

# ALGORITMOS GENÉTICOS – SISTEMA PARA A BUSCA DA MENOR DISTÂNCIA ENTRE CIDADES SEDES DA COPA DO MUNDO DE 2014

Andre Geraldo da Silva<sup>1</sup> <andregeraldo.silva@gmail.com>

Thiago Bandeira<sup>2</sup> <thiagobandeirars@gmail.com>

Universidade Luterana do Brasil (Ulbra) – Curso de Ciência da Computação – Câmpus Gravataí  
Av. Itacolomi, 3.600 – Bairro São Vicente – CEP 94170-240 – Gravataí - RS

28 de Junho de 2013

## RESUMO

Este artigo descreve o processo de desenvolvimento de um sistema para a busca da menor distância entre as cidades sedes da copa do mundo de 2014 utilizando Algoritmos Genéticos.

**Palavras-chave:** AG; Algoritmos Genéticos.

## ABSTRACT

**Title:** “Genetic Algorithms – System for search of less distance between of seats world cup 2014”

*This article describes the process of developing a system for searching the shortest distance between the cities hosting World Cup 2014 using genetic algorithms.*

**Key-words:** AG; Genetic Algorithms.

## 1 INTRODUÇÃO

As pesquisas e estudos sobre o desenvolvimento de sistemas utilizando Algoritmos Genéticos é recente. Essa técnica da IA (Inteligência Artificial) possibilita grandes vantagens quando bem aplicada. Os Algoritmos Genéticos são inspirados no princípio da evolução das espécies de Charles Darwin. São algoritmos probabilísticos que fornecem um mecanismo de busca paralela e adaptativa baseado no princípio da sobrevivência, onde os melhores indivíduos de uma população tem mais chances de reproduzir uma população ainda melhor que a sua.

Atualmente os Algoritmos Genéticos são muito utilizados para resolver Problemas de Otimização Combinatória (POC), nestes problemas o objetivo é atribuir valores a um conjunto de variáveis de decisão, de tal modo que uma determinada função objetivo seja otimizada, atendendo um determinado conjunto de restrições. O Problema do Caxeiro Viajante, o Problema da Mochila e o do roteamento de veículos, são exemplos de Problemas de Otimização Combinatória.

O sistema desenvolvido utiliza esta técnica para buscar a menor distância para percorrer todas as cidades sedes da copa do mundo de 2014. A seguir será apresentado o objetivo do sistema, a sua arquitetura, as etapas do algoritmo e conclusões.

## 2 OBJETIVO

O sistema consiste em percorrer todas as cidades sedes da copa do mundo de 2014 a fim de encontrar a menor rota, iniciando e terminando em Porto Alegre.

A distância entre as cidades é conhecida e já esta organizada nas estruturas de dados do sistema.

---

1 Aluno da disciplina de Inteligência Artificial II do curso de Ciência da Computação na Ulbra Gravataí..

2 Aluno da disciplina de Inteligência Artificial II do curso de Ciência da Computação na Ulbra Gravataí.

### 3 ARQUITETURA

O sistema foi desenvolvido na linguagem *Java*, e sua arquitetura foi estruturada com as técnicas de orientação a objetos onde foram definidos os seguintes objetos:

- *AG* – Classe principal do sistema onde são instanciados os objetos para a execução do algoritmo;
- *População* – Classe que possui os objetos da população do algoritmo genético, possui como seus atributos o tamanho da população e uma lista de objetos *Cromossomo*. O tamanho da população é definido em 11 cromossomos;
- *Cromossomo* – Classe que possui todos os atributos de um cromossomo, contém uma lista de genes, onde cada gene é um texto que identifica uma cidade sede. Cada cromossomo possui uma lista de 11 genes;
- *DistanciaCidades* – Classe que contém como atributos todas as cidades sedes e também a distância que cada cidade possui em relação as outras. Cada cidade é um *HashMap*, onde a chave é o texto da cidade e o valor é a distância para chegar na outra cidade;
- *Log* – Classe que salva em arquivos as gerações geradas pelo algoritmo. Cada geração é salva com o nome gera;
- *PercorreCidadesAG* – Classe que possui o método main. Instância o objeto *AG* e executa o algoritmo genético em busca da melhor solução possível.

### 4 FUNÇÃO DE APTIDÃO

A função de aptidão é definida pelo melhor cromossomo de uma população. A parada do algoritmo é definida quando o resultado da função de aptidão se repetir em um determinado número de vezes ou quando o limite de geração estourar. A cada geração é feita uma busca na população para retornar o melhor cromossomo desta população.

Abaixo uma lista de algumas constantes importantes do algoritmo.

- Limite de gerações: 100 mil gerações;
- Quantidade de repetições do melhor cromosso: 10 mil repetições;
- Tamanho da população: 11 cromossomos;
- Tamanho do cromossomo: 11 genes.

## 5 CRUZAMENTO E MUTAÇÃO

### 5.1 Cruzamento

O cruzamento é realizado escolhendo-se dois pais aleatoriamente e executando um torneio entre eles, o melhor é escolhido e adicionado em uma lista de pais. Após a escolha dos pais, se faz o cruzamento dos cromossomos e com isso é gerados dois filhos. O cruzamento é feito utilizando o operador OX (*Order Crossover*). Esse operador gera filhos escolhendo uma subsequência de um pai e preservando a ordem relativa das cidades do outro pai.

A Figura 1 apresenta um exemplo utilizando este operador de cruzamento.

