Trabalho de Pesquisa Operacional

PROBLEMA DE ROTEAMENTO DE VEÍCULOS (VRPTW)

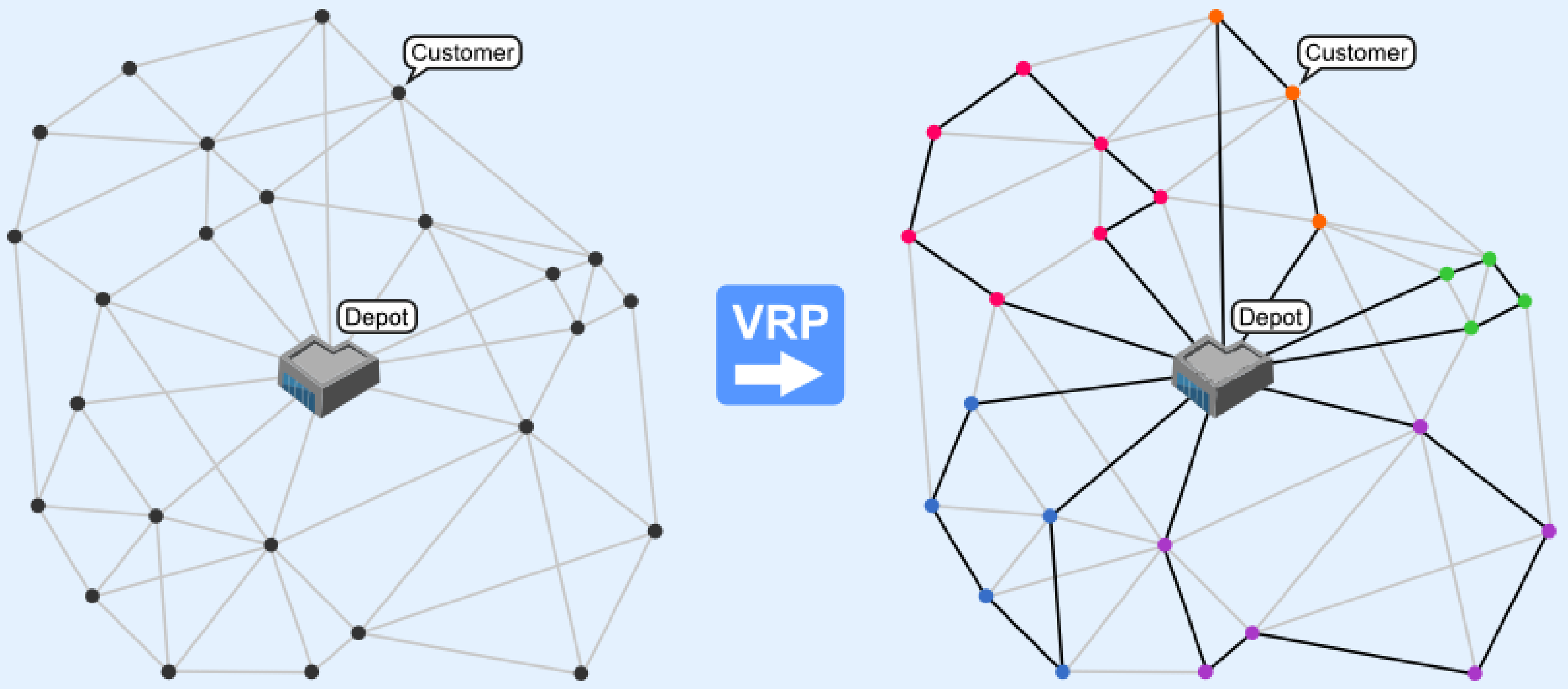
Professor: Lucas de Souza Batista

Integrantes do grupo:

- Lucas Pimenta Braga
- Mateus de Souza Gontijo
- Victor Gabriel dos Santos Silva

DESCRIÇÃO DO PROBLEMA

- O Problema de Roteamento de Veículos (VRP) busca determinar um conjunto de rotas de menor custo para uma frota que parte de um depósito, visita clientes e retorna.
- No VRPTW, há duas restrições adicionais:
  - Capacidade máxima de cada veículo;
  - Janela de tempo (Time Window): **Cada cliente tem um tempo de atendimento**.
- Objetivo: minimizar o custo total percorrido respeitando capacidade e tempo.



