Parameter-less Genetic Algorithm: A Worst-case Time and Space Complexity Analysis

Martin Pelikan
Dept. of General Engineering
and Dept. of Computer Science
University of Illinois, Urbana, IL 61801
pelikan@illigal.ge.uiuc.edu

Fernando G. Lobo
Dept. of Environmental Engineering
Universidade Nova de Lisboa
2825 Monte da Caparica, Portugal
fgl@mail.fct.unl.pt

A parameter-less genetic algorithm (Harik & Lobo, 1999) is an alternative to a common trial-and-error method of tweaking the values of the parameters of the genetic algorithm in order to find a set-up to accurately and reliably solve a given problem. The algorithm fixes the selection pressure and crossover rate according to a simplified schema theorem. The selection pressure s is set to $s = 4$ (the best solution in a population gets 4 copies in the selected set of the same size as the population) and the crossover rate p_c is set to $p_c = 0.5$ (half of the pairs of solutions in the selected set undergo crossover). No mutation is used.

To eliminate the population size, the parameter-less genetic algorithm manages a number of independent runs of the genetic algorithm with different population sizes with the remaining parameters set to fixed values as outlined above. The population size used in the first run is set to a constant. The population size of each next run doubles. Runs with larger populations require proportionally more fitness evaluations to process each generation than the runs with smaller populations, and therefore it would be "unfair" to process each of the runs at the same speed with respect to the number of performed generations. Moreover, by running an infinite number of runs at the same speed an infinite loop would be inevitable. The problem can be evaded by allowing each run to use the same number of fitness evaluations throughout the entire computation.

This poster presents a worst-case analysis of the recently proposed parameter-less genetic algorithm. Time and space complexity as a function of a time required by the genetic algorithm with an optimal population size is computed.

The results suggest, that in the worst case, the paramter-less genetic algorithm requires $O\left(E_{opt} \log E_{opt}\right)$ fitness evaluations to find the solution of a required quality, where E_{opt} is the number of fitness evaluations required by the genetic algorithm with an optimal population size. The space requirements can grow as $O(G_{opt}N_{opt}) = O(E_{opt})$, where G_{opt} is the expected number of generations until convergence of the genetic algorithm with an optimal population size N_{opt}. An increased pressure can be put on runs with bigger populations, making the run with a double population use m time less function evaluations than the run with the smaller population (in the above we set $m = 2$). Using a higher value of m would change the space complexity to $O(G_{opt}N_{opt}^{log_m 2})$.

The analysis provided in this work is very useful because it gives estimates of worst-case time and space complexities of the parameter-less genetic algorithm. However, the reader should be very careful in interpreting the results given herein. The analysis does not take into account a cumulative effect of running multiple independent runs. The more runs (however badly configured), the bigger the chance to find the solution. Moreover, in actual practice, the worst-case scenario is very unlikely to occur. For a proof of the above theoretical result, please see Pelikan and Lobo (1999).

References

Harik, G., & Lobo, F. (1999). A parameter-less genetic algorithm. In Banzhaf, W., Daida, J., Eiben, A. E., Garzon, M. H., Honavar, V., Jakiela, M., & Smith, R. E. (Eds.), *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99*, Volume I (pp. 258–265). Orlando, FL: Morgan Kaufmann Publishers, San Fransisco, CA.

Pelikan, M., & Lobo, F. G. (1999). *Parameter-less genetic algorithm: A worst-case time and space complexity analysis* (IlliGAL Report No. 99018). Urbana, IL: University of Illinois at Urbana-Champaign, Illinois Genetic Algorithms Laboratory.