Niching an Archive-based Gaussian Estimation of Distribution Algorithm via Adaptive Clustering

Yongsheng Liang, Zhigang Ren
Autocontrol Institute, Xi'an Jiaotong University
Xi'an, Shaanxi, 710049, P.R. China
liangyongsheng@stu.xjtu.edu.cn, renzg@mail.xjtu.edu.cn

Bei Pang, An Chen
Autocontrol Institute, Xi'an Jiaotong University
Xi'an, Shaanxi, 710049, P.R. China
beibei@stu.xjtu.edu.cn, chenan123@stu.xjtu.edu.cn

ABSTRACT

Traditional Gaussian estimation of distribution algorithm (EDA) may suffer from premature convergence and has a high risk of falling into local optimum when dealing with multimodal problem. In this paper, we first attempt to improve the performance of EDA by utilizing historical solutions and develop a novel archive-based EDA variant. The use of historical solutions not only enhances the search efficiency of EDA to a large extent, but also significantly reduces the population size so that a faster convergence could be achieved. Then, the archive-based EDA is further integrated with an novel adaptive clustering strategy for solving multimodal optimization problems. Taking the advantage of the clustering strategy in locating different promising areas and the powerful exploitation ability of the archive-based EDA, the resultant algorithm is endowed with strong capability in finding multiple optima. To verify the efficiency of the proposed algorithm, we tested it on a set of niching benchmark problems, the experimental results indicate that the proposed algorithm is competitive.

CCS CONCEPTS

• **Mathematics of computing** → *Evolutionary algorithms;*

KEYWORDS

estimation of distribution algorithm, archive, clustering, multimodal optimization

1 INTRODUCTION

Estimation of distribution algorithm (EDA) [1] is a special branch of evolutionary algorithm (EA). Since it came into being, EDA has attracted increasing research effort and achieved great success in diverse domains. Continuous EDA usually adopts Gaussian model as the basic probability distribution model. However, traditional Gaussian EDA (GEDA) may suffer from premature convergence [2] and has difficulty in dealing with multimodal problems [3]. To solve that, this paper first proposes an novel archive-based EDA variant named EDA^2, then EDA^2 is further incorporated into an adaptive clustering strategy. The resultant algorithm, referred to as C-EDA^2, shows appealing performance in dealing with multimodal problems.

2 DESCRIPTION OF C-EDA^2

2.1 Archive-based EDA: EDA^2

Continuous EDAs generally employ Gaussian model as the basic probability distribution model. The Gaussian probability density function for an n-dimensional random vector $\boldsymbol{x}$ can be parameterized by its mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{C}$. $\boldsymbol{\mu}$ and $\boldsymbol{C}$ for the next generation are generally estimated according to the maximum likelihood estimation (MLE) method [2].

Instead of only utilizing solutions in current generation as traditional EDA, EDA^2 maintains an archive to store some high-quality historical solutions. For each generation, the archive $\boldsymbol{A}^t$ is defined as follows:

$$\boldsymbol{A}^t = \boldsymbol{S}^{t-1} \cup \boldsymbol{S}^{t-2} \cup ... \cup \boldsymbol{S}^{t-l} , \tag{1}$$

where $\boldsymbol{S}^{t-i}$ denotes the set of solutions selected at the $(t-i)$ th generation, l is a nonnegative integer and denotes the length of the archive. Once $\boldsymbol{A}^t$ is determined, EDA^2 estimates its covariance matrix as follows:

$$\overline{\boldsymbol{C}}^{t+1} = \frac{1}{|\boldsymbol{H}^t|}\sum_{i=1}^{|\boldsymbol{H}^t|}(\boldsymbol{H}_i^t - \overline{\boldsymbol{\mu}}^{t+1})(\boldsymbol{H}_i^t - \overline{\boldsymbol{\mu}}^{t+1})^{\mathrm{T}},\ \boldsymbol{H}^t = \boldsymbol{A}^t \cup \boldsymbol{S}^t , \tag{2}$$

where the new mean $\overline{\boldsymbol{\mu}}^{t+1}$ is still estimated using only $\boldsymbol{S}^t$ as in the traditional EDA.

