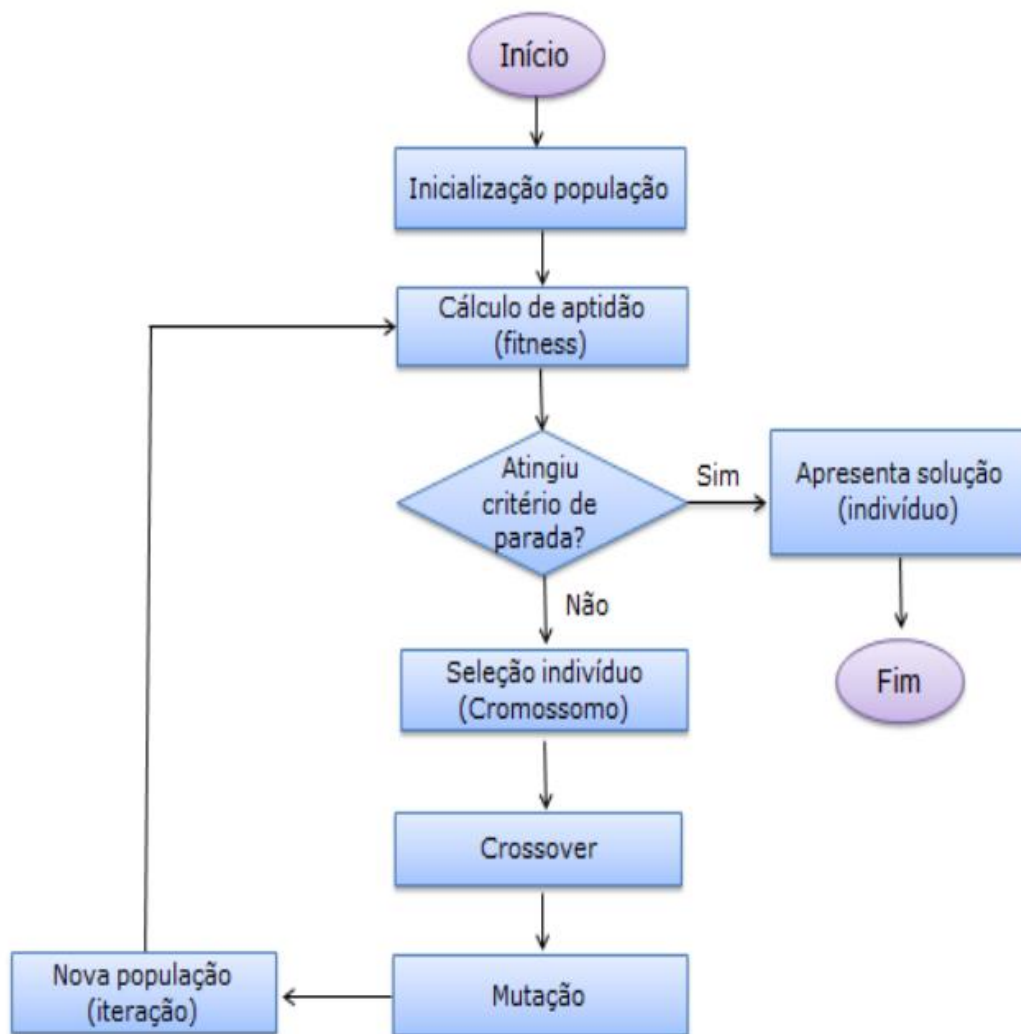
Com base na teoria evolutiva das espécies, John H. Holland, um pesquisador da Universidade de Michigan, propôs nos anos 60 a construção de um algoritmo matemático para otimização em sistemas complexos, sendo denominado de Algoritmo Genético e apresenta algumas vantagens que fazem o uso dele ter crescido muito nos últimos anos, como por exemplo (GOLDBERG, 1989):

- i - Não requer informações sobre o gradiente da superfície de resposta;
- ii - Eventuais descontinuidades da superfície de resposta não afetam a performance da otimização;
- iii - A presença de mínimos locais não reduz a eficiência do algoritmo;
- iv - A performance do algoritmo tem apresentado excelentes resultados para problemas de otimização de grande escala.

O algoritmo reproduz artificialmente o que foi proposto por Charles Darwin. Funciona da seguinte forma: Uma população é iniciada com um número definido de indivíduos. Para cada indivíduo se atribui um conceito de aptidão ou *fitness*, que indica o quão apto é o indivíduo dentro daquela população, o que fazendo uma conexão com a seleção natural, seria numerar a probabilidade desse indivíduo de sobreviver e se reproduzir. Desta forma, os indivíduos com as maiores aptidões são selecionados e passam suas características para a próxima geração, o que dentro do AG é denominado como uma iteração. Também, nesse momento, pode ser aplicado um método de *crossover* e de mutação dentro desses indivíduos, o que busca uma nova geração que seria ainda mais apta, o que no caso do AG, significa trazer melhores soluções. Esse procedimento segue até que o critério de parada seja atingido (LIMA, 2015).

A FIGURA 3 ilustra como o processo funciona.

FIGURA 3 - Fluxograma da representação dos Algoritmos Genéticos



Fonte: Lima (2015)

E o QUADRO 1 mostra a comparação dos termos na biologia com os termos usados na execução do algoritmo.

QUADRO 1 - Analogia de termos usados na biologia e no AG

Termos Biológicos	Algoritmo Genético
Cromossomo	Vetor de caracteres (solução)
Gene	Característica do problema
Alelo	Valor da característica
Loco	Posição no vetor
Genótipo	Estrutura
Fenótipo	Estrutura submetida ao problema
Indivíduo	Solução
Geração	Iteração, ciclo

Fonte: Pacheco (1999)

O cromossomo é composto por genes, que determinam a característica do cromossomo. Os genes na maioria das vezes são codificados como códigos binários, pelas vantagens computacionais que essa determinação proporciona.

No exemplo da FIGURA 4, o cromossomo tem 2 genes divididos em 8 posições, ou simplesmente 8 *bits*. Então os primeiros 4 *bits* representam uma característica e os próximos 4 *bits* uma outra característica. Cada um desses *bits* dentro do gene pode ser denominado como alelo.

