

Uso da Ferramenta *OR-Tools* na Geração de Rotas Otimizadas em Agência dos Correios Utilizando Algoritmos para o Problema do Caixeiro Viajante

Eolo Charles da Silva¹, Diego Rocha Lima¹, Luan Cosme dos Santos¹,
Fred Daniel Varelo da Silva¹

¹Instituto Federal de Educação, Ciência e Tecnologia do Ceará. IFCE - *Campus* Aracati
Rodovia CE-040, Km 137,1 s/n Aeroporto, Aracati - CE, 62800-000

{eolocharles, drlima7, luancosmefilho, fdvarelo}@gmail.com

Abstract. *The Brazilian Post and Telegraph Company is the largest logistics operator in Brazil and the main provider of delivery services for products purchased through e-commerce. They use the Traffic Management System to develop optimized routes in the largest operational units, but in the smaller ones it is up to the experience of the postmen. This work addressed postal deliveries in Russas-CE as a Traveling Salesman Problem and used Google OR-Tools to prepare the routes. We show its viability in developing optimized routes compared to currently used methods, minimizing costs in the delivery of objects. We conclude that the use of Google OR-Tools can increase the operational efficiency of deliveries.*

Resumo. *A Empresa Brasileira de Correios e Telégrafos é a maior operadora logística do Brasil e principal fornecedora do serviço de entregas dos produtos adquiridos no comércio eletrônico. Usam o Traffic Management System para elaborar rotas otimizadas nas maiores unidades operacionais, mas nas menores fica a cargo da experiência dos carteiros. Este trabalho abordou as entregas dos Correios em Russas-CE como um Problema do Caixeiro Viajante e utilizou o Google OR-Tools para elaboração das rotas. Mostramos sua viabilidade na elaboração de rotas otimizadas em comparação com os métodos atualmente utilizados, minimizando custos na entrega de objetos. Concluímos que a utilização da Google OR-Tools pode aumentar a eficiência operacional das entregas.*

1. Introdução

A ECT (Empresa Brasileira de Correios e Telégrafos) é a maior operadora logística do Brasil. Em 2021, foram entregues mais de 14,2 milhões de objetos postais por dia, sendo 11,8 milhões de correspondências e 2,3 milhões de encomendas nacionais e internacionais em 5.556 municípios brasileiros. Para garantir esse volume de entregas, foram percorridos mais de 1,7 milhões de quilômetros, usando mais de 23 mil veículos [Correios 2022].

Segundo o relatório integrado Correios 2021 divulgado em maio de 2022, a participação de mercado de encomendas dos operadores postais varia entre os países, normalmente na faixa de 20% a 40%. Os Correios encontram-se dentro dessa faixa, com 36% de participação em 2021. Ainda segundo o mesmo relatório o grupo formado pelos segmentos encomenda, logística e internacional vem aumentando sua participação

na receita total dos Correios (que contempla a receita operacional, as receitas financeiras e outras receitas operacionais), representando 66,8% em 2021 contra 57,4% em 2020 e 53,3% em 2019. Isso reduz a vulnerabilidade decorrente de a empresa ter receitas oriundas de serviços substituíveis pela utilização crescente de mídias eletrônicas para comunicação [Correios 2022].

Para lidar com um volume tão grande de entregas diárias, e sendo estas uma das principais fontes de receita, os Correios utilizam o Traffic Management System (TMS), uma ferramenta proprietária para elaborar rotas otimizadas na entrega de encomendas, contudo ainda há unidades onde não foi implantado o uso deste sistema. Neste caso a experiência dos carteiros é a principal forma de elaborar tais rotas nestas unidades operacionais da empresa. Após receberem a carga proveniente dos Centros de Tratamento de Cartas e Encomendas (CTCE), os carteiros são responsáveis por realizarem a Triagem por Distrito (TD) e Separação por Logradouro (SL), bem como elaborar a ordem de entrega dos objetos postais do dia.

