



Algoritmos genéticos para otimização de funções

Genetic Algorithms for Function Optimization

Isaac Vinicius Fontes Sampaio

Hugo Valadares Siqueira

RESUMO

Algoritmo genético (AG) é uma subclasse dos algoritmos evolutivos, derivado da computação bio-inspirada, ou seja, seu funcionamento é inspirado em conceitos da natureza. O algoritmo procura emular conceitos biológicos em códigos com a finalidade de efetuar otimizações. Para isso, usou-se um problema típico de benchmark com relação direta com tarefas da área de logística e otimização inteira, visando a diminuição do custo de deslocamento do viajante (caixeiro viajante). Para essa situação foi feita uma simulação com população de 400, elitismo de 55, mutação de 0,001 e com 1000 gerações de novas populações. Nota-se que o método é capaz de adequadamente resolver a tarefa.

Palavras-chave: Algoritmos, Genética, Otimização.

ABSTRACT

Genetic algorithm (GA) is a subclass of evolutionary algorithms, derived from bio-inspired computing, that is, its operation is inspired by concepts from nature. Algorithm seeks to emulate biological concepts in codes in order to carry out optimizations. For this, we used a typical benchmark problem with a direct relationship with tasks in the entire logistics and optimization area, aiming to reduce the traveler's travel cost (traveling salesman). For this situation, a simulation was made with a population of 400, elitism of 55, mutation of 0.001 and 1000 generations of new populations. Please note that the method is capable of properly solving the task.

Keywords: Algorithms, Genetics, Optimization.

INTRODUÇÃO

Algoritmo genético (AG) é uma subclasse dos algoritmos evolutivos, derivados da computação bio-inspirada, ou seja, seu funcionamento é inspirado em conceitos da natureza, nesse caso em seleção natural (HOLLAND, 1992), (LOPES, 2013).

Esse processo procura emular conceitos biológicos em códigos com a finalidade de efetuar otimizações. Os algoritmos possuem influência da genética, logo a principal fonte é a teoria da evolução proposta por Charles Darwin em meados do século XIX. Visto que cada planta ou animal apresentam diferenças individuais, ele apontou que as espécies podem passar por mutações espúrias que eventualmente levam a uma vantagem de sobrevivência ao se adaptarem melhor ao ambiente, o que culmina em maiores probabilidades de sobrevivência para a próxima geração. Este fenômeno tem sido chamado de sobrevivência do mais adaptado, também conhecido como seleção natural (DE CASTRO, 2006), (HILLIER, 2006).

A genética moderna fornece uma explicação mais elaborada para o processo evolutivo. Em qualquer espécie que se reproduz por meio da reprodução sexuada, cada filho herda alguns cromossomos de cada genitor



e os genes nos cromossomos determinam as características individuais. Um filho que herde as melhores características de seus pais tem mais probabilidade de viver até a idade adulta e então pode transmitir tais características para a próxima geração (GOLDBARG, 2005), (RUSSELL, 2013). Por meio desse processo, a população tende a melhorar seu grau de adaptação lentamente com o tempo.

O segundo fator de influência é uma taxa de mutação aleatória de baixo nível do DNA cromossômico, de modo que mutações ocasionais podem alterar as características herdadas dos pais pela prole. Embora a maioria das mutações não tenha efeito ou sejam desvantajosas, algumas delas fornecem melhorias desejáveis. A prole, neste cenário, tem maior probabilidade de sobreviver e contribuir para o futuro banco genético da espécie (LOPES, 2013).

Essas ideias inspiraram John Holland (HOLLAND, 1992) a modelar este processo com vistas a solucionar problema de otimização. Uma solução viável para um determinado problema corresponde a um indivíduo de uma determinada espécie, em que a aptidão de cada um é medida pelo valor da função objetivo, score este conhecido como fitness. Em vez de lidar com uma solução experimental de cada vez (como a forma básica de pesquisa tabu e maleabilidade de simulação), agora usa-se toda a população de solução experimental.

PROCEDIMENTOS OPERACIONAIS

Antes do algoritmo genético ser implementado, algumas questões precisam ser respondidas, a fim de guiar a busca para seu funcionamento. Isso se mostra através da escolha de parâmetros abaixo:

1. Qual deve ser o tamanho da população?
2. Como devem ser selecionados os membros da população atual para se tornarem pais?
3. Como as características dos filhos devem ser derivadas das características dos pais?
4. Como as mutações devem ser injetadas nas características dos filhos?
5. Qual regra da parada deve ser usada?

Claramente, tais respostas dependem da estrutura do problema em questão.

A implementação do AG dentro de linguagens de programação pode ser realizada de forma manual, isto é, sem o uso de bibliotecas específicas para desenvolvimento do algoritmo. Mas nos dias atuais há uma gama bibliotecas denominadas “evolutivas” que constituem um conjunto de ferramentas específicas que, no geral, simulam os passos descritos do AG (DEAP, Pyevolve; etc.).

Exemplos de aplicação de um AG se concentram em soluções para a minimização/maximização de funções custo, ou seja, a otimização. Aqui iremos exemplificar este processo através do problema do caixeiro viajante.