FIGURA 4 - Representação de um cromossomo

0	0	1	0	1	1	0	0
---	---	---	---	---	---	---	---

Fonte: O autor (2022)

Apresentado o conceito do cromossomo, agora pode se apresentar o conceito de população, que pode ser definida como um conjunto de cromossomos. A população inicial pode ser iniciada aleatoriamente ou por um conjunto de soluções aproximadas já conhecidas. Neste trabalho, o Algoritmo Genético entrará como uma melhoria para a heurística de C&W, então a população inicial será a solução obtida

após a resolução pelo método de C&W. O número de indivíduos da população determina o tamanho da população. Uma população pequena pode não explorar todo o espaço de soluções enquanto uma população muito grande pode perder a eficiência pelo grande custo computacional. A FIGURA 5 representa uma população com 4 indivíduos.

FIGURA 5 - Representação de uma população

0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	1
0	0	0	0	1	0	1	0
0	0	0	0	0	1	0	0

Fonte: O autor (2022)

O *fitness* (ou aptidão) avalia a capacidade que o problema tem de oferecer uma solução satisfatória. Com o uso dela, alguns indivíduos são selecionados para passar pelo processo de recombinação e é com uso dela que é representado o ganho para um problema de maximização ou minimização. Quando é alcançado um número máximo de gerações, quando há uma perda de diversidade ou quando não houve convergência dos valores nas últimas gerações, se diz que se atingiu um critério de parada do algoritmo.

Para escolher os indivíduos que participarão do processo de reprodução é realizada uma seleção, essa seleção deve ser feita com base na aptidão de cada cromossomo de forma que os cromossomos com a maior aptidão tenham maior forma de serem escolhidos. Uma das formas de escolher os indivíduos é com o uso de uma roleta, que determinará a probabilidade de o cromossomo ser escolhido de acordo com a equação (16):

$$p_i = \frac{f_i}{\sum_{j=1}^n f_j} \quad (16)$$

Onde:

- p_i : probabilidade do indivíduo (cromossomo) i ser escolhido
- n : tamanho da população
- f_i : valor fitness do indivíduo i

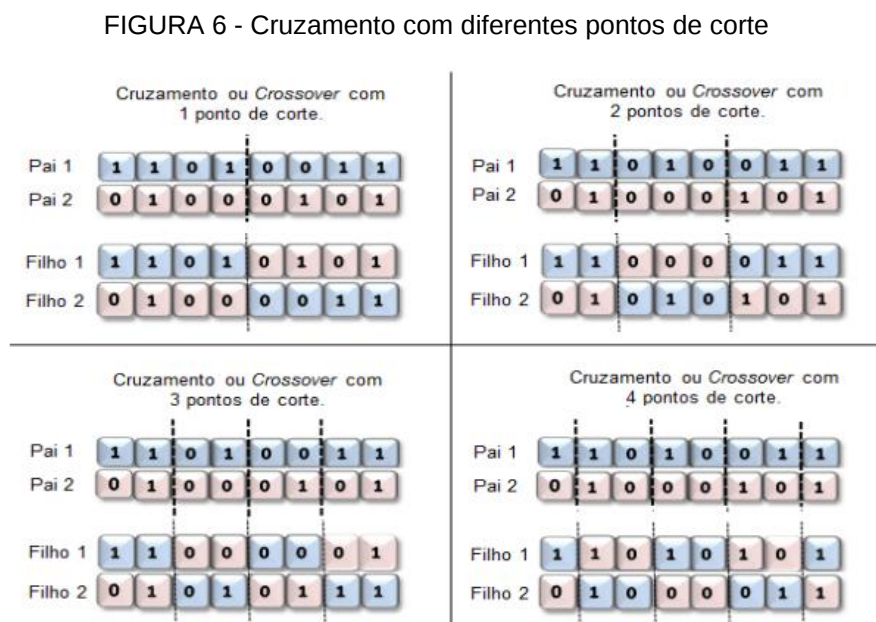
- $\sum_{j=1}^n f_j$: valor acumulado de fitness de todos os indivíduos da população

O sorteio é feito por meio de um jogo, onde a probabilidade de cada indivíduo ser incluído na população é proporcional ao seu *fitness*.

E por fim, os operadores genéticos, que são responsáveis por transformar a população por meio das iterações (ou gerações) até que se chegue em um resultado satisfatório. A população é diversificada mantendo as características que foram adquiridas nas gerações anteriores. Os principais são Cruzamento (*Crossover*) e Mutação.

O *crossover* consiste em cruzar informações dos 2 cromossomos escolhidos no processo de seleção e trocar materiais genéticos, trocar informações que estão na mesma posição, de forma a gerar novos indivíduos (filhos) que sejam melhores que os gerados (pais). A posição da troca de material genético é chamada de ponto de corte e dentro do processo de *crossover* podem existir 1 ou mais pontos de corte.

A FIGURA 6 ilustra o procedimento de cruzamento dos cromossomos considerando de 1 a 4 pontos de corte.



Fonte: Lima (2015)

E a mutação consiste em alterar de forma aleatória um dos genes do cromossomo para oferecer a possibilidade para que mais pontos sejam avaliados, a mutação evita a estagnação em ótimos locais. A mutação deve ser aplicada em uma

taxa não muito alta, caso contrário a busca ficará aleatória, o que dificulta a convergência.

