

Avaliação da aplicação do algoritmo Hill Climbing para resolução do problema de Roteamento de Veículos

Fabiane Kuhn¹, Gabriel Colling¹, Pedro Bohlmann Cascaes Silva¹, Samuel Armbrust Freitas²

¹Curso Bacharelado em Ciência da Computação – Universidade do Vale do Rio dos Sinos (UNISINOS)
Av. Unisinos, 950 – 93020-190 – São Leopoldo – RS – Brasil

²Curso Bacharelado em Engenharia da Computação – Universidade do Vale do Rio dos Sinos (UNISINOS)
Av. Unisinos, 950 – 93020-190 – São Leopoldo – RS – Brasil

fabianekuhn@edu.unisinos.br, pcascaess@unisinos.br, gabrielcolling@edu.unisinos.br, samuelaf@edu.unisinos.br

Abstract. *This meta-paper describes the implementation of a Hill Climbing algorithm to solve the Capacitated vehicle routing problem (CVRP). The algorithm was developed with Python and served as the basis for the performance analysis compared to the global maxima within the problem*

Resumo. *Esse artigo descreve a implementação de um algoritmo Hill Climbing para solucionar o problema de Roteamento de Veículos Capacitados (CVRP). O algoritmo foi desenvolvido na linguagem Python e serviu como base para a análise de desempenho comparado com os máximos globais dentro do problema.*

1. Roteamento de Veículos Capacitados (CVRP)

O problema de Roteamento de Veículos, também conhecido como Capacitated vehicle routing problem (CVRP), é representado por um conjunto de n nós e a distância entre cada par de nós. Os nós representam todos os clientes e um depósito para a carga. Nesse contexto, cada cliente possui uma demanda pelas mercadorias fornecidas pelo depósito e entregues pelos caminhões, sendo que todos os veículos possuem a mesma capacidade de carga.

Para resolver o problema é necessário encontrar um roteamento capaz de atender a demanda de cada cliente, sempre respeitando a capacidade de carga dos caminhões. O roteamento deve apresentar quais clientes cada caminhão irá visitar e em qual ordem fará as entregas. Além de realizar as entregas, a solução deve visar minimizar a quantidade de caminhões utilizados e a soma da distância percorrida por todos os caminhões.

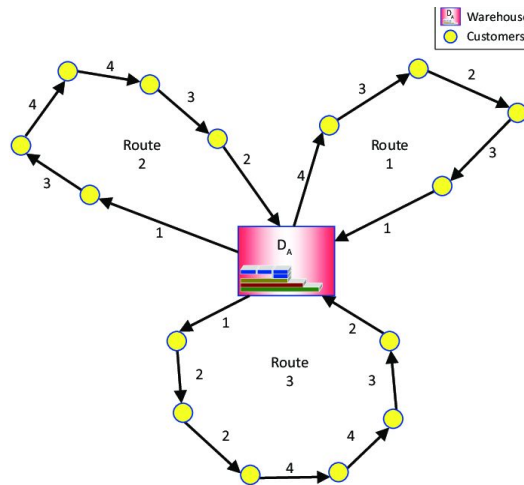


Figura 1. Exemplo de roteamento com três veículos

2. Hill Climbing

Hill Climbing é um método heurístico de otimização matemática, uma das suas maiores aplicações é a resolução de problemas computacionais que possuem múltiplas soluções. É um método iterativo e pertence a categoria de pesquisa local que começa com uma solução aleatória e melhora iterativamente essa solução, um elemento por vez, até chegar em alguma solução otimizada.

O algoritmo Hill Climbing possui algumas variantes, tais como:

- Reinício aleatório - O algoritmo é rodado várias vezes, até encontrar a solução ótima.
- Primeira escolha - Escolhe o primeiro vizinho melhor.
- Estocástico - O melhor nó é escolhido aleatoriamente (ou proporcional à melhora).

O diagrama de espaço de estados é uma representação gráfica do conjunto de estados que o algoritmo de busca pode atingir em relação ao valor da função objetivo (a função que desejamos maximizar).

- Eixo X: denota o espaço de estados, isto é, estados ou configurações que o algoritmo pode alcançar.
- Eixo Y: denota os valores da função objetivo correspondentes a um estado particular.

A melhor solução será o espaço de estados onde a função objetivo tem valor máximo (máximo global).