Com o advento do comércio eletrônico, a demanda por serviços de entrega de mercadorias e distribuição de serviços a domicílio só aumenta a cada ano. Apenas no primeiro semestre de 2021 o faturamento em vendas *online* no Brasil cresceu 31% em comparação com o ano anterior [ebit 2021]. Segundo dados da 6ª rodada da Pesquisa de Impacto no Transporte – Covid-19, realizada pela Confederação Nacional do Transporte (CNT), a maioria das empresas entrevistadas acreditam que não haverá mudanças no cenário nos próximos seis meses. Das 580 organizações de cargas e passageiros de todos os modais de transporte ouvidas, 28,6% estão otimistas e vislumbram um crescimento da demanda no setor [cnt 2021].

Diante deste cenário e com aumento da concorrência, buscar soluções que tragam ganhos de eficiência é imprescindível para as empresas deste setor se manterem crescendo de forma sustentável. Segundo pesquisa da Capterra os clientes estão cada vez mais buscando por serviços que ofereçam prazos menores e preços mais acessíveis [Capterra 2022].

Assim, este trabalho pretende testar a viabilidade do uso do conjunto de *softwares Google OR-Tools* para gerar rotas otimizadas no distrito especial da AC-Russas, a partir do algoritmos disponibilizados para resolver o modelo básico do Problema do Caixeiro Viajante. Neste caso serão utilizados dados geográficos reais de entregas a serem efetuadas pela unidade. Além disso, as rotas feitas atualmente e as geradas pela ferramenta são comparadas, verificando os ganhos de eficiência.

O presente trabalho está estruturado da seguinte maneira: a Seção 2 revisa todos os conceitos chaves relacionados à temática; já a Seção 3 apresenta alguns trabalhos relacionados a este estudo; a Seção 4 descreve a metodologia utilizada para desenvolvimento do estudo proposto; na Seção 5 são apontados os resultados. Na Seção 6 estão contidas as considerações finais.

2. Fundamentação Teórica

2.1. Definições

O Problema do Caixeiro Viajante em sua versão clássica consiste em um vendedor que partindo de qualquer uma das cidades de um dado conjunto de cidades a visitar, deve

passar por cada uma delas uma única vez tendo como destino final a cidade de onde partiu. A rota é realizada a partir de um critério pré-estabelecido, tal como, distância, tempo, custo, entre outros.

Dado um grafo $G = (V, A)$, onde V e A , são conjuntos formadores de um sistema em que $V = \{v_1, \dots, v_n\}$ é o conjunto dos n vértices/nós do grafo e $A = \{a_1, \dots, a_p\} \subset V \times V$ o conjunto das m arestas/arcos do grafo. A variável x_{ij} representa a utilização do arco entre os nós i e j . Quando $x_{ij} = 1$ o arco da cidade i para a cidade j é utilizado, sendo $x_{ij} = 0$ no caso contrário. Sendo c_{ij} a distância (ou o custo) relativo ao arco da cidade i para a cidade j , o problema pode ser formulado da seguinte forma:

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

$$\text{sujeito a } \sum_{j=1}^n x_{ij} = 1, i = 1, \dots, n, \quad (2)$$

$$\sum_{i=1}^n x_{ij} = 1, j = 1, \dots, n, \quad (3)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1, \forall S \subset V : 1 < |S| < n - 1, \quad (4)$$

$$x_{ij} \in \{0, 1\} \quad (5)$$

Em (1) temos a função objetivo. As restrições 2 e 3 garantem que para cada cliente haja apenas uma aresta de entrada e uma de saída respectivamente. Em 4 garantimos a não formação de sub-rotas e em 5 temos o domínio das variáveis. Vale salientar ainda que o PCV pode ser classificado como Simétrico (PCVS) se o $c_{ij} = c_{j,i}, \forall i, j \in V$ ou Assimétrico (PCVA) caso não seja [Oliveira 2015].

