**Histórias Complexas e Reais para Aplicação de Algoritmos de Otimização de Rotas no Projeto A-MoVeR**

**Contexto Geral**

O projeto A-MoVeR (Advanced Mobility for e-Delivery Routing) surge como uma iniciativa pioneira em Vila Urbana para transformar o sistema logístico urbano através da adoção de motas elétricas em serviços profissionais. O objetivo é otimizar a mobilidade de empresas locais, melhorar a eficiência de serviços públicos e reduzir a pegada ecológica. Dentro deste ecossistema, o trabalho desenvolvido por Rodrigo Fernandes incide sobre o núcleo algorítmico de otimização de rotas, interligando-se a três projetos paralelos: interface de gestão, API de fornecimento de serviços e base de dados logística.

**Cenário 1: Rede Local de Distribuição Alimentar (Empresa DelivFresh)**

**História introdutória:**  
Todas as manhãs, às primeiras luzes do dia, a empresa DelivFresh inicia a sua operação de entrega de produtos frescos pelas ruas da Vila Urbana. Os clientes exigem produtos como pão ainda quente, bolos acabados de sair do forno e queijos regionais diretamente do produtor. A mota elétrica tem de cumprir um percurso exigente antes das 08:00 para manter a frescura e qualidade do serviço.

**Descrição detalhada:**  
A mota parte às 06:30 do armazém DelivFresh (nó 0) e deve entregar encomendas a seis estabelecimentos:

* Café Aurora (nó 1): 5kg de pão alentejano
* MiniMercado Sol Nascente (nó 2): 2kg de croissants e 2kg de bolas de berlim
* Padaria Real (nó 3): 3kg de bolas de berlim
* Escola Secundária Central (nó 4): 6kg de queijo de cabra curado
* Cantina da Câmara Municipal (nó 5): 5kg de pão rústico
* Quiosque Verde (nó 6): 2kg de queijo fresco e 2kg de croissants

Capacidade da mota: 25 kg | Autonomia: 10 kWh | Janela temporal: 06:30 – 08:00

**Algoritmo aplicado:** Tabu Search

**Justificação:** Dada a diversidade de produtos e exigência de cumprir uma janela de entrega restrita, é essencial utilizar uma metaheurística capaz de otimizar a rota com múltiplas restrições. O Tabu Search adapta-se bem a cenários com elevada complexidade combinatória.

**Resultado:**

✔ Roteamento: [[0, 1, 2, 3, 4, 5, 6, 0]]

📏 Distância total: 45.38 km

🔋 Energia estimada: 9.08 kWh

**Análise:** A rota cumpre todos os requisitos logísticos, assegurando a entrega completa dos produtos dentro do tempo e dos limites técnicos da mota.

**Cenário 2: Circuito de Recolha de Análises Clínicas (Unidade MobileSaúde)**

**História introdutória:**  
O laboratório MobileSaúde opera uma mota elétrica equipada com sistema refrigerado que percorre vários centros de saúde e escolas para recolher amostras clínicas. Este serviço é essencial para manter o fluxo de análises diárias e garantir diagnósticos em tempo útil. Cada paragem tem uma janela de tempo definida e a mota precisa de respeitar tanto os prazos como as condições térmicas exigidas.

**Descrição detalhada:**  
A mota parte do laboratório central (nó 0) às 08:00 e tem até às 11:30 para visitar cinco locais:

* Centro Saúde Norte (nó 1): 3kg de sangue (até às 09:00)
* Unidade de Apoio Escolar (nó 5): 3kg de sangue (entre 09:00 e 10:00)
* Centro Saúde Este (nó 2): 4kg de sangue e urina (até às 10:00)
* Hospital Local (nó 4): 6kg de material misto (entre 09:30 e 11:00)
* Extensão Poente (nó 3): 2kg de PCRs (até às 11:30)

Capacidade da mota: 18 kg | Autonomia: 10 kWh | Rota sensível à temperatura e tempo

**Algoritmo aplicado:** GRASP

**Justificação:** A aleatoriedade controlada do GRASP permite gerar múltiplas soluções e selecionar a melhor dentro das janelas temporais e de capacidade.