Então, o AG pode ser resumido em 6 passos:

- 1) Gerar a população inicial;
- 2) Avaliar o *fitness*;
- 3) Avaliar o critério de parada;
- 4) Selecionar os indivíduos gerados da próxima geração;
- 5) Utilizar os operadores genéticos;
- 6) Substituir a população e voltar ao passo 2;

E ele só tem início após ser definido os seguintes parâmetros:

- Tamanho da população inicial;
- Tipo de seleção;
- Quantidade de pontos de cruzamento (se houver cruzamento);
- Taxa de mutação (se houver mutação);
- Critério de parada

3 METODOLOGIA

A metodologia adotada constituiu de levantamento biográfico, implementação computacional, aplicação em problemas práticos e avaliação de soluções. Na etapa de levantamento biográfico pesquisas foram realizadas a partir de estudos já publicados, como livros e artigos, para se familiarizar com as modelagens existentes e as técnicas utilizadas para resolução aproximada. Na parte de implementação computacional a heurística será implementada em VBA para encontrar soluções viáveis em Problema de Roteamento de Veículos Capacitados (PRVC) e o modelo matemático será usado para encontrar a solução exata utilizando o software Gurobi. Na etapa de aplicação em problemas práticos, será aplicada a heurística implementada em um problema em busca de uma solução viável e a busca pela solução exata usando o Gurobi. E por fim, a etapa de avaliar as soluções, tanto a encontrada pelo método de C&W, usando a versão paralela, junto com Algoritmo Genético quando a encontrada no Gurobi, avaliando tempos computacionais e a diferença entre a solução ótima e a solução aproximada.

Neste trabalho a pesquisa se iniciou definindo o Problema de Caixeiro Viajante e o Problema de Roteamento de Veículos, sendo um PRV um tipo de PCV. E dentro do PRV existem os problemas que contam com veículos com limitação de carga, sendo chamado de Problema de Roteamento de Veículos Capacitados. A heurística de C&W será aplicada para busca de soluções viáveis de um PRVC e contará com a ajuda do Algoritmo Genético, já que ela consegue fazer a busca em diferentes áreas do espaço solução por trabalhar com uma população e não com um único ponto, também garantindo a melhoria de cada solução por iteração. Tais métodos, quando utilizados da forma proposta nesse trabalho, acabam por diminuir o esforço computacional quando comparado a *softwares* que buscam a solução ótima.

Três problemas (também chamados de instâncias) da biblioteca de Problema de Veículos Capacitados da PUC - RJ (CVRPLIB, 2022) foram escolhidos para aplicação desta pesquisa. O portal disponibiliza a solução ótima, mesmo assim o PRVC será implementado e resolvido de forma a encontrar a melhor solução viável dada a limitação da capacidade computacional e o tempo colocado com limite por meio do *software* de otimização *Gurobi*, tal ferramenta é conhecida por resolver problemas de programação linear e foi escolhida devido a sua capacidade de resolução de problemas maiores em comparação com outros solvers de otimização

(como por exemplo o Lingo ou solver do Excel) e por possuir licença gratuita no meio acadêmico. A seguir, a heurística de C&W será implementada na ferramenta Microsoft Excel, mais especificamente em VBA assim será possível obter uma solução aproximada para o problema. Tal ferramenta foi escolhida devida a facilidade do autor com esta linguagem de programação e por ser uma linguagem amplamente utilizada por empresas do setor produtivo. E por fim, será implementado o método de Algoritmos Genéticos com o objetivo de se obter uma melhoria da solução encontrada em C&W.

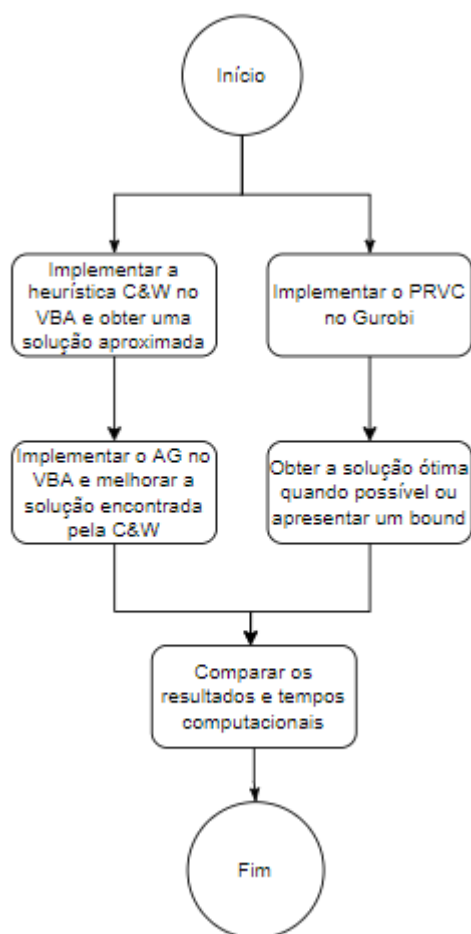
Como apresentado na metodologia, alguns parâmetros precisam ser definidos para que o AG seja inicializado. Neste trabalho o tamanho da população será a quantidade de rotas que a solução de C&W obteve. O tipo de seleção escolhido foi o método da roleta e o *fitness* de cada indivíduo equivalerá a demanda total da rota dividido pelo custo total da rota, desta forma, espera-se que quanto maior o *fitness* maior a oportunidade de encontrar um cromossomo que se comporte de melhor forma em outro indivíduo e melhore a função objetivo. O método para criação de novas gerações será apenas cruzamento, não haverá mutações. Haverá apenas um ponto de corte e é importante destacar que no PRVC os indivíduos podem ter tamanhos diferentes, já que por exemplo um veículo pode ter uma rota com 8 clientes a serem atendidos enquanto outro pode ter apenas 5, então a cada iteração a posição de corte será definida aleatoriamente, não podendo ser maior do que o tamanho do menor indivíduo.

Além da condicional de custo, ou seja, o somatório dos custos dos indivíduos antes do cruzamento tem que ser menor do que depois para o cruzamento ser válido, também há a condicional de demanda, então após ser verificado se o custo foi melhorado será feita a verificação se a demanda de cada um dos novos indivíduos não ultrapassa a capacidade do veículo.

O critério de parada será três vezes a instância do problema, ou seja, um problema com 100 clientes terá 300 iterações. Ao final das iterações será analisada o custo total além de cada uma das rotas para observar se houve melhoria comparado a solução inicial.

A FIGURA 7 ilustra como será o processo da metodologia empregada nesta pesquisa.