The new estimation method naturally integrates the evolution direction information into the estimated covariance matrix, which endows EDA^2 with better search direction and greater search scope. The use of historical solutions also reduces the population size of EDA^2 as solutions selected from $(l+1)$ generations are used to estimate the covariance matrix. Since the performance of EDA is enhanced by exploiting Evolution Direction information hidden in the Archive, we named this algorithm EDA^2. Algorithm 1 presents the detailed steps of EDA^2.

2.2 Adaptive Clustering Strategy

Decision space and target space (DS-TS) information based clustering [4] is adopted. Considering the characteristics of optimization problem, its idea consists in that cluster centers are solutions with better fitness value and farther relative distance. Procedures of DS-TS clustering can be found in[4].

GECCO'18 Companion, July 15-19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07.
https://doi.org/10.1145/3205651.3205798

Algorithm 1: Procedure of EDA²

1. Initialize parameters, including population size p, selection ratio τ, and archive length l;
2. Set $t = 0$, $i = 0$, and $\boldsymbol{A}^t = \emptyset$, randomly and uniformly generate the initial population $\boldsymbol{P}^t$;
3. Evaluate $\boldsymbol{P}^t$ and update the best solution $\boldsymbol{b}^t$ obtained so far;
4. Output $\boldsymbol{b}^t$ if the stopping criterion is met;
5. Select the best $\lfloor \tau p \rfloor$ solutions from $\boldsymbol{P}^t$ and store them into $\boldsymbol{S}^t$;
6. Estimate the mean $\overline{\boldsymbol{\mu}}^{t+1}$ with $\boldsymbol{S}^t$ using MLE;
7. Estimate covariance matrix $\overline{\boldsymbol{C}}^{t+1}$ with $\boldsymbol{S}^t$ and $\boldsymbol{A}^t$ according to (2);
8. **If** $i < l$ **then**
9. set $\boldsymbol{A}^{t+1} = \boldsymbol{A}^t \cup \boldsymbol{S}^t$ and set $i \leftarrow i + 1$;
10. **Else**
11. $\boldsymbol{A}^{t+1} = \boldsymbol{A}^t \cup \boldsymbol{S}^t \setminus \boldsymbol{S}^{t-l}$;
12. Set $t \leftarrow t + 1$, build a probability model G^t based on $\overline{\boldsymbol{\mu}}^t$ and $\overline{\boldsymbol{C}}^t$;
13. Generate $p - 1$ new solutions by sampling from G^t and store them into $\boldsymbol{M}^t$;
14. Set $\boldsymbol{P}^t = \boldsymbol{M}^t \cup \boldsymbol{b}^{t-1}$ and goto step 3.

2.3 Procedure of C-EDA²

The design of C-EDA² is to cluster a group of selected solutions into clusters by DS-TS clustering, then EDA² is utilized to evolve these clusters independently and find their optima. Algorithm 2 presents the procedure of C-EDA². In step 6, EDA² would be stopped if one of the following termination criteria is met: 1) the improvement of the median of population values is smaller than the defined accuracy level in the last 5 generations; 2) the maximum number of function evaluations (MaxFEs) is reached.

Algorithm 2: Procedure of C-EDA²

1. Set the initial solution number N, selection ratio τ, initialize EDA² and DS-TS clustering, set *Output*= Ø;
2. **while** the MaxFEs is not reached **do**
3. Randomly and uniformly initialize N solutions and select the best $\lfloor \tau N \rfloor$ solutions of them;
4. Use DS-TS clustering strategy to divide the selected solutions into different clusters;
5. **for** k = 1 to *the number of clusters* **do**
6. Use EDA² to evolve the solutions in the k th cluster until the termination criteria is met;
7. Store the best solution obtained into *Output*;
8. **end**
9. **end**
10. Output the solutions in *Output*.

3 EXPERIMENTS AND CONCLUSIONS

C-EDA² is tested on the CEC'2013 multimodal function set [5] and compared with three efficient niching algorithms, including LMCEDA [3], LMSEDA [3] and RS-CMSA [6]. The peak ratio (PR) is used to evaluate their performance.

The initial solution number N in C-EDA² is set as N=1000+10D^2, where D is the problem dimension. The selection ratio τ is set to 0.35, and the threshold factor α of DS-TS clustering is set to 0.8. As for EDA², its population size p and archive length l are set as $p = 4(D+1)$ and $l = 5$.