2.2. Métodos de Solução do PCV

Para a solução de um modelo básico basta explorarmos todas as rotas possíveis e descobrirmos qual delas nos fornece o menor valor para a função objetivo (por exemplo, distância percorrida em quilômetros). Se visitarmos $(n - 1)$ cidades em sequência, sendo n o total de cidades a visitar, então o número total de possíveis rotas (circuitos fechados) é dado por uma permutação de $(n - 1)!$. Para um valor baixo de n essas rotas podem ser definidas explicitamente, mas aumentando-se o número de cidades visitadas o número total de rotas aumenta drasticamente. Por exemplo, para $(n) = 6$ cidades, há 720 circuitos possíveis, para $(n) = 12$, há 479.001.600 e para $(n) = 18$, há $6,402 \times 10^{15}$ possíveis rotas [Filip and Otakar 2011]. Devido a complexidade do problema, os métodos exatos em geral não representam uma alternativa computacionalmente viável, sendo comum o uso de heurísticas e meta-heurísticas, algoritmos que permitem obter boas soluções em tempo aceitável. Nesse sentido, a *OR-Tools* apresenta uma boa maneira de resolver problemas combinatórios através de ferramentas robustas.

2.3. OR-Tools

OR-Tools é um *software* de código aberto para otimização combinatória disponibilizado pela Google para resolver diversos problemas de Pesquisa Operacional como

Programação linear e Inteira, Roteamento, Empacotamento, Fluxo e muitos outros (<https://developers.google.com/optimization>). Na maioria dos casos, problemas como esses têm um grande número de soluções possíveis muitas para um computador pesquisar todas elas. Para superar isso, o *OR-Tools* usa algoritmos no estado da arte para restringir o conjunto de buscas, a fim de encontrar uma solução ótima (ou próxima da ótima) [Perron and Furnon]. Para este trabalho foi utilizada a biblioteca especializada em problemas de roteamento, a qual oferece um conjunto de opções com vários parâmetros configuráveis que serão usados e descritos a seguir.

O *First Solution Strategy* é o método usado pelo solucionador para encontrar uma solução inicial, ele contém um conjunto de opções, dentre elas as seguintes usadas neste estudo: *AUTOMATIC*, permite que o solucionador detecte qual estratégia usar de acordo com o modelo que está sendo resolvido; *SAVINGS*, algoritmo de poupança [Clarke and Wright 1964]; *CHRISTOFIDES*, algoritmo Christofides (na verdade, uma variante do algoritmo Christofides usando uma correspondência maximizada em vez de uma correspondência máxima, o que não garante o fator $3/2$ da aproximação em um caixeiro viajante métrico); funciona em modelos genéricos de roteamento de veículos, estendendo uma rota até que nenhum nó possa ser inserido nela [Christofides 1976]; *PARALLEL_CHEAPEST_INSERTION*, constrói iterativamente uma solução inserindo o nó mais barato em sua posição mais barata; o custo de inserção é baseado na função de custo do arco; *GLOBAL_CHEAPEST_ARC*, conecta iterativamente dois nós que produzem o segmento de rota mais barato [Perron and Furnon].

As opções de busca local utilizadas na *OR-Tools* foram: (I) *GUIDED_LOCAL_SEARCH*, uma busca local guiada para escapar dos mínimos locais, meta-heurística eficiente para roteamento de veículos. (II) *TABU_SEARCH*, usa busca tabu para escapar dos mínimos locais.

3. Trabalhos Relacionados

No contexto de entregas e logística, alguns trabalhos utilizam do PCV para formular problemas reais e buscar soluções aplicáveis a essas tarefas, propondo novos métodos ou testando os já existentes.

Em [Shi et al. 2020] é proposto um algoritmo baseado no PCV para planejamento de caminho de logística terminal. Devido as particularidades do problema de entrega foram utilizadas técnicas de Programação Dinâmica, pois essa estratégia permite que o problema seja dividido em várias etapas inter-relacionadas possibilitando que a otimização das decisões ocorram em cada uma dessas fases. Em geral, se tratando de entregas as decisões tomadas têm relação com o tempo, desse modo, o estado atual afeta diretamente o próximo daí o uso de uma abordagem "dinâmica", já que o processo para obter solução está em constante mudança. O trabalho apresenta uma equação que atende às especificidades dos processos de coleta e entrega, e almeja fornecer uma solução que sirva de base teórica para um novo modelo de entrega inteligente.