FIGURA 7 - Fluxograma da Metodologia



Fonte: O autor (2022)

Para realização dos experimentos foram utilizadas as seguintes configurações de *hardware*: processador Intel ® Core ™ i7-1165G7; placa NVIDIA GeForce MX330; 16GB de memória RAM; sistema operacional Windows 11 PRO 64 bits.

4 RESULTADOS

Neste capítulo, após implementar o PRVC e colocar no formato padrão para execução no software Gurobi, implementação da heurística de C&W no VBA assim como implementação do Algoritmo Genético também em VBA foram testados três problemas da Biblioteca de Problemas de Veículos Capacitados da PUC-RJ. Os três problemas são: A-n32-k5, P-n16-k8 e A-n45-k7, onde o número após a letra “n” representa o número de pontos de demanda, e o número após a letra “k” representa o número mínimo de veículos. A capacidade dos veículos para cada um dos problemas é 100, 35 e 100, respectivamente. Foram levados em consideração o tempo de processamento (em segundos) assim como a solução encontrada, que nesse caso é puramente o somatório da distância percorrida por todos os veículos em cada problema. Também será apresentada as rotas de cada um dos veículos em cada um dos problemas. Alguns dos arcos são representados em reta pontilhada apenas para facilitar a visualização.

4.1 RESULTADOS COMPUTACIONAIS

A TABELA 1, TABELA 2 e TABELA 3 mostram os resultados obtidos quando utilizados cada um dos métodos citados no problema e o tempo limite. Na primeira coluna está a fonte da solução, na segunda coluna o tempo de execução em segundos, na terceira coluna a solução obtida e na quarta coluna o *Gap* (Intervalo) entre a solução obtida e a solução ótima. Para os testes no software Gurobi foram testados 6 limites de tempo máximo para execução de cada um dos problemas. Os resultados obtidos com o Algoritmo Genético não foram apresentados por não mostrarem melhoria com a utilização após a heurística de C&W.

Tabela 1 - Resultados problema A-n32-k5

A-n32-5k			
Fonte	Tempo (s)	Solução	Gap Solução ótima
Solução ótima (CVRPLIB)	-	784	-
C&W - VBA	1,0270	843	8%
Gurobi	120	868	11%
Gurobi	300	843	8%
Gurobi	600	816	4%
Gurobi	1800	816	4%
Gurobi	3600	816	4%
Gurobi	18000	816	4%

Fonte: O autor (2022)

Tabela 2 - Resultados problema P-n16-k8

P-n16-k8			
Fonte	Tempo (s)	Solução	Gap Solução ótima
Solução ótima (CVRPLIB)	-	450	-
C&W - VBA	0,4492	464,39	3%
Gurobi	120	451,34	0,3%
Gurobi	300	451,34	0,3%
Gurobi	600	451,34	0,3%
Gurobi	1800	451,34	0,3%
Gurobi	3600	451,34	0,3%
Gurobi	18000	451,34	0,3%

Fonte: O autor (2022)

Tabela 3 - Resultados problema A-n45-k7

A-n45-k7			
Fonte	Tempo (s)	Solução	Gap Solução ótima
Solução ótima (CVRPLIB)	-	1146	-
C&W - VBA	2,0938	1213,48	6%
Gurobi	120	-	-
Gurobi	300	1631	42%
Gurobi	600	1491	30%
Gurobi	1800	1433	25%
Gurobi	3600	1305	14%
Gurobi	18000	1290	13%

Fonte: O autor (2022)

É possível observar em que em todos os problemas que foram testados a solução obtida pela heurística de C&W não ultrapassou de 2,1 segundos e que o *Gap* em nenhum dos casos passou de 8%. Inclusive esse intervalo percentual pequeno entre a solução obtida e a solução ótima é uma das hipóteses para o Algoritmo Genético não ter encontrado nenhuma melhoria. Nas soluções obtidas pelo Gurobi, chama a atenção que no primeiro problema, A-n32-5k, foram necessários 300 segundos de processamento para se obter o mesmo resultado obtido pela heurística de C&W em apenas 1,02 segundos utilizando o VBA. No segundo problema, P-n16-k8, apesar da solução heurística ter sido encontrada em apenas 0,44 segundos, a solução encontrada pelo Gurobi em 120 segundos ficou com um *Gap* de apenas 0,3% do que a solução encontrada no site. Por fim, no último problema, A-n45-k7, chama a atenção que a melhor solução obtida pelo Gurobi, foi com 18000 segundos (5 horas) e mesmo assim o gap ficou relativamente alto, 13%, quando comparada a solução fornecida pela biblioteca, o que mostra uma certa dificuldade quando os problemas são um pouco maiores.

4.2 ROTAS ENCONTRADAS

A seguir serão apresentadas as rotas encontradas em cada uma das instâncias, assim como sua representação gráfica no plano cartesiano.

4.2.1 Problema A-n32-k5

TABELA 4 - Rotas CVRPLIB – Problema A-n32-k5

CVRPLIB	Rotas
Rota 1	1-22-32-20-18-14-8-27-1
Rota 2	1-13-2-17-31-1
Rota 3	1-28-25-1
Rota 4	1-30-19-9-10-23-16-11-26-6-21-1
Rota 5	1-15-29-12-5-24-4-3-7-1

Fonte: O autor (2022)

TABELA 5 - Rotas C&W - Problema A-n32-k5

C&W	Rotas
Rota 1	1-21-6-26-11-16-30-28-1
Rota 2	1-24-3-4-18-20-32-22-1
Rota 3	1-17-8-14-2-13-1
Rota 4	1-27-7-19-29-5-12-9-10-23-15-1
Rota 5	1-25-31-1

