

Problema do Caixeiro Viajante Euclideano

Convex Hull

O algoritmo da casca convexa consiste em criar uma casca inicial dos pontos dados e esta casca é tratada como uma sub-rota. A partir daí, ir buscando rotas melhores onde dados dois pontos, busca-se um ponto que a distância do novo caminho, seja menor que a distância original. O algoritmo pára quando todos os vértices internos forem adicionados a sub-rota.

$O(n^2 \log n)$.

Closest First

O algoritmo do Closest First é similar ao Prim. Ele funciona da seguinte forma: dado uma raiz, o algoritmo vai caminhando e construindo a rota com o ponto mais próximo.

$O(n+m)$

Otimização

- 2-opt

A otimização 2-opt, funciona de forma local, reduzindo a distância entre dois pares de vértices próximos e que, inicialmente, possuem arestas que se sobrepõem. O 2-opt, checa se há possibilidade de combinar esses pontos de forma que as novas arestas não se cruzam.

$O(n^3)$

Problemas encontrados:

Encontramos um gargalo no 2-opt, na implementação do swap, onde cairia de $O(n)$ para $O(1)$.

O Algoritmo da Casca Convexa se mostrou ineficiente pois requer um tempo de execução muito alto.

Estudo de Caso

Grafo	Time (s)	Minimum Distance
points-5.txt	0,00158	1.728.245
points-10.txt	0,0091	2.921.770,25
points-20.txt	0,04686	4.199.096,00
points-50.txt	0,36033	6.571.397,00
points-100.txt	2,71764	9.140.787,00
points-200.txt	15,5747	12.699.448,00
points-500.txt	228,531	18.026.410,00
points-1000.txt	1496,16	25.698.446,00
points-2000.txt	532,466	37.692.660,00
points-5000.txt	-	inf
points-7500.txt	-	inf
points-10000.txt	-	inf