

Prática Investigativa

Artigo:

The Traveling Salesman Problem: An overview of exact and approximate algorithms, Gilbert Laporte, European Journal of Operational Research 59 (1992) 231-247

Luiza Ávila

O artigo apresenta os principais algoritmos do problema do caixeiro viajante, definindo o problema, sua complexidade e suas aplicações atuais. O autor enuncia os diferentes algoritmos usados para auxiliar a resolução do problema (inteiros lineares, branch-and-bound, spanning tree, entre outros) e explica o uso deles em algoritmos exatos. O autor também ressalta a importância do problema TSP (caixeiro viajante) no desenvolvimento teórico computacional.

Para mim, os pontos positivos do artigo são a apresentação de aplicações que também podem ser consideradas TSP e a explicação de forma consistente; e a menção casos especiais de TSP que podem ser resolvidos em tempo polinomial.

Como pontos negativos, achei que o artigo no geral possuía uma linguagem confusa, o nível de matemática do artigo não consegue ser entendido por um graduando no primeiro ano como o autor pretendia e ele poderia ter explicado melhor no final por que os melhores algoritmos são os melhores.

Algoritmos Exatos:

- Integer linear programming formulations

Todas as combinações são checadas, e se a combinação for um circuito que envolve todos os vértices do grafo passando apenas 1 vez por cada um, a soma é feita e o menor resultado entre todas as combinações válidas é escolhido.

O autor diferencia os de dois artigos, DFJ e MTZ, pontuando que entre todas formulações propostas e comparadas a que possui o relaxamento linear mais forte é o de DFJ.

- The assignment lower bound and related branch-and-bound algorithms
Algoritmos branch-and-bound são os mais utilizados em problemas TSP.

Divide o problema em duas partes: um assignment problem (AP) que acha um limite para ser aplicado a um branch-and-bound, e o próprio branch-and-bound, que usa subtours e subproblemas para encontrar arestas impróprias.

O autor apresenta três algoritmos que usam o método: Carpaneto and Toth (1980), Balas and Christofides (1981) and Miller and Pekny (1991).

- The shortest spanning arborescence bound and a related algorithm

Esta abordagem também usa branch-and-bound, mas para o valor inicial, é gerada a arborescência mínima, e então a anti-arborescência desta, de forma que com as duas ligadas há um circuito inicial que serve de limite para o branch-and-bound.

O autor apresenta o exemplo de Fischetti and Toth (1991).

- The shortest spanning tree bound and related algorithms

Similar ao método anterior, mas otimizado para grafos simétricos. O valor inicial, ao contrário do anterior, é obtido calculando a menor 1-árvore (árvore contendo apenas 1 vértice) mais duas arestas distintas que incidem a este vértice.

- The 2-matching lower bound and related algorithms

Valida várias restrições antes de ramificar em variáveis fracionárias, aumentando o valor do relaxamento, e limitando o crescimento da árvore de pesquisa. Elimina subrotas, depois procura pares de desigualdades correspondentes, combinando-as e forma o clique na árvore.

Algoritmos Heurísticos

- Heuristics with guaranteed worst-case performance

O algoritmo primeiramente transforma o grafo em um grafo direcionado, depois busca eliminar as arestas que incidem sobre vértices já visitados.

Uma alternativa para melhorar esta solução é ligar os vértices que possuem grau ímpar, visando obter um atalho caso precise depois.

O autor cita Christofides (1976) como um dos principais contribuintes.

- Heuristics with good empirical performance

O autor explica que os melhores métodos são os algoritmos compostos, que combinam procedimentos de construção de tour e procedimento de melhoria de tour.

- Tour construction procedures

A ideia principal do algoritmo é gradualmente construir uma solução pela adição de um novo vértice em cada passo. Ele apresenta 3 principais algoritmos:

- 1) Vizinho mais próximo: constrói um caminho possível tomando a decisão mais vantajosa para cada passo.
- 2) Algoritmo de inserção: consiste em cada passo inserir um vértice no caminho que satisfaça um critério definido.
- 3) Algoritmo de patch para TSPs assimétricos: explorando o problema de os vértices estarem uniformemente distribuídos, ligando-os até que se forme apenas um caminho.

- Tour improvement procedures

A ideia do algoritmo é melhorar uma solução viável executando várias trocas. Divido em 3 principais algoritmos:

1. Algoritmo r-opt: remove um número r de ligações (consecutivas ou não) do caminho e tenta reconectá-los querendo obter um caminho de menor custo.

2. Simulado de anelamento: busca mover de uma solução inicial para outra com o menor custo, analisando todas as soluções vizinhas e reduzindo o número de mudanças permitidas à medida que o algoritmo avança.
3. Pesquisa tabu: busca evitar a criação de ciclos, fazendo com que as soluções que já foram examinadas sejam deixadas de lado e colocadas na “lista Tabu”.
 - Composite algorithms

O método baseia-se na junção dos dois últimos. O autor apresenta 2 algoritmos principais

1. CCAO heuristic: explora uma propriedade do problema, que são os vértices localizados na parte convexa do grafo, visitando-os na ordem em que aparecem.
2. GENIUS: insere os vértices na solução de maneira mais cautelosa, e não na ordem que aparecem, além de uma otimização posterior, reinserindo os vértices.