When Hillclimbers Beat Genetic Algorithms in Multimodal Optimization

Fernando G. Lobo
DEEI-FCT & CENSE, Universidade do Algarve
Campus de Gambelas, 8005-139 Faro, Portugal
fernando.lobo@gmail.com

Mosab Bazargani
DEEI-FCT & CENSE, Universidade do Algarve
Campus de Gambelas, 8005-139 Faro, Portugal
mbazargani@gmail.com

ABSTRACT

We show that multistart next ascent hillclimbing compares favourably to crowding-based genetic algorithms when solving instances of the multimodal problem generator. We conjecture that it is unlikely that any practical evolutionary algorithm is capable of solving this type of problem instances faster than the multistart hillclimbing strategy.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

Keywords

multimodal optimization; niching; multimodal problem generator; hillclimbing; genetic algorithms

1. THE MULTIMODAL PROBLEM GENERATOR

The multimodal problem generator was originally proposed in [1] and used by several researchers in subsequent studies. The generator creates problem instances with a certain number of peaks. For a problem with n peaks, n bit-strings of length L are uniformly randomly generated. Each of these strings is a peak (a local optimum) in the landscape. Different heights can be assigned to different peaks. To evaluate an individual x, first locate its nearest peak in Hamming space (with ties broken uniformly at random). The fitness of x is the number of bits the string has in common with its nearest peak, divided by L, and scaled by the height of the nearest peak.

2. HILLCLIMBING AND GENETIC ALGORITHMS WITH NICHING

We applied a multistart next ascent hillclimbing (MS-NAHC) algorithm to instances of the multimodal problem generator. Starting from a random solution, the algorithm reaches the top of a peak using next ascent hillclimbing (NAHC). Once there, it restarts from another random solution, and keeps doing that until a given stop criterion is satisfied. NAHC explores the neighbourhood of the current solution in a randomly generated order. As soon as a better neighbour is found, that neighbour becomes the current solution. This process is repeated until no neighbour improves upon the current solution. In this paper we consider the neighbourhood of a string x to be the set of strings whose Hamming distance to x is 1.

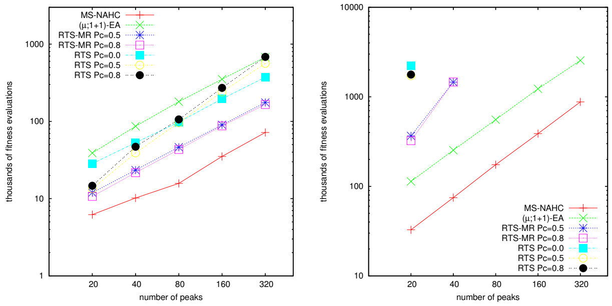
A standard genetic algorithm (sGA) without diversity preservation techniques is unable to reliably reach the top of the best peak on instances of the multimodal problem generator unless very large population sizes are used, being much less efficient than multistart next ascent hillclimbing [4].

Diversity preservation techniques, commonly referred as *niching*, are especially useful for solving multimodal optimization problems. By maintaining diversity in a population of solutions it is expected that an evolutionary algorithm (EA) can maintain basins of attraction of several optima for long periods of time allowing it to obtain multiple optimal or near-optimal solutions in a single run. Here we explore crowding-based niching techniques, namely restricted tournament selection (RTS) [2], and variations inspired on it. RTS incorporates the notion of local competition within a steady-state EA forcing a new individual to compete with an existing population member that is similar to it.

RTS does not have a mating restriction mechanism preventing solutions from different basins of attraction to mate with each other. As observed in [4], crossover is only beneficial in these problem instances when it crosses solutions near the same peak. To address this issue, we implemented a mating restriction mechanism on top of RTS. We name the resulting method RTS-MR. As opposed to RTS, only one solution is randomly chosen from the population, call it A. Ideally A should mate with a solution that is not to far away from it, i.e., a solution in the same basin of attraction. The obvious way to achieve that is to implement the same method employed by RTS for finding a not dissimilar individual to compete with, and use it for the mating phase as well. As such, instead of picking the second solution B at random, we scan w individuals at random from the population and pick the one that is most similar (but whose distance to it is at least 2 bits) to mate with it. The 2-bit minimum distance restriction is used because crossing two bit strings whose Hamming distance is less than 2 always produces children identical to the parents, regardless of the crossover operator used. The remaining part of the algo-

GECCO '15 July 11-15, 2015, Madrid, Spain

ACM ISBN 978-1-4503-3488-4/15/07.
DOI: http://dx.doi.org/10.1145/2739482.2764666



(a) Unequal-height peaks (b) Equal-height peaks

Figure 1: Average number of fitness evaluations needed for to reach the top of the best peak for the unequal-height peak instances, and to reach the top of all peaks for the equal-height peak instances.

rithm is exactly the same as in RTS. In other words, RTS-MR implements both a mating restriction policy as well as a competition/replacement restriction.

