Adaptive Selection of Helper-Objectives with Reinforcement Learning

Arina Buzdalova
St. Petersburg National Research University
of Information Technologies, Mechanics and Optics
49 Kronverkskiy prosp.
Saint-Petersburg, Russia, 197101
Email: abuzdalova@gmail.com

Maxim Buzdalov
St.Petersburg National Research University
of Information Technologies, Mechanics and Optics
49 Kronverkskiy prosp.
Saint-Petersburg, Russia, 197101
Email: mbuzdalov@gmail.com

***Abstract*—In this paper a previously proposed method of choosing auxiliary fitness functions is applied to adaptive selection of helper-objectives. Helper-objectives are used in evolutionary computation to enhance the optimization of the primary objective. The method based on choosing between objectives of a single-objective evolutionary algorithm with reinforcement learning is briefly described. It is tested on a model problem. From the results of the experiment, it can be concluded that the method allows to automatically select the most effective helper-objectives and ignore the ineffective ones. It is also shown that the proposed method outperforms multi-objective evolutionary algorithms, that were used with helper-objectives originally.**

I. Introduction

This paper is dedicated to improvement of the efficiency of single-objective evolutionary computation. Usually the aim of the evolutionary algorithms (EA) is to find an individual that maximizes the *target objective*, or the target fitness function in terms of evolutionary computation.

Sometimes additional fitness functions, in other words, helper-objectives or *helpers* [1], can be used in order to enhance the efficiency of an optimization algorithm. Such approach is presented in [1]–[3], where some single-objective optimization problems are multi-objectivised and solved with various multi-objective evolutionary algorithms. Development of an instrument that allows to automatically select the optimal helpers is an open question [1].

In this paper a method that automatically selects helpers in a single-objective evolutionary algorithm is briefly described and tested on a model problem. It chooses the most efficient fitness function at each generation of the algorithm, so as the target fitness function grows faster. There is no prior knowledge about the helpers properties. We do not aim to maximize the helpers, they are just used to increase the efficiency of the target objective optimization. The method was previously proposed in [4], [5]. In the present paper its ability to ignore inefficient helpers is revealed. It is shown that the method allows to select the most efficient helpers from an arbitrary set.

II. Method description

The proposed method is based on guiding an evolutionary algorithm (EA) by choosing fitness functions with reinforcement learning (RL) [6]. There are the target objective that should be maximized by the EA and a set of helpers. In terms of evolutionary computation, objectives are called *fitness functions*.

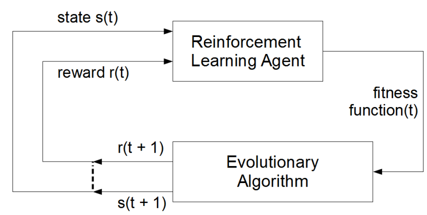

Fig. 1. Scheme of the proposed method

In our method the EA is considered as the RL *environment* [6]. The *action* is to choose the objective to be used at each generation of the EA. Either the target objective or a helper can be chosen. The fitness function is chosen each time when a next generation should be evolved. The reward is based on the difference of the target objective values in two sequential generations. The method is illustrated in Fig. 1

The method will be further referred to as *EA + RL*. To denote the particular applications of the method, we use a notation *A*+*L* where *A* is the name of the used EA and *L* is the particular reinforcement learning method. For example, the name of the genetic algorithm (*GA*) guided with *Q-learning* is *GA + Q-learning*.

III. H-IFF optimization problem

The proposed method was applied to the hierarchical if-and-only-if (H-IFF) function optimization problem [2]. H-IFF formula f is given below, where B is a bit string individual, B_L and B_R are its left and right halves respectively.

$$f(B) = \begin{cases} 1 & \text{if } |B| = 1, \\ |B| + f(B_L) + f(B_R) & \text{if } \forall i\{b_i = 0\} \| \forall i\{b_i = 1\}, \\ f(B_L) + f(B_R) & \text{otherwise.} \end{cases}$$

H-IFF optimization problem can be multi-objectivized in order to avoid getting stuck in local optima [2]. Multi-objectivized H-IFF optimization problem is called MH-IFF.

978-0-7695-4913-2/12 $26.00 © 2012 IEEE
DOI 10.1109/ICMLA.2012.159

TABLE I
H-IFF OPTIMIZATION RESULTS

Algorithm	Best fitness	Average fitness	σ	Successful runs
H-IFF problem				
(1+5) ES	216	179.07	16.99	0%
H-IFF problem with f_0 and f_1 helpers				
(1+5) ES + R-learning	448	448.00	0.00	100%
PESA-II	448	448.00	0.00	100%
H-IFF problem with f_0, f_1 and θ helpers				
(1+5) ES + R-learning	448	439.45	36.32	92%
PESA-II	312	277.83	20.07	0%

It can be efficiently solved with multi-objective evolutionary algorithms. Helpers corresponding to the MH-IFF problem are f_0 and f_1 (below).

$$f_k(B) = \begin{cases} 0 & \text{if } |B| = 1 \text{ and } b_1 \neq k, \\ 1 & \text{if } |B| = 1 \text{ and } b_1 = k, \\ |B| + f_k(B_L) + f_k(B_R) & \text{if } \forall i\{b_i = k\}, \\ f_k(B_L) + f_k(B_R) & \text{otherwise.} \end{cases}$$

In this research we also use an inefficient helper θ. It counts the number of overlaps with a bit mask of alternating ones and zeros: $1010\ldots10$. Optimizing such function destroys blocks of equally valued bits searched in the H-IFF problem.