MODELAGEM

CONJUNTOS

$N = \{1, 2, \ldots, n\}$: conjunto de clientes

$N_0 = N \cup \{0\}$: conjunto de todos os nós (clientes + depósito)

K : número máximo de veículos disponíveis

PARÂMETROS

d_i : demanda do cliente i

s_i : tempo de serviço no cliente i

$[e_i, l_i]$: janela de tempo do nó i

Q : capacidade do veículo

c_{ij} : custo/tempo de deslocamento entre os nós i e j

M : constante Big-M suficientemente grande

MODELAGEM

VARIÁVEIS DE DECISÃO:

$$x_{ij} = \begin{cases} 1, & \text{se um veículo percorre o arco } (i,j), \\ 0, & \text{caso contrário} \end{cases} \quad \forall i,j \in N_0,\ i \neq j$$

$$t_i \geq 0 \quad : \text{tempo de início de serviço no nó } i$$

$$u_i \geq 0 \quad : \text{carga acumulada ao chegar no nó } i$$

FUNÇÃO OBJETIVO

$$\min \sum_{i \in N_0} \sum_{\substack{j \in N_0 \\ j \neq i}} c_{ij}\, x_{ij}$$

RESTRIÇÕES

3.1. Atendimento único de cada cliente

Cada cliente deve ter exatamente **uma saída**:

$$\sum_{\substack{j \in N_0 \\ j \neq i}} x_{ij} = 1, \quad \forall i \in N$$

Cada cliente deve ter exatamente **uma entrada**:

$$\sum_{\substack{j \in N_0 \\ j \neq i}} x_{ji} = 1, \quad \forall i \in N$$

3.2. Limite de veículos que partem e retornam ao depósito

$$\sum_{j \in N} x_{0j} \leq K$$

$$\sum_{i \in N} x_{i0} \leq K$$

RESTRIÇÕES

3.3. Janelas de tempo (Big-M)

Garantir consistência temporal:

$$t_j \geq t_i + s_i + c_{ij} - M(1 - x_{ij}), \quad \forall i, j \in N_0,\, i \neq j$$

Respeito aos limites individuais:

$$e_i \leq t_i \leq l_i, \quad \forall i \in N_0$$

3.4. Capacidade e eliminação de subtours (MTZ com carga)

Condições iniciais e limites:

$$u_0 = 0$$

$$0 \leq u_i \leq Q, \quad \forall i \in N_0$$

Transição de carga ao longo das rotas:

$$u_j \geq u_i + d_j - Q(1 - x_{ij}), \quad \forall i, j \in N_0,\, i \neq j$$

FERRAMENTAS & METODOLOGIA

Linguagem: Python 3

- Fácil integração com bibliotecas matemáticas

Solver: Gurobi Optimizer

- Branch-and-Cut nativo
- Interface gurobipy
- Licença acadêmica UFMG

Validação:

- Comparação com BKS (literatura)

Entrada: Instâncias Solomon

- Coordenadas dos clientes
- Demandas e janelas de tempo
- Tempos de serviço

Saída:

- Rotas ótimas (sequência de clientes)
- Distância total, número de veículos
- Tempos de atendimento
- Arquivos JSON com solução completa
- Gráficos PNG das rotas