No trabalho de [Sanguansat 2019] o autor faz uso da API¹ de Matriz de distâncias do *Google* para obter dados em tempo real e utiliza esses dados em conjunto com o *Google OR-Tools*, uma biblioteca de código aberto para problemas de otimização, para resolver

¹<https://maps.googleapis.com/maps/api/distancematrix/>

três tipos de problemas de roteamento de veículos (PVR), o clássico, onde apenas a quantidade de veículos e os endereços são considerados, o PRVC, no qual são acrescentadas restrições de capacidade dos veículos e demanda dos clientes, por fim o PRVC-JT que obedece às mesmas restrições do anterior, porém com restrições de janelas de tempo para cada cliente. São apresentados os resultados dos experimentos com dados da cidade de Bangkok, para instâncias com trinta endereços.

Quando ao PCV em serviços postais, [Ferreira et al. 2017] traz a aplicação de um conjunto de algoritmos para otimização do percurso de distribuição de encomendas dos Correios na cidade de Coromandel/MG. Para elaboração do estudo foi utilizado o *Open Street Map* uma base de dados geográficos gratuitos mantida de forma colaborativa, uma heurística baseado em um algoritmo genético em conjunto com o Algoritmo de Dijkstra usando a linguagem MATLAB. A solução proposta apresentou rotas condizentes em custos de percurso com as mesmas elaboradas por profissionais experientes.

Ainda no contexto de serviços postais [Ji and Chen 2007] nos apresenta um estudo sobre otimização de rotas de veículos no serviço postal de Hong Kong. Neste trabalho a partir de um PRV é proposto um modelo de programação linear de inteiros visando a maximização de recursos e minimização de custos de operação. Para lidar com a complexidade é utilizada uma ferramenta comercial *CPLEX*, atualmente na versão 20.1. O trabalho se concentra em elaborar rotas para coleta de correspondências em trinta e quatro pontos de operação espalhados pela ilha, obedecendo as restrições de minimização da distância, capacidade dos veículos e demanda de cada ponto de coleta.

4. Metodologia

A Figura 1 apresenta um fluxo básico da metodologia e os meios usados em cada fase, Iniciando na Modelagem do Problema e passando por todas as etapas intermediárias até chegar na etapa de Avaliação de Desempenho.



Figura 1. Fases de criação do sistema.

4.1. Modelagem do Problema

Na AC-Russas o distrito especial abrange toda à área com distribuição na cidade. O veículo utilizado possui a capacidade de transportar toda a carga diária da agência, desse modo, o carteiro ao elaborar sua rota, o faz de modo a acomodar todos os objetos de uma só vez no compartimento destinado a carga. O tempo para entrega corresponde ao horário comercial com o intervalo para almoço de uma hora, contudo não há a necessidade de

levar em consideração a parada para almoço pois o carteiro não leva em conta o local de almoço na elaboração da rota. Os correios ainda não realizam entregas com horário agendado. Para implementação de um programa básico capaz de resolver instâncias do problema foi utilizado o *Google Colab*² em conjunto com a linguagem de programação *Python* na versão 3.7 e o pacote *OR-Tools*. A escolha deste conjunto de ferramentas se deu pela facilidade de instalação, configuração e flexibilidade para utilizar o *OR-Tools* e *Python* no ambiente do *colab*, não sendo necessário utilizar um *hardware* dedicado de alto desempenho para a tarefa.

4.2. Aquisição e Pré-processamento dos Dados

A coleta dos dados foi realizada na AC-Russas em um intervalo de dez dias úteis consecutivos junto ao carteiro responsável pela distribuição no distrito especial. Os dados coletados consistem em dez rotas, sendo a menor uma sequência de cinquenta e dois endereços e a maior contém um total de oitenta e três endereços. Excluindo-se os endereços repetidos, foram coletados um total de quinhentos e vinte e sete endereços de entrega. Os Correios utilizam o Sistema de Rastreamento de Objetos (SRO). Após a preparação da rota pelo carteiro cada objeto é lançado por meio do seu endereço e é gerada uma Lista de Objetos Entregues ao Carteiro (LOEC) que é carregada na versão móvel do SRO que roda nos *Smartphones* usados para baixa dos objetos. Com essa lista em mãos o carteiro cria a sua rota.

