

AULA 6: Exercício teórico complexidade de problemas

Aluno: Gian Franco Joel Condori Luna

September 24, 2024

Exercices

- 1 (0.5) Pesquise um problema que pode ser resolvido em tempo polinomial (classe P) diferente do problema de ordenação já visto. Explique do que trata o problema, qual o(s) algoritmo(s) proposto(s) para esse problema e qual a complexidade dele (linear, $n \log n$, quadrática, etc). Cite as fontes consultadas!.

Solução:

Problema do caminho mais curto em grafos

Explicação do Problema

Dado um grafo ponderado, o objetivo é encontrar o caminho com a menor soma de pesos entre um vértice de origem e um vértice de destino. Esse problema tem várias aplicações práticas, como em redes de transporte, rotas de navegação e telecomunicações.

Algoritmos Propostos

- Algoritmo de Dijkstra: É um dos algoritmos mais conhecidos para resolver o problema do caminho mais curto, que funciona para grafos com arestas de pesos não negativos. O algoritmo usa uma abordagem gulosa para explorar os vértices do grafo e calcular o menor caminho possível desde a origem até os outros vértices.
- Algoritmo de Bellman-Ford: Também pode ser utilizado, sendo mais geral, pois lida com arestas de pesos negativos.

Complexidade

- O Algoritmo de Dijkstra tem complexidade $O(V^2)$, onde V é o número de vértices, em sua implementação mais simples com uma matriz de adjacências. Entretanto, ao utilizar uma fila de prioridades (heap), a complexidade pode ser reduzida para $O(V \log V + E)$, onde E é o número de arestas.
- O Algoritmo de Bellman-Ford tem complexidade $O(V \cdot E)$, pois para cada vértice ele realiza um relaxamento das arestas.

Fontes Consultadas

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press.
- ChatGPT

- 2 (0.5) Pesquise um problema que não pode ser resolvido em tempo polinomial (classe NP). Explique do que trata o problema, qual o n máximo que o algoritmo por 'força bruta' já foi testado, se existe algoritmo(s) heurístico(s) que são aplicados ao problema para soluções parciais. Cite as fontes consultadas!.**

Solução:

Problema do Caixeiro Viajante (TSP - Travelling Salesman Problem)

Explicação do Problema

O Problema do Caixeiro Viajante (TSP) é um dos problemas mais conhecidos da classe NP-completo. O problema consiste em, dado um conjunto de cidades e as distâncias entre cada par de cidades, encontrar o caminho mais curto que permita visitar todas as cidades exatamente uma vez e retornar à cidade de origem. O TSP tem diversas aplicações práticas, como otimização de rotas de entrega, circuitos eletrônicos e planejamento logístico.

Este problema é NP-completo, o que significa que não existe, até o momento, um algoritmo que o resolva em tempo polinomial, e que, dado uma solução candidata, podemos verificar sua correção em tempo polinomial. Os algoritmos conhecidos para resolver o TSP de forma exata, como o método de força bruta, têm uma complexidade exponencial.

Algoritmo por Força Bruta

O algoritmo de força bruta para o TSP tenta todas as permutações possíveis das cidades para encontrar o menor caminho, resultando em uma complexidade de

$O(n!)$, onde n é o número de cidades. Isso torna o problema computacionalmente intratável para grandes valores de n .

Até o momento, o maior número de cidades para o qual o TSP foi resolvido de forma exata por força bruta é pequeno (em torno de 20 cidades), devido ao rápido crescimento da quantidade de combinações possíveis.

Algoritmos Heurísticos e Aproximações

Devido à complexidade exponencial, vários algoritmos heurísticos e de aproximação foram desenvolvidos para resolver instâncias maiores do TSP de forma aproximada. Alguns dos algoritmos mais conhecidos são:

- Algoritmo Guloso: seleciona a próxima cidade mais próxima da atual, mas não garante a solução ótima.
- Algoritmo de Vizinho Mais Próximo: começa em uma cidade e vai visitando a cidade mais próxima disponível. A solução é rápida, mas geralmente longe da ideal.
- Algoritmos Genéticos: simulam o processo de seleção natural para gerar soluções aproximadas.
- Algoritmo de Simulated Annealing: inspirado no processo de resfriamento de metais, faz pequenas alterações na solução atual para encontrar soluções melhores.
- Algoritmo de Colônia de Formigas: simula o comportamento de formigas em busca de comida para otimizar a rota.

Esses algoritmos heurísticos são amplamente usados para instâncias grandes do TSP (com centenas ou até milhares de cidades) e conseguem encontrar soluções próximas da ótima em tempo razoável.

Fontes Consultadas

- Applegate, D. L., Bixby, R. E., Chvátal, V., & Cook, W. J. (2006). The Traveling Salesman Problem: A Computational Study. Princeton University Press.
- ChatGPT