Programação Evolutiva, EP

3.6.1 Taxonomia

Programação Evolutiva (Evolutionary Programming) é um algoritmo de Otimização Global e uma instância de Algoritmo Evolutivo no campo da Computação Evolutiva. A abordagem é uma técnica irmã de outros Algoritmos Evolutivos, como o Algoritmo Genético (Seção 3.2) e os Sistemas de Classificação por Aprendizado (Seção 3.9). Às vezes, é confundida com a Programação Genética, dada a semelhança no nome (Seção 3.3), e, mais recentemente, mostra uma forte semelhança funcional com as Estratégias Evolutivas (Seção 3.4).

3.6.2 Inspiração

A Programação Evolutiva é inspirada pela teoria da evolução por meio da seleção natural. Especificamente, a técnica é inspirada pelo processo evolutivo em nível macro, ou seja, no nível de espécie (fenótipo, hereditariedade, variação), sem se preocupar com os mecanismos genéticos da evolução (genoma, cromossomos, genes, alelos).

3.6.3 Metáfora

Uma população de uma espécie se reproduz, criando descendentes com pequenas variações fenotípicas. Os descendentes e os pais competem com base em sua adequação ao ambiente, onde os membros mais aptos geralmente constituem a próxima geração e são os que têm a oportunidade de se reproduzir. Esse processo se repete, melhorando a adequação adaptativa entre a espécie e o ambiente.

3.6.4 Estratégia

O objetivo do algoritmo de Programação Evolutiva é maximizar a adequação de uma coleção de soluções candidatas no contexto de uma função objetivo de um determinado domínio. Esse objetivo é perseguido usando um modelo adaptativo com substitutos para os processos de evolução, especificamente hereditariedade (reprodução com variação) sob competição. A representação utilizada para as soluções candidatas é diretamente avaliável por uma função de custo ou objetivo do domínio.

3.6.5 Procedimento

O Algoritmo 3.6.1 fornece um pseudocódigo do algoritmo de Programação Evolutiva para minimizar uma função de custo.

3.6.6 Heurísticas

A representação das soluções candidatas deve ser específica ao domínio, como números reais para a otimização de funções contínuas.

O tamanho da amostra para a seleção por torneio durante a competição geralmente varia entre 5% e 10% do tamanho da população.

Tradicionalmente, a Programação Evolutiva utiliza apenas o operador de mutação para criar novas soluções candidatas a partir das existentes. O operador de crossover, utilizado em outros Algoritmos Evolutivos, não é empregado na Programação Evolutiva.

A Programação Evolutiva se preocupa com a ligação entre as soluções candidatas pais e filhos, sem utilizar substitutos para mecanismos genéticos.

A otimização de funções contínuas é uma aplicação popular da abordagem, onde representações de valores reais são utilizadas com um operador de mutação baseado em distribuições gaussianas.

Os parâmetros específicos de mutação utilizados na aplicação do algoritmo para otimização de funções contínuas podem ser adaptados em conjunto com as soluções candidatas .

3.6.7 Listagem de Código

A Listagem 3.5 fornece um exemplo do algoritmo de Programação Evolutiva implementado na linguagem de programação Ruby. O problema de demonstração é uma instância de otimização de função contínua que busca minimizar f(x), onde f = Σx², −5.0 < xi < 5.0 e n = 2. A solução ótima para essa função de bacia é (x₀, …, xn-1) = 0.0. O algoritmo é uma implementação da Programação Evolutiva baseada na implementação clássica para otimização de funções contínuas de Fogel et al. , com variância adaptativa por variável, conforme descrito por Fogel em uma variação auto-adaptativa na página 160 de seu livro de 1995.

3.6.8 Fontes Primárias

A Programação Evolutiva foi desenvolvida por Lawrence Fogel, descrita em artigos iniciais (como ) e, posteriormente, tornou-se o foco de sua dissertação de doutorado . Fogel focou no uso de um processo evolutivo para o desenvolvimento de sistemas de controle utilizando representações de Máquinas de Estados Finitos (FSM). O trabalho inicial de Fogel sobre Programação Evolutiva culminou em um livro (coautorado com Owens e Walsh) que elaborou a abordagem, concentrando-se na evolução de máquinas de estado para a previsão de símbolos em séries temporais de dados .

Saiba Mais

O campo da Programação Evolutiva permaneceu relativamente inativo por 30 anos até ser revivido pelo filho de Fogel, David. Trabalhos iniciais consideraram a aplicação da Programação Evolutiva a sistemas de controle , e, posteriormente, à otimização de funções (identificação de sistemas), culminando em um livro sobre a abordagem e na dissertação de doutorado de David Fogel . Lawrence Fogel colaborou na revitalização da técnica, incluindo revisões e extensões que se tornaram o foco da abordagem na otimização de funções .

Yao et al. forneceram um estudo seminal da Programação Evolutiva, propondo uma extensão e comparando-a com a abordagem clássica em um grande número de problemas de teste . Finalmente, Porto fornece uma excelente visão geral contemporânea do campo e da técnica .