

ANÁLISE DE ROTAS SEGURAS PARA VEÍCULOS DE TRANSPORTE DE VALORES POR MEIO DE GRAFOS E ALGORITMOS DE OTIMIZAÇÃO (AEGIS)

Amanda Duarte Meneghini do Carmo; Andy Hyong Tae Choi Youn; Renato Gonçalves Machado; Sedra Alhendi; Bruno Henrique Oliveira Mulina (Dr).

Universidade Anhembi Morumbi.

12522192773@ulife.com.br.

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema para análise de rotas seguras destinadas a Veículos Especiais de Transporte de Valores. A pesquisa utilizou grafos como representação da malha urbana e aplicou estruturas de dados como LinkedList, pilhas e MinHeap para organizar as conexões e processar informações de risco e distância. Os algoritmos de Dijkstra e Tarjan foram empregados para identificar caminhos eficientes e localizar componentes fortemente conectados. Os resultados mostram que a abordagem permite avaliar trajetos com base em múltiplos critérios, oferecendo suporte ao planejamento seguro e otimizado de deslocamentos urbanos. O sistema demonstrou bom desempenho na identificação de áreas críticas e no cálculo de rotas adequadas ao contexto operacional de transporte de valores.

Palavras-chave: grafos, algoritmos, segurança viária

INTRODUÇÃO

O transporte de valores requer planejamento rigoroso, já que envolve grandes quantias de dinheiro e bens de alto risco. Veículos Especiais de Transporte de Valores dependem de rotas que reduzam ameaças, encurtem o deslocamento e atendam às exigências operacionais das empresas. A análise manual desses trajetos tende a ser lenta e sujeita a falhas, o que motivou a criação de ferramentas computacionais capazes de apoiar esse processo.

Este estudo teve como objetivo desenvolver um software capaz de analisar a malha urbana de forma estruturada, utilizando distância e nível de risco como parâmetros para selecionar trajetos adequados. A proposta buscou integrar estruturas de dados e algoritmos clássicos de análise de rotas, permitindo uma avaliação conjunta de eficiência e segurança. O trabalho se destaca por aplicar métodos consolidados da computação em um problema real, oferecendo uma solução de apoio à tomada de decisão em um setor sensível.

MÉTODOS

A pesquisa foi conduzida por meio da construção de um sistema baseado na modelagem da malha urbana como um grafo. Cada cruzamento foi tratado como um nó, e cada rua como uma aresta com peso. Esses pesos foram definidos com base em duas variáveis: distância física e nível de risco associado à região. Os dados foram previamente catalogados para permitir a simulação das rotas.

Para armazenar as relações entre os nós, utilizou-se uma LinkedList, escolhida por sua eficiência na manipulação de elementos adjacentes. O programa também fez uso de pilhas para detectar a presença de ciclos, informação útil para a identificação de regiões com possíveis entradas e saídas repetidas. Essa característica ajudou a avaliar vulnerabilidades operacionais.

O MinHeap foi utilizado na priorização de caminhos durante a execução do algoritmo de Dijkstra, garantindo menor custo computacional na busca pelos trajetos de menor peso. O algoritmo de Tarjan foi aplicado para reconhecer componentes

fortemente conectados, o que permitiu localizar áreas críticas em termos de circulação e segurança.

A implementação foi desenvolvida em Java e validada com dados simulados que representavam uma malha urbana típica de operações de transporte de valores. Os testes foram repetidos até a estabilidade dos resultados e a consistência da rota sugerida pelo sistema.

RESULTADOS E DISCUSSÕES

Os testes realizados com o sistema mostraram que a combinação entre estruturas de dados clássicas e algoritmos de grafos funciona bem para analisar rotas de transporte de valores. O algoritmo de Dijkstra conseguiu encontrar caminhos que equilibram distância e nível de risco, entregando rotas estáveis mesmo quando o grafo incluía regiões com peso elevado. O uso do MinHeap reduziu o tempo de processamento e manteve o desempenho em cenários com muitos nós. A aplicação do algoritmo de Tarjan também trouxe ganhos. Ele permitiu identificar áreas com forte conexão interna, ou seja, regiões em que o carro poderia entrar e sair repetidas vezes. Isso ajudou a destacar pontos que merecem atenção extra no planejamento por poderem facilitar perseguições ou emboscadas.