4.3. Construção das Matrizes de Distância e Duração

Na passagem dos dados na implementação do programa é necessário fornecer uma matriz de distâncias como a mostrada abaixo, ou matriz de duração caso o objetivo de minimização escolhido seja o tempo gasto para percorrer a rota. A matriz de distância é uma matriz cuja entrada (i, j) é a distância da localização i à localização j em metros, onde os índices da matriz correspondem às localizações de cada ponto a ser visitado. A matriz de duração segue a mesma linha, mas ao invés de distância traz uma estimativa da duração do percurso de i à j . O *Google* disponibiliza uma API de matriz de distâncias, esse serviço oferece a possibilidade de consultarmos a distância entre dois pontos e nos devolve a duração da viagem e a distância em metros. Os dados são disponibilizados em formato JSON como no exemplo abaixo. Abaixo exemplos respectivos da matriz de distâncias e dados de consulta em JSON.

```
1 data = {}
2 data['distance_matrix'] = [
3     [0, 2451, 713, 1018, 1631,],
4     [2451, 0, 1745, 1524, 831],
5     [713, 1745, 0, 355, 920 ],
6     [1018, 1524, 355, 0, 700],
7     [1631, 831, 920, 700, 0 ],]
8 data['num_vehicles'] = 1
9 data['depot'] = 0
10 return data
```

```
{
1  "destination_addresses": ["San Francisco, CA, USA"],
2  "origin_addresses": ["Vancouver, BC, Canada"],
3  "rows": [{"elements": [{"
4      "distance": { "text": "1,723 km", "value": 1723247 },
5      "duration": { "text": "3 days 21 hours", "value": 335356 },
6      "status": "OK",
7  }],
8  }],
9  }
```

²<https://colab.research.google.com/>

A API possui uma série de parâmetros que permitem definir o tipo de veículo utilizado, tipos de via a serem evitadas o sistema de medidas para as distâncias, opções de tráfego, entre outros, que permitem um ajuste ao modelo de problema a ser resolvido, na construção das nossas matrizes utilizamos a configuração padrão que vem ajustada para o modo *driving* devolvendo distâncias usando a rede rodoviária, além disso, para as distâncias foi utilizado o sistema métrico de unidades, não foi configurado nenhum parâmetro extra.

4.4. Geração das Rotas

O programa recebe como entrada a matriz de distância ou duração da rota e retorna como resultado o valor da função objetivo de minimização escolhida, neste caso a distância total da rota ou o tempo estimado para o percurso. O programa também retorna como saída uma rota indicando a sequência de entrega dos clientes para a rota gerada, sendo esta representada pelos índices da ordem dos endereços usados para gerar a matriz de distâncias. Para solução das dez instâncias, após uma análise prévia a fim de descobrir quais métodos ofereciam as melhores soluções, foram aplicados os métodos com as opções presentes na Seção 2.3. *First solution strategy*: AUTOMATIC, SAVINGS, CHRISTOFIDES, PARALLEL CHEAPEST INSERTION, GLOBAL CHEAPEST ARC. Local Search Metaheuristic: GUIDED LOCALSEARCH, TABU SEARCH. Por se tratar de um contexto de entregas optamos por adotar como objetivo de minimização a distância total da rota. Também foram feitos testes com a minimização do tempo para realizar o percurso. Para o objetivo tempo gasto para percorrer a rota, aplicamos o método padrão *First solution strategy* na opção AUTOMATIC para dez instâncias, já para a distância total da rota foram aplicados todos os métodos descritos anteriormente na Seção 2.3