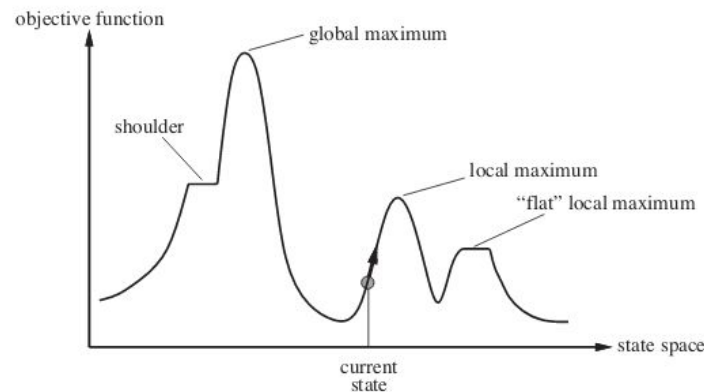


Figura 2. Descrição do Hill Climbing

Na figura 2 está apresentado o funcionamento do algoritmo de Hill Climbing, levando em consideração os seguintes pontos:

- Máximo local: É um estado que é melhor que seu estado vizinho, no entanto, existe um estado que é ainda melhor (máximo global).
- Máximo global: é o melhor estado possível no diagrama do espaço de estados.
- Flat local maximum: É uma região plana do espaço de estados onde os estados vizinhos têm o mesmo valor.
- Ridge: É a região que é mais alta que seus vizinhos, mas tem uma inclinação. É um tipo especial de máximo local.
- Current state: A região do diagrama do espaço de estados onde o algoritmo está no momento da pesquisa.
- Shoulder: É um plano que tem uma grande subida.

3. Aplicação prática

Para a análise da resposta do algoritmo Hill Climbing na solução do problema de roteamento de veículos foi realizada a implementação do algoritmo utilizando a linguagem Python, partindo de um arquivo de entrada que contém os dados referentes ao número de caminhões e suas capacidades, além do número de locais, suas coordenadas e demandas.

Considerando que um caminhão deve partir do depósito, realizar uma rota dentro do seu limite e, então, voltar ao depósito, primeiramente foram calculadas as distâncias entre os nós, para que as rotas de cada caminhão pudessem ser traçadas sem ultrapassar o seu limite. A partir disso, os caminhões foram destinados a rotas aleatórias dentro de sua capacidade, de forma que todos os nós fossem cobertos com o número disponível de caminhões. Em seguida, foram simuladas trocas entre os nós cobertos por uma mesma solução, ou seja, nós dentro da mesma rota de um determinado caminhão, e efetivadas aquelas que melhoraram o custo total da rota. Por fim, as simulações mencionadas foram realizadas também entre os caminhões, trocando nós entre as rotas, isto é, removendo um nó da rota de um caminhão e adicionando à rota de outro.

Como resultados temos os seguintes valores de custos para os problemas, antes e depois da aplicação do algoritmo Hill Climbing:

Problema	Caminhões	Soluções ótimas	Execução 1		Execução 2		Execução 3	
			Antes	Depois	Antes	Depois	Antes	Depois
att48	4	12649	159123.94	156401.19	166151.72	145237.48	175868.90	139267.37
eil33	4	910	1685.10	1591.20	1776.86	1652.27	1613.18	1501.09
<u>eilc76</u>	8	745	2336.24	2295.37	2310.19	2130.30	2197.83	2089.07

Tabela 1. Resultados de três execuções distintas do algoritmo para cada cenário

A partir destes resultado é possível identificar que algumas das execuções apresentaram resultados muito satisfatórios na identificação de ótimos locais da solução e em alguns casos os valores inclusive ficam próximos aos ótimos globais.

De qualquer forma é importante ressaltar os resultados distintos de cada execução devido a aleatoriedade da solução inicial e portanto em certos casos em que a solução inicial aproxima-se um ótimo local, a melhoria possível do algoritmo é bastante reduzida, tal como pode ser percebido na execução 1 do cenário att48, de acordo com a tabela 1.

Em relação as melhores configurações para a execução do algoritmo, foi possível realizar uma análise de desempenho para diferentes números de repetições da rotina, destacando os resultados do caso de teste eil33 nas figuras 3 e 4.