INSTÂNCIAS TESTADAS

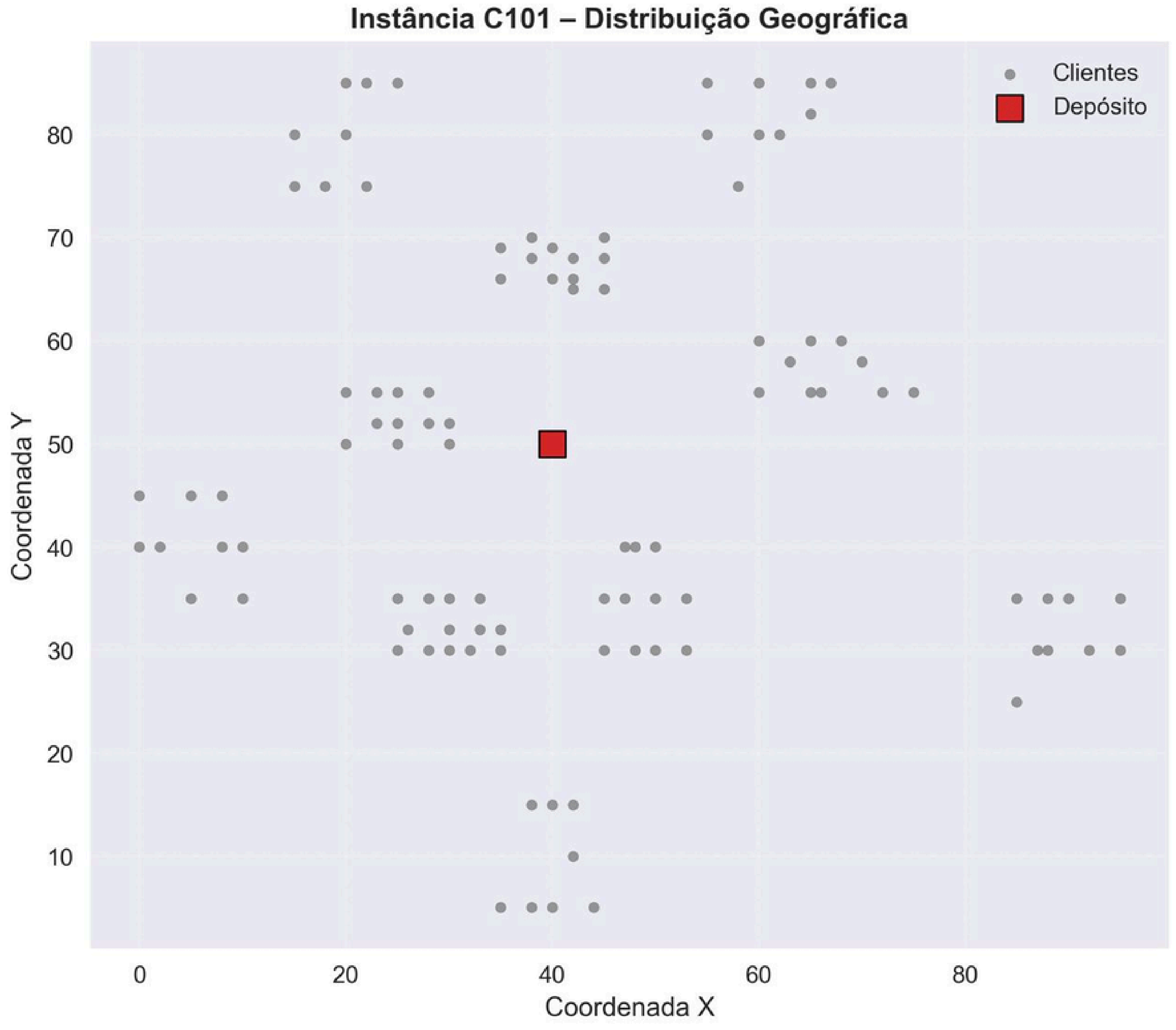
INSTÂNCIAS DE SOLOMON (100 CLIENTES CADA)

