Evolutionary Algorithms

Delft University of Technology

Practical Assignment 1

Deadline: Thursday, May 16, 2019, 23:59

Deliverables:

1. **.zip** file with
   a. source code
   b. scripts to compile your source code
   c. scripts to re-run your final (reported) experiments
2. **.pdf** file with written report in English; **NOTE: Use at most 3 pages of text using a similar font(size) and similar margins as this document! You may however use additional pages (up to a total of 10) for graphs and tables.**

Background:

The primary goal of both practical assignments is that you gain hands-on experience with EA design, implementation, and experimentation, complementing the more fundamental and theoretical aspects presented during the lectures. In addition, the practical assignments are meant to test your (ability to gain) insights into the workings of EAs and to design experiments that can support claims that you can make about this. The written report for each assignment should contain exactly such claims and (results of) supporting experiments.

Each assignment is to be performed in groups of 2 students. Practical assignment 1 will account for 10% of the final course grade. Practical assignment 2 will account for 30%.

Assignment:

Optimization problem

In this assignment, you will be solving the following binary optimization problem that calls for maximization of a function denoted $f_{P1}(\boldsymbol{x}, k, m, d)$ (to reflect the fact that this function pertains to practical assignment 1), i.e.:

$$\max_{\boldsymbol{x} \in \{0,1\}^{\ell}} \{f_{P1}(\boldsymbol{x}, k, m, d)\}$$

where

$$f_{P1}(\boldsymbol{x}, k, m, d) = \sum_{i=0}^{m-1} f_{P1}^{Sub}\big((\boldsymbol{x}_{ik}, \boldsymbol{x}_{ik+1}, \dots, \boldsymbol{x}_{ik+k-1}), k, d\big)$$

with

$$f_{P1}^{Sub}(\boldsymbol{b}, k, d) = \begin{cases} 1 & if\ u(\boldsymbol{b}) = k \\ (1-d)\dfrac{k-1-u(\boldsymbol{b})}{k-1} & otherwise \end{cases}$$

and

$$u(\boldsymbol{b}) = \sum_{i=0}^{|\boldsymbol{b}|-1} \boldsymbol{b}_i$$

and where $\ell = |\boldsymbol{x}| = mk$ for some $m \in \mathbb{N}^+$ and $k \in \{3,5,10\}$, i.e. the number of binary variables is a multiple of k.

EA

The EA that you should consider, is the following "modern" variant (P+O, tournament selection with tournament size 4) of the simple Genetic Algorithm (sGA) that was also presented during the lectures:

$modernSimpleGA(n, p_c, p_m)$ // population size n must be even

1 $t \leftarrow 0$
2 $\boldsymbol{P}^t \leftarrow createAndEvaluateInitialIndividuals(n)$
3 $\boldsymbol{while}$ $terminationCriterionNotSatisfied(\boldsymbol{P}^t)$ $\boldsymbol{do}$
  3.1 $\boldsymbol{O}^t \leftarrow \emptyset$
  3.2 $\boldsymbol{\pi} \leftarrow randomPermutation(n)$
  3.3 $\boldsymbol{for}$ $i \in \{0,1,\dots,(n/2)-1\}$ $\boldsymbol{do}$
    3.3.1 $\boldsymbol{O}^t \leftarrow \boldsymbol{O}^t \cup crossoverAndMutation(P^t_{\pi_{2i}}, P^t_{\pi_{2i+1}}, p_c, p_m)$
  3.4 $evaluateAllSolutions(\boldsymbol{O}^t)$
  3.5 $\boldsymbol{Q}^t \leftarrow \boldsymbol{P}^t \cup \boldsymbol{O}^t$
  3.6 $\boldsymbol{P}^{t+1} \leftarrow \emptyset$
  3.7 $\boldsymbol{for}$ $j \in \{0,1\}$ $\boldsymbol{do}$
    3.7.1 $\boldsymbol{\pi} \leftarrow randomPermutation(2n)$
    3.7.2 $\boldsymbol{for}$ $i \in \{0,1,\dots,(2n/4)-1\}$ $\boldsymbol{do}$
      3.7.2.1 $\boldsymbol{P}^{t+1} \leftarrow \boldsymbol{P}^{t+1} \cup bestIndividual\left(Q^t_{\pi_{4i}}, Q^t_{\pi_{4i+1}}, Q^t_{\pi_{4i+2}}, Q^t_{\pi_{4i+3}}\right)$
  3.8 $t \leftarrow t+1$

Experimental research

The goal is to research the (in)capability of the simple GA to solve the aforementioned optimization problem, for different values of parameters d, k, m, and n, and for different variation operators.

- Regarding d, consider (at least) two cases: $d = 1/k$ and $d = 1 - 1/k$.
- Regarding k, consider (at least) the case of $k = 5$.
- Regarding m, consider (at least) the cases of $m \in \{1,2,4,8,16\}$.
- Regarding n, determine the value for which the simple GA performs best, meaning that the smallest number of evaluations (or time, your choice) is required to find the optimal solution for the first time (i.e., the first time an optimal solution is evaluated in the fitness function).
- Regarding variation, for crossover you should consider both the *uniform crossover* and the *one-point crossover* operator, each in the setting that 2 offspring are generated from 2 parents. Moreover, in this practical assignment, the crossover probability p_c (the probability that one such crossover event happens) will be set to 1 (i.e., crossover happens for each pair of parents) and the mutation probability p_m (the probability that *any* bit in *any* offspring solution is flipped) will be set to 0 (i.e., no mutation is used).

Implementation

To perform this practical, you will need an implementation of the simple GA and the optimization problem defined above. To help you get started, an implementation of the simple GA is provided to you in Java, but with certain components missing: the fitness function and the crossover operators. You must finish the implementation yourself. If you wish, you can also make your own implementation completely from scratch. For the coding language then you may choose C, C++, Java, Python, or MATLAB.

Besides the base code, you will also need to document and implement the scripts that execute the final experiments that you will report on. For the sake of reproducibility, in your final submission, include not only instructions on how to compile and run your code, but also the scripts (in separate folders) that clearly identify how you ultimately ran the simple GA to support the findings in your report.

Goal

The experiments amount to **scalability analyses**. These are closely related to the notion of computational complexity, as they enable observing what happens to the required resources of an algorithm as the problem size increases.

The final goal, i.e., what should be written down in your report, is to determine under ***what*** conditions the simple GA solves the optimization problem efficiently, **as well as** to (do your best to) explain ***why***. When drawing conclusions, be sure to describe (and reason about) the influence of the different parameters and variation operators. Support your observations and claims by summarizing the results of your experiments in graphs and/or tables. Note: do not simply plot everything there is to plot and put it all in your report but include only graphs and/or tables that support your reasoning.