4.5. Avaliação de Desempenho

A atratividade visual é um critério que se provou eficaz para facilitar a implementação prática e a colaboração dos setores de uma organização envolvidos nos níveis de planejamento e operacional no transporte [Rocha et al. 2022]. Como conclui [Rossit et al. 2019] em sua revisão de literatura, onde nos apresenta um conjunto de métricas aplicáveis a vários problemas de roteamento de veículos, elas são classificadas em quatro categorias: compactação; proximidade; sobreposição e cruzamentos de rotas; e complexidade. Desse modo, optamos por utilizar duas das recomendadas pelos autores devido sua adequação ao caso em estudo. (i) Métrica de compactação descrita em [Matis 2008]: $COMP_i = \frac{AvgDist_i}{AvgMaxDist_i}$, onde $AvgDist_i$ é a distâncias média entre dois clientes consecutivos na rota i , e $AvgMaxDist_i$ é a distância média das 20% maiores distâncias entre dois clientes consecutivos na rota i , assim quanto maior o valor mais compacta é a solução. (ii) Número de cruzamentos em uma mesma rota como em [Poot et al. 2002]: neste caso quanto menor o número de cruzamentos melhor é a solução.

5. Resultados

Os resultados foram obtidos a partir da execução das instâncias de acordo com a modelagem prévia para cada um dos métodos utilizados, incluindo a solução do carteiro.

Apresentamos também os resultados das métricas escolhidas para avaliação da atratividade visual de cada uma das rotas geradas. Na Tabela 1 temos os valores das distâncias totais em metros e a quantidade de clientes para cada uma das rotas com o respectivo método de solução. Cada linha represta o percurso de um dia do carteiro com a quantidade de clientes atendidos daquele dia. O menor valor está destacado, mostrando onde diminuição no percurso total das rotas se comparado a rota feita pelo carteiro.

N° Clientes	Carteiro	FIRST SOLUTION STRATEGY					LOCAL SEARCH OPTIONS	
		AUTOMATIC	SAVINGS	CHRISTOFIDES	PARALLEL_CHEAPEST_INSERTION	GLOBAL_CHEAPEST_ARC	GUIDED_LOCAL_SEARCH	TABU_SEARCH
67	42363	33068	31624	32316	31345	32836	30882	30917
57_a	37147	28851	29259	28474	29701	28496	28055	28620
57_b	36933	31188	29053	28998	29792	28625	28521	28851
52	38252	31041	30280	29530	30295	31251	29456	29830
61	37766	27597	28817	28175	30232	27213	27139	27597
64	41236	30069	29726	30665	31401	30128	29470	29453
60	34665	29417	27723	28350	33494	29417	27463	27723
56	35965	28534	26911	26926	27624	27555	25851	26765
83	46533	37015	33897	35897	34887	34231	32706	35166
69	39845	29969	29963	29122	32222	30840	29059	29644

Tabela 1. Distância total das rotas para cada método de solução.

A Tabela 2 traz a diferença percentual entre as distâncias totais da rota do carteiro e as rotas geradas usando o *OR-tools*, para cada um dos métodos usados. No geral houve uma diminuição média de 23,16% o que pode significar uma economia relevante nos custos operacionais.

N°clientes	FIRST SOLUTION STRATEGY					LOCAL SEARCH OPTIONS	
	AUTOMATIC	SAVINGS	CHRISTOFIDES	PARALLEL_CHEAPEST_INSERTION	GLOBAL_CHEAPEST_ARC	GUIDED_LOCAL_SEARCH	TABU_SEARCH
66	-21.94	-25.35	-23.72	-26.01	-22.49	-27.10	-27.02
57_a	-22.33	-21.23	-23.35	-20.04	-23.29	-24.48	-22.95
57_b	-15.56	-21.34	-21.48	-19.34	-22.49	-22.78	-21.88
52	-18.85	-20.84	-22.80	-20.80	-18.30	-22.99	-22.02
61	-26.93	-23.70	-25.40	-19.95	-27.94	-28.14	-26.93
64	-27.08	-27.91	-25.64	-23.85	-26.94	-28.53	-28.57
60	-15.14	-20.03	-18.22	-3.38	-15.14	-20.78	-20.03
56	-20.66	-25.17	-25.13	-23.19	-23.38	-28.12	-25.58
83	-20.45	-27.15	-22.86	-25.03	-26.44	-29.71	-24.43
69	-24.79	-24.80	-26.91	-19.13	-22.60	-27.07	-25.60

Tabela 2. Diferença percentual entre as rotas do carteiro e as geradas usando *OR-tools*.