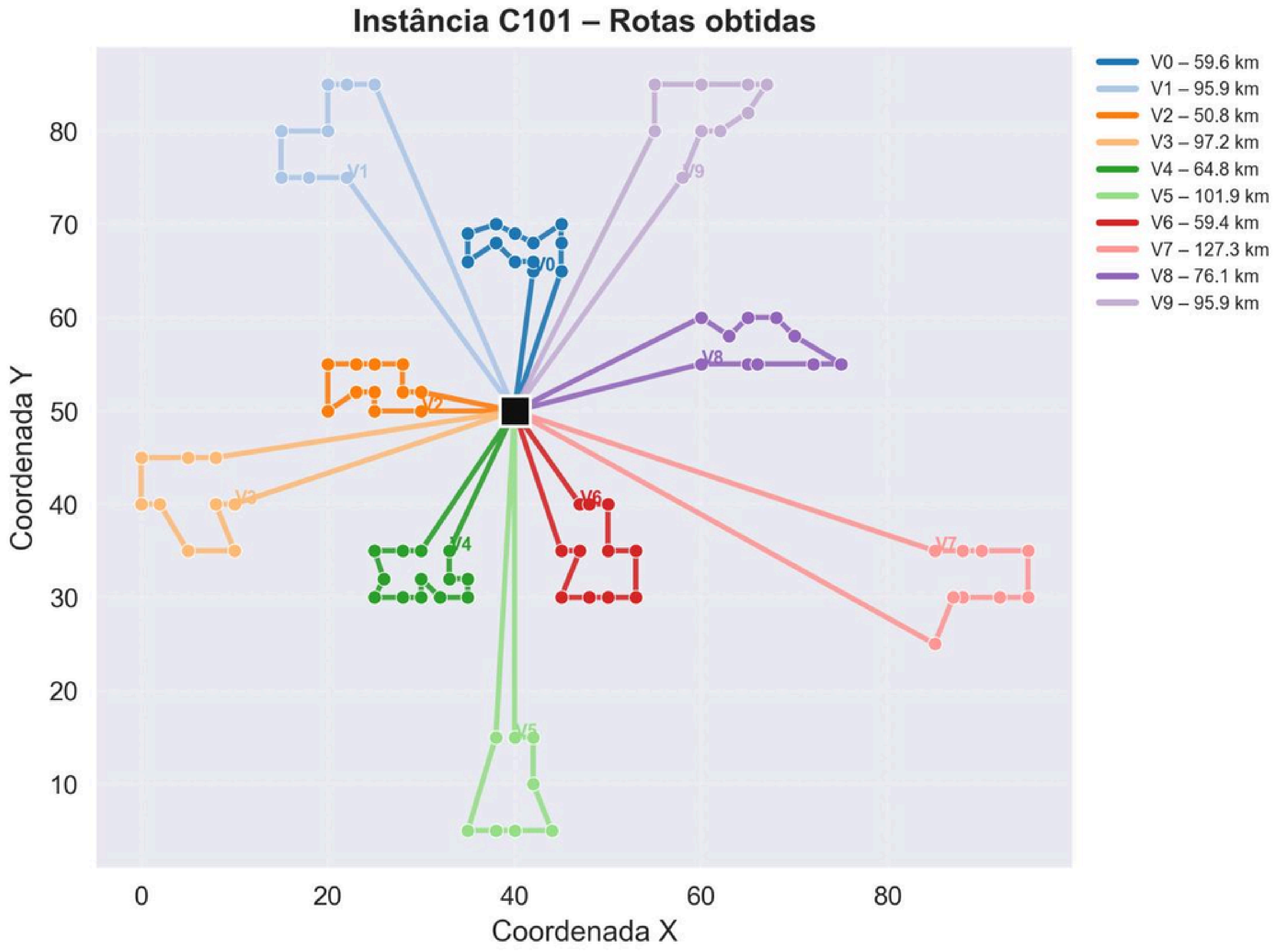
C101	R101	RC101
• Clientes agrupados em clusters • Fácil: vizinhos estão próximos • Requer ≈10 veículos • Mais rápido de resolver	• Clientes aleatoriamente distribuídos • Difícil: sem estrutura espacial • Requer ≈19-20 veículos • Mais lento de resolver	• Mix de clusters + pontos aleatórios • Complexidade intermediária • Requer ≈14-15 veículos • Tempo intermediário

