A Closer Look to the Scouting-Inspired Evolutionary Algorithm

Konstantinos Bousmalis; Supervisor: Gillian Hayes

K.Bousmalis@sms.ed.ac.uk

Abstract: The work presented is an extension of the research that introduced the Scouting-Inspired Evolutionary Algorithm(SEA)[1] and attempts to shed more light on the behavior of the SEA.

I. Introduction

An Evolutionary Algorithm(EA) is a technique that finds optimal solutions to a given problem by evolving a set of solutions. When the search space, however, has numerous peaks, a traditional EA often gets trapped in a suboptimal solution (**premature convergence**). The Scouting-Inspired Evolutionary Algorithm(SEA) is a recently introduced technique, which helps a traditional EA avoid entrapment in local optima.[1]

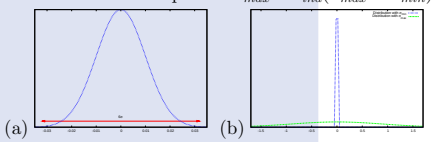
II. Background

A Traditional Evolutionary Algorithm

1. Initialization of a set (**population**) of potential solutions (**individuals**). Each individual is a series of parameters (**genes**).
2. The individuals are assigned a measure of merit(a **fitness** value).
3. A set number of individuals-parents is selected for reproduction, based on their rank and via a **selection** policy.
4. The parents are recombined by exchanging genes (**crossover**) and/or mutated to produce the next generation of individuals. When genes are represented with real values, **mutation** is achieved by adding a small value to one or more genes.
5. The process repeats from step 2 for a large number of generations.

Scouting-Inspired Evolutionary Algorithm

- In simple mutation, a small value is often randomly drawn from a Gaussian probability distribution with a preset variance σ and mean 0.
- Scouting keeps an experience database of all individuals created.
- Given a new individual, an estimated fitness is calculated by using a weighted k-nearest neighbor algorithm.
- The absolute difference of the estimated and actual fitness values is the **surprise** value for that individual, s_{ind}.
- Scouting-driven Mutation:
  1. The variance, σ, is mapped to surprise, s_{ind}. The higher the surprise, the smaller the σ, and vice versa.
  2. Given σ_{min} and σ_{max}, the σ modulator used by Pfaffmann et al[1] is the simple inverse relationship: $\sigma = \sigma_{max} - s_{ind}(\sigma_{max} - \sigma_{min})$



(a) (b)

Figure 1: (a)Gaussian distribution with σ=0.107 and mean 0. (b)The distributions corresponding to σ_{min}=0.107 and σ_{max}=0.9.

III. A Closer Look to the SEA

New variance(σ) modulator

Reproduction and analysis of the results in [1] showed that ca. 99.97% of all surprise values are in the lowest 10% of the surprise range. This results in the SEA behaving very much like an EA with σ_{max}. To achieve an even fairer comparison with the traditional EA, the modulator was changed to $\sigma = \sigma_{min} + (1 - s_{ind})(\sigma_{max} - \sigma_{min})$. σ_{max} should be large enough to allow for the necessary jump across the search space, and s_{ind} should be scaled to [0,1].

The Effect of Crossovers

Pfaffmann et al[1] introduced Scouting to an EA that used only mutation. Figure 1(a) shows the probability distribution used by the traditional EA, and Figure 1(b) the range of distributions used by the SEA, as outlined in [1]. The effect of crossovers to the SEA is also examined here.

Fitness Uniform Selection

- The traditional EA introduced in [1] uses the Roulette Wheel selection policy, which favors the fittest individuals.
- Hutter et al[2] introduced the Fitness-Uniform Selection Scheme (FUSS), which, very much like the SEA, avoids premature convergence by promoting genetic diversity in the pool of individuals.
- FUSS achieves that, by uniformly choosing a fitness level, and subsequently selecting the individual closest to that level.
- The introduction of FUSS to the SEA is examined here.

Experimental Setup

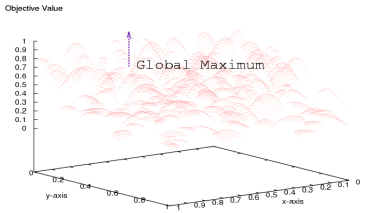


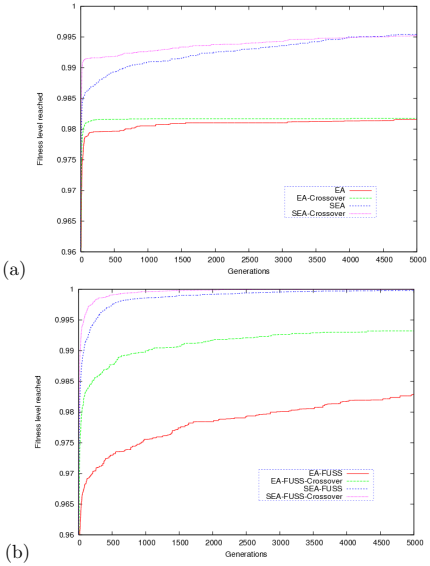
Figure 2: The test case objective function landscape

All sets of experiments presented below were run 150 times using a 2-dimensional test case with 150 peaks (Fig. 2) created by the TCG2 package [3]. They were run for 5000 generations, with populations of sizes 10, 20, 30 and 100. Only the results for 100 individuals are presented here, due to lack of space. The results for 10, 20 and 30 individuals scale down accordingly. The sets of experiments run and presented are:

1. Traditional EA using Roulette Wheel, mutation σ=0.107, no crossover.
2. SEA using Roulette Wheel, σ_{min}=0.107, σ_{max}=1.7, no crossover.
3. 1 and 2 with 50% rate of crossover and 50% rate of mutation.
4. 1, 2, and 3 using FUSS instead of Roulette Wheel.

IV.Results-Conclusion

As the results in Figure 3 show, scouting helped all other EA techniques to find fitter individuals faster. SEA performs well with its new modulator and exhibits similar improvement to EAs with or without crossovers. The combination of FUSS and SEA gives the best results, as it was hypothesized.



(a) (b)

Figure 3: The fitness level reached by generation (on average) for populations of size 100, using (a)Roulette Wheel and (b)FUSS

Future work will include a scouting-driven crossover and experimentation with a variety of problems, especially multi-dimensional ones.

References

[1] Pfaffmann, J.O.; Bousmalis, K.; Colombano, S., "A scouting-inspired evolutionary algorithm," Evolutionary Computation, 2004. CEC2004. Congress on, vol.2, no.pp. 17061712 Vol.2, 1923 June 2004

[2] Hutter, M.; Legg, S., "Fitness uniform optimization," Evolutionary Computation, IEEE Transactions on , vol.10, no.5pp. 568589, Oct. 2006

[3] M. Schmidt and Z. Michalewicz, "Test-case generator TCG2 for nonlinear parameter optimization," in Parallel Problem Solving from Nature-PPSN VI, Proceedings of the 6th International Conference, September 1820 Paris, France, vol. 1917 of Lecture Notes in Computer Science, K. Deb et. al., Eds. Heidelberg, Germany: SpringerVerlag, 2000, pp. 539-548.

School of informatics