A Tabela 3 apresenta uma estimativa do tempo, valores em minutos, para percorrer as rotas, tendo como base a matriz de duração fornecida pela API. Neste caso aplicamos apenas o Método *AUTOMATIC*, onde a ferramenta escolhe qual método usar. Os valores apresentam uma economia semelhante em termos percentuais, comparado com os valores obtidos quando o objetivo de minimização escolhido é a distância. Após análise dos resultados podemos notar que as maiores diferenças se apresentam nas instâncias com uma maior quantidade de pontos de entrega, como evidenciado na tabela 2, cuja instância com oitenta e três clientes usando *GUIDED_LOCAL_SEARCH* apresenta uma redução na distância total da rota de 29,71%, em relação à gerada pelo carteiro. Quando o objetivo de minimização é o tempo, foi observado o mesmo comportamento, como visto na Tabela 3, onde usando o método *FirstSolutionStrategy.AUTOMATIC*, a instância com oitenta e três clientes, teve redução de 23,49% na estimativa do tempo total da rota. Podemos inferir que

Nº clientes	Carteiro	FIRST SOLUTION STRATEGY .AUTOMATIC	Diferença(%)
66	120	94.76	-21.03
57_a	98.13	83.65	-14.76
57_b	108.06	90.25	-16.48
52	101.56	86.71	-14.62
61	109.85	87.4	-20.44
64	122.53	95.7	-21.90
60	101.15	86.6	-14.38
56	106.43	85.26	-19.89
83	129.2	98.85	-23.49
69	115.38	92.11	-20.17

Tabela 3. Estimativa da duração das rotas e diferença percentual.

essa diferença se dá pela maior dificuldade do carteiro em lidar com instâncias grandes, já que os métodos apresentam desempenho semelhante para as demais instâncias.

A Tabela 4 traz valores das métricas de atratividade visual para cada instâncias resolvidas utilizando minimização das distâncias das rotas, para os métodos utilizados. As métricas apresentaram resultados melhores ou iguais aos das rotas dos carteiros indicando que as rotas geradas estão de acordo em termos visuais, compactação e quantidades de cruzamentos. A Figura 2 apresenta uma rota do carteiro e uma obtida usando *GUIDED LOCAL SEARCH* para instância com cinquenta e dois clientes, onde é possível observar as melhorias em uma das rotas definidas pela *OR-Tools*.

Nº Clientes	Carteiro		FIRST SOLUTION STRATEGY										LOCAL SEARCH OPTIONS			
			AUTO-MATIC		SAVINGS		CHRIS-TOFIDES		PARALLEL-CHEAPEST-INSERTION		GLOBAL-CHEAPEST-ARC		GUIDED-LOCAL-SEARCH		TABU-SEARCH	
			Cp	Cr	Cp	Cr	Cp	Cr	Cp	Cr	Cp	Cr	Cp	Cr	Cp	Cr
67	0.38	9	0.42	8	0.42	5	0.43	8	0.45	4	0.42	5	0.43	5	0.43	9
57_a	0.36	5	0.39	4	0.4	11	0.39	3	0.41	4	0.38	3	0.39	3	0.38	2
57_b	0.38	8	0.43	6	0.44	2	0.45	3	0.43	3	0.43	1	0.43	1	0.43	2
52	0.34	11	0.35	3	0.36	2	0.36	2	0.36	4	0.34	3	0.37	0	0.36	2
61	0.38	2	0.43	2	0.46	4	0.44	4	0.45	4	0.43	2	0.44	2	0.43	2
64	0.37	10	0.42	2	0.41	5	0.41	2	0.42	7	0.42	5	0.44	1	0.43	5
60	0.41	8	0.48	6	0.49	6	0.51	8	0.51	15	0.48	6	0.52	4	0.49	5
56	0.38	11	0.47	8	0.47	1	0.46	6	0.47	3	0.47	4	0.46	1	0.46	3
83	0.33	12	0.36	21	0.36	8	0.33	16	0.36	14	0.35	12	0.35	12	0.36	18
69	0.35	10	0.39	5	0.41	5	0.41	5	0.41	6	0.42	3	0.4	3	0.41	5

Tabela 4. Métricas de compactação (Cp) e cruzamento (Cr).



