

Análise de soluções para o Problema do Caixeiro Viajante (Travelling Salesman Problem)

Avelar Ribeiro Hostalácio

Universidade Federal de Minas Gerais

1 Introdução

O Problema do Caixeiro Viajante (TSP) é um problema muito relevante nos campos de Ciência da Computação e Pesquisa Operacional, com importantes aplicações em diversas áreas, como logística e redes de computadores. O problema consiste em encontrar a menor rota para percorrer uma série de cidades, retornando à cidade de origem. O TSP pertence à classe de problemas NP-difíceis, o que significa que não se conhece uma solução eficiente em termos de tempo para instâncias grandes. Sua relevância prática reside na necessidade de otimizar rotas de entrega, reduzir custos operacionais e maximizar a eficiência em diversas situações do mundo real.

Este trabalho visa testar e comparar diferentes alternativas de encontrar uma solução para o TSP em tempo hábil. Os algoritmos utilizados são o Branch and Bound, Twice Around the Tree, e o algoritmo de Christofides.

2 Implementação

2.1 Estruturas de Dados

As instâncias do problema são armazenadas em grafos não direcionados. Cada nó do grafo representa uma cidade e cada aresta representa uma rota entre duas cidades. A implementação da estrutura foi feita com a utilização da biblioteca `networkx` para manipulação e análise de grafos.

2.2 Algoritmos

2.2.1 Branch and Bound

Esse algoritmo consiste em uma maneira sistemática de abordar o problema, explorando a árvore de possíveis soluções e podando ramos que não resultam em uma solução ótima. Por possuir uma complexidade assintótica de $O(n!)$, essa solução, apesar de fornecer a solução ótima é adequada apenas para instâncias de tamanho menor.

A abordagem escolhida para o algoritmo é uma best-first search, priorizando nós com melhor limite estimado. Essa abordagem foi escolhida para que o caminho mais promissor na árvore seja seguido primeiro, aumentando a probabilidade de podar ramos com soluções inferiores.

2.2.2 Christofides

Esse algoritmo consiste em uma abordagem heurística que fornece uma solução até 1.5 vezes pior do que a ótima. A complexidade do algoritmo é derivada do custo de encontrar um emparelhamento mínimo sendo $O(n^3)$, o que o torna apropriado para instâncias de tamanho moderado.

2.2.3 Twice Around The Tree

Este método fornece uma solução aproximada para o problema, através da construção de uma árvore geradora mínima ($O(E \log V)$) seguida de uma busca em profundidade ($O(V+E)$). Este algoritmo é indicado para obter uma solução inicial rápida ou para instâncias de tamanho grande.

2.3 Experimentos e Resultados

Os experimentos foram conduzidos com instâncias do Problema do Caixeiro Viajante disponibilizadas na biblioteca TSPLIB95. Os algoritmos foram avaliados com os seguintes critérios: tempo de execução, uso de memória e precisão da solução.

2.3.1 Branch and Bound

O algoritmo Branch and Bound se mostrou inviável até mesmo para o menor dataset ($n=50$) devido à sua complexidade.

2.3.2 Comparação entre Christofides e Twice Around the Tree

Esta seção utiliza gráficos para comparar os algoritmos em cada uma das métricas de desempenho em função do tamanho da entrada.

Ambos os algoritmos apresentam um aumento no uso de memória aproximadamente linear de acordo com o crescimento da entrada.

Em termos de tempo de execução, o Twice Around the Tree se mantém baixo em comparação ao Christofides, que apresenta um aumento expressivo conforme cresce o dataset.

Para avaliar a qualidade das soluções dos algoritmos, foi calculado o desvio médio das soluções em relação ao ótimo do problema. O algoritmo de Christofides apresentou um desvio médio de 11.76% enquanto o Twice Around the Tree apresentou um desvio de 34.70%. Logo, conclui-se que o algoritmo de Christofides possui uma precisão maior.

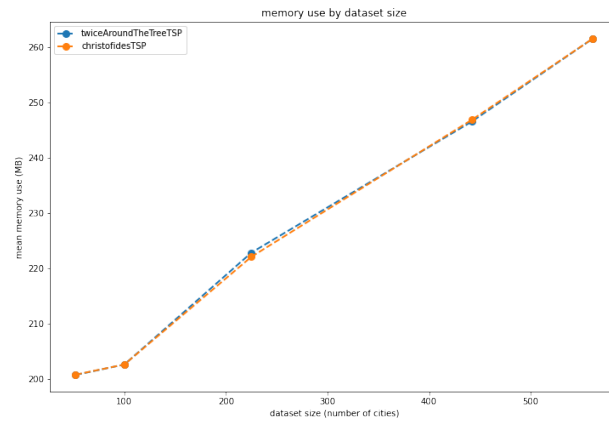


Figure 1: Uso de memória dos algoritmos

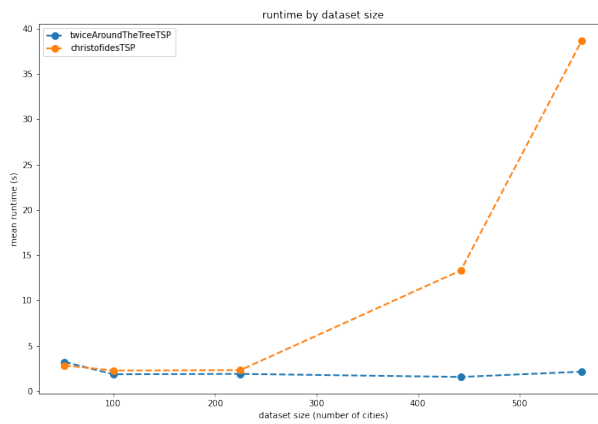


Figure 2: Tempo de execução dos algoritmos

2.4 Conclusão

Em conclusão, a escolha entre esses métodos para resolver o PCV depende da natureza específica da instância do problema em questão, bem como das necessidades práticas do contexto em que está sendo aplicado.

Enquanto o Branch and Bound oferece garantias de otimalidade, ele demonstra limitações significativas em termos de aplicações práticas devido à alta complexidade assintótica. o Christofides oferece soluções aproximadas viáveis em um tempo computacional mais acessível, possuindo um baixo desvio médio. Já o Twice Around the Tree se destaca por ser extremamente eficiente, podendo fornecer uma solução rapidamente mesmo para valores altos de n .

Portanto, a seleção do método mais apropriado deve ser cuidadosamente ponderada em função das características específicas de cada aplicação e das demandas de eficiência computacional.

References

- [1] Wikipedia. Travelling salesman problem. In *Wikipedia, The Free Encyclopedia*. [Online; accessed 09-December-2023] Retrieved from https://en.wikipedia.org/wiki/Travelling_salesman_problem