RESULTADOS VS BNS

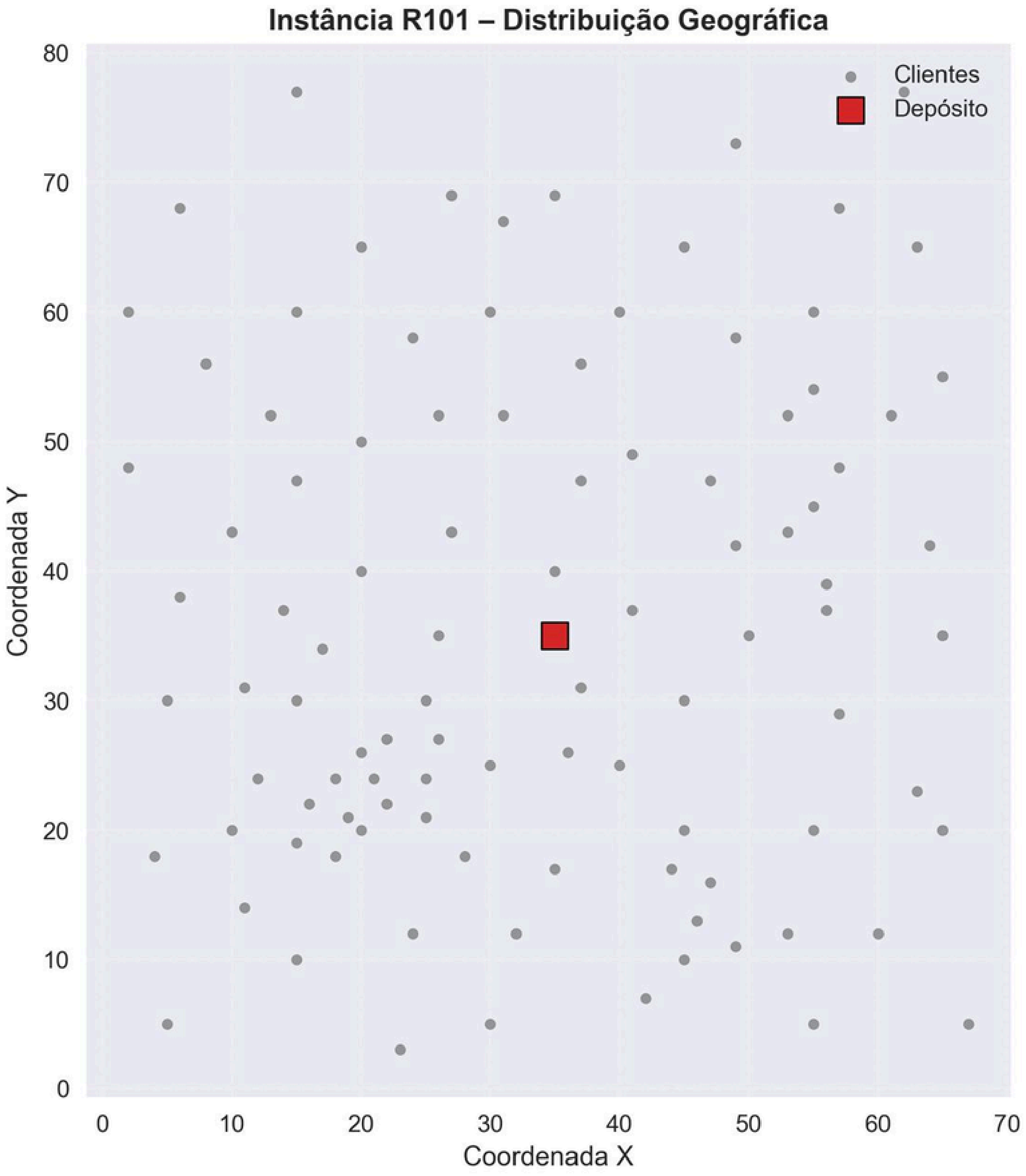
Instâncias	Veículos	Distância	BKS dist.	Gap dist.	Tempo
C101	10	829.94	828.94	0.00%	1.93
R101	20	1642.88	1650.80	0.48%	1.73
RC101	15	1623.58	1696.94	4.32%	901.33

