



Ciência da Computação & Sistemas de Informação

APS - Atividades Práticas Supervisionadas

5^o e 6^o Semestre – 2026/1

Inteligência Artificial: Espaço de Busca do TSP

Sumário

1	Projeto e Desenvolvimento	2
2	Tema e Objetivos	3
2.1	Tema	3
2.2	Objetivo Geral	3
2.3	Objetivos Específicos	3
3	Conteúdo do Projeto	4
3.1	Bases de Dados (Datasets)	4
3.2	Desenvolvimento e Conceitos Computacionais	4
3.2.1	Busca em Espaço de Estados (Solução Ótima)	4
3.2.2	Branch and Bound (Solução Ótima Otimizada)	5
3.2.3	Heurística (Solução Aproximada)	5
4	Estrutura do Trabalho Escrito	6
	Instruções para o Desenvolvimento	7
5	Avaliação	8
	Modelo Ficha	9

Capítulo 1

Projeto e Desenvolvimento

O Projeto e Desenvolvimento tem como finalidade proporcionar ao aluno a aplicação prática dos conhecimentos adquiridos durante o curso, promovendo a integração entre teoria e prática.

Neste semestre, o projeto será desenvolvido com foco na elaboração de um algoritmo e programa para resolver o TSP, um problema clássico NP-Completo.

O objetivo é entender que uma solução ótima do TSP, para uma aplicação do mundo real, não é viável. Sendo preciso procurar uma solução alternativa que não seja ótima.

Capítulo 2

Tema e Objetivos

2.1 Tema

Análise Comparativa de Algoritmos para o Problema do Caixeiro Viajante (TSP)

2.2 Objetivo Geral

O objetivo deste trabalho é explorar a resolução do Problema do Caixeiro Viajante (Traveling Salesman Problem - TSP), um problema clássico de otimização combinatória classificado como NP-difícil.

O desafio consiste em encontrar a rota mais curta que visite um conjunto de cidades exatamente uma vez e retorne à cidade de origem. O grupo deve implementar e comparar três abordagens distintas, analisando o *trade-off* entre tempo de processamento e qualidade da solução.

2.3 Objetivos Específicos

- Implementar um grafo;
- Ler um grafo de um arquivo;
- Implementar corretamente uma busca em um espaço de estados do TSP;
- Compreender que um algoritmo ótimo do TSP é inviável;
- Implementar uma heurística;
- Comparar os resultados significativos;

Capítulo 3

Conteúdo do Projeto

3.1 Bases de Dados (Datasets)

As instâncias para teste e validação dos algoritmos devem ser extraídas das seguintes fontes acadêmicas:

- TSPLIB95: <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>
- World TSP (University of Waterloo): <https://www.math.uwaterloo.ca/tsp/world/countries.html>
- Repositório do professor Vinicius de Novaes com alguns exemplos organizados das fontes anteriores: https://github.com/viniciusdenovaes/TSPLib-APS_IA

3.2 Desenvolvimento e Conceitos Computacionais

O trabalho será dividido em três etapas principais de implementação:

3.2.1 Busca em Espaço de Estados (Solução Ótima)

Nesta etapa, o grupo deve implementar um algoritmo que faça uma busca exaustiva no Espaço de Estados (podendo ser uma busca em Profundidade ou Largura).

Conceito: Busca em Espaço de Estados (Solução Ótima) Nesta etapa, o grupo deve implementar um algoritmo que faça uma busca exaustiva no Espaço de Estados (podendo ser uma busca em Profundidade ou Largura).

Análise: Deve-se discutir a explosão combinatória de complexidade $O(n!)$. Neste caso teremos $n!$ possibilidades, sendo n o número de vértices no grafo, mostrando como o tempo de execução cresce exponencialmente com o aumento do número de cidades (n).

3.2.2 Branch and Bound (Solução Ótima Otimizada)

A aplicação da técnica de Branch and Bound visa encontrar **a mesma** solução ótima do passo anterior, porém de forma mais eficiente.

Conceito: A técnica utiliza a árvore de busca onde cada nó representa uma solução parcial. O "Branch" (ramificação) gera os filhos de cada vértice, enquanto o "Bound" (limite) calcula um limite inferior de custo para cada ramo.

Poda (Pruning): Se o limite inferior de um ramo for maior que a melhor solução já encontrada (upper bound), o ramo é descartado ("podado"), evitando a exploração de caminhos inúteis e reduzindo o tempo de processamento em relação à busca exaustiva simples.

Complexidade: Deve-se discutir a melhora no tempo de execução em relação ao anterior.

3.2.3 Heurística (Solução Aproximada)

O grupo deve implementar uma técnica que não garante a otimalidade, mas entrega uma solução "boa o suficiente" em tempo polinomial. Para este item, sugere-se o uso de Soluções iniciais Greedy Algoritmos Genéticos, Colônia de Formiga ou Busca Local (ex: 2-opt). O grupo pode escolher qualquer um.

Conceito: Diferente dos métodos exatos, as heurísticas focam na eficiência computacional. Uma solução Greedy, por exemplo, pode escolher apenas as menores rotas disponíveis até que uma solução seja formada. No caso de Algoritmos Genéticos, utiliza-se conceitos de evolução biológica (população, cruzamento, mutação e seleção natural) para iterativamente melhorar as rotas.

Análise: O foco aqui é demonstrar que, para instâncias grandes (onde os passos 1 e 2 falham por tempo/memória), a heurística consegue fornecer resultados viáveis rapidamente.

Capítulo 4

Estrutura do Trabalho Escrito

O trabalho escrito deverá conter:

- Capa;
- Introdução;
- Descrição do problema tratado;
- Descrição das soluções propostas;
- Descrição das técnicas utilizadas;
- Prints das telas desenvolvidas;
- Conclusão;

Instruções para o Desenvolvimento e Entrega

- O trabalho deve ser realizado em grupo de até 5 alunos;
- Cada grupo deverá apresentar oralmente o projeto.
- A entrega do trabalho escrito deverá ser realizada exclusivamente pelo sistema de trabalhos acadêmicos da UNIP, disponível em: <https://trabalhosacademicos.unip.br/entrega/>.
- A entrega do trabalho deverá ser feita até a Data-limite para a entrega ou postagem dos trabalhos acompanhados das fichas de registro das Atividades Acadêmicas (APS, AC, EC, TC e Extensão) e resolução de exercícios (ED) no sistema acadêmico, pelos alunos. Disponível em: <https://www.unip.br/servicos/aluno/manual/tradicional/calendarios.aspx>.

Capítulo 5

Avaliação

A avaliação será baseada nos seguintes critérios:

- Organização e estrutura do código com leitura do grafo (2,0 pontos);
- Implementação correta da busca em um espaço de estados (2,0 pontos);
- Otimização da busca (2,0 pontos);
- Heurística (2,0 pontos);
- Documentação (2,0 pontos);

Total: 10,0 pontos.