**Resultado:**

✔ Rota encontrada: [0, 1, 5, 2, 4, 3, 0]

📏 Custo total: 28.88 km

**Análise:** A solução respeita a ordem temporal das recolhas e mantém os limites de carga e autonomia, sendo apropriada para serviços médicos móveis.

**Cenário 3: Assistência Rápida a Equipamentos Urbanos (Serviço TecFast)**

**História introdutória:**  
Durante a tarde, uma avaria num painel solar na Escola Técnica ativa o protocolo de intervenção rápida da TecFast. Uma mota elétrica equipada com ferramentas parte imediatamente da garagem central. A cidade exige uma resposta célere e eficiente, respeitando os limites de bateria disponíveis no momento.

**Descrição detalhada:**  
A mota parte da garagem central (nó 0) às 15:12 após alerta para avaria na Escola Técnica (nó 3). Não transporta carga significativa, apenas ferramentas. Tem apenas 5 kWh disponíveis.

Destino: Escola Técnica (nó 3) | Janela: intervenção imediata | Autonomia limitada

**Algoritmo aplicado:** Dijkstra

**Justificação:** Como é necessária a rota mais curta possível com base na autonomia, o algoritmo de Dijkstra é ideal para encontrar o caminho mínimo entre dois pontos.

**Resultado:**

✔ Caminho encontrado: [0]

📏 Distância total: 8.6 km

🔋 Energia estimada: 1.72 kWh

**Análise:** O resultado indica rota direta. É importante garantir que o JSON define o nó de destino corretamente para que o algoritmo seja útil em outros casos semelhantes.

**Cenário 4: Logística de Armazéns Descentralizados (Parceria EcoPost)**

**História introdutória:**  
A EcoPost distribui material reciclado a escolas e papelarias a partir de dois depósitos, Norte e Sul. Cada mota parte do seu depósito com uma rota distinta e encomendas específicas. A separação logística e a otimização da rota são fundamentais para garantir sustentabilidade e eficiência energética.

**Descrição detalhada:**  
Duas motas partem simultaneamente dos depósitos Norte (nó 0) e Sul (nó 5):

* Mota Norte:
  + Escola 1 (nó 1): 8kg de kits de reciclagem
  + Papelaria Laranja (nó 2): 7kg de caixas A4 recicladas
* Mota Sul:
  + Escola 2 (nó 3): 6kg de tinteiros regenerados
  + Papelaria Azul (nó 4): 5kg de papel reciclado e kits

Capacidade de cada mota: 30 kg | Autonomia: 8 kWh | Objetivo: máxima eficiência logística

**Algoritmo aplicado:** Savings (Clarke & Wright modificado)

**Justificação:** Este algoritmo é ideal para agrupamento local de entregas com economias de percurso, especialmente em contextos com vários depósitos.

**Resultado:**

✔ Caminho encontrado: [0, 1, 2, 0]

Distância total: 25.62 km

✔ Caminho encontrado: [5, 3, 4, 5]

Distância total: 21.04 km

**Análise:** A alocação de rotas é equilibrada e compatível com a capacidade de cada mota, minimizando o tempo e o consumo total.

**Cenário 5: Visita Técnica a Microempresas (Serviço ProLocal)**

**História introdutória:**  
A empresa ProLocal presta serviços de manutenção técnica preventiva a pequenos estabelecimentos em Vila Urbana. Como o número de clientes por rota é reduzido, é possível otimizar perfeitamente a ordem de visita com um algoritmo exato. A mota parte da oficina e deve visitar 6 locais antes do meio-dia.

**Descrição detalhada:**  
A mota parte da oficina (nó 0) às 09:00 com ferramentas leves e visita:

* Barbearia Estilo (nó 1)
* Cafetaria Aroma (nó 2)
* Papelaria Central (nó 3)
* Loja EcoVida (nó 4)
* Alfaiataria Lino (nó 5)
* Mercearia 25 de Abril (nó 6)

Capacidade: não relevante  
Autonomia: 10 kWh  
Janela temporal: 09:00 – 12:00  
Restrições: foco em distância mínima

**Algoritmo aplicado:** Branch and Bound

**Justificação:**  
O Branch and Bound é ideal para instâncias pequenas (até 6 clientes), permitindo encontrar a rota exata de menor custo.