Table 1 summarizes the average PRs obtained by the four algorithms at accuracy level $\varepsilon = 10^{-4}$. From Table 1, we can observe that C-EDA² achieves better performance than LMCEDA and LMSEDA on most functions. Concretely, C-EDA² obtains better, same or worse results than LMCEDA on 8, 10 and 2 functions, and the corresponding numbers for LMSEDA are 8, 9 and 3. So C-EDA² has an edge over the two algorithms. Compared to RS-CMSA, C-EDA² obtains the same results with it on 8 functions, but is surpassed by it on 11 functions.

In summary, C-EDA² indicates better performance than LMCEDA and LMSEDA, which is achieved with much simple algorithmic framework and less parameters. But C-EDA² also has some shortages, there is still room to improve its performance. The major disadvantage of C-EDA² lies in the restart mechanism. Independent restarts may improve its performance to some extent, it is still very likely to revisit previously explored regions. While in RS-CMSA, taboo method is adopted to reduce the chance of revisiting in restarts, which makes RS-CMSA more efficient. Similar idea could also be introduced to C-EDA² to further enhance its performance.

Table 1: Average PRs obtained by 4 algorithms.

Fun.	LMCEDA	LMSEDA	RS-CMSA	C-EDA²
F_1	1.000	1.000	1.000	1.000
F_2	1.000	1.000	1.000	1.000
F_3	1.000	1.000	1.000	1.000
F_4	1.000	1.000	1.000	1.000
F_5	1.000	1.000	1.000	1.000
F_6	0.990	0.972	0.999	1.000
F_7	0.734	0.673	0.997	0.711
F_8	0.347	0.613	0.871	0.839
F_9	0.284	0.248	0.730	0.300
F_{10}	1.000	0.998	1.000	1.000
F_{11}	0.667	0.892	0.997	0.667
F_{12}	0.750	0.990	0.948	0.663
F_{13}	0.667	0.667	0.997	0.667
F_{14}	0.667	0.667	0.810	0.667
F_{15}	0.696	0.738	0.748	0.740
F_{16}	0.667	0.670	0.667	0.667
F_{17}	0.456	0.620	0.703	0.660
F_{18}	0.657	0.660	0.667	0.667
F_{19}	0.451	0.458	0.503	0.500
F_{20}	0.059	0.248	0.483	0.248

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 61105126, in part by the Postdoctoral Science Foundation of China under Grants 2014M560784 and 2016T90922, and in part by the Postdoctoral Science Foundation of Shaanxi Province.

REFERENCES

[1] P. Larrañaga and J. A. Lozano (Eds.). 2002. *Estimation of Distribution Algorithms : A New Tool for Evolutionary Computation.* Springer Science & Business Media, vol. 2.

[2] P.A.N. Bosman, J. Grahl, and D. Thierens. 2008. Enhancing the Performance of Maximum-Likelihood Gaussian EDAs Using Anticipated Mean Shift. In *Proceedings of the 10th international conference on Parallel Problem Solving from Nature* (PPSN X). Springer-Verlag, Berlin, Heidelberg, 133-143.

[3] Q. Yang, W.-N. Chen, Y. Li, C.L.P. Chen, X.-M. Xu and J. Zhang. 2017. Multimodal Estimation of Distribution Algorithms. *IEEE Transactions on Cybernetics*, 47(3), 636-650.

[4] S. Huang, Z. Ren, M. Wang and C. Sun. 2016. A Two Stages Invasive Weed Optimization via a New Clustering Strategy. In *Proceedings of the Genetic and Evolutionary Computation Conference* (GECCO'16). ACM, Denver, 1001-1005.

[5] X. Li, A. Engelbrecht, and M.G. Epitropakis. *2013. Benchmark Functions for CEC'2013 Special Session and Competition on Niching Methods for Multimodal Function Optimization.* Technical Report. Evolutionary Computation and Machine Learning Group, RMIT University, Australia. 1–10 pages.

[6] A. Ahrari, K. Deb and M. Preuss. 2017. Multimodal Optimization by Covariance Matrix Self-Adaptation Evolution Strategy with Repelling Subpopulations. *Evolutionary Computation*, 25(3), 439-471.