(a) Rota do carteiro. Distância total 38252 metros. 11 cruzamentos.



(b) Rota OR-tools. Distância total 29456 metros. 0 cruzamentos.

Figura 2. Trajeto das rotas.

6. Conclusão

Neste trabalho analisou-se o uso dos algoritmos disponibilizados pelo *Google OR-tools* na criação de rotas a serem usadas pela unidade operacional dos Correio de Russas-CE. Os métodos são baseados no Problema do Caixeiro Viajante. Os resultados indicam a viabilidade técnica e operacional do uso da ferramenta, demonstrando eficiência no uso de soluções *open source* para automatizar a geração de rotas nos Correios. Cabe ressaltar, contudo, que o serviço de entregas possui os cenários mais diversos e que em suma não é possível generalizar os resultados para além das restrições aqui delimitadas. Desse modo, as condições em que o estudo foi realizado são fatores cruciais para avaliarmos os dados obtidos por este trabalho. Nesse sentido seriam necessários estudos em outros tipos de distrito e uma posterior comparação com a Traffic Management System.

Referências

- Capterra (2022). Quick commerce: 95% dos consumidores gostariam de reduzir os prazos de entrega. <https://www.capterra.com.br/blog/2365/quick-commerce>.
- Christofides, N. (1976). Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report, Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group.
- Clarke, G. and Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations research*, 12(4):568–581.
- cnt (2021). Pesquisa de impacto no transporte – covid-19, 6ª rodada.
- Correios (2022). relatorio-integrado-correios-2021.
- ebit (2021). 44ª ed.webshoppers. Realizado pela Ebit desde 2001, o Webshoppers é o relatório de maior credibilidade sobre o comércio eletrônico brasileiro e é considerado a principal referência para os profissionais do segmento.
- Ferreira, M. F., Dias, T. D. d. M. T., Ferreira, J. B. F. J. B., da Silva, F. H. B., da Silveira Santos, I. A., Ribeiro, L. C., et al. (2017). Otimização do percurso de distribuição de encomendas dos correios na cidade de coromandel-mg. *Revista Agroveterinária, Negócios e Tecnologias*, 2(2):42–55.
- Filip, E. and Otakar, M. (2011). The travelling salesman problem and its application in logistic practice. *WSEAS Transactions on Business and Economics*, 8(4):163–173.
- Ji, P. and Chen, K. (2007). The vehicle routing problem: the case of the hong kong postal service. *Transportation Planning and Technology*, 30(2-3):167–182.
- Matis, P. (2008). Decision support system for solving the street routing problem. *Transport*, 23(3):230–235.
- Oliveira, A. F. M. d. A. (2015). *Extensões do problema do caixeiro viajante*. PhD thesis, Universidade d Coimbra.
- Perron, L. and Furnon, V. Or-tools.
- Poot, A., Kant, G., and Wagelmans, A. P. M. (2002). A savings based method for real-life vehicle routing problems. *Journal of the Operational Research Society*, 53(1):57–68.

- Rocha, D., Aloise, D., Aloise, D. J., and Contardo, C. (2022). Visual attractiveness in vehicle routing via bi-objective optimization. *Computers & Operations Research*, 137:105507.
- Rossit, D. G., Vigo, D., Tohmé, F., and Frutos, M. (2019). Visual attractiveness in routing problems: A review. *Computers & Operations Research*, 103:13–34.
- Sanguansat, P. (2019). Developing applications for vehicle routing problems with real time data acquisition. *origins*, 13(100.532713&destinations):13–847881.
- Shi, K., Zhang, H., Zhang, Z., and Zhou, X. (2020). The algorithm of terminal logistics path planning based on tsp problem. In *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, pages 130–133.