Durante a pesquisa, foi identificado que já existem estudos sólidos voltados ao roteamento de veículos. Um exemplo é o trabalho de Oliveira e Coelho, *Abordagens heurísticas de solução para Problemas de Roteamento de Veículos (PRV)*, que apresenta heurísticas capazes de gerar boas soluções para problemas de logística. Esses métodos são amplamente usados por empresas que precisam reduzir custos e melhorar a eficiência das rotas. A comparação com a literatura foi importante porque mostrou que o objetivo do nosso projeto não era competir com sistemas comerciais nem reproduzir ferramentas existentes. A proposta aqui era aprender e aplicar estruturas de dados sem recorrer a rotinas prontas como ArrayList ou HashMap, seguindo o critério definido na disciplina.

Além disso, o foco deste trabalho não foi resolver o PRV de maneira ampla, mas sim adaptar técnicas de grafos para um problema mais específico: encontrar rotas

mais seguras para o abastecimento de caixas eletrônicos. A escolha do tema surgiu da busca por um exemplo real que permitisse explorar teoria de grafos, manipulação manual de estruturas como LinkedList e pilhas, e implementação de algoritmos de forma íntegra. O resultado mostrou que, mesmo sem heurísticas avançadas, é possível criar um sistema útil e coerente com o objetivo da pesquisa.

No geral, o sistema criado cumpriu sua função. Ele analisou a malha urbana simulada, identificou áreas críticas e apresentou rotas compatíveis com os critérios de segurança definidos. A comparação com estudos da área mostrou que o trabalho segue uma linha diferente, mais voltada ao aprendizado de estruturas e lógica do que à criação de um otimizador comercial. Ainda assim, os testes indicam que a abordagem pode ser expandida no futuro e integrada a modelos mais completos de roteamento.

CONCLUSÕES

O desenvolvimento do sistema mostrou que a modelagem por grafos é uma ferramenta consistente para analisar rotas usadas no transporte de valores. As estruturas de dados implementadas manualmente, como LinkedList, pilhas e MinHeap, funcionaram bem no apoio aos algoritmos escolhidos. O Dijkstra foi eficiente na identificação de caminhos que equilibram distância e risco, enquanto o algoritmo de Tarjan ajudou a reconhecer áreas com maior complexidade estrutural, úteis para avaliar pontos de atenção no planejamento.

O projeto atingiu o objetivo proposto na disciplina: aplicar estruturas de dados sem recorrer a bibliotecas prontas e construir uma solução funcional baseada em lógica e análise. Embora o sistema não tenha a intenção de competir com ferramentas comerciais ou com heurísticas avançadas presentes na literatura, ele demonstrou que a abordagem escolhida é capaz de gerar resultados coerentes com o problema de abastecimento seguro de caixas eletrônicos.

As conclusões obtidas se aplicam ao conjunto de dados usado nos testes.

Trabalhos futuros podem ampliar o sistema, incluir mapas reais, ajustar pesos de

risco e integrar outros métodos de otimização para aproximar o modelo de cenários operacionais mais complexos.

REFERÊNCIAS

W3SCHOOLS. **DSA Linked Lists**. Disponível em:

<https://www.w3schools.com/dsa/dsa_theory_linkedlists.php>. Acesso em: 17 nov. 2025.

W3SCHOOLS. **DSA Graphs**. Disponível em:

<https://www.w3schools.com/dsa/dsa_theory_graphs.php>. Acesso em: 17 nov. 2025.

W3SCHOOLS. **DSA Stacks**. Disponível em:

<https://www.w3schools.com/dsa/dsa_data_stacks.php>. Acesso em: 17 nov. 2025.

W3SCHOOLS. **DSA Dijkstra's Algorithm**. Disponível em:

<https://www.w3schools.com/dsa/dsa_algo_graphs_dijkstra.php>. Acesso em: 17 nov. 2025.

MENEGHINI, Amanda; HYOUNG, Andy; MACHADO, Renato; ALHENDI, Sedra;.

GitHub - AmandaMeneghini/aegis-core-logic: Base project for the “Aegis” system.

This repository contains custom data structures (LinkedList, Stack, MinHeap) and core graph algorithms (Dijkstra, Tarjan's) built from scratch in pure Java. This project serves as the foundational logic engine for a future Spring Boot API. Disponível em:

<<https://github.com/AmandaMeneghini/aegis-core-logic>>. Acesso em: 17 nov. 2025.

DE OLIVEIRA, Joaquim Rodrigo; COELHO, Antonio Sérgio. Abordagens heurísticas de solução para Problemas de Roteamento de Veículos (PRV). **Revista ESPACIOS | Vol. 36 (Nº 13) Año 2015, 2015**. Disponível

em:<<https://www.revistaespacios.com/a15v36n13/15361303.html>>. Acesso em: 17 nov. 2025.