Resolvendo Caixeiro Viajante com Algoritmo Genético

Projeto e Análise de Algoritmos

Alexandre Mendonça Fava[1]; Gustavo Diel[1]

[1]Mestrado Acadêmico em Computação Aplicada - PPGCA

3 Dezembro de 2019

UDESC

Caixeiro Viajante

- É um problema **NP-Hard**
- Dada uma lista de cidades, encontrar o menor caminho que passe por todas uma únca vez e termine no início

UDESC

Caixeiro Viajante - Exemplo

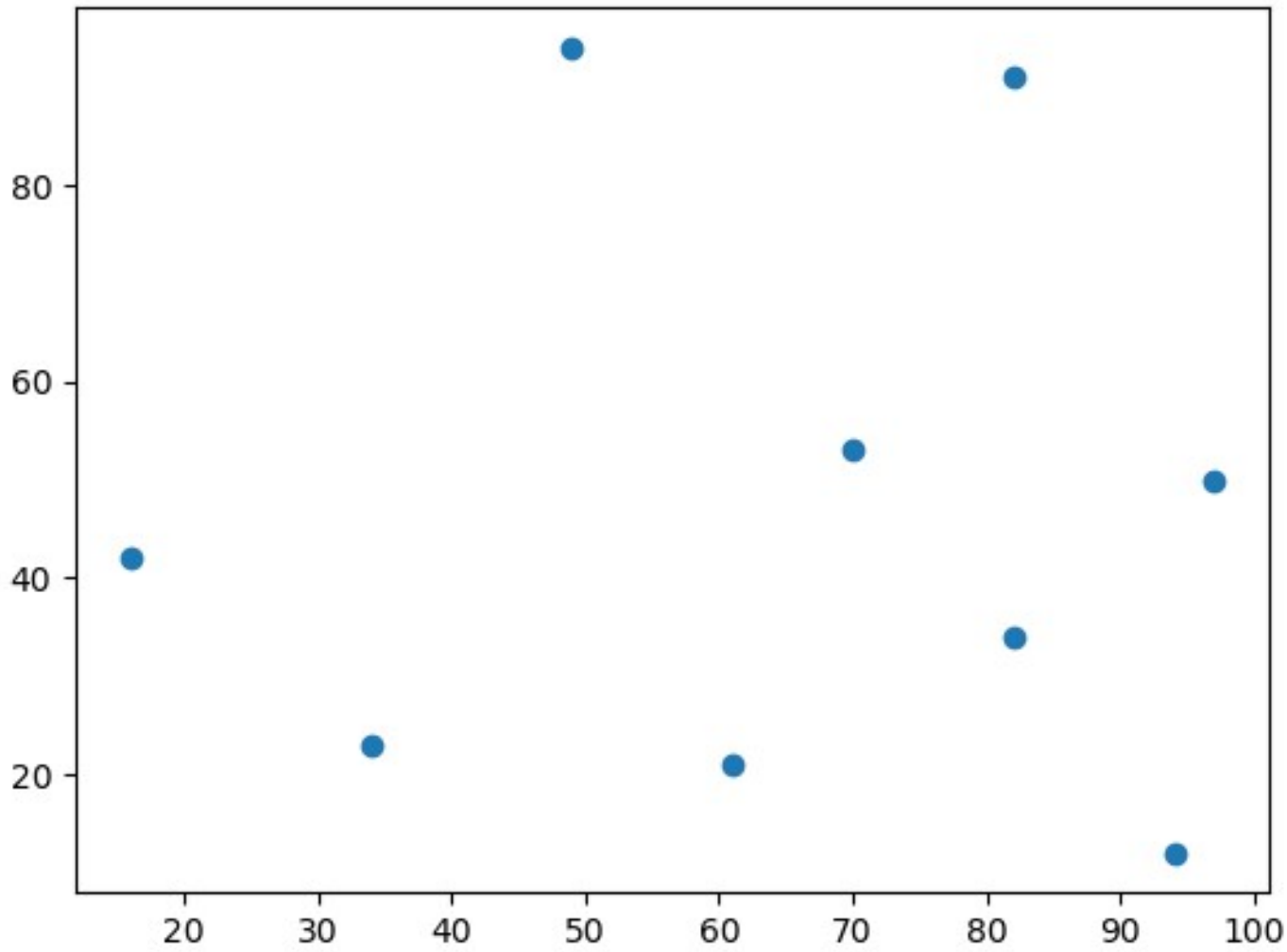


FIGURE – Mapa das cidades

Caixeiro Viajante - Exemplo resolvido

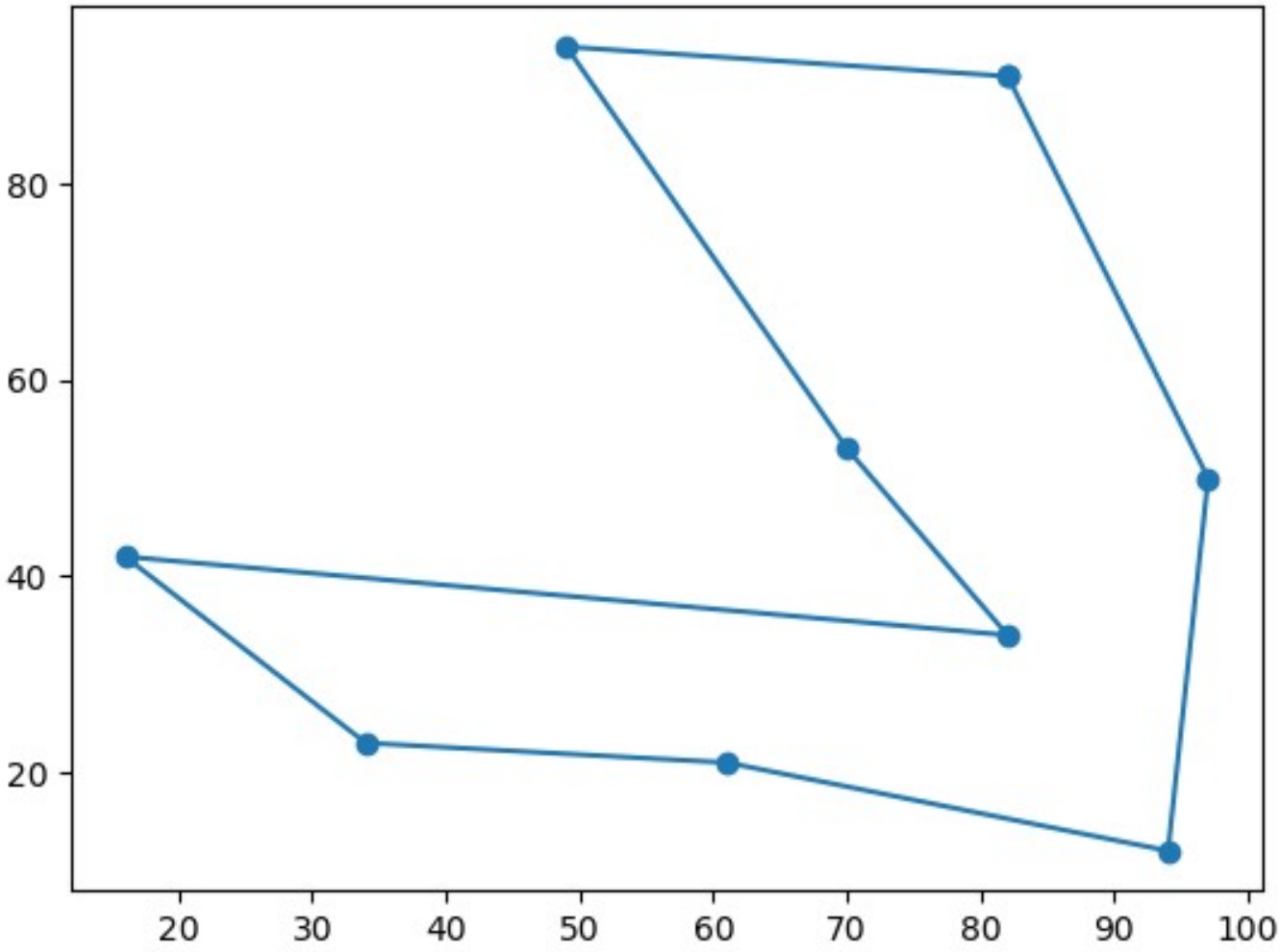


FIGURE – Resolução

Caixeiro Viajante - Força Bruta

```
1   def find_paths(current_city, cities, path, distance, routes):
2     path.append(current_city)
3
4     if len(path) > 1:
5       distance += Distance(path[-2], current_city)
6
7     if (len(cities) == len(path)):
8       intial_city = path[0]
9       path.append(intial_city)
10      distance += Distance(last_city, intial_city)
11      routes.append([distance, path])
12      return
13
14    for city in cities:
15      if (city not in path):
16        find_paths(city, cities, path.copy(), distance, routes)
```

UDESC

Algoritmos Genéticos

- Inspirados nas teorias evolutivas do Darwinismo
- **Os mais fortes sobrevivem !**

- População com **n** indivíduos
- Cada indivíduo tem seus cromossomos aleatórios
- Depois de cada iteração, novos indivíduos baseados nos anteriores são gerados

