

# Otimização por Colônia de Formigas Aplicado ao Caxeiro Viajante

Gustavo Silveira Dias  
Bacharelado em Engenharia de Computação  
Instituto Federal de Minas Gerais  
Inteligência Artificial  
Prof. Dr. Ciniro Nametala

## SUMÁRIO

<b>I</b>	<b>Introdução</b>	1
<b>II</b>	<b>Definição do problema</b>	1
<b>III</b>	<b>Metodologia</b>	1
III-A	Representação . . . . .	1
III-B	Parametrização . . . . .	1
<b>IV</b>	<b>Resultados</b>	1
IV-A	Análise do melhor caso . . . . .	1
IV-B	Análise de experimentos . . . . .	1
<b>V</b>	<b>Conclusões</b>	2
	<b>Referências</b>	2

## LISTA DE FIGURAS

1	Convergência no experimento de melhor caso . . . . .	2
---	--	---

# Otimização por Colônia de Formigas Aplicado ao Caxeiro Viajante

**Resumo**—Neste relatório técnico são discutidos detalhes de implementação, testes e resultados obtidos com o algoritmo de Otimização por Colônia de Formigas quando aplicado ao problema do Caxeiro Viajante. Buscando-se balancear parâmetros foram realizadas avaliações empíricas utilizando um grafo composto de 190 arestas. Após definição de parâmetros, o algoritmo balanceado foi executado 20 vezes.

## I. INTRODUÇÃO

O algoritmo de Otimização por Colônia de Formigas é uma heurística baseada em probabilidade que utiliza conceitos biológicos para resolver problemas de otimização. Ele funciona por meio de formigas artificiais que caminham aleatoriamente por um território, depositando feromônio ao longo do caminho. O feromônio é uma substância que atrai outras formigas, fazendo com que elas sejam mais propensas a escolher os caminhos com maior quantidade de feromônio. Com o tempo, os caminhos com mais feromônio tendem a ser os mais escolhidos, o que leva à convergência para uma solução ótima ou quase ótima para o problema.

O ACO é uma técnica eficaz para resolver uma variedade de problemas de otimização, incluindo o problema do caixeiro viajante, o problema de roteamento de veículos e o problema de alocação de recursos.

Esse trabalho apresenta nas seções a seguir uma implementação do algoritmo Otimização por Colônia de Formigas aplicado ao Caxeiro Viajante focado na resolução do maior caminho de um grafo. O objetivo foi realizar manualmente a parametrização do código para que se tornasse possível averiguar a aplicabilidade do mesmo, já balanceado, em uma sequência de 20 experimentos. O grafo possui 190 arestas poderadas por valores inteiros. O valor ótimo é 168. Importante ressaltar também que foram realizados testes em grafos menores com valores de tipo ponto flutuante.

## II. DEFINIÇÃO DO PROBLEMA

Considerando um grafo  $G(V,E)$  onde  $V$  é um conjunto que contém  $u,v$  sendo  $u$  um vértice de origem e  $v$  um vértice de destino e, para cada  $V$  está associado uma ponderação  $E$ , sendo  $E$  um conjunto de valores inteiros, pretende-se encontrar o conjunto  $P^*$  que contém a sequência de vértices que determinam o caminho com maior somatório dos pesos de  $E$  partindo de um  $u$  inicial qualquer e chegando-se a um  $v$  final qualquer que tenha como próximo e último elemento da sequência o próprio  $u$ , passando-se por todos os vértices.

## III. METODOLOGIA

O Otimização por Colônia de Formigas implementado para avaliação neste relatório seguiu os mesmos passos propostos

e já comentados na seção de Introdução, contudo, algumas estratégias foram adicionadas buscando-se atingir melhorias na resolução do problema proposto. As principais são descritas a seguir.

### A. Representação

A representação utilizada foi um dicionário. Cada chave do dicionário corresponde a um vértice de origem do grafo. Cada chave tem associado uma lista de tuplas contendo o vértice destino e o custo da operação.