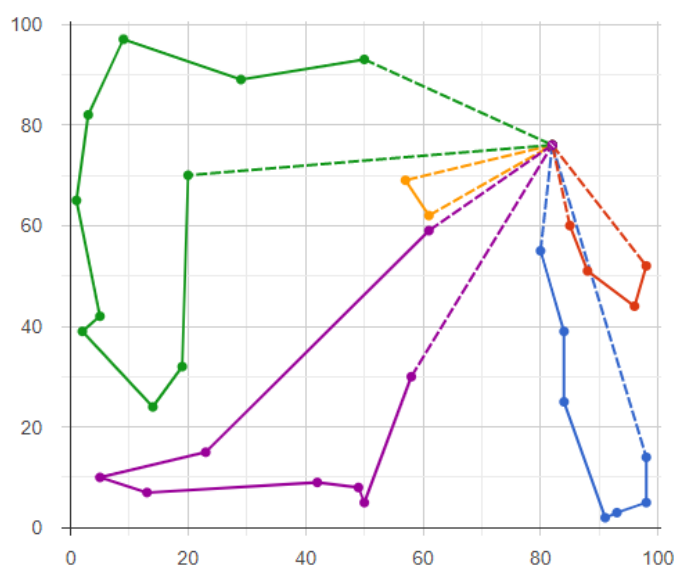
Fonte: O autor (2022)

TABELA 6 - Rotas Gurobi - Problema A-n32-k5

Gurobi	Rotas
Rota 1	1-21-6-26-11-16-30-28-1
Rota 2	1-7-24-29-5-12-9-19-10-23-1
Rota 3	1-27-22-32-20-18-3-4-1
Rota 4	1-31-15-25-1
Rota 5	1-13-2-14-8-17-13-1

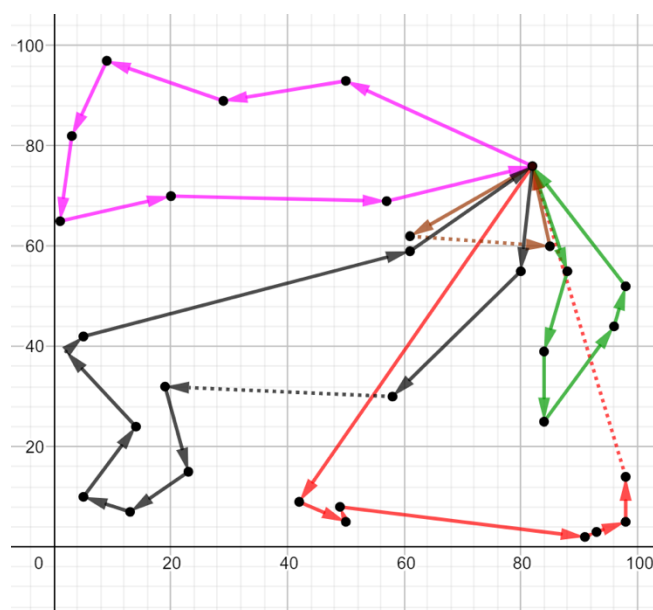
Fonte: O autor (2022)

FIGURA 8 - Rotas CVRPLIB - Problema A-n32-k5



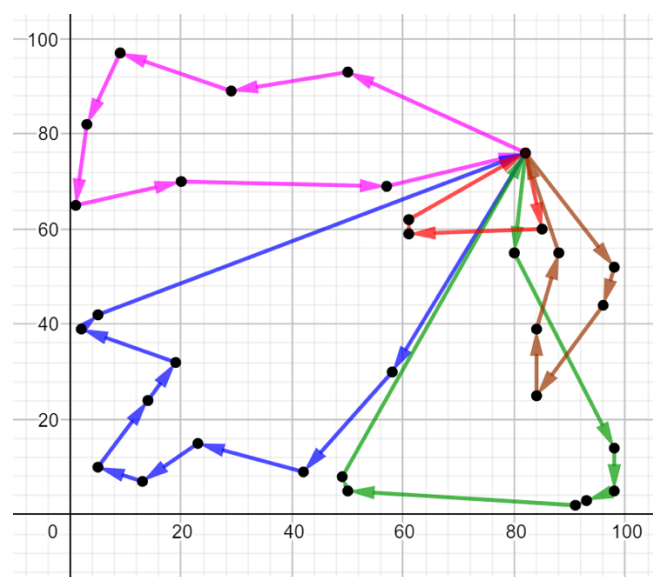
Fonte: CVRPLIB (2022)

FIGURA 9 - Rotas C&W - Problema A-n32-k5



Fonte: O autor (2022)

FIGURA 10 - Rotas Gurobi - Problema A-n32-k5



Fonte: O autor (2022)

Importante destacar que no caso do Gurobi, tanto quanto a tabela de rotas quanto a representação gráfica das rotas é de teste de 18000 segundos. É possível perceber que a configuração das rotas pela solução ótima difere um pouco das soluções obtidas via heurística e pelo Gurobi.

4.2.2 Problema P-n16-k8

TABELA 7 - Rotas CVRPLIB – Problema P-n16-k8

CVRPLIB	Rotas
Rota 1	1-3-1
Rota 2	1-7-1
Rota 3	1-9-1
Rota 4	1-16-13-11-1
Rota 5	1-15-6-1
Rota 6	1-14-10-8-1
Rota 7	1-12-5-1
Rota 8	1-4-2-1

Fonte: O autor (2022)

TABELA 8 - Rotas C&W – Problema P-n16-k8

C&W	Rotas
Rota 1	1-3-1
Rota 2	1-7-1
Rota 3	1-14-9-1
Rota 4	1-13-16-12-1
Rota 5	1-10-15-1
Rota 6	1-6-8-1
Rota 7	1-5-11-1
Rota 8	1-2-4-1

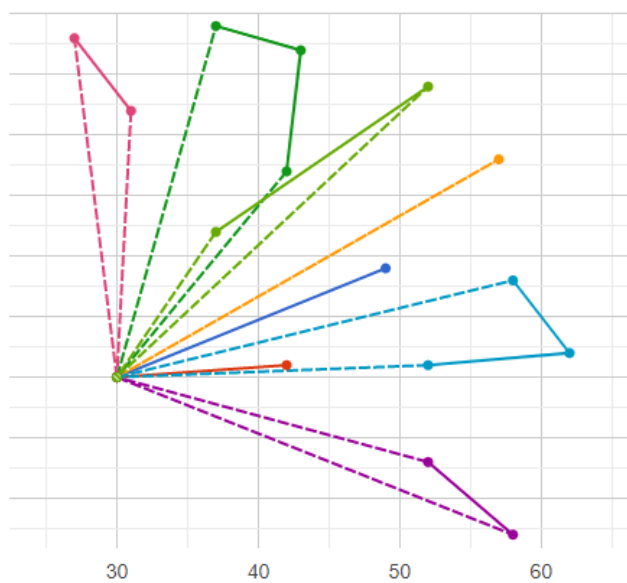
Fonte: O autor (2022)