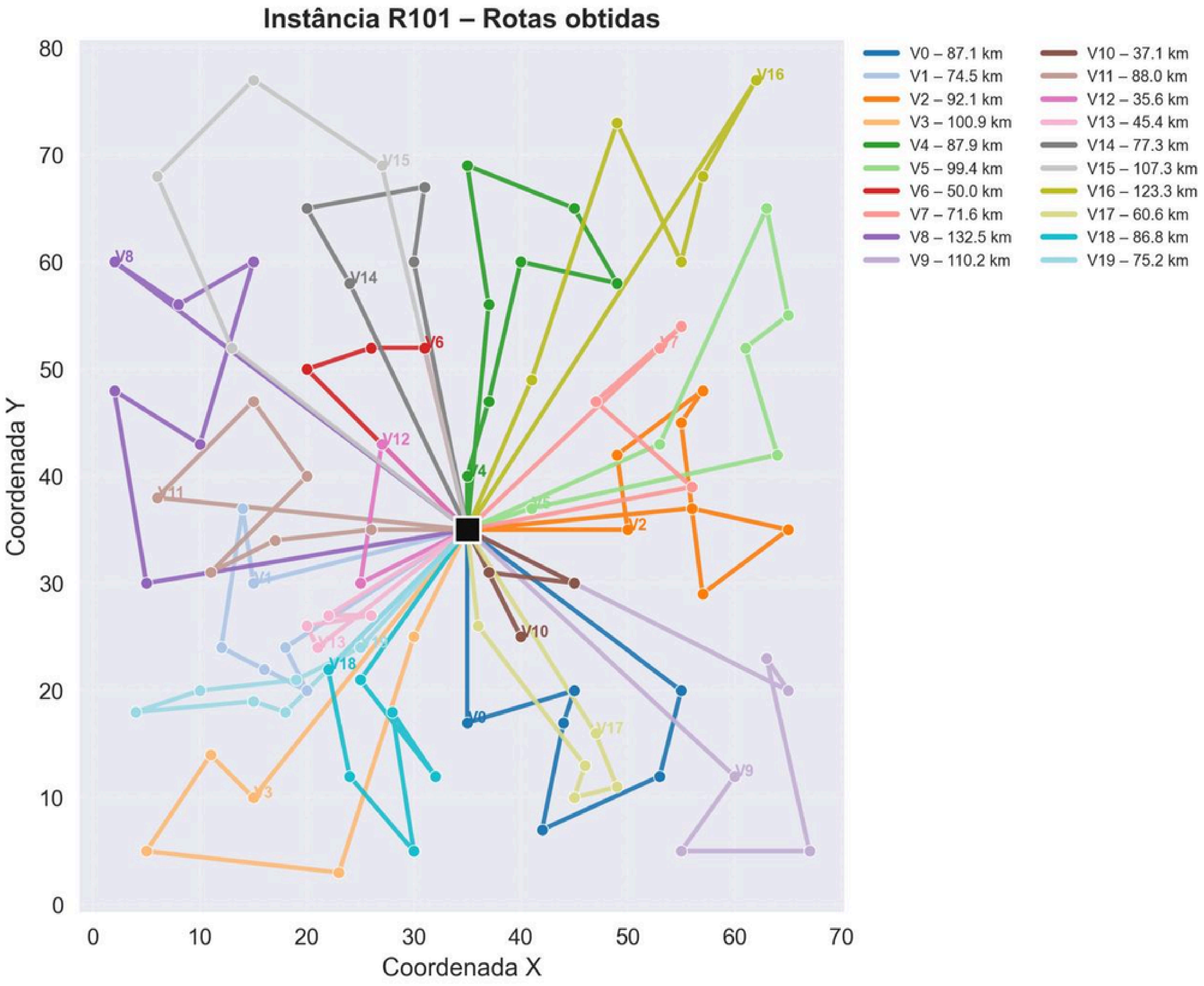
ROTAS OBTIDAS – C101



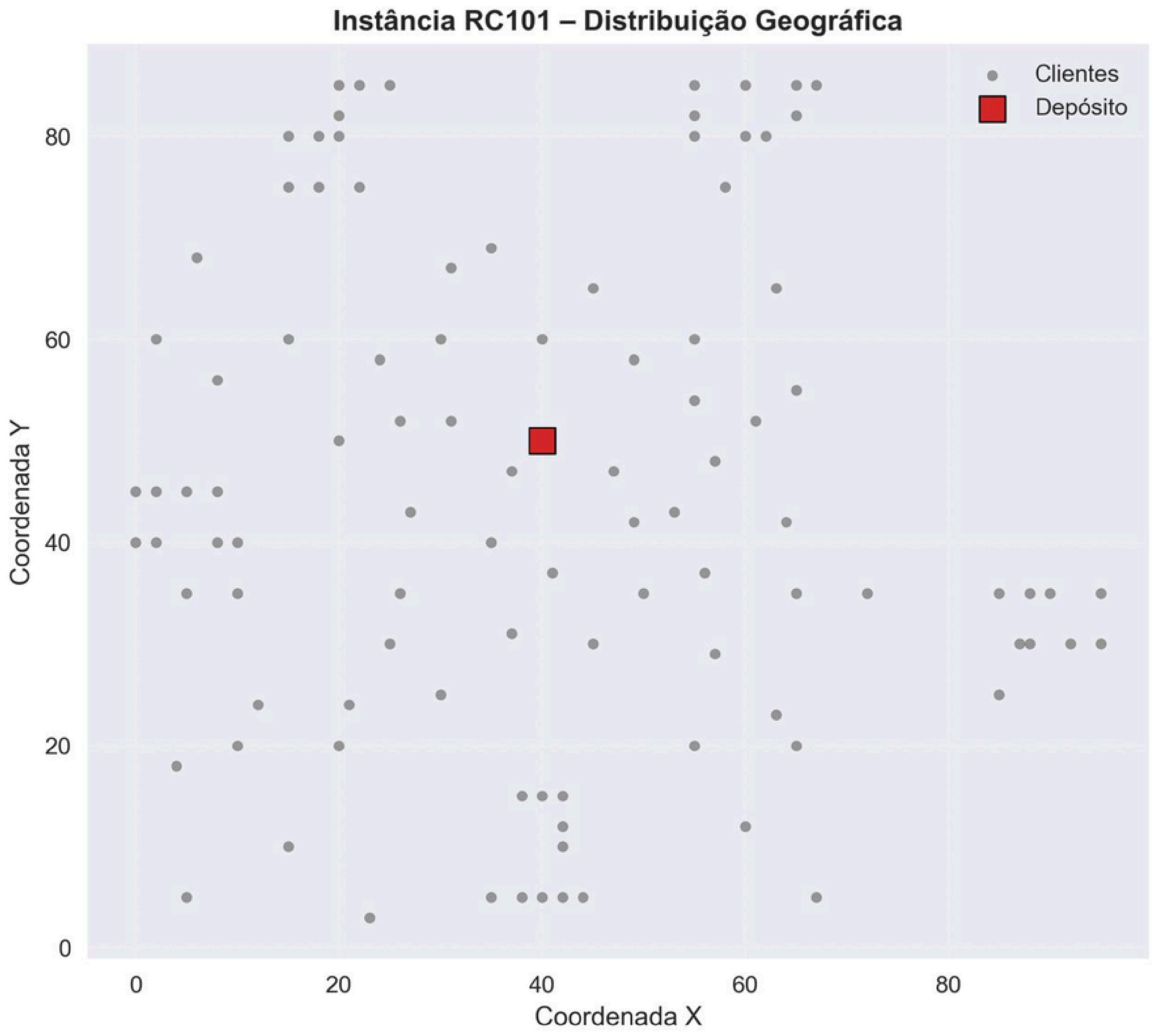


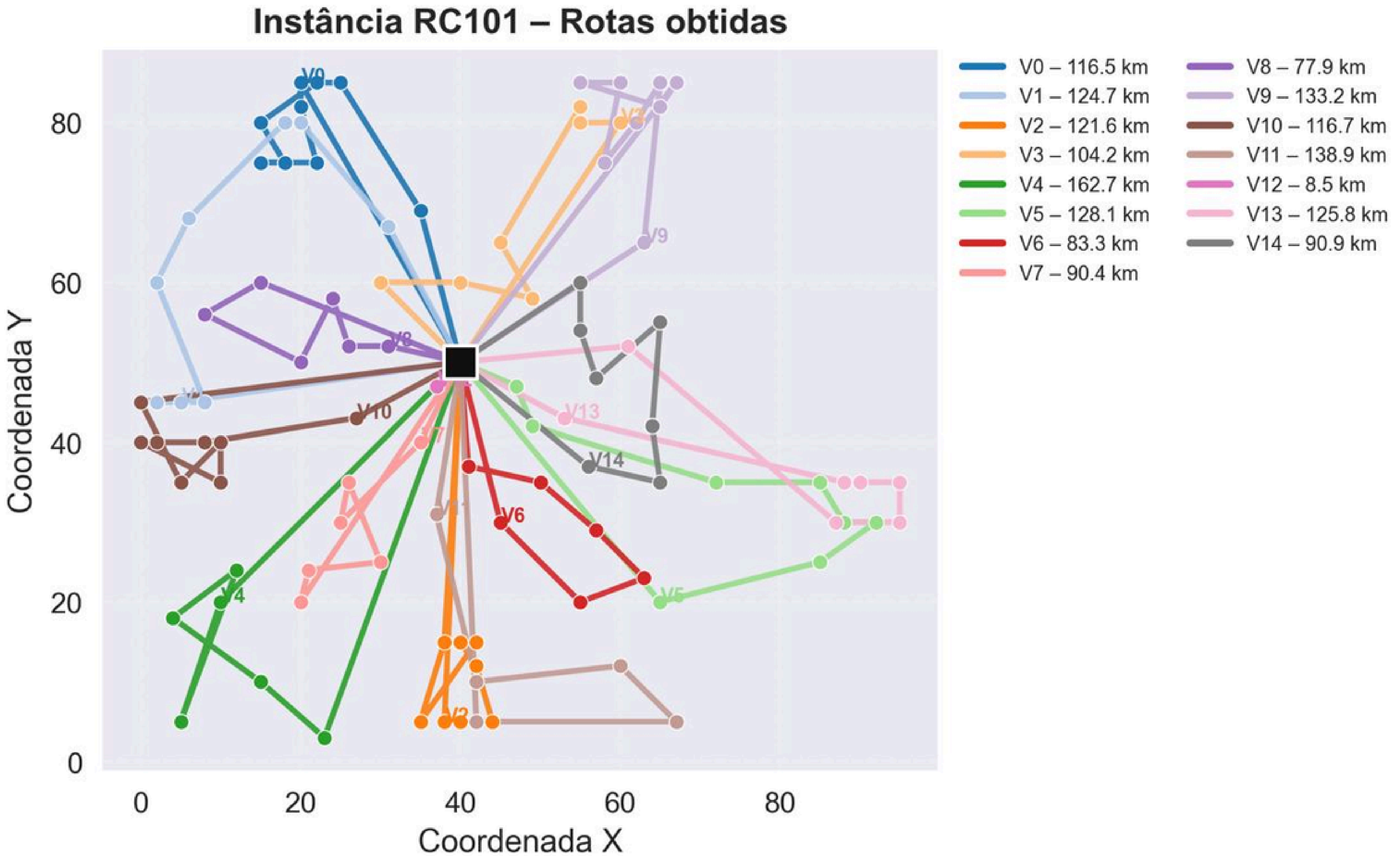
ROTAS OBTIDAS – R101





ROTAS OBTIDAS – RC101





CONCLUSÕES

✓ Modelo MILP resolve instâncias de 100 clientes eficientemente;

✓ Validado pela literatura;

✓ Formulação MTZ adaptada é didática e eficaz para este porte;

✓ Topologia dos clientes impacta drasticamente tempo de solução:

- Clusterizada (C101): < 2s;
- Híbrida (RC101): 15 min.

✓ Aprendizado sobre Big-M: Q como constante de relaxação para capacidade é matematicamente correto.