UDESC

Preparando o problema

Inicialmente uma população com **n** indivíduos é gerada :

1 7 10 4 5 3 9 8 1 2 6
2 4 5 8 3 1 9 4 6 10 2
⋮
n 5 3 7 6 1 8 9 4 2 10

Os indivíduos são os cromossomos, ou seja, soluções do problema

Cada gene do cromossomo equivale a uma instância do problema

UDESC

Preparando o problema

Posteriormente é realizado o cálculo da aptidão de cada indivíduo :

1	7 10 4 5 3 9 8 1 2 6	[84%]
2	4 5 8 3 1 9 4 6 10 2	[24%]
⋮		
n	5 3 7 6 1 8 9 4 2 10	[57%]

Como em muitos casos não se sabe os limites do problema, o cálculo da aptidão é um valor aproximado

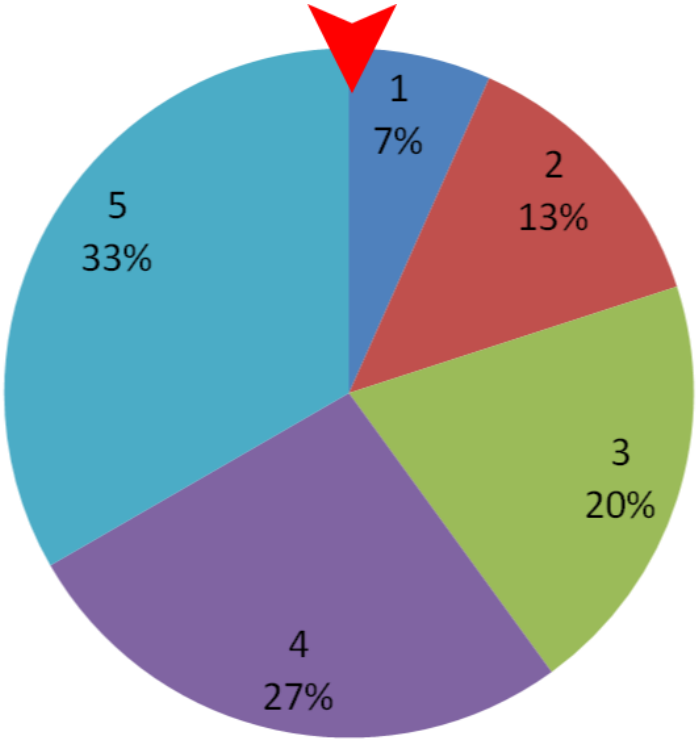
Para o problema do Caixeiro Viajante, a aptidão é cálculada por meio da distância euclidiana :

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

UDESC

Preparando o problema

Em seguida é realizando um processo de seleção :



5 vs 4

1 vs 8

1 vs 2

FIGURE – Métodos de Seleção

Preparando o problema

Depois de selecionados, é realizado o cruzamento :

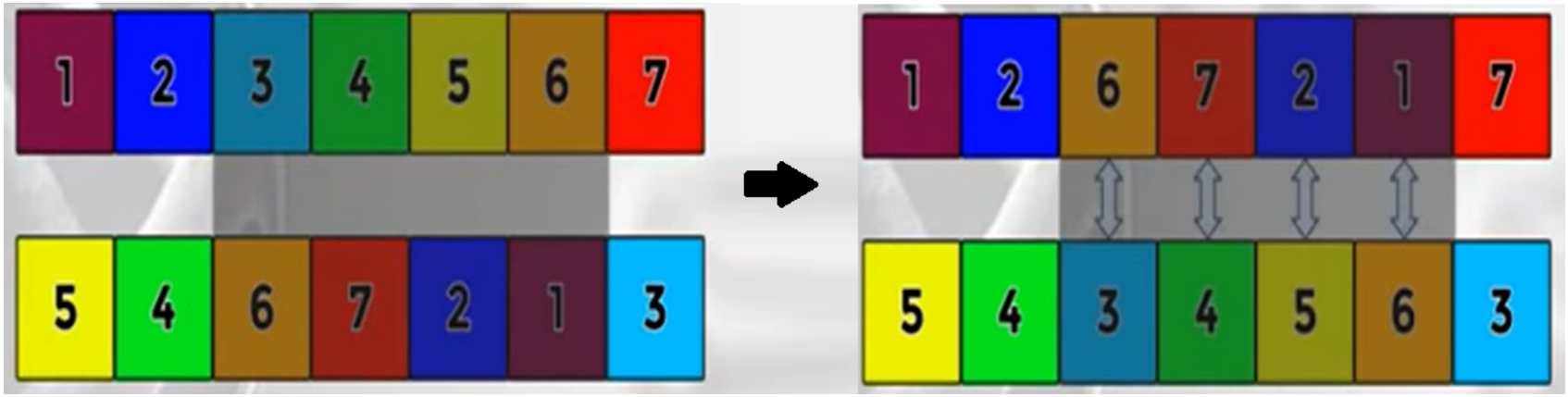


FIGURE – Cruzamento

Para alguns problemas quando utilizada a recombinação em dois pontos, é preciso realizar o PMX (Partially Matched Crossover)

