Swarm and Evolutionary Computation xxx (2018) 1–10

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

journal homepage: www.elsevier.com/locate/swevo

Information fusion in offspring generation: A case study in DE and EDA

Hui Fang [a], Aimin Zhou [a,*], Hu Zhang [b,**]

[a] Shanghai Key Laboratory of Multidimensional Information Processing, Department of Computer Science and Technology, East China Normal University, Shanghai 200062, China

[b] Beijing Electro-mechanical Engineering Institute, Beijing 100074, China

ARTICLE INFO

Keywords:
Differential evolution
Estimation of distribution algorithm
Hybrid algorithm
Information fusion

ABSTRACT

Both differential evolution (DE) and estimation of distribution algorithm (EDA) are popular and effective evolutionary algorithms (EAs) in solving global optimization problems. The two algorithms utilize different kinds of information for generating offspring solutions. In the former, the mutation and crossover operators use the individual information to create trial solutions, while in the later, a probabilistic model is built for sampling new trial solutions, which extracts the population distribution information. It is therefore natural to make use of both kinds of information for generating solutions. In this paper, we propose an algorithm that hybridizes DE and EDA, named as DE/GM, which utilizes both DE crossover/mutation operators and a Gaussian probabilistic model based operator for offspring generation. The basic idea is to generate some of trial solutions by the EDA operator, and to generate the rest by the DE operator. To validate the performance of DE/GM, a test suite of 13 benchmark functions is employed, and the experimental results suggest that DE/GM is promising.

1. Introduction

In this paper, we consider the following *box-constrained continuous optimization problem*.

$$\begin{cases} \min \ f(x) \\ \text{s.t.} \quad x \in \Omega \end{cases} \tag{1}$$

where $x = (x^1, x^2, \ldots, x^d)^T$ is a decision variable vector, $\Omega = [a^i, b^i]^d$ is the feasible region of the decision space, $a^i < b^i$, $a^i \in R$ and $b^i \in R$ are the lower and upper boundaries of the ith dimension in the decision space, respectively. $f(x) : \Omega \rightarrow R$ is a continuous mapping from Ω to the objective space R.

There exists a variety of methods to deal with the global optimization problems. Among them, the *evolutionary algorithms (EAs)* [1] have been attracting much attention partly due to their weak assumptions and global search ability. Different EAs have been proposed, such as *genetic algorithm (GA)* [2], *differential evolution(DE)* [3–5], *particle swarm optimization (PSO)* [6], and *estimation of distribution algorithm (EDA)* [7,8]. DE is a popular EA that creates new offspring solutions by combining several parent solutions through crossover and mutation operators [9,10]. It exhibits remarkable performance in diverse fields of science and engineering, such as cluster analysis [11], robot control [12], controller design [13], and graph theory [14]. However, in practice it has been shown that DE is sensitive to the mutation strategies [15–17] and the control parameters, i.e., the population size, the scaling factor F, and the crossover rate CR [18,19]. EDA is another popular EA paradigm that has a different mechanism from DE for offspring generation. Instead of combining several parent solutions directly, EDAs explicitly extract the distribution information of a set of parent solutions by probabilistic models and sample new solutions from the models [8,20,21]. Compared with traditional EAs, EDAs have their own advantages for dealing with hard problems when there are linkages or dependencies between decision variables [22]. However, EDAs have also been criticized for their high computational cost and low efficiency due to inadequate probabilistic models [21].

Information fusion is a natural way to improve algorithm performance for handling hard optimization problems. In the case of offspring generation, the information fusion strategies can be roughly classified into three levels, i.e., the population, individual and chromosome levels. For the population level, it means that in some generations, the population is generated by one operator while in the other generations, the population is generated by other operators. In Ref. [23], the crossover/mutation operators and a model based operator are called alternatively in different generations to generate trial solutions. For the

* Corresponding author.
** Corresponding author.
E-mail addresses: amzhou@cs.ecnu.edu.cn (A. Zhou), jxzhanghu@126.com (H. Zhang).

https://doi.org/10.1016/j.swevo.2018.02.014
Received 29 January 2017; Received in revised form 29 November 2017; Accepted 23 February 2018
Available online XXX
2210-6502/

Please cite this article in press as: H. Fang, et al., Information fusion in offspring generation: A case study in DE and EDA, Swarm and Evolutionary Computation (2018), https://doi.org/10.1016/j.swevo.2018.02.014

individual level, it means that some of the solutions in a population are generated by one operator and the others are generated by other operators. Most of hybrid EAs fall into this category. Some hybridize different operators [24,25], and some hybridize global search operators with local search operators [26]. For the chromosome level, it means that some of the elements of a chromosome are generated by one operator and the others by other operators. In Ref. [27], a hybrid algorithm, called DE/EDA, is proposed and some of the elements of the decision vector are from DE and the rest are from EDA. Following this idea in Ref. [28], some of the elements are from EDA and the others are from local search.

