

Relatório técnico para a disciplina de algoritmos em grafos ministrada pelo Prof. Dr. Ademir A. Constantino

Algoritmos construtivos e melhorativos para o problema do caixeiro viajante assimétrico

Dalvan T. Oliveira¹, Luiz H. C. M. Marques¹

¹ Universidade Estadual de Maringá (UEM)
Centro de Tecnologia – Departamento de Informática
Programa de Pós-Graduação em Ciência da Computação
Maringá – PR – Brasil

`dalvan.oliveira@outlook.com, luhenrique06@hotmail.com`

Resumo. Neste trabalho foi implementado algoritmos em grafos para resolução do problema do caixeiro viajante usando técnicas construtivas e melhorativas. É feita uma breve explicação do problema e das técnicas utilizadas. A linguagem de programação C foi usada para implementação dos algoritmos e foi resolvido as instâncias assimétricas da base de dados TSBLIB.

Introdução

pequena introdução sobre o que vamos falar nessa seção...

O problema do caixeiro viajante

O problema do caixeiro viajante - PCV consiste em visitar v_1, \dots, v_n pontos distintos, partindo de um ponto inicial $v_0 = v_i$, para algum $i = 1, \dots, n$, de tal forma que não passe mais de uma vez em um mesmo ponto e além disso, ao término, deve-se voltar ao ponto inicial v_0 . Além disso, todos os pontos estão mutualmente conectados e essas conexões possuem custos.

Pode-se considerar os pontos como clientes. Os caminhos entre os clientes tem um custo, que podemos associar à distância. Considera-se que somente um caixeiro sairá do ponto inicial v_0 e visitará os demais $n - 1$ pontos, retornando ao ponto inicial v_0 ao término do percurso.

Resumindo, o objetivo é encontrar uma viagem de ida e volta, com menor custo possível, sem passar pelo mesmo ponto/cliente. Essa rota é conhecida na literatura como ciclo Hamiltoniano.

Neste trabalho, foi usado o conceito de grafos para resolver o PCV. Um grafo $G(V, E)$, é constituído por um conjunto $V = \{v_1, \dots, v_n\}$ de vértices e um conjunto de arestas $E = \{(v_i, v_j); v_i, v_j \in V, v_i < v_j\}$ onde (v_i, v_j) denota a aresta (ou reta) que liga os vértices v_i e v_j .

Logo, os clientes são tratados como vértices de um grafo $G(V, E)$ e o caminho entre dois clientes é a aresta que liga os dois vértices.

Métodos heurísticos

Métodos heurísticos são técnicas que buscam encontrar uma solução para o problema. São geralmente usados quando o custo para encontrar uma solução ótima para o problema é muito elevado e é aceitável uma solução aproximada.

Tratando do PCV, é usual classificar as heurísticas em dois grupos: construtivas e melhorativas.

Heurísticas construtivas

Heurísticas construtivas são usadas pra construir uma solução inicial. Dentre algumas heurísticas clássicas encontradas na literatura podemos citar:

- Inserção vizinho mais próximo;
- Inserção mais barata;
- Inserção mais distante;
- Algoritmo de Clarke e Wright

Foi usado a **heurística construtiva de inserção no vizinho mais próximo** para obter a solução inicial. Consiste em construir uma rota, onde a cada passo, adiciona-se o vértice mais próximo do último vértice inserido, de modo que não se repita vértices.

- 1 *Inicialização* - Comece com uma rota com apenas uma vértice v_i , escolhido aleatoriamente;
- 2 *Seleção* - Seja (v_1, \dots, v_k) a rota parcial atual. Encontre o vértice v_{k+1} que ainda não está na rota e que esteja mais perto de v_k ;
- 3 *Inserção* - Insira v_{k+1} no final da rota parcial;
- 4 Se todos vértices foram inseridos então pare. Caso contrário volte para o passo 2.

A Figura 1 exemplifica o funcionamento desse algoritmo.

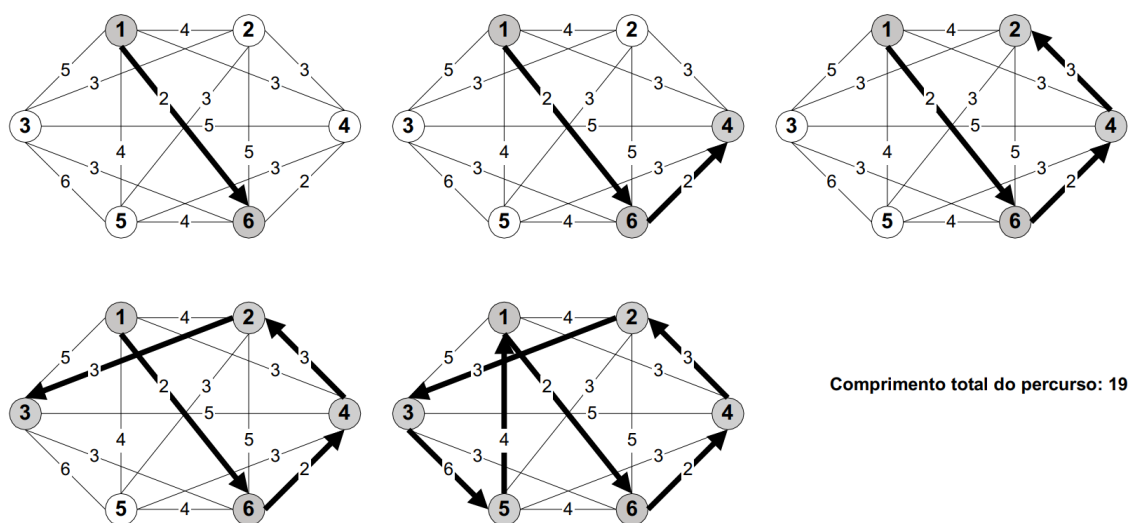


Figura 1. Execução da heurística de inserção do vizinho mais próximo

Heurística melhorativa

Dado uma solução inicial obtida pela heurística construtiva, a ideia é usar uma heurística melhorativa para fazer alterações na solução inicial buscando melhorar a qualidade da solução. Para isso foi usado a heurística melhorativa 2-opt.