UDESC

Preparando o problema

Partially Matched Crossover :

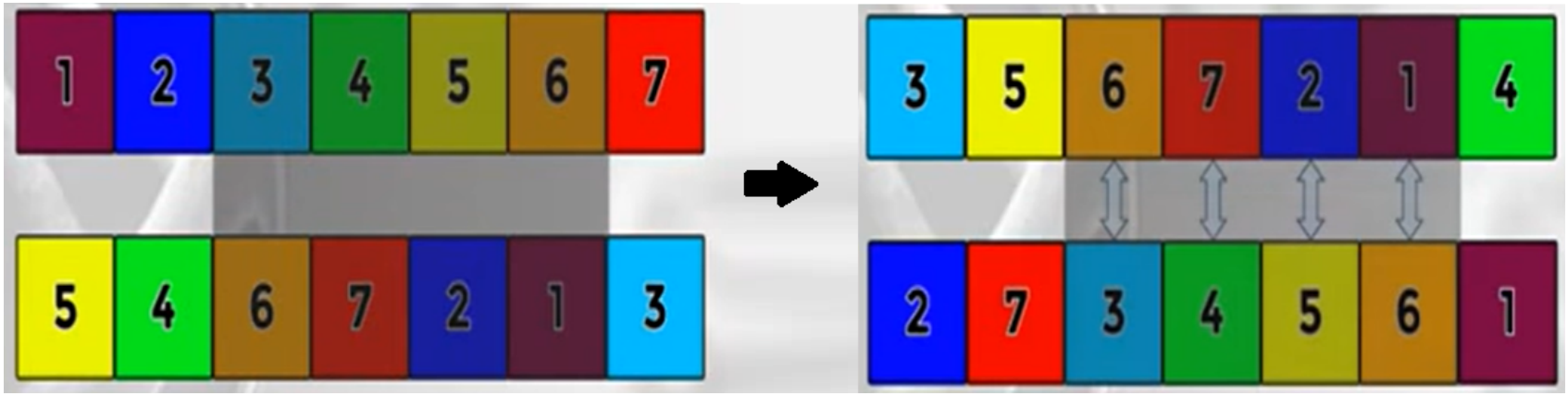


FIGURE – Cruzamento

À esquerda os pais e à direita os filhos

UDESC

Preparando o problema

Ao término os filhos passam por um processo de mutação :