TABELA 9 - Rotas Gurobi - Problema P-n16-k8

Gurobi	Rotas
Rota 1	1-3-1
Rota 2	1-7-1
Rota 3	1-6-10-4-1
Rota 4	1-12-5-1
Rota 5	1-11-13-16-1
Rota 6	1-2-1
Rota 7	1-9-14-1
Rota 8	1-8-15-1

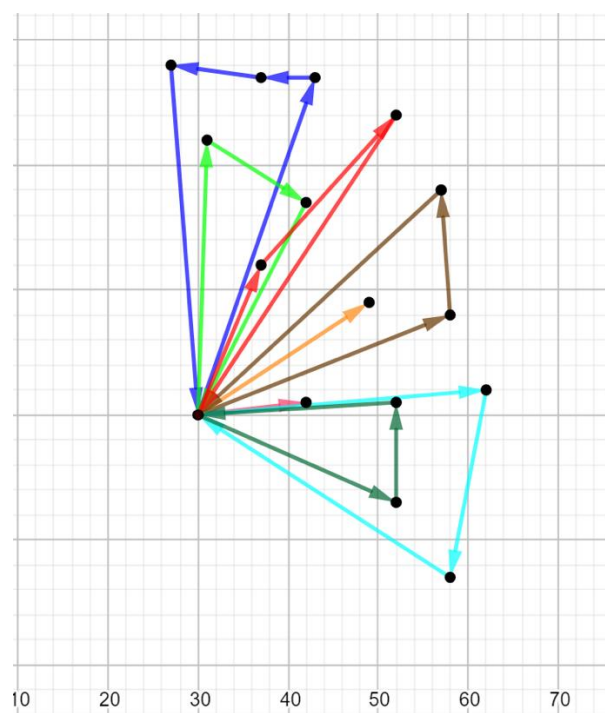
Fonte: O autor (2022)

FIGURA 11 - Rotas CVRPLIB – Problema P-n16-k8



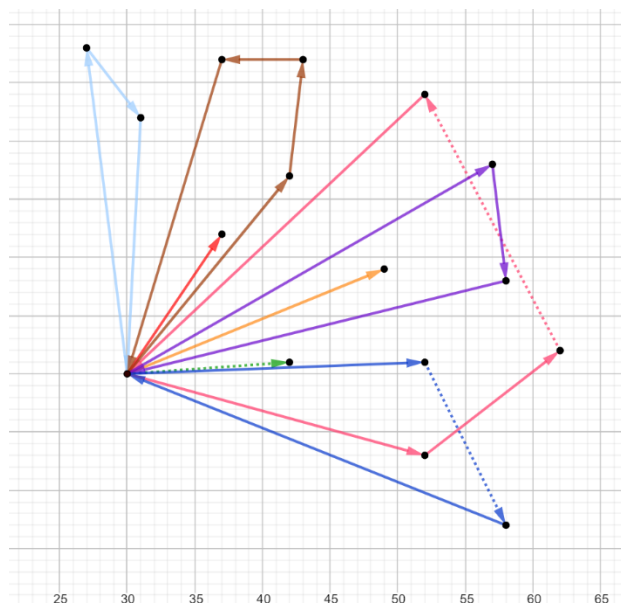
Fonte: CVRPLIB (2022)

FIGURA 12 - Rotas C&W - Problema P-n16-k8



Fonte: O autor (2022)

FIGURA 13 - Rotas Gurobi - Problema P-n-16-k5



Fonte: O autor (2022)

No caso do problema P-n16-k8 é possível encontrar um pouco mais de semelhanças entre as rotas, é o caso das rotas 1-3-1 e 1-7-1, vistas nos três casos. Isso acontece porque tanto o ponto 3 quanto o ponto 7 tem demandas muito altas e em nenhum dos casos poderia ser ligada a outro ponto.

4.2.3 Problema A-n45-k7

TABELA 10 - Rotas CVRPLIB – Problema A-n45-k7

CVRPLIB	Rotas
Rota 1	1-9-43-2-38-31-6-23-1
Rota 2	1-14-39-18-24-26-16-11-1
Rota 3	1-19-8-17-21-41-1
Rota 4	1-40-22-25-45-15-10-33-1
Rota 5	1-34-30-20-37-32-1
Rota 6	1-28-42-44-12-4-29-7-3-1
Rota 7	1-13-5-27-35-36-1

Fonte: O autor (2022)

TABELA 11 - Rotas C&W – Problema A-n45-k7

C&W	Rotas
Rota 1	1-3-5-27-35-36-40-1
Rota 2	1-13-33-19-41-1
Rota 3	1-10-39-18-24-26-21-17-8-14-1
Rota 4	1-11-23-34-45-22-1
Rota 5	1-15-25-12-44-4-29-7-1
Rota 6	1-9-16-43-2-38-31-6-1
Rota 7	1-28-42-30-20-37-32-1

