

Разработка эффективных операторов скрещивания и мутации в мультиплоидных алгоритмах

Коханов А.И., Университет ИТМО, г. Санкт-Петербург
Научный руководитель — Буздалов М.В., к.т.н., научный сотрудник ФИТиП
Университета ИТМО

Введение

Генетические алгоритмы [1] основываются на принципах естественной эволюции и обычно оперируют гаплоидными особями, состоящими из одной аллели. Однако многие биологические виды обладают диплоидными или мультиплоидными геномами, в которых активность каждого гена в первом приближении определяется с помощью доминирования (гены могут быть доминантными или рецессивными). Мультиплоидность помогает запоминать ранее эффективные адаптации. Таким образом, она полезна в ситуациях, когда функция приспособленности нестационарна (меняется со временем), так как мультиплоидные особи быстро реагируют и эффективнее приспособляются, храня в себе рецессивные гены. Становясь рецессивными, гены могут защитить сами себя от уничтожения в процессе развития популяции.

Одной из проблем многих эволюционных алгоритмов является стагнация и недостаточно быстрое развитие особи. Алгоритмам, использующим и операторы скрещивания и операторы мутации, сложно быть быстрее, чем унарные алгоритмы (использующие только операторы мутации). Недавние теоретические исследования полностью опровергли данную точку зрения, что, вероятно, должно быть очень на руку алгоритмам, использующим мультиплоидную схему, ведь в их работе операторы скрещивания крайне необходимы.

Цель работы

Целью данной работы является разработка эффективного мультиплоидного алгоритма, работающего быстрее алгоритмов оптимизации (1+1) EA [2] или RLS [3] и позволяющего достичь высокого разнообразия особей.

Промежуточные результаты

Были изучены и проанализированы основные способы оптимизаций кроссовера на основе “функции прыжка” [4] и жадного алгоритма. В частности, последний проигрывает унарным алгоритмам за счет того, что кроссовер может генерировать особи, идентичные их предкам.

Описание предлагаемого подхода

Предлагается реализовать генетический алгоритм на базе кроссовера с предложенной оптимизацией [5]. В зависимости от результата применения кроссовера меняется вероятность распределения бит при мутации. Асимптотически данная модификация не влияет на оценку алгоритма, зато значительно уменьшает константу.

Список литературы

1. Sean Luke, Essentials of Metaheuristics – George Mason University, 2015
2. Droste, S., Jansen, T., Ingo, W.: On the analysis of the (1 + 1) evolutionary algorithm – Theoretical Computer Science 276 (2002), 51–81
3. Pourhassan M., Roostapour V., Neumann F. – Runtime Analysis of RLS and (1+1) EA for the Dynamic Weighted Vertex Cover Problem – March 9, 2019

4. Jansen, T., Wegener, I.: The analysis of evolutionary algorithms - a proof that crossover really can help – *Algorithmica* 34, 47–66 (2002)
5. Sudholt Dirk, Crossover Speeds Up Building-Block Assembly – GECCO'12, 2012