Antes : 3 5 6 7 2 1 4

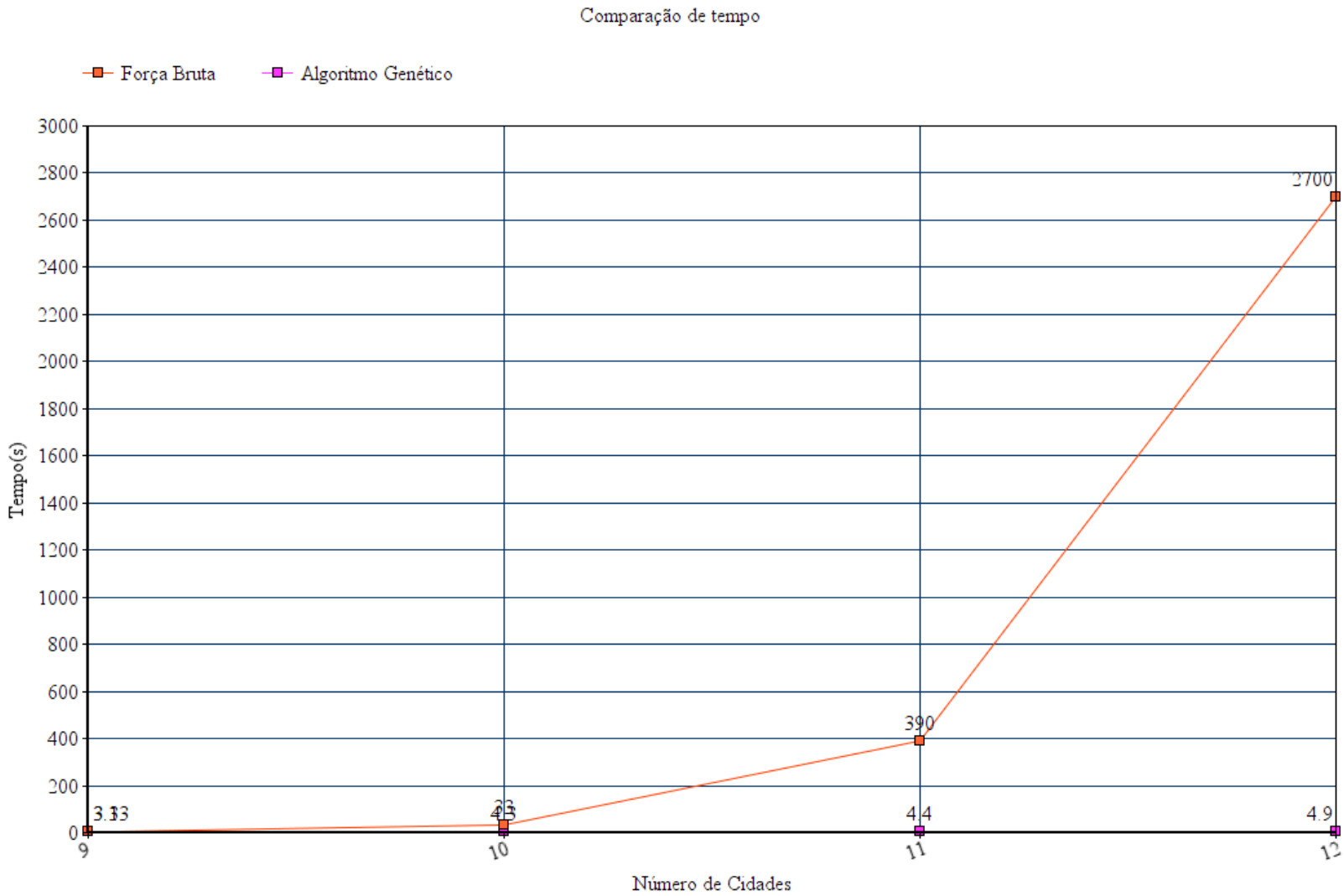
Depois : 3 1 6 7 2 5 4

Alterar o gene de número 5 para 1
Obrigatoriamente altera o gene de número 1 para 5

Como etapa adicional, pode ser incluido um processo de elistimo

UDESC

Comparação de tempo



Caixeiro Viajante com 15, 29 e 38 cidades

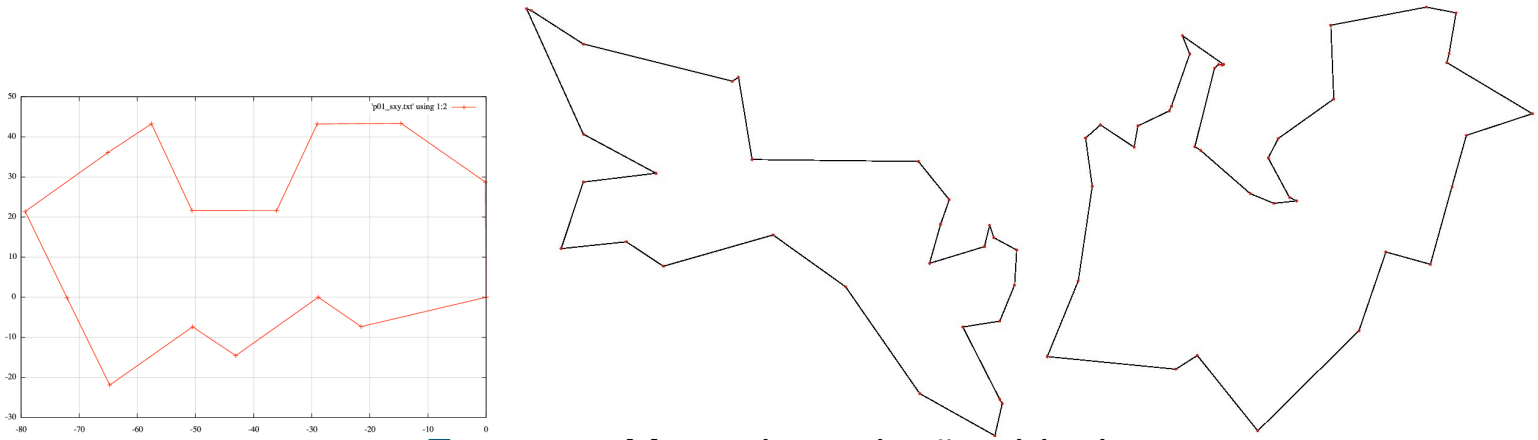


FIGURE – Mapa das soluções ideais

Fonte (15) :
`https://people.sc.fsu.edu/~jburkardt/datasets/tsp/tsp.html`
Fonte (29 e 38) :
`http://www.math.uwaterloo.ca/tsp/world/countries.html`

