
PROJETO 1

Marcos Godinho Filho
cc22142@g.unicamp.br

Éric Carvalho Figueira
cc22156@g.unicamp.br

1 Introdução

1.1 Definição e explicação

O problema dos múltiplos caixeiros viajantes, ou Multiple Travelling Salesman Problem (mTSP), é definido da seguinte forma: dado um conjunto de N cidades e M caixeiros, qual a melhor forma dos M caixeiros percorrerem as N cidades, sendo a soma dos caminhos percorridos por cada caixeiro a menor possível? O problema apresenta, ainda, as seguintes restrições: a) cada caixeiro deve sair de uma mesma cidade inicial; b) cada caixeiro deve passar por um número indefinido de cidades, visitando cada cidade uma única vez; c) um caixeiro não pode percorrer uma cidade que já foi visitada por outro caixeiro; d) cada caixeiro deve retornar à cidade inicial. A pergunta acima pode ser respondida pelo conjunto de percursos feitos por cada caixeiro, sendo cada caminho uma ordenação das cidades que o constituem.

1.2 Aplicação e relevância

O mTSP é usado, majoritariamente, para encontrar boas rotas em problemas de transportes, apesar de seu uso se estender para "agendamento de tarefas, armazenamento de produtos ou posicionamento de objetos"[1], tornando-o um objeto de estudo extremamente relevante.

Entre as possíveis aplicações do problema, vale citar:

- O planejamento de rotas de ônibus escolares, onde os veículos têm de passar por múltiplas paradas da maneira mais otimizada possível;
- E a realização de sessões de observação entre receptores de sinais de satélite, que deve ser feita na melhor ordem possível, visando determinar a posição geográfica de um ponto.

2 Heurística construtiva

A heurística construtiva pensada é formada pelas seguintes etapas:

- Inicialmente, encontramos a cidade cuja soma das distâncias euclidianas entre ela e cada outra cidade do conjunto é a menor possível, que será chamada de centroide.
- Em seguida, a ideia é formar um polígono que contenha o centroide como seu ponto interno e cujo perímetro seja formado por conexões entre as demais cidades, de modo que elas circundem a cidade inicial. Para alcançar esse objetivo, fazemos os procedimentos a seguir:

1. Criamos conexões entre o centroide e cada outra cidade do conjunto. Para entender a intenção por trás desse item, imagine um ângulo \hat{A} , que possui como vértice o centroide, e cujos segmentos que o constituem possuem, cada um, os seguintes pontos: duas cidades quaisquer entre as demais. Queremos que, iniciando de uma cidade C , a próxima cidade P a ter uma ligação com C seja a que gere o menor ângulo \hat{A} entre C e P .

Em outras palavras, queremos que o polígono gerado tenha sido criado imitando o sinal de um radar, que gira ao redor de um ponto (o centroide) e encontra objetos (as cidades) conforme avança (sendo as cidades conectadas na ordem em que foram encontradas partindo de um ponto inicial). As ligações entre as cidades e o centroide, então, servem para simular o sinal desse radar, permitindo futuras verificações de colisão entre caminhos (que aumentam a eficiência do algoritmo).

2. Para cada cidade A mais próxima do centroide, em ordem crescente de proximidade, visitamos as duas cidades mais próximas desta e criamos ligações entre elas e A , considerando, nesse processo, as restrições abaixo:
 - As ligações a serem criadas não podem gerar intersecções com nenhuma outra conexão entre cidades*. Isso é feito para evitar que um caminho se sobreponha ao outro, o que diminui a eficiência do algoritmo.

*OBS: Caso não haja mais cidades que permitam produzir um caminho sem colisões, essa restrição é ignorada, prevenindo que certas cidades não sejam devidamente ligadas.

- A ligação a ser criada não pode gerar um ciclo**, já que isso isolaria um grupo de cidades de outro, comprometendo futuras etapas do algoritmo relacionadas ao

particionamento do mesmo.

****OBS:** Caso o ciclo gerado seja o polígono desejado, essa restrição é ignorada, prevenindo que a última cidade não seja conectada.

- Por fim, com o polígono formado, particionamos o caminho entre os diferentes caixeiros, fazendo com que cada um percorra K cidades, partindo do centroide e voltando para o mesmo. Nesse caso, K , em uma situação ideal, vale a divisão inteira de N cidades por M caixeiros.

Para casos em que o valor não é exato, distribuímos o restante, ou o resto da divisão de N por M , entre os diferentes caixeiros, de modo que a diferença de cidades visitadas por um caixeiro e outro não passe de 1.

3 Experimentos computacionais

3.1 Instâncias de testes

O conjunto de testes é formado por 9 instâncias, variando em cada uma o número N de cidades (entre 13 e 92), o número M de caixeiros (entre 1 e 5), o número K máximo de cidades a serem visitadas por cada caixeiro (entre 13 e 20), e as coordenadas das cidades (todas no intervalo $[0,1000]$).

3.2 Resultados

Instância	N	M	K	Solução (em tese) Ótima	Solução Encontrada
1	13	1	13	3071	3013
2	17	1	17	3948	4146
3	19	1	19	4218	4686
4	32	3	11	5841	8378
5	48	3	16	6477	9931
6	60	3	20	6786	11769
7	72	5	15	8618	12447
8	86	5	17	9565	15770
9	92	5	19	9586	16432

3.2.1 Discussão

Analisando-se os resultados, nota-se uma piora crescente na solução encontrada conforme aumenta-se o número de cidades e de viajantes nos casos de teste, demonstrando que o algoritmo possui uma eficiência melhor com parâmetros menores.

Isso se deve ao fato de a heurística construída gerar, inicialmente, um polígono ao redor de um ponto, o que se adequa muito bem ao problema do caixeiro viajante único, mas não a múltiplos caixeiros, tendo em vista que à medida em que se aumenta M , o número de conexões com o centroide aumenta tanto quanto.

Outra desvantagem do algoritmo é que, apesar de quase não haver intersecções e sobreposições entre os caminhos formados, os mesmos em grande parte do tempo possuem forma de ziguezague, o que cria percursos desnecessariamente longos que poderiam ser reduzidos.

Referências

- [1] OLIVEIRA, André Filipe Maurício de Araújo. **Extensões do problema do caixeiro viajante**. 2015. Dissertação de Mestrado. Disponível em: https://estudogeral.uc.pt/bitstream/10316/31684/1/Tese_AndreOliveira.pdf. Acesso em 24 mar. 2024.