IV. Experiment description

Three variations of the H-IFF problem were solved with different algorithms. Firstly, the original H-IFF without any helpers was optimized with (1 + 5) evolution strategy (ES). The corresponding mutation operator flipped one randomly chosen bit of each individual.

Secondly, two efficient helpers f_0, f_1 were added. The corresponding MH-IFF problem was solved with a multi-objective evolutionary algorithm PESA-II [7] and the proposed ES + R-Learning method that adjusted the same (1 + 5) evolution strategy. The parameters of the R-learning algorithm were $\alpha = 0.5$ and $\beta = 0.35$ [8]. The ε-greedy exploration strategy [6] with $\varepsilon = 0.25$ was used. All parameter values were chosen manually during the preliminary experiment.

Finally, the inefficient helper θ was added and the corresponding problem was solved with PESA-II and ES + R-learning again.

The length of an individual was 64 bits, so the optimal fitness was 448. 30 runs of each algorithm were performed. In each run 500000 fitness calculations were made. The statistics shown further were based on the best individuals from the last generation of each run.

A. Experiment results

The experiment results of optimizing H-IFF with different objectives are presented in Table I. The runs in which the ideal individual of fitness 448 was evolved are called *successful*. (1 + 5) ES, that used no helpers, appeared to be unsuccessful in all runs.

Applying the (1 + 5) ES + R-learning variation of the proposed method with f_0, f_1 helpers led to the increase of the ES efficiency and allowed to evolve an ideal individual in each run. PESA-II also appeared to be effective in this case.

In the last two rows of the Table I the inefficient helper objective θ is added to the objectives set. The proposed (1 + 5) ES + R-learning method is still more effective than the (1 + 5) ES, evolving an ideal individual in 92% of runs.

Notice that PESA-II is not effective anymore in the last part of the experiment. At the same time, the proposed EA + RL method was able to ignore the inefficient helper, because it learned that its application is profitless. So the proposed method can be more useful than multi-objectivization techniques when we have incomplete knowledge of helpers.

V. Conclusion

The previously proposed method, which performs adaptive selection of helper-objectives in single-objective optimization, is investigated. It is shown that the method manages to automatically ignore inefficient objectives and outperforms multi-objective evolutionary algorithms, which are commonly used in the helper objectives technique. The future work on the proposed method involves finding a heuristic for setting parameters of reinforcement learning, as setting the parameters during preliminary experimentation requires additional effort. It is also can be useful to modify the proposed method for dynamic helpers selection in multi-objective evolutionary algorithms.

VI. Acknowledgments

The research was supported by Ministry of Education and Science of Russian Federation in the context of Federal Program “Scientific and pedagogical personnel of innovative Russia”.

References

[1] M. T. Jensen, “Helper-objectives: Using multi-objective evolutionary algorithms for single-objective optimisation: Evolutionary computation combinatorial optimization,” *Journal of Mathematical Modelling and Algorithms*, vol. 3, no. 4, pp. 323–347, 2004.

[2] J. D. Knowles, R. A. Watson, and D. Corne, “Reducing local optima in single-objective problems by multi-objectivization,” in *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, ser. EMO ’01. London, UK: Springer-Verlag, 2001, pp. 269–283.

[3] D. F. Lochtefeld and F. W. Ciarallo, “Helper-objective optimization strategies for the job-shop scheduling problem,” *Appllied Soft Computing*, vol. 11, no. 6, pp. 4161–4174, 2011.

[4] A. Afanasyeva and M. Buzdalov, “Choosing best fitness function with reinforcement learning,” in *Proceedings of the Tenth International Conference on Machine Learning and Applications, ICMLA 2011*, vol. 2. Honolulu, HI, USA: IEEE Computer Society, 2011, pp. 354–357.

[5] A. Afanasyeva and M. Buzdalov, “Optimization with auxiliary criteria using evolutionary algorithms and reinforcement learning,” in *Proceedings of 18th International Conference on Soft Computing MENDEL 2012*, Brno, Czech Republic, 2012, pp. 58–63.

[6] A. Gosavi, “Reinforcement learning: A tutorial survey and recent advances,” *INFORMS Journal on Computing*, vol. 21, no. 2, pp. 178–192, 2009.

[7] D. W. Corne, N. R. Jerram, J. D. Knowles, M. J. Oates, and M. J., “Pesa-II: Region-based selection in evolutionary multiobjective optimization,” in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2001)*. Morgan Kaufmann Publishers, 2001, pp. 283–290.

[8] A. Schwartz, “A reinforcement learning method for maximizing undiscounted rewards,” in *Proceedings of the Tenth International Conference on Machine Learning*, 1993, pp. 298–305.