UDESC

Caixeiro Viajante com 15, 29 e 38 cidades

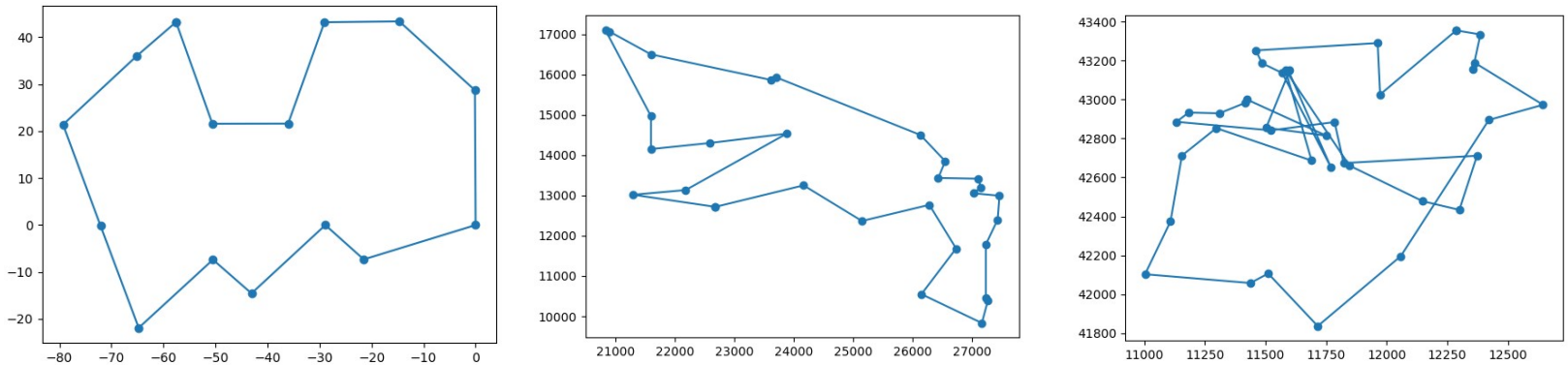


FIGURE – Melhores soluções do AG

Melhor 15 : 291 (ótima) 291 (AG)
Melhor 29 : 27.603 (ótima) 29.654 (AG)
Melhor 38 : 6.656 (ótima) 9.354 (AG)

