Adaptive Parameter Selection in Evolutionary Algorithms by Reinforcement Learning with Dynamic Discretization of Parameter Range

Arkady Rost
ITMO University
Saint Petersburg, Russia
arkrost@gmail.com

Irina Petrova
ITMO University
Saint Petersburg, Russia
petrova@rain.ifmo.ru

Arina Buzdalova
ITMO University
Saint Petersburg, Russia
abuzdalova@gmail.com

ABSTRACT

Online parameter controllers for evolutionary algorithms adjust values of parameters during the run. Recently, a new efficient parameter controller based on reinforcement learning was proposed by Karafotias et al. In this method parameter ranges are discretized into several intervals before the run. However, performing adaptive discretization during the run may increase efficiency of an evolutionary algorithm. Aleti et al. proposed another efficient controller with adaptive discretization.

In this paper we propose a parameter controller based on reinforcement learning with adaptive discretization. The proposed controller is compared with the existing parameter adjusting methods on different configurations of an evolutionary algorithm. Results show that the new controller outperforms the other controllers on most of the considered test problems.

Keywords

evolutionary algorithms; parameter control; Q-learning.

1. INTRODUCTION

Efficiency of evolutionary algorithm (EA) depends on the parameter choice. Values of the parameters can be set before a run. However, as optimal values of algorithm parameters may change over the course of the run, adaptive parameter adjustment is required.

We consider parameters with continuous values. When adjusting such parameters, parameter ranges are usually discretized into some intervals. Parameter ranges can be discretized a priori, in this case the chosen segmentation is kept during a run. Dynamic discretization may improve algorithm's performance [1]. Aleti et al. proposed entropy-based adaptive range parameter controller (EARPC) [1] which is one of the most efficient controllers with dynamic discretization [3].

Recently Karafotias et al. proposed another efficient parameter controller [2,3] based on reinforcement learning (RL) [4]. We call this method K-controller. Unfortunately, this method was not compared to EARPC. In the K-controller a priori discretization is used. We propose two new methods which combine RL and dynamic discretization. The first of them combines the K-controller and EARPC. The second one splits parameter ranges using RL and Kolmogorov-Smirnov criterion.

GECCO '16 July 20-24, 2016, Denver, CO, USA

ACM ISBN 978-1-4503-4323-7/16/07.

DOI: http://dx.doi.org/10.1145/2908961.2908998

2. PROPOSED METHODS

In the K-controller, dynamic state space segmentation is used [2]. However, the ranges of the parameters being adjusted are discretized a priori. We propose to improve the K-controller by using the EARPC method for dynamic discretization of parameter ranges. We call this method E+K.

Preliminary experiments showed that there was no significant improvement of the EA efficiency when many states were used. Thus the second proposed method does not use the dynamic state space segmentation, instead a single state of the RL environment is used. The parameter range is discretized using Kolmogorov-Smirnov criterion and it is rediscretized if the expected rewards are close to each other for all actions of the agent. We call this method KS+RL. The detailed description of the proposed methods is available at arXiv[1].

3. EXPERIMENTS AND RESULTS

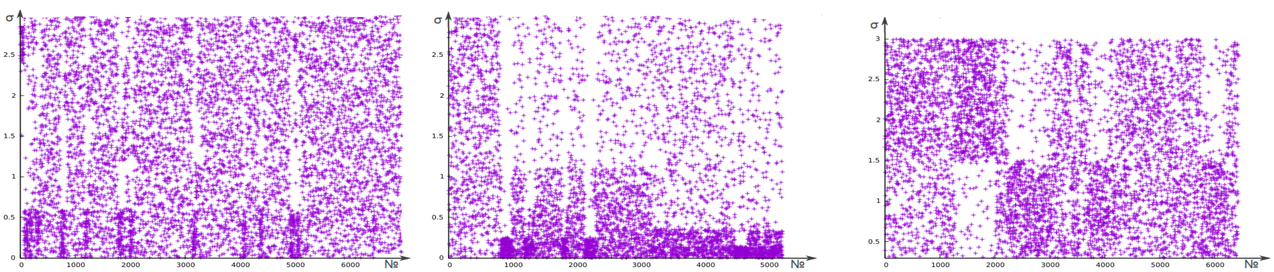
We use $(\mu + \lambda)$ evolution strategy. Mutation strength σ is the adjustable parameter. We expect that σ should become smaller as the global optimum is approached. The range of σ is $[0, k]$, where k is a constant. As k grows, it becomes harder to find the optimal value of σ.

The average number of generations needed to reach the optimum using different parameter controllers is presented in Table 1. The first three columns contain values of EA parameters k, μ and λ. The next 20 columns contain results of optimizing four functions with different landscapes: Sphere, Rastrigin, Levi and Rosenbrock. For each function, we present the results of the following parameter controllers: the proposed method KS+RL (K+R), the Q-learning algorithm (Q), the K-controller (K), the EARPC algorithm (E) and the proposed method E+K (E+K). The last row contains the total number of the EA configurations on which the corresponding algorithm outperformed the other algorithms. The gray background highlights the best result for

[1] http://arxiv.org/abs/1603.06788

Table 1: Averaged number of runs needed to reach the optimum using the proposed method KS+RL (K+R), the Q-learning algorithm (Q), the K-controller (K), the EARPC algorithm (E) and the proposed method E+K (E+K)