**Resultado:**

✔ Rota ótima: [0, 1, 3, 5, 4, 6, 2, 0]  
📏 Distância total: 21.4 km

**Análise:**  
Como o problema envolve um número reduzido de clientes, o algoritmo conseguiu avaliar todas as permutações possíveis de forma eficiente e determinou a rota exata de menor distância. Esta abordagem garante máxima eficiência logística em operações técnicas locais com baixo volume.

**Cenário 6: Entregas Expresso em Zonas Próximas (Serviço FastDrop)**

**História introdutória:**  
O serviço FastDrop realiza entregas expresso em zonas densamente povoadas de Vila Urbana. A prioridade é minimizar o tempo de deslocação com decisões simples e imediatas — ideal para casos em que o sistema precisa decidir rapidamente a rota baseada apenas na proximidade.

**Descrição detalhada:**  
A mota parte do centro logístico (nó 0) às 10:00 e deve visitar:

* Quiosque do Parque (nó 1)
* Loja BioMais (nó 2)
* Café MiraSol (nó 3)
* Papelaria Nova (nó 4)
* Florista RosaLinda (nó 5)
* MiniMercado Verde (nó 6)

Capacidade: 20 kg  
Autonomia: 10 kWh  
Janela temporal: até às 12:00  
Objetivo: decisão rápida com base na proximidade

**Justificação:**  
O algoritmo *Nearest Neighbor* é ideal para cenários com pressão de tempo e complexidade reduzida. Apesar de não garantir a rota ótima global, gera rapidamente soluções muito próximas do ideal. Com a aplicação do 2-opt, é possível melhorar a rota inicial obtida pela heurística.

**Algoritmo aplicado:** Nearest Neighbor

**Resultado:**

✔ Rota: [0, 1, 6, 4, 5, 2, 3, 0]  
📏 Distância: 17.63 km  
🔋 Energia estimada: 3.53 kWh

**Análise:**  
O algoritmo *Nearest Neighbor* é ideal para cenários com pressão de tempo e complexidade reduzida. Apesar de não garantir a rota ótima global, gera rapidamente soluções muito próximas do ideal. Com a aplicação do 2-opt, é possível melhorar a rota inicial obtida pela heurística.

**Conclusão Geral**

Este trabalho teve como objetivo aplicar e validar diferentes algoritmos de otimização de rotas no contexto urbano do projeto A-MoVeR, focado na utilização de motas elétricas em serviços logísticos. Através de quatro cenários distintos e baseados em situações realistas, foi possível perceber como cada abordagem algorítmica responde a restrições concretas como janelas temporais, capacidade de carga, autonomia energética e urgência de resposta.

Os resultados mostraram que não existe um algoritmo “melhor” de forma absoluta — a escolha depende fortemente do problema a resolver. O **Tabu Search** mostrou-se adequado para problemas com elevada complexidade combinatória e múltiplas exigências logísticas. O **GRASP** foi útil em situações com várias soluções viáveis onde é preciso encontrar rapidamente uma boa alternativa. O **Dijkstra** cumpriu bem o papel em rotas ponto-a-ponto com foco na minimização da distância, e o **Savings** demonstrou ser eficaz em distribuições logísticas com múltiplos depósitos.

Do lado da implementação prática, os scripts em Python desenvolvidos — nomeadamente o branch\_and\_bound.py e o nearest\_neighbor.py — permitiram não só testar soluções, mas também criar uma base modular para validação de diferentes heurísticas e estratégias. Mesmo com limitações (como o tempo de execução no Branch and Bound ou a simplicidade do Vizinho Mais Próximo), estes algoritmos provaram ser úteis enquanto ferramentas de análise e simulação.

No final, o objetivo foi cumprido: analisar, comparar e aplicar algoritmos a problemas reais, com dados concretos, dentro de um contexto técnico coerente. O trabalho não pretende reinventar a roda — mas contribui com solidez e clareza para o tema da otimização logística em mobilidade elétrica.