A Conical Area Differential Evolution with Dual Populations for Constrained Optimization

Bin Wu[1], Weiqin Ying[1](✉), Yu Wu[2], Yuehong Xie[1], and Zhenyu Wang[1]

[1] School of Software Engineering, South China University of Technology, Guangzhou 510006, China
yingweiqin@scut.edu.cn
[2] School of Computer Science and Educational Software, Guangzhou University, Guangzhou 510006, China

Abstract. During the last decade, multiobjective approaches to deal with constraints in evolutionary algorithms have drawn more and more attention from researchers. In this paper, an efficient conical area differential evolution algorithm (CADE) with dual populations is proposed for constrained optimization by borrowing the ideas of cone decomposition for multiobjective optimization. In CADE, a feasible subpopulation and a conical one are designed not only to heighten the diversity of population, but also to search the global feasible optimum, respectively, along the feasible segment and the Pareto front in a cooperative way. The feasible subpopulation is ranking according to tolerance-based sorting, while the conical subpopulation aims to construct and utilize the Pareto front by the conical area indicator and a biased cone decomposition strategy in geometric proportion. Afterwards, neighbors in both subpopulations are fully exploited to help each other. The performance of CADE is assessed on 13 benchmark test functions and the result reveals that CADE is capable of producing the significantly competitive solutions for constraint optimization problems compared with the other popular approaches.

Keywords: constrained optimization, differential evolution, multiobjective optimization, cone decomposition, dual populations

1 Introduction

In real world applications, most optimization problems are subject to different types of constraints which are known as constrained optimization problems (COPs) [1, 2]. In general, a COP can be formulated as (1).

$$\begin{aligned} &\mathit{minimize}\ f(\mathbf{x}) \\ &\mathit{subject\ to}\ g_i(\mathbf{x}) \le 0, i = 1, 2, ..., q \\ &\qquad\qquad h_j(\mathbf{x}) = 0, j = q + 1, ...m \\ &\qquad\qquad \mathbf{x} = \{x_1, x_2, \cdots, x_n\} \in \Omega \end{aligned} \tag{1}$$

where Ω is the decision space defined by the parametric constraints. $g_i(\mathbf{x})$ is the i-th inequality constraint while the $h_j(\mathbf{x})$ is the j-th equality constraint. when a solution satisfies all constraints, it is regard as a feasible one.

In the past decades, exhibiting significant performance, evolutionary algorithms (EAs) have been widely used in solving COPs. As unconstrained optimization technique, EAs require additional mechanisms to resolve constraints and a large number of constraint-handling techniques for EAs have been developed.

As the most common constraint handling techniques, penalty function approaches punish an infeasible solution according to constraint violation so that it has smaller chance to survive into the next generation. Based on the preference of feasible solutions over infeasible solutions, Deb [3] proposed a feasibility rule which is the most common method to compare individuals: (1) if one solution is infeasible and the other one is feasible, the feasible solution is preferred; (2) between two feasible solutions, the one having a better objective function value is preferred; (3) between two infeasible solutions, the one having a smaller degree of constrain violation is better. Moreover, Stochastic Ranking (SR) is the approach proposed by Runarsson and Yao which combines the penalty function and the feasibility rule to solve COPs [4]. And on the basis of SR, lots of new approaches have been proposed such as SRDE [5], ASR [6] and so on. But for some complex problems, it is difficult for these methods to find optimal solution.

In addition, more and more researchers have used multiobjective evolutionary algorithms (MOEAs) when solving COPs. Proposed by Wang, CMODE [7] is the approach of great performance which combines multiobjective optimization with differential evolution to deal with constrained optimization problems. But CMODE has not constructed PF systematically to assist population in searching.

Recently, more and more MOEAs have been proposed to solve multiobjective optimization problems (MOPs) by decomposing a MOP into a number of scalar objective optimization subproblems and the most common MOPs is the bi-objective optimization problems (BOPs). CAEA, proposed by Ying [8], not only decomposes the BOP into N scalar subproblems, but also assigns each subproblem an exclusive decision subset Ω_i, which has shown more significant performance and higher efficiency compared with the other popular decomposition approaches.