### B. Parametrização

Levando-se em conta que a versão do algoritmo utilizada foi exatamente a clássica, foram realizados testes empíricos variando-se, vez a vez, cada um dos parâmetros para se observar comportamentos como velocidade de convergência e qualidade do melhor indivíduo obtido. Após diversos testes, a parametrização utilizada foi a seguinte:

- Feromônio inicial: 0.01
- Quantidade de formigas: 150
- Taxa de evaporação ( $p$ ): 0.07
- Quantidade de iterações: 100

Foram realizados, com estas configurações, um total de 20 execuções. Os resultados mostrados na seção a seguir dizem a respeito a um destes experimentos (melhor caso) e, na sequência, a todos.

## IV. RESULTADOS

### A. Análise do melhor caso

Tomando-se por base um dos experimentos realizados (melhor caso) pode-se observar na Figuras 1, respectivamente, a convergência da melhor solução, iteração a iteração, e, a rota encontrada ao final de 100 iterações.

A rota que apresentou o valor de 122 e foi composta da seguinte sequência de vértices  $P^* = 16.0, 6.0, 12.0, 17.0, 1.0, 19.0, 9.0, 4.0, 7.0, 13.0, 2.0, 5.0, 15.0, 18.0, 10.0, 11.0, 8.0, 14.0, 20.0, 3.0$

Como pode-se perceber, o algoritmo converge, no entanto, prematuramente e, após encontrar o maior caminho, não há uma observação de melhora.

### B. Análise de experimentos

Vale ressaltar que na apresentação em sala de aula o código apresentava erros no cálculo do maior caminho. Após a apresentação me dediquei em resolver o erro para entregar da melhor forma que conseguiria. Todos os testes que foram realizados foi com base na correção do código.

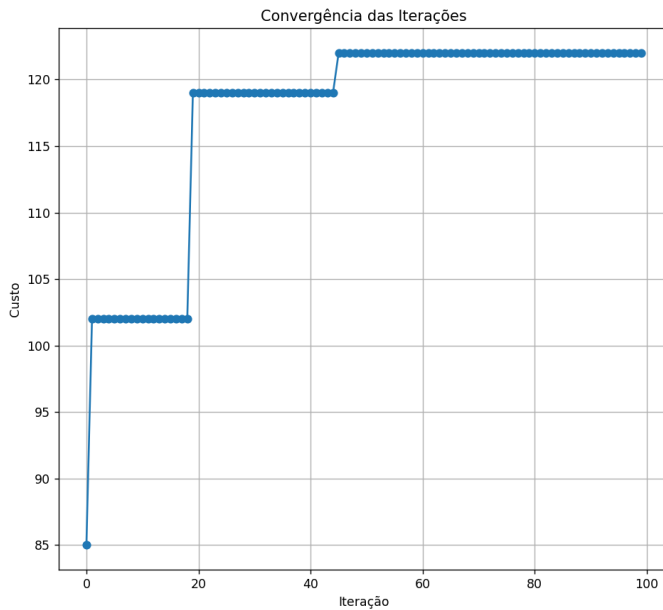


Figura 1: Convergência no experimento de melhor caso

Tomando-se por base 20 execuções do algoritmo obteve-se quanto aos valores das melhores rotas, média de 113,4, mediana de 112 e desvio padrão de 4,25. A pior rota encontrada foi de 107.

Os custos encontrados nas 30 execuções foram os seguintes, sequencialmente: 109, 108, 113, 111, 112, 122, 112, 119, 112, 111, 107, 110, 113, 122, 119, 115, 111, 112, 115 e 115.

Por fim, utilizando-se a linguagem de programação Python 3 e realizando-se experimentos sob um processador Intel i5 de 10ª geração com 2.1GHz, o pior tempo de processamento obtido em todas as 20 execuções foi de 34.95 segundos.

## V. CONCLUSÕES

Neste relatório foi apresentada uma análise do Algoritmo Otimização por Colônia de Formigas quando submetido ao problema do Caxeiro Viajante.

Os principais resultados observados deixaram claro que o algoritmo é capaz de identificar pontos ótimos, no entanto, na sua versão clássica o algoritmo demonstrou grande dificuldade em sair de máximos locais apresentando uma convergência prematura.

O melhor resultado obtido foi de 122, em 20 execuções. Dado que o melhor resultado possível é 168, houve considerável distância para o ótimo global.

## REFERÊNCIAS

- [1] DORIGO, M. Maniezzo; COLORNI, A. The ANt System: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man and Cybernetics - Part B, v. 26, n. 1, p. 1-13, 1996.