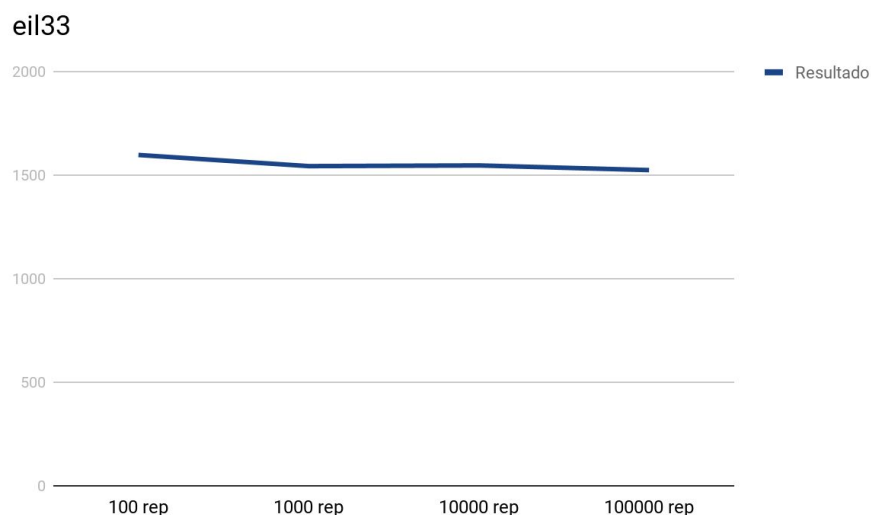


Figura 3. Análise de resultado com diferentes repetições

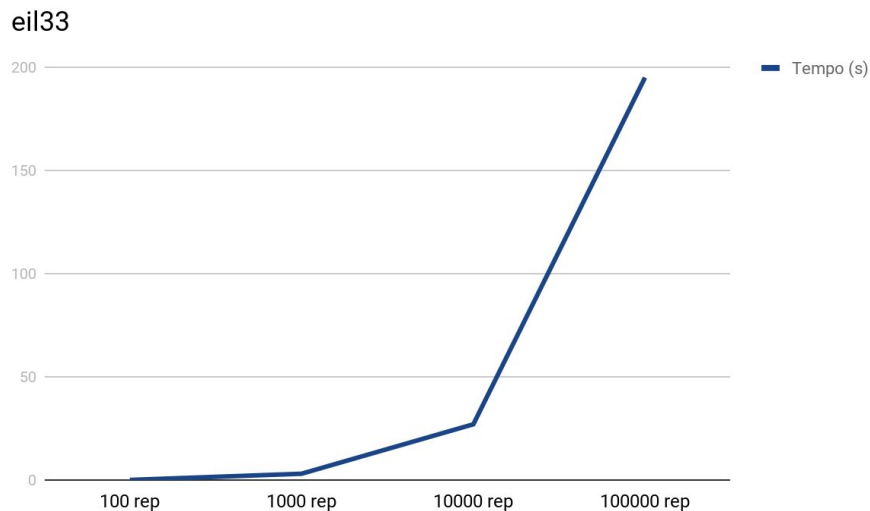


Figura 4. Análise do tempo de execução com diferentes repetições

Foi possível constatar que a melhora é pouco significativa quando alterado o número de repetições realizadas pela rotina, sendo favorável uma configuração com poucas repetições para manter um bom desempenho.

4. Conclusões

Com a aplicação do algoritmo de Hill Climbing para a melhoria do roteamento de veículos com duas estratégias distintas foi possível identificar em grande parte das execuções, melhoria considerável no custo total das rotas.

Considerando as duas estratégias de vizinhança adotadas, a troca da ordem dos nós visitados pelos caminhões provou-se a estratégia com melhor melhoria, pois permitiu a maior diminuição do custo total do trajeto. A segunda estratégia, que consiste na troca de nós entre os caminhões provou-se pouco ágil para solucionar um problema de tamanha complexidade.

Dessa forma, no contexto geral, foi possível abordar as partes principais de um algoritmo de busca, que consiste na definição do problema, identificado do escopo de movimentação para encontrar as vizinhanças, função de custo, definição aleatória da solução inicial e função de execução do hill climbing e dessa forma chegar em performance parecida com o algoritmo ideal.

Enfim, como uma avaliação holística do problema, o presente trabalho obteve êxito em provar o conceito teórico do Hill Climbing e atingir resultados satisfatórios.

Referências

Introduction to Hill Climbing | Artificial Intelligence - GeeksforGeeks (2019). <https://www.geeksforgeeks.org/introduction-hill-climbing-artificial-intelligence/>, Acessado em: 20/04/2019.

ROJAS-CUEVAS, I-D., Caballero-Morales, S-O., et al. 2018. '**Capacitated vehicle routing problem model for carriers**', Journal of Transport and Supply Chain Management. 12(0), a345. Disponível em: <https://jtscm.co.za/index.php/jtscm/article/view/345>. Acessado em: 19/04/2019.