O algoritmo 2-opt consiste em remover duas arestas não adjacentes da rota e reconectá-las usando duas outras arestas, de modo a reconstruir a rota e verificar se a houve melhora com a nova rota obtida. Repetindo esse processo para todos os pares de arestas possíveis, ao término, realizamos a melhor troca.

O algoritmo melhorativo 2-opt consiste em, dado uma rota.....

falar código

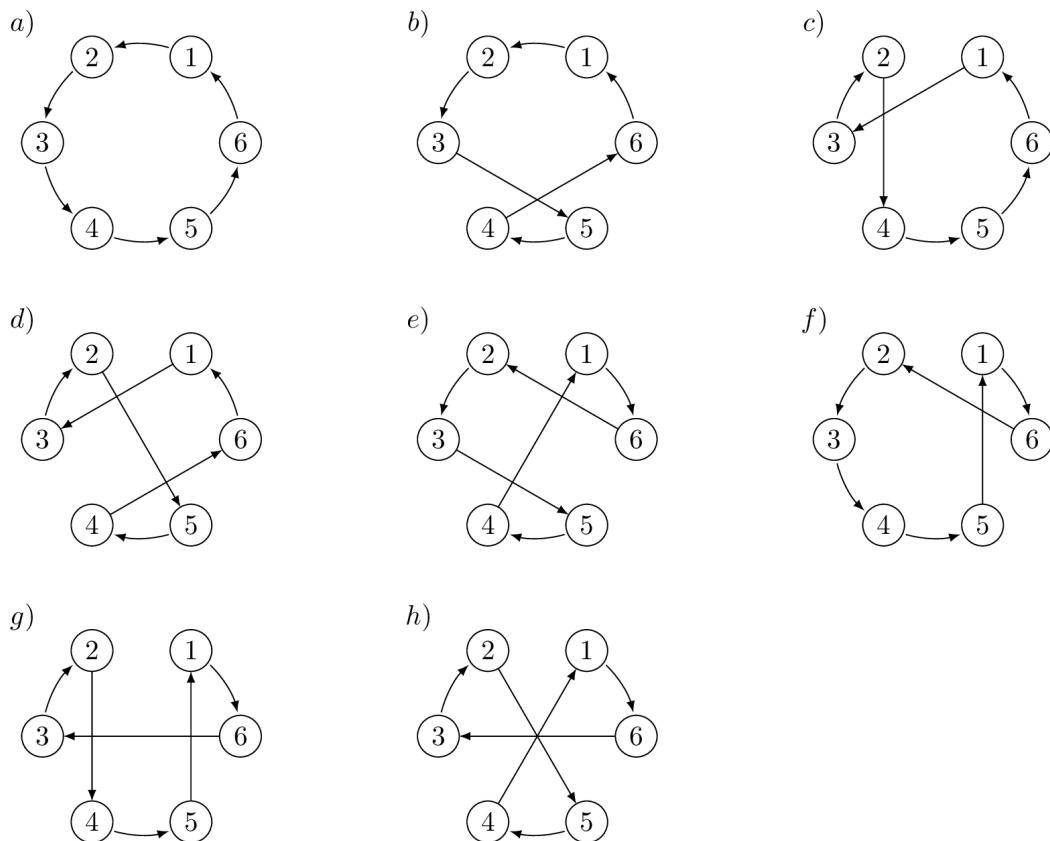


Figura 2. Execução da heurística 2-opt

Base da dados

Os algoritmos implementados foram executados usando as instâncias assimétricas do PCV da biblioteca TSPLIB.

Como executar algoritmo

.....

Dificuldades etc

.....

Resultados

Aplicações

Tabela 1. Valores de Similaridade por Usuário

Instâncias	Valor
br17	1
ftv33	1
ftv35	1
ftv38	1
p43	1
ftv44	1
ftv47	1
ry48p	1
ft53	1
ftv55	1
ftv64	1
ft70	1
ftv70	1
kro124p	1
ftv170	1
rbg323	1
rbg358	1
rbg403	1
rbg443	1

Figures and Captions

Modo de citacao 1: (Figure 3)

Modo de citacao 2: Figure 4.

Referências



Figura 3. A typical figure

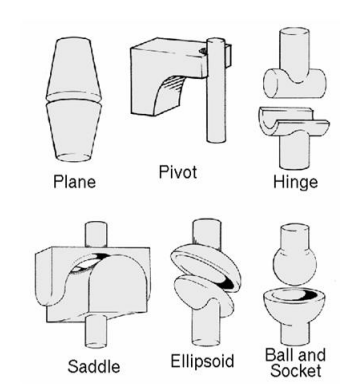


Figura 4. This figure is an example of a figure caption taking more than one line and justified considering margins mentioned in Section 7.

Tabela 2. Variables to be considered on the evaluation of interaction techniques

	Chessboard top view	Chessboard perspective view
Selection with side movements	6.02 ± 5.22	7.01±6.84
Selection with in- depth movements	6.29±4.99	12.22±11.33
Manipulation with side movements	4.66± 4.94	3.47±2.20
Manipulation with in- depth movements	5.71 ±4.55	5.37 ±3.28