Trabalhos Futuros:

- Testar mais instâncias (C2, R2, RC2) com mais clientes;
- Comparar com heurísticas (Genetic Algorithm, Tabu Search).

REFERÊNCIAS

- ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. Pesquisa Operacional para Cursos de Engenharia. 1. ed. Rio de Janeiro: Elsevier, 2007. ISBN 978-85-352-5193-7.
- SOLOMON, M. M. Algorithms for the Vehicle Routing and Scheduling Problem with Time Window Constraints. Operations Research, v. 35, n. 2, p. 254–265, 1987.
- GUROBI OPTIMIZATION, LLC. Gurobi Optimizer Reference Manual. 2023.
- CORDEAU, J.; DESAULNIERS, G.; DESROSIERS, J.; SOLOMON, M. M.; SOUMIS, F. The Vehicle Routing Problem with Time Windows. In: The Vehicle Routing Problem. SIAM, 2002. p. 157–193.
- BATISTA, L. S.; CARRANO, E. G. Slides de aula da disciplina. Departamento de Engenharia Elétrica, Universidade Federal de Minas Gerais (UFMG), 2023.
- Vehicle Routing Problem. In: Wikipedia: The Free Encyclopedia. Disponível em: https://en.wikipedia.org/wiki/Vehicle_routing_problem. Acesso em: 24 out. 2025.

DÚVIDAS?