Caixeiro viajante é um problema benchmark, mas com relação direta com tarefas da área de logística e otimização inteira, visto que ele retrata o deslocamento do viajante. O intuito dele é determinar qual o menor caminho possível para se alcançar todos os destinos desejados. Dessa forma, podemos mapear esta ideia para resolver uma instância deste problema como segue:

- Gene: uma cidade representada por coordenadas;
- Indivíduo: uma rota única que satisfaz as condições do problema;
- População: conjunto de indivíduos, ou seja, uma coleção de rotas possíveis;
- Pais: combinação de duas rotas para criar uma rota nova;
- Fitness: uma função que avalia cada rota (no caso, a rota que tem menor distância);
- Mutação: uma forma de variar a população trocando de forma aleatória duas cidades em uma rota;
- Elitismo: um fator que permite passar para próximas gerações indivíduos com melhor desempenho.



RESULTADOS

Fazendo uma simulação com um código da biblioteca Colab, foram inseridos os seguintes valores: população de 400, elitismo de 55, mutação de 0,001 e com 1000 geração de novas populações. O gráfico da figura 1 mostra a melhor de 30 execuções do algoritmo:

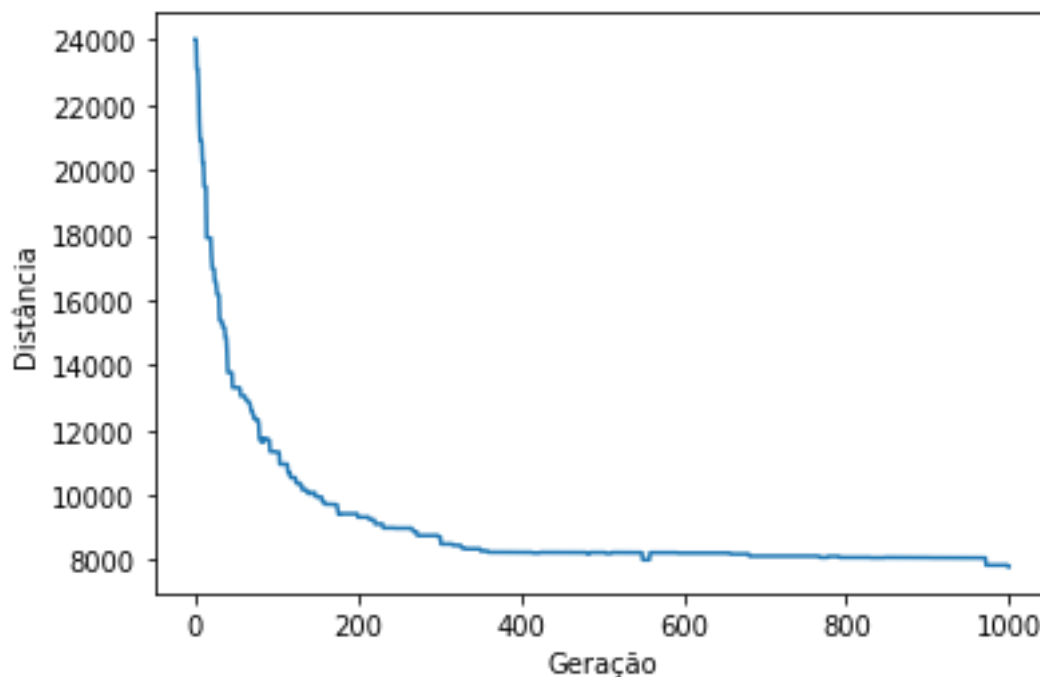


Figura 1- Distância mínima obtida no problema do caixeiro viajante após 1000 gerações

Fonte: Autoria própria

No gráfico pode-se perceber que conforme as gerações vão passando, a distância total percorrida vai diminuindo. Reparando no gráfico entre a geração 0 e 400, pode-se observar que a otimização funciona como uma exponencial com grande impacto na diminuição da distância. Após a geração 400, pode-se perceber que a variação da distância tem um comportamento próximo do linear, embora ainda sim perceptível, mas não tão agressivo. Isso prova que com os passar das gerações, o AG vai se adaptando e melhorando o seu comportamento para a realização do objetivo afim de encontrar uma solução ótima.

CONCLUSÃO

Este trabalho apresentou uma investigação acerca das etapas construtivas de um algoritmo genético para solução de problemas de mundo real. Foram discutidos os passos envolvidos no processo, apresentadas as inspirações provenientes da natureza, bem como modos de implementação.

Como estudo de caso foi apresentada uma instância do problema do caixeiro viajante, o qual foi solucionado com sucesso para os objetivos didáticos desta proposta.

Como trabalhos futuros pretende-se investigar outros algoritmos da literatura, bem com a aplicação em outros problemas relevantes de otimização.



SEI-SICITE 2021

Pesquisa e Extensão para um
mundo em transformação

XI Seminário de Extensão e Inovação
XXVI Seminário de Iniciação Científica e Tecnológica
08 a 12 de Novembro - Guarapuava/PR



REFERÊNCIAS

- HOLLAND, J. H. **Genetic algorithms**. *Scientific American*, 1992
- LOPES, H. S.; RODRIGUES, L. C. A. **Meta-heurísticas em pesquisas operacionais**. Curitiba, PR, Omnipax, 2013.
- DE CASTRO, L. N. **Fundamentals of natural computing: basic concepts, algorithms, and applications**, CRC Press, 2006.
- HILLIER, F. S. **Introdução à Pesquisa Operacional**. São Paulo, 2006. McGraw-Hill.
- GOLDBARG, M. C.; LUNA, H. **Otimização Combinatória e Programação Linear**. Rio de Janeiro, Elsevier, 2005.
- RUSSELL, S. **Inteligência Artificial**. Rio de Janeiro, Elsevier, 2013.