UDESC

Caixeiro Viajante com 48 cidades

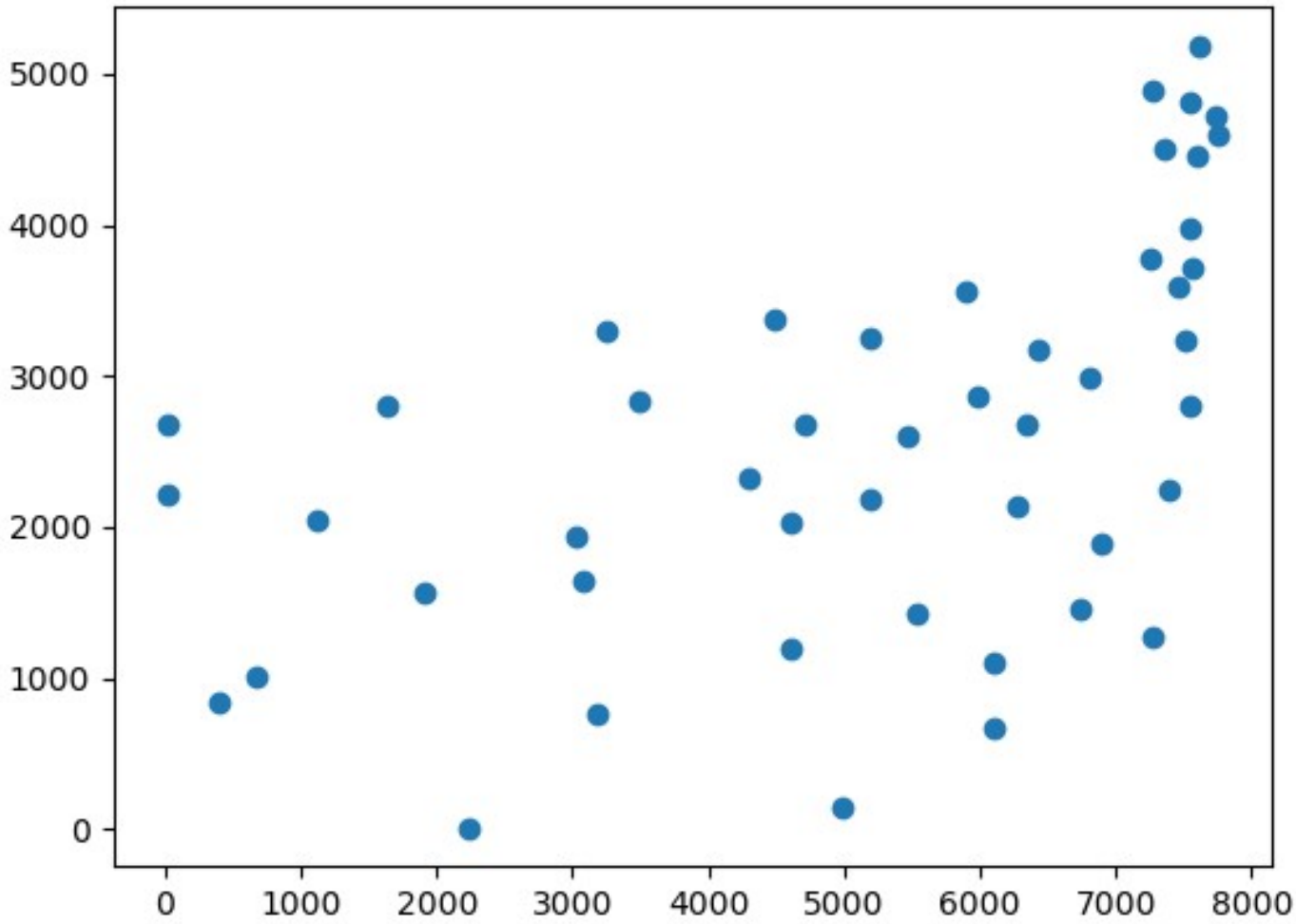


FIGURE – Mapa das 48 cidades

Caixeiro Viajante com 48 cidades

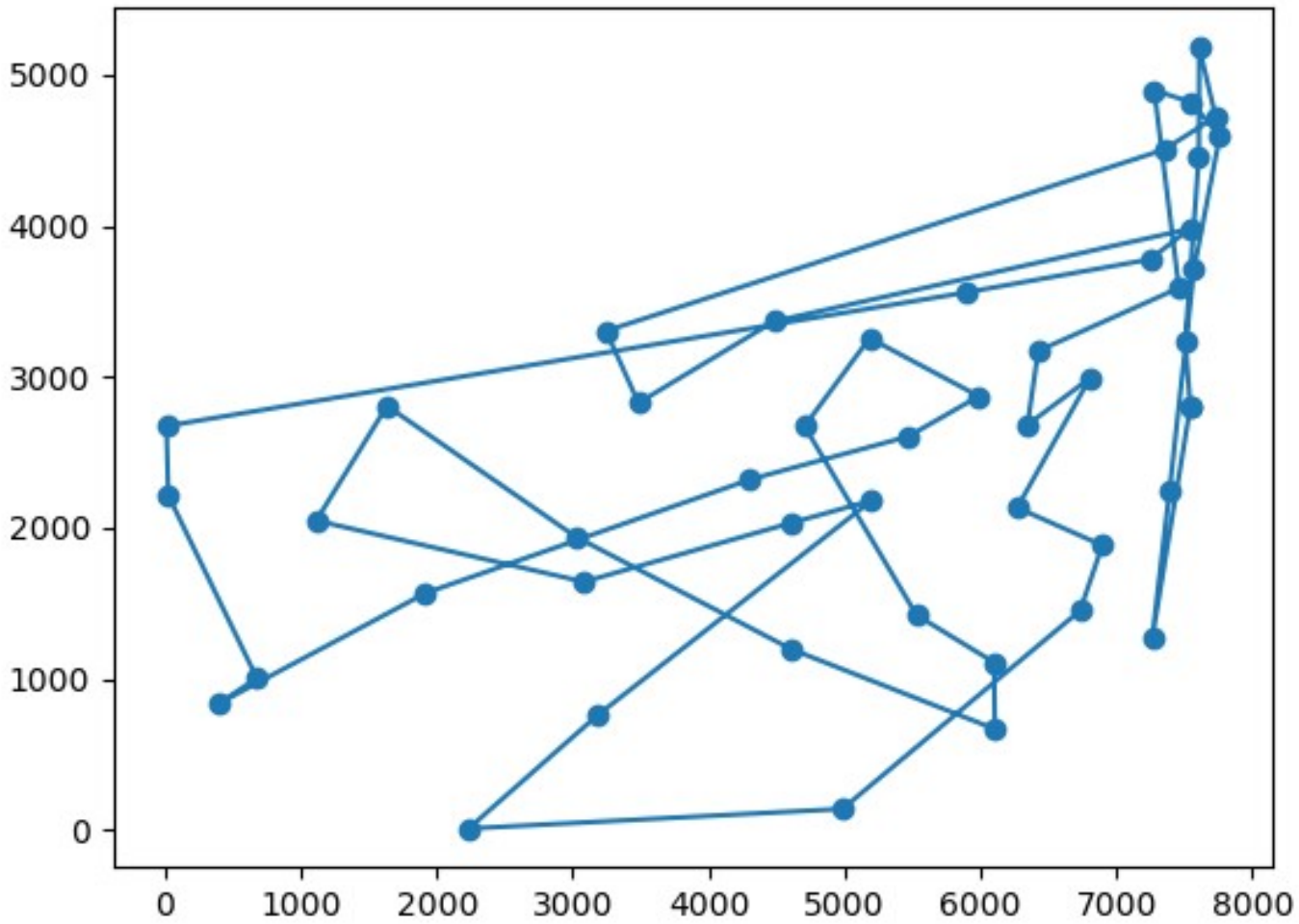


FIGURE – Resposta encontrada

Caixeiro Viajante com 48 cidades

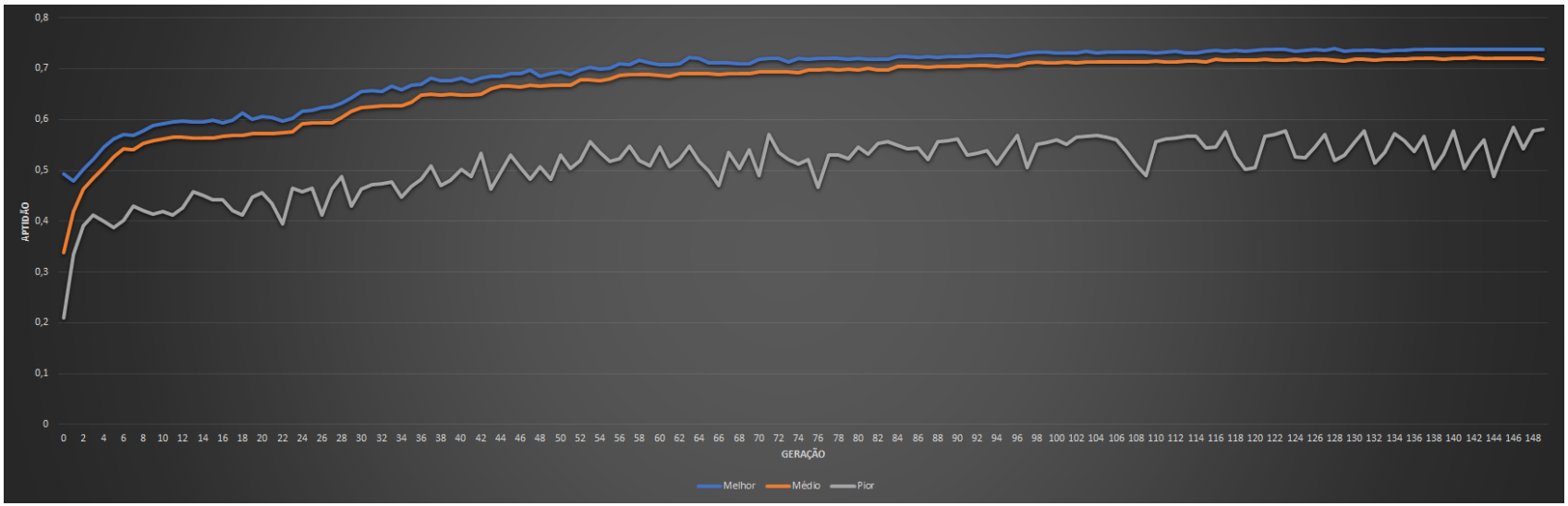


FIGURE – Fitness

Conclusão

- Utilizando o algoritmo genético, é possível resolver instâncias grandes do caixeiro viajante
- Como é possível determinar os parâmetros do algoritmo, tais como numero de iterações, taxa de mutação ou tamanho da população, resultados mais precisos podem ser obtidos
- Deve-se lembrar que existem métodos com heurísticas que também são rápidos e entregam soluções melhores
- Contudo, destace-se inúmeras aplicações para os algoritmos genéticos, desde a medicina até a engenharia

UDESC

"Não é o mais forte que sobrevive, nem o mais inteligente. Quem sobrevive é o mais disposto à mudança" - Charles Darwin

Obrigado !

Alexandre Mendonça Fava - alexandre.fava@hotmail.com
Gustavo Diel - gustavodiel@hotmail.com

UDESC