			Sphere function					Rastrigin function					Levi function					Rosenbrock function				
k	μ	λ	K+R	Q	K	E	E+K	K+R	Q	K	E	E+K	K+R	Q	K	E	E+K	K+R	Q	K	E	E+K
1	1	1	2434	8769	8048	5258	4830	3631	9653	8385	7794	8689	3496	7200	7265	7986	14092	5124	15058	13418	9003	12905
1	1	3	2207	4221	2683	3942	3070	1776	2226	2069	3148	3610	1980	3305	2903	2789	3688	2301	3914	4167	3553	2701
1	1	7	878	1085	2620	1653	2247	1226	1605	1757	1422	1422	1321	1584	1600	1923	3820	1411	1791	2330	2296	1941
1	5	1	1450	1664	2076	4472	3893	1666	1706	2281	4341	4530	1778	1898	1865	2372	2246	1859	2311	2809	5730	4453
1	5	3	569	706	824	1589	845	918	894	942	1958	2197	855	862	794	1632	2171	1053	869	748	1393	2749
1	10	1	703	959	747	1438	728	1008	1103	1105	2681	2926	884	840	804	2353	1451	1617	1497	1593	2213	3085
1	10	3	331	378	358	534	400	622	604	665	1460	1485	533	502	624	1491	1134	683	488	616	1161	1225
2	1	1	4342	25192	28523	31738	16182	4744	23663	21043	27865	19142	4947	13420	14789	25721	30653	5165	23371	27239	20005	20819
2	1	3	2333	7681	6478	5152	4233	1839	6405	6825	9748	8997	2020	6884	6601	7477	4612	3461	7295	7169	7166	13342
2	1	7	1464	3360	3739	1688	2956	1160	2388	2183	3806	4085	1205	3521	2370	2533	2967	1753	3488	2968	5874	7870
2	5	1	1891	3814	3468	4908	5173	2467	3944	4029	5961	5750	1935	3517	3369	7222	5365	1990	8076	5810	11153	11856
2	5	3	974	994	1163	1631	946	1128	1614	1780	2293	1720	1085	1231	1326	3039	2943	1396	2116	2705	3775	4542
2	10	1	1164	2397	1833	778	876	1722	2108	2255	2411	2807	1803	1884	2197	5139	3072	2022	3415	2787	8768	4603
2	10	3	459	445	465	719	788	988	1420	1116	1486	1234	887	988	1006	1045	2381	1181	1037	1106	3719	1648
3	1	1	8055	36698	29112	54868	30710	5159	29082	23305	24249	27327	5216	21470	21953	28900	35119	6535	28702	36597	32533	43832
3	1	3	2826	7845	9115	12446	11985	2821	7977	10726	6541	16040	2900	9029	7071	6966	10883	3391	11427	9873	13061	10068
3	1	7	1427	3907	4813	10118	9207	1517	4680	4266	4144	5408	1700	4139	4019	3375	2062	2573	5820	6462	9797	9775
3	5	1	2447	8328	5886	3348	12313	2704	6208	5419	5953	7665	3105	6966	6742	10480	9059	3148	12438	6791	8952	11581
3	5	3	1445	2790	2222	1379	1848	1544	2514	2096	2823	2304	1808	2467	2664	4593	3714	1721	2289	3673	4434	6799
3	10	1	1996	2398	2531	3173	3745	2048	3747	3258	5095	3873	2071	2901	2943	5210	4814	2352	4473	6531	8320	10539
3	10	3	1053	1074	1206	1114	171	1296	1740	1523	1013	1028	1330	1462	1832	3140	2389	1677	2000	1650	2611	2479
Summary			17	1	0	2	1	18	2	0	1	0	18	1	2	0	0	16	3	2	0	0



(a) (b) (c)

Figure 1: Selected values of σ by the K-controller (a) and proposed methods: KS+RL (b) and E+K (c) on Rastrigin function.

each EA configuration. The KS+RL method outperformed the other considered methods on most problem instances.

Fig. 1 shows values of σ selected by the two proposed methods and the K-controller during optimization of Rastrigin function. The horizontal axis refers to the current iteration count, the vertical axis refers to the selected value of σ. The selected σ convergences to the optimal value in the proposed method KS+RL (Fig. 1b). The other methods do not seem to show such performance.

4. CONCLUSION

We proposed two parameter controllers which discretize parameter range dynamically. One of the proposed methods is based on two existing parameter controllers: EARPC and the K-controller. In the second approach the parameter range is discretized using Kolmogorov-Smirnov criterion and it is re-discretized if the expected rewards are close to each other for all actions of the agent.

The proposed methods were compared with EARPC, the K-controller and the Q-learning algorithm. We tested controllers with 18 EA configurations on four test problems. On most problem instances, the second proposed approach outperformed the other methods. This method improves the parameter value during the whole optimization process contrary to the other methods.

This work was partially financially supported by the Government of Russian Federation, Grant 074-U01.

5. REFERENCES

[1] A. Aleti and I. Moser. Entropy-based adaptive range parameter control for evolutionary algorithms. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 1501–1508, 2013.

[2] G. Karafotias, Á. E. Eiben, and M. Hoogendoorn. Generic parameter control with reinforcement learning. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 1319–1326, 2014.

[3] G. Karafotias, M. Hoogendoorn, and A. Eiben. Parameter control in evolutionary algorithms: Trends and challenges. *Evolutionary Computation, IEEE Transactions on*, PP(99):1–1, 2014.

[4] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, USA, 1998.