

# Experimento - Problema do Caixeiro Viajante

Eliton Zimmermann, Gabriel Filippi, Lucas Costa

Univille - Universidade da Região de Joinville  
Caixa Postal 246 – 89201-972 – Joinville – SC – Brazil

{eliton.zimmermann, gabriel.filippi,  
lucas.fernandes.1}@univille.br

***Resumo.** Neste artigo será feito a explicação do experimento “Problema do caixeiro viajante” atividade da disciplina de Introdução a Teoria da Computação, ministrado pelo professor Leanderson André.*

## 1. Caixeiro Viajante

O Problema do Caixeiro Viajante (PCV) é um problema clássico de otimização trabalhado por diversos pesquisadores pela sua simplicidade na formulação, complexidade na resolução e sua larga aplicabilidade no mundo real, e sua resolução podem trazer inúmeros benefícios para a sociedade.

O caso que será tratado na atividade será um conjunto de cidades, todas elas sendo representadas por pontos nos grafos. Cada trajeto ou ligações do grafo terão um custo de distância e valores numéricos diferentes. Para ser concluído o trajeto deve-se passar por todas as cidades sem repetir nenhuma, no entanto, tentando ter o menor custo possível para fazer o processo.

O ponto de saída é aleatório e o ponto de chegada é variável, dependendo quando passar por todas as cidades, caso um ponto vá para ele mesmo, o custo será 0, como no caso de viagem da cidade para própria cidade.

A ideia para este experimento é executar três algoritmos ao menos 30 vezes e identificar a média, o desvio de padrão e assim qual a melhor solução dentre esses três.

## 2. Descrição dos Algoritmos

### 2.1. Algoritmo Aleatório

Com o algoritmo aleatório, como o próprio nome diz, não existe muita lógica nos caminhos escolhidos, são apenas usados de forma aleatória. Logo o algoritmo sairá de um ponto aleatório para um caminho aleatório como um viajante sem rumo.

### 2.2. Algoritmo Guloso

Já o algoritmo guloso, a partida também é de uma cidade aleatória, será feito a escolha dos caminhos com base na próxima menor distância, desta forma o algoritmo irá ficar preso muitas vezes, não tendo a melhor solução.

### 2.3. Algoritmo Semi-Guloso ou Híbrido

O algoritmo híbrido é uma combinação entre o algoritmo aleatório e o algoritmo guloso. Neste caso é sorteado um número inteiro, que irá selecionar qual vai ser a forma de escolha do próximo caminho do algoritmo, aleatório ou guloso. Um exemplo para demonstrar como funciona é colocar um range de 0 a 200, caso o número sorteado por exemplo for 20 e o valor do caminho for maior escolhemos a forma guloso caso o contrário será escolhido o aleatório

## 3. Resultados

Ao executar 30 vezes cada um dos algoritmos, os seguintes resultados foram obtidos utilizando da média aritmética, desvio padrão e diagrama de caixa:

Aleatório – Média(Distância): 3138,97KM – Desvio padrão (Distância): 1415.68KM

Guloso - Média(Distância): 409.97KM – Desvio padrão (Distância): 1042.13KM

Híbrido - Média(Distância): 1041.82KM – Desvio padrão (Distância): 1042.17KM

#### **4. Conclusões**

Após a finalização dos testes e analisando os resultados, é possível concluir que quase se equivale aos algoritmos de execução. A relação pode ser feita com a quantidade de dados processados, pois não foi inserido os algoritmos em testes de estresse, para que seja possível a validação algoritmo demoaria mais para encontrar a solução com uma maior quantidade de dados. Por costume, o algoritmo aleatório tende a ser o que pode achar uma solução mais rápida devido a sua lógica, onde ao pegar qualquer caminho diferente da distância não analisará o ponto mais relavante desse experimento que trata de encontrar a melhor solução. Com a analise dos algoritmos podemos observar que:

Em todos os acontecimentos o algoritmo aleatório apresentou o pior para determinar a rota mais curta. Foi o o que obteve a média e desvio padrão ultrapassando a soma dos resperctivos valores dos outros dois tipos de algoritmos. É possível visualizar também que o algoritmo aleatório gera as soluções capazes de se igualar ao algoritmo guloso e híbrido.

No algoritmo guloso foi onde gerou as menores totas, mesmo considernado o risco de ter problemas nos “ótimos locais”. Dentre os três algoritmos, foi o que gerou as melhores soluções, com base nos valores de média e desvio, quando ao diagrama da caixa, onde é possível analisar que parte de suas soluções ficam em valores próximos.

O algoritmo híbrido foi o que apresentou ser o mais estabilizado dentre os os três tipos, quando é analisado o diagrama da caixa. Ao analisar a média e desvio padrão, ele se aproxima ao algoritmo aleatório, o que mostra que não compensa ser utilizado, porém já vendo o diagrama da caixa, podemos verificar que foi gerado valores altos devido seus outliers.

Diante de tudo isso, pode ser concluído que esse experimento, levando em consideração os dados utilizados, o melhor algoritmo a ser utilizado é o guloso, porém é necessário informar que é possível acontecer problemas de ótimos locais, podendo ser gerado outliers. Com o algoritmo guloso foi onde apresentou a melhor solução devido pela quantidade de daados consumidos, o número de tantativas.