Algorithm 1 DE/GM Framework.

```
   // initialize population
1  Initialize the population $pop = \{x_1, \cdots, x_N\}$ and
   evaluate them.
   // terminate condition
2  while $fe < FE$ do
      // sort the population
3     Sort $pop$ by an ascending order of the objective
      values.
      // local search to $x_1$ by mean shift
4     $y' \leftarrow MeanShift(pop)$.
      // population partition
5     Partition $pop$ into $K$ classes$\{C_1, C_2, ..., C_K\}$.
6     foreach $k \in \{1, \cdots, K\}$ do
         // EDA model building and sampling
7        $y'' \leftarrow GaussianModel(C_k)$;
         // chromosome level fusion
8        Set $y_k^j = \begin{cases} y'^j & if\ rand(0,1) < P_c \\ y''^j & \text{otherwise} \end{cases}$, for
         $j = 1 \cdots d$.
         // offspring repair
9        Repair $y_k$.
         // environmental selection
10       Set $x_{N-k+1} = y_k$ if $f(y_k) < f(x_{N-k})$.
11    end
12    Set $pop' = \{x_1, \cdots, x_{N-K}\}$.
13    foreach $i \in \{1, \cdots, N\text{-}K\}$ do
         // DE based offspring generation
14       $y_i \leftarrow DE(x_i, pop')$.
         // offspring repair
15       Repair $y_i$.
         // environmental selection
16       Set $x_i = y_i$ if $f(y_i) < f(x_i)$.
17    end
18 end
```

It is clear that DE and EDA represent two different offspring generation mechanisms. The former uses the individual location information while the latter uses the global population distribution information. To use more information in offspring generation, it is natural to combine the two search strategies as it has been done in DE/EDA [27] which falls into the chromosome level category. Following the idea of DE/EDA and our previous work [29], which uses the mean-shift method as a local search to improve the performance of DE, we propose a new algorithm in this paper. The proposed approach, called DE/GM, utilizes both the DE operator and a Gaussian probabilistic model based EDA operator for generating offspring solutions. In each generation, some of trial solutions are generated by the DE operator and the rest are generated by the EDA operator. The major differences between DE/GM and DE/EDA are as follows:

- DE/GM is based on both the individual information fusion and the chromosome information fusion while DE/EDA is only based on the chromosome information fusion.
- In DE/GM, the EDA operator is based on the Gaussian probabilistic model and a mean-shift based local search while in DE/EDA, the EDA operator is based on a univariate marginal distribution model.

The rest of this paper is organized as follows. In Section 2, the DE/GM algorithm framework is introduced, and the details of the EDA operator are presented. The experimental results and analysis are given in Section 3. Finally, this paper is concluded in Section 4.

2. Proposed algorithm

2.1. DE/GM framework

As mentioned above, the basic idea of our approach is hybridizing DE and EDA for offspring generation in both the individual level and the chromosome level. In each generation, the population is sorted and the DE operator is applied to the best individuals. The EDA part works as follows: firstly, the population is partitioned into several clusters, then for each cluster a multivariate Gaussian probabilistic model is built and a trial solution is sampled, after that the trial solutions are combined with a solution, which is an improved one of the best solution by a mean-shift based local search [29], to form offspring solutions, finally the offspring solutions will replace the worst solutions in the current population.

In each generation, DE/GM maintains

- a set of N solutions $\{x_1, x_2, \ldots, x_N\}$,
- their objective values $\{f(x_1), f(x_2), \ldots, f(x_N)\}$.

The algorithm framework of the proposed DE/GM is shown in Algorithm 1. We would like to make some comments on the algorithm as follows.

- *Population Initialization:* In *Line 1*, the initial population is uniformly and randomly sampled from the feasible region Ω.
- *Termination Condition:* In *Line 2*, the algorithm stops when the number of function evaluations fe reaches the preset maximum number FE.
- *Offspring Generation:* A hybrid strategy is applied to generate offspring solutions. The EDA operator is used in *Lines 3–11*: first, the best solution, i.e., x_1 in the sorted population, is improved by the mean-shift based local search to obtain a candidate solution y', then for each cluster a Gaussian probabilistic model is built and a candidate solution y'' is sampled from the model, finally the elements of the two candidate solutions are combined to form an offspring solution y^k where the ratio is controlled by the parameter P_c. The DE operator is used for the best $N - K$ solutions in *Line 14.*
- *Environmental Selection:* The selection is based on the objective values. In *Line 10*, the solutions generated by the EDA operator try to replace the worst solutions in each generation. In *Line 16*, the solutions generated by the DE operator try to replace the corresponding parent solutions.
- *Offspring Repair:* The newly generated offspring solution y might be infeasible, and it is repaired in *Lines 9* and *15* as follows.

$$y_j = \begin{cases} rand(a_j, x_j) & if\ \ y_j < a_j \\ rand(x_j, b_j) & if\ \ y_j > b_j \\ y_j & \text{otherwise} \end{cases} \tag{2}$$

where $j = 1, \ldots, n$, $rand(s, t)$ returns a random number from $[s, t]$, and x is the corresponding parent of y.

It should be noted that the offspring combination strategy in *Line 8* is the same as in Refs. [27,30], which is a chromosome level information fusion strategy. The major part uses an individual level information fusion strategy. Some details of DE/MG will be discussed shortly in the following sections.