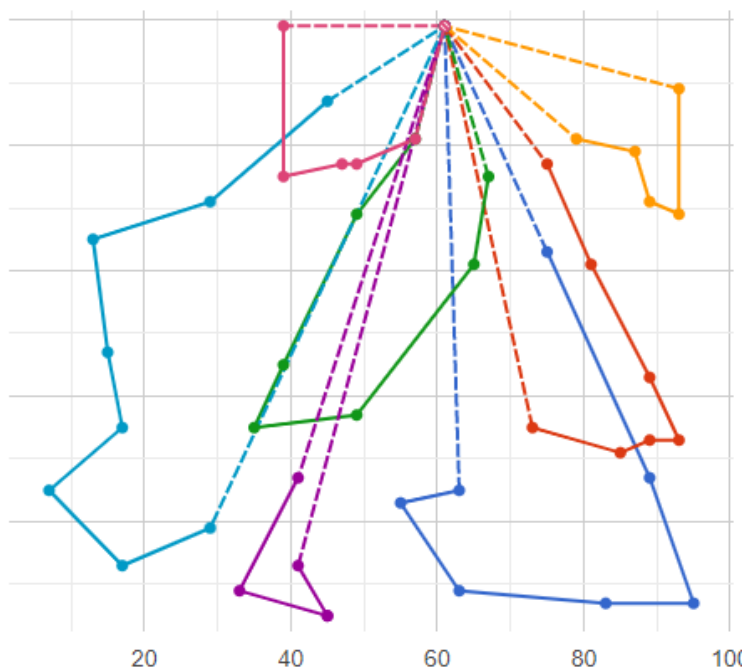
Fonte: O autor (2022)

TABELA 12 - Rotas Gurobi – Problema A-n45-k7

Gurobi	Rotas
Rota 1	1-13-19-8-17-21-41-1
Rota 2	1-27-7-29-4-45-25-1
Rota 3	1-33-18-24-26-16-9-1
Rota 4	1-6-20-37-31-23-1
Rota 5	1-3-5-12-44-28-35-1
Rota 6	1-14-39-11-38-2-43-40-36-1
Rota 7	1-22-34-42-30-32-15-10-1

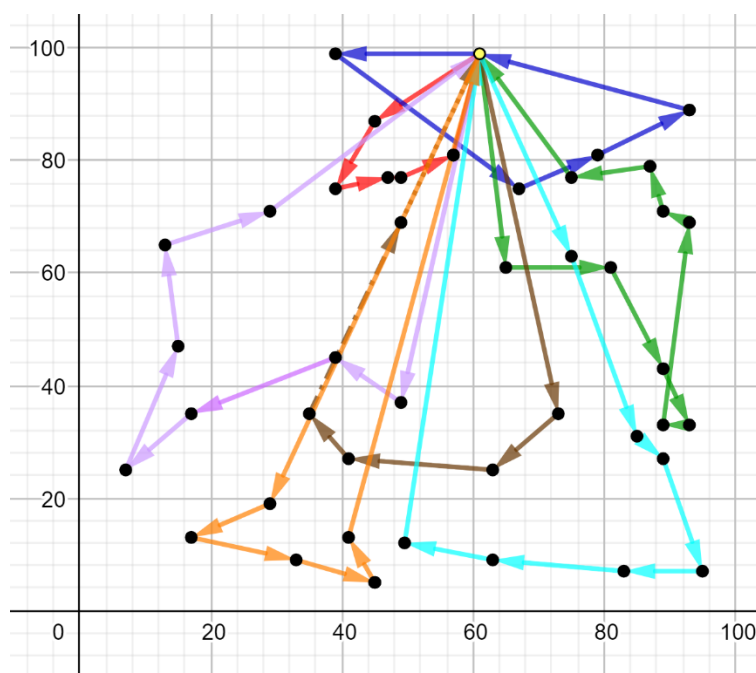
Fonte: O autor (2022)

FIGURA 14 - Rotas CVRPLIB - Problema A-n45-k7



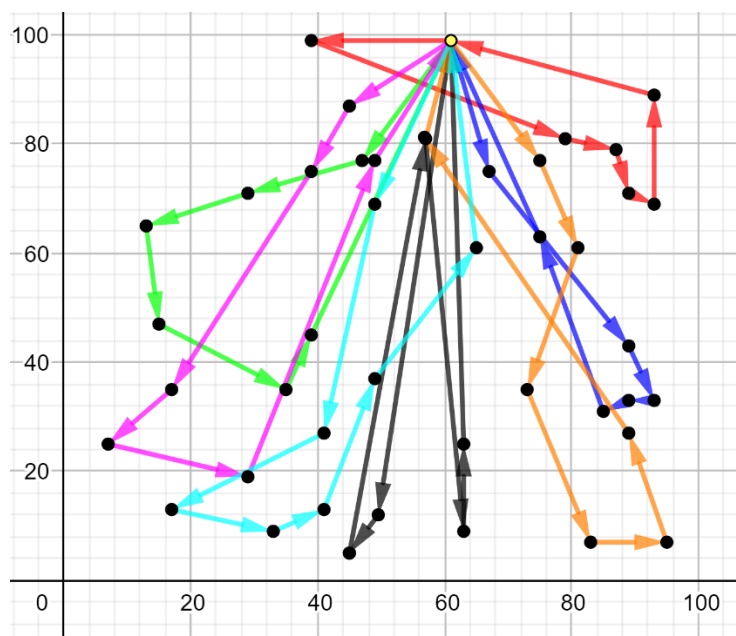
Fonte: CVRPLIB (2022)

FIGURA 15 - Rotas C&W - Problema A-n45-k7



Fonte: O autor (2022)

FIGURA 16 - Rotas Gurobi – Problema A-n45-k7



Fonte: O autor (2022)

Por fim, o problema A-n45-k7 concentra a maior diferença entre as rotas, na FIGURA 16 por exemplo é possível perceber que a Rota 6, que está representada pela cor cinza, apresenta um comportamento diferente, já que ela liga a um ponto que

está mais próximo do depósito e depois volta ao ponto F. Esse com certeza é um dos motivos que fez o teste no Gurobi ficar com custo maior do que C&W.

5 CONSIDERAÇÕES FINAIS

O objetivo do trabalho foi implementar VBA (linguagem de programação do Excel) a heurística de C&W, assim como a do Algoritmo Genético e comparar os resultados com o PRVC no Gurobi. Os resultados foram satisfatórios, já que em nenhum dos casos o Gap entre as soluções ficou maior do que 11%, mas o desapontamento ficou por conta de o Algoritmo Genético não ser capaz de melhorar os resultados em nenhum dos casos, mas como já comentado uma das hipóteses é que por o Gap se pequeno, o Algoritmo realmente não consegue encontrar melhoria, já que revisando outros trabalhos, a heurística de melhoria conseguiu encontrar soluções quando o Gap era maior.