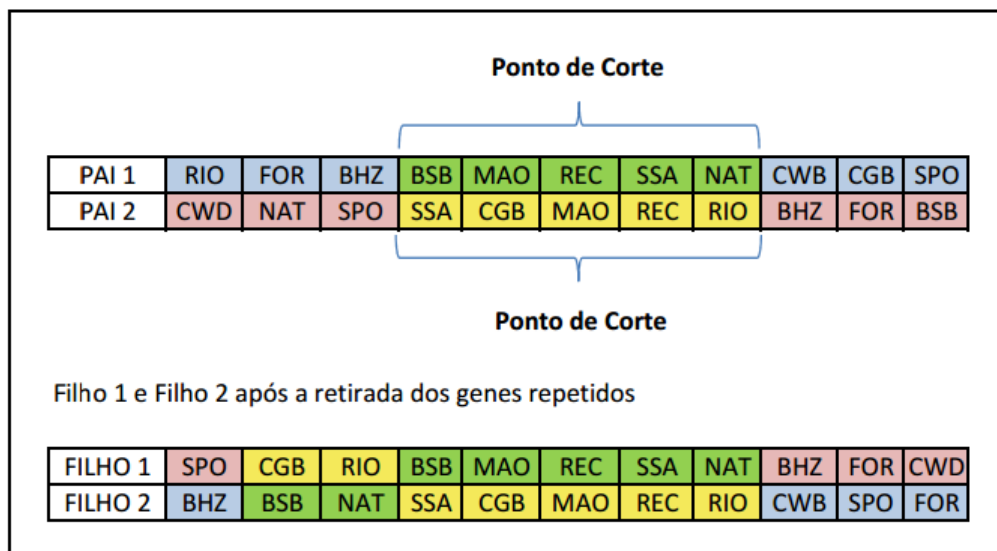


Figura 1 – Cruzamento dos cromossomos.

### 5.2 Mutação

A mutação é uma etapa realizada após o cruzamento. Para realizar a mutação neste sistema, foi escolhido dois genes aleatoriamente, e logo troca-se as suas posições no cromossomo.

A Figura 2 apresenta um exemplo da mutação dos cromossomos filhos.

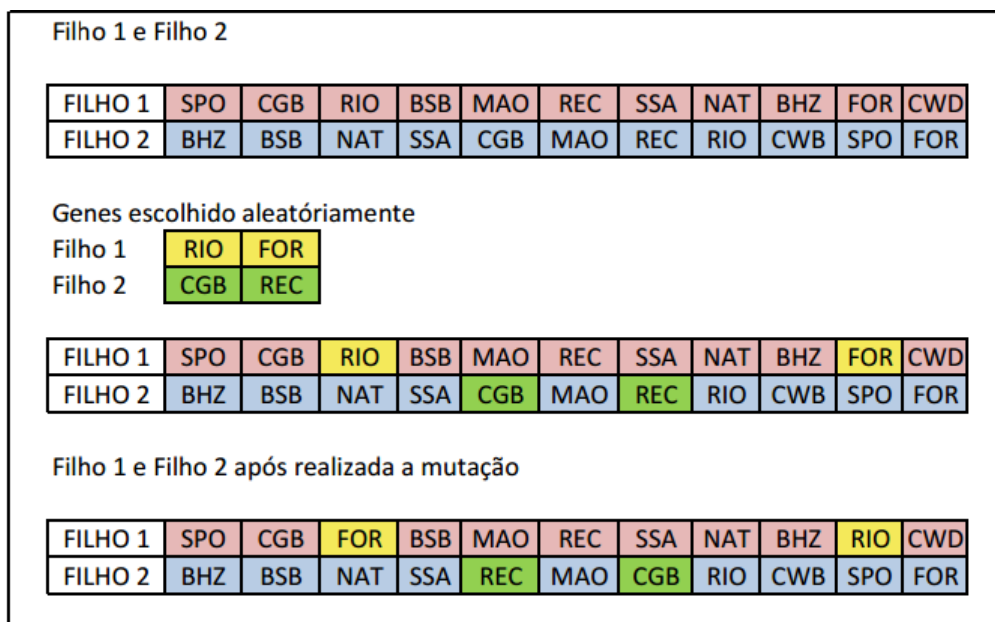


Figura 2 – Mutação dos cromossomos filhos.

## 6 TESTES

Em média o algoritmo encontra uma solução aceitável entre 10 mil á 14 mil gerações.

Abaixo a tabela 1 apresenta os resultados gerados em 12 testes executados.

**Tabela 1 – Resultados dos testes.**

	<b>Distância percorrida</b>	<b>Qtd gerações</b>	<b>Qtd repetições melhor cromossomo</b>
Teste 1	10211 km	10493	10000
Teste 2	10211 km	11082	10000
Teste 3	10142 km	10657	10000
Teste 4	10142 km	10023	10000
Teste 5	10211 km	14057	10000
Teste 6	10255 km	10056	10000
Teste 7	10142 km	10268	10000
Teste 8	10211 km	10156	10000
Teste 9	10211 km	10165	10000
Teste 10	10142 km	11351	10000
Teste 11	10235 km	10122	10000
Teste 12	10142 km	10384	10000

## 7 CONCLUSÃO

O desenvolvimento de software utilizando Algoritmos Genéticos não é algo complexo, mas existem algumas variáveis e métodos que devem ser considerados e analisados para um melhor desempenho da solução.

Este artigo apresentou uma solução otimizada para o problema proposto pelo professor em aula, foram aplicados diversos conhecimentos adquiridos em aula para que este objetivo fosse alcançado.