We also tested a third algorithm that uses mutation alone and has a replacement strategy that enforces a crowding-like mechanism when mutation rates are low. The algorithm is very simple. A solution A is drawn at random from the population. That solution undergoes mutation yielding a new solution A'. Then A' competes with A and whichever is best is allowed to stay in the population. With a low mutation rate, A and A' should be similar to each other, and the competition between them enforces a crowding mechanism, just like in RTS. We name this algorithm $(\mu; 1+1)$-EA, due to its resemblance to the classical $(1+1)$ and $(\mu+1)$ EAs.

3. EXPERIMENTS

We run MS-NAHC, RTS, RTS-MR, and $(\mu; 1+1)$-EA, on randomly generated instances of the multimodal problem generator with 20, 40, 80, 160, and 320 peaks. A string length L of 100 bits is used on all instances. We did experiments with equal- and unequal-height peak instances. For the unequal-height case, the heights were linearly interpolated between 0.5 and 1.0 and the stopping criterion was to reach the top of the best peak. For equal-height instances, the stopping criterion was to reach the top of all peaks.

The implementation of RTS, RTS-MR, and $(\mu; 1+1)$-EA, was done in such a way that new individuals were only evaluated if absolutely necessary. Whenever a newly created individual was identical to one of the parent individuals, no fitness evaluation was spent. Similarly, during the local competition on RTS and RTS-MR, if the two competing individuals are identical, no fitness evaluation is spent. We did our best to use near-optimal parameter settings for the three EAs so that they could perform as best as possible.

Uniform crossover was used on all experiments. On instances of the multimodal problem generator, crossover is only effective when crossing strings in the same basin of attraction. In such cases it is as if the problem had only a single peak and that would make it equivalent to the classical *onemax* problem for which uniform crossover provides better mixing and faster convergence. We tested three crossover rates $P_c = 0.0, 0.5, 0.8$ and used bit-flip mutation with probability $1/L$. For RTS and RTS-MR, the window size w was set to 4 times the number of peaks following the recommendations given in [2] or to the population size, whichever was minimum.

With respect to population sizing we used the bisection method [5] to obtain the minimum population size that allows the algorithms to reach the target goal on 100/100 independent runs. 30 independent bisection runs were performed, yielding a total of $30 * 100 = 3000$ runs, per algorithm and per problem instance. On all runs we imposed a limit of 5 million fitness evaluations, upon which we considered the run to be unsuccessful. This limit is more than 3 times larger than the number of evaluations needed by the worst of the 100 independent runs of MS-NAHC when solving the most difficult instance: reaching the top of all peaks on a 320 equal-height peak instance.

Figure 1 reports the average number of fitness evaluations needed by the various algorithms for increasing number of peaks. MS-NAHC needs on average substantially less fitness evaluations than tuned crowding-based EAs. Additional results and their discussion can be found in [3].

4. SUMMARY AND CONCLUSIONS

This paper showed that conventional niching and mating restriction techniques incorporated in an EA were not sufficient to make them competitive with a multistart next ascent hillclimbing strategy, when solving instances of the multimodal problem generator.

We conjecture that it is unlikely that any practical evolutionary algorithm is capable of solving this type of problem instances faster than the multistart hillclimbing strategy. The reason for this claim is due to the observation that the various optima are uniformly randomly generated, and therefore completely unrelated to each other. In such cases EAs are unable to do an effective search in the space of local optima.

5. ACKNOWLEDGEMENTS

This work was supported by Fundação para a Ciência e Tecnologia under grant PTDC/EEI-AUT/2844/2012.

6. REFERENCES

[1] K. A. De Jong, M. A. Potter, and W. M. Spears. Using problem generators to explore the effects of epistasis. In T. Bäck, editor, *Proceedings of the International Conference on Genetic Algorithms (ICGA 1997)*, pages 338–345, San Francisco, 1997. Morgan Kaufmann.

[2] G. R. Harik. Finding multimodal solutions using restricted tournament selection. *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 24–31, 1995.

[3] F. G. Lobo and M. Bazargani. When hillclimbers beat genetic algorithms in multimodal optimization. arXiv Report No. 1504.06859, 2015.

[4] F. G. Lobo and C. F. Lima. On the utility of the multimodal problem generator for assessing the performance of evolutionary algorithms. In M. Keijzer et al., editors, *Proceedings of the ACM SIGEVO Genetic and Evolutionary Computation Conference (GECCO-2006)*, pages 1233–1240. ACM Press, 2006.

[5] K. Sastry. Evaluation-relaxation schemes for genetic and evolutionary algorithms. Master's thesis, University of Illinois at Urbana-Champaign, Urbana, IL, 2001.