Os resultados mostram que quanto maior a quantidade de instâncias do problema, mais a heurística se destaca quando comparada ao Gurobi, trazendo resultados melhores além de que é uma ferramenta mais acessível que os softwares de otimização.

Para trabalhos futuros, sugere-se testar problemas maiores para ver como a heurística se comporta comparada com softwares de otimização, já que no último caso tem-se a heurística trazendo um resultado melhor em um tempo muito melhor. Além disso, em problemas maiores é esperado que o Gap com a solução obtida na heurística aumente e o AG poderá trazer efeito na melhoria das soluções, caso essa melhoria não ocorra ou não seja suficiente, técnicas como 2-opt e 3-opt podem ser implementadas. Também é sugerido usar o método de inclusão e exclusão além de testar o método *Callback* do software Gurobi, usando esse método o PRVC não precisa ter restrições de sub-rotas pois ele é capaz de inserir restrições conforme as sub-rotas aconteçam e então as evitam.

REFERÊNCIAS

- ARENALES, M.; ARMENTANO, V; MORABITO, R., YANASSE, H., **Pesquisa Operacional**. 2011.
- ARROYO, J. E. C. **Heurísticas e metaheurísticas para otimização combinatória multiobjetivo**. Tese (Doutorado) - Faculdade de Engenharia Elétrica e de Computação, Departamento de Engenharia de Sistemas, Universidade Estadual de Campinas, Campinas, SP, 2002.
- BELFIORE, P. P. **Scatter Search para problema de roteirização de veículos com frota heterogênea, janela de tempo e entregas fracionadas**. 2006, 188f. Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Produção, São Paulo, SP, 2006.
- BOWERSOX, D. J.; CLOSS, D. J. **Supply Chain Logistics Management**. 2001
- BJARNODÓTTIR, A. S. **Solving the vehicle routing problem with genetic algorithms**, 2004
- CARRETERO, M. M.; RANGEL, S. **Grafos Hamiltonianos**. 2007, Universidade Estadual Paulista, Campus de São José do Rio Preto, SP, 2007.
- CARVALHO, A. P. L. F; **Algoritmos Genéticos**. ICMC/USP. Notas de aula, 2009.
- CUNHA, C. B. **Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais**. Transportes (Rio de Janeiro), Rio de Janeiro, v.8, n. 2, p. 51-74, 2000.
- CVRPLIB. Capacitated Vehicle Routing Problem Library. Puc-RJ. Acesso em 09 de abril de 2022. Disponível em: <<http://vrp.galgos.inf.puc-rio.br/index.php/en/>>
- GAMA, M. B. **Roteirização de veículos: Implementação e melhoria de método de Clarke e Wright**. 93 f. TCC (Graduação) - Curso de Engenharia de Produção, Universidade Federal do Vale do São Francisco, Juazeiro, BA, 2011.
- GOLDBERG, D. E. **Genetic Algorithms in search, optimization and machine learning**. U.S.A., Addison-Wesley Publishing Company, 1989.
- GOUVEIA, L. PIRES, JM. **The asymmetric travelling salesman problem and a reformulation of the Miller-Tucker-Zemlin constraints**, Eur. J. Oper. Res. 112 (1999)
- LAPORTE, G. **The vehicle routing problem: an overview of exact and approximate algorithms**. European Journal of Operational Research, v. 59, n. 3, p. 345-358. 1992.
- LIMA, S. J. A. **Otimização do problema de roteamento de veículos capacitado usando algoritmos genéticos com heurísticas e representações cromossômicas**

alternativas. Tese (Mestrado) - Programa de Pós-Graduação em Engenharia de Produção da Universidade Nove de Julho, São Paulo, SP, 2015.

Mastercard SpendingPulse 2021. Acesso em 4 de junho de 2021. Disponível em: <<https://www.mastercard.com/news/latin-america/pt-br/noticias/comunicados-de-imprensa/pr-pt/2021/abril/com-crescimento-de-75-em-2020-e-commerce-brasileiro-chegou-a-representar-11-das-vendas-do-varejo-revela-estudo-da-mastercard/>>

ONCAN, T.; ALTINEL, I. K; LAPORTE, G. **A comparative analysis of several asymmetric traveling salesman problem formulation.** Department of Industrial Engineering, Galatasaray University, Ortaköy, 'Istanbul 34357, Turkey. Department of Industrial Engineering, Bo ğaziçi University, Bebek, 'Istanbul 34342, Turkey. Canada Research Chair in Distribution Management, HEC Montréal, 3000, chemin de la Côte-Sainte-Catherine, Montréal, Canada, H3T 2A7. 2007

OLIVEIRA, R. A. C.; DELGADO, K. V. **Sistema para toreamento de veículos capacitados aplicando Métodos de Monte Carlo.** XI Simpósio Brasileiro Sistemas da Informação, Goiânia, GO, 2015.

OLIVEIRA, R. M.; DOMINGOS, A. P.; JUNIOR, P. S. L. **Aplicação dos Algoritmos Genéticos e Meméticos ao Problema do Caixeiro Viajante.** 2001. Departamento de Telemática, Departamento de Eng. de Sistemas e Departamento de Automação e Computação, Universidade Estadual de Campinas, Campinas, SP, 2001.

PACHECO, M. A. C. **Algoritmos genéticos: princípios e aplicações.** ICA: Laboratório de Inteligência Computacional Aplicada, Departamento de Engenharia Elétrica Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, 1999.

SIQUEIRA, P. H. **Metaheurísticas e Aplicações. Parte III – Outras Metaheurísticas.** PPGMNE/UFPR. Notas de Aula, 2014

TEIXEIRA, R. G.; CUNHA, C.B.; **Heurísticas para o problema de dimensionamento e roteiração de uma frota heterogênea utilizando o algoritmo out-of-kilter.** Transportes, v.10, 2002.

VIEIRA, H. P. **Metaheurística para a solução de problemas de roteamento de veículos com janela de tempo.** 2013, 108f. Dissertação (Mestrado em Matemática Aplicada) – Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, Campinas, SP, 2013.