In this paper, we proposed a conical area differential evolution algorithm with a dual-population scheme for solving COPs to improve the competitiveness of solutions. At first, CADE adopts a dual-population scheme where the feasible subpopulation and conical subpopulation are designed to search the optimal feasible solution, respectively, along the feasible segment and the Pareto front. In order to solve COPs, CADE parts the decision space with a biased cone decomposition strategy different from CAEA, which implies that it needs to maintain more individuals near the feasible region. In addition, CADE can also get a better diversity of population for COPs.

2 Dual-Population Scheme

In many multiobjective optimization techniques to deal with constraints, the degree of constraint violation of an individual $\mathbf{x}$ on the i-th constraint is defined as

$$G_i(\mathbf{x}) = \begin{cases} max\{0, g_i(\mathbf{x})\}, & 1 \leq i \leq q \\ max\{0, |h_i(\mathbf{x})| - \delta\}, & q+1 \leq i \leq m \end{cases} \tag{2}$$

where δ is the positive tolerance value for equality constraints. Thus, $G(X) = \sum_{j=1}^{m} G_i(\mathbf{x})$ reflects the degree of constraint violation of the individual $\mathbf{x}$.

To solve COPs by multiobjective optimization techniques, a COP is in general converted into a biobjective optimization problem in which two objectives are considered: the first objective f_1 is to minimize the degree of constraint violation $G(\mathbf{x})$, while the second one f_2 is to optimize the original objective function $f(\mathbf{x})$. Fig. 1 demonstrates the BOP converted from a COP. As shown in Fig. 1, all feasible solutions are on the feasible segment while the non-dominated solutions are located on the Pareto front (PF). In particular, the intersection between the feasible segment and the Pareto front is just the desired global feasible optimum.

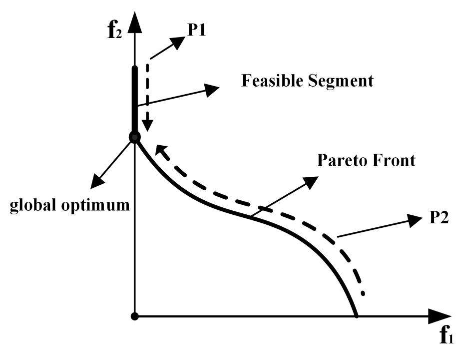

Fig. 1. Dual-population Scheme for a BOP converted from a COP.

The advantage of multiobjective approaches such as CMODE is that it can take advantage of the non-dominated solutions to guide the population to search the global optimum from infeasible regions to feasible regions. And how to utilize the non-dominated solutions has become the key of multiobjective approaches in the search for global optimum. Based on multiobjective technique, CMODE is able to create competitive results to solve COPs, but it mainly keeps no-dominated solutions and doesn't construct PF systematically. In the proposed CADE, a bias cone decomposition strategy is employed to construct PF in a systematic way and a conical area indicator is used to find a local nondominated solution in its associated decision subset. Fig. 1 also illustrates the dual-population scheme in CADE. This scheme consists of two subpopulations: the feasible one and the conical one, written as P1 and P2, which are designed to search the global feasible optimum, respectively, along the feasible segment and the Pareto front, as shown in Fig. 1.

2.1 Conical Subpopulation and Biased Cone Decomposition

To partition the decision space Ω, an utopia point and a nadir point are first required in the objective space. Given a set of solutions $A \in \Omega$, the utopia point and the nadir point can be formulated as follows:

$$\begin{aligned} \mathbf{F}^{min}(A) &= (f_1^{min}, f_2^{min}), f_i^{min} = \min\nolimits_{\mathbf{x}\in A}(f_i(\mathbf{x})) \\ \mathbf{F}^{max}(A) &= (f_1^{max}, f_2^{max}), f_i^{max} = \max\nolimits_{\mathbf{x}\in A}(f_i(\mathbf{x})) \end{aligned} \tag{3}$$

For convenience, we transform an objective vector $\mathbf{y}$ by $\overline{\mathbf{y}} = \mathbf{y} - \mathbf{F}^{min}(A)$. After the transformation, the utopia point is at the origin $(0,0)$.

Definition 1 (Observation vector). *The observation vector for any transformed point* $\overline{\mathbf{y}} = (\overline{y_1}, \overline{y_2})$ *is* $\mathbf{V}(\overline{\mathbf{y}}) = (v_1, v_2)$ *where* $v_i = \frac{\overline{y_i}}{\overline{y_1}+\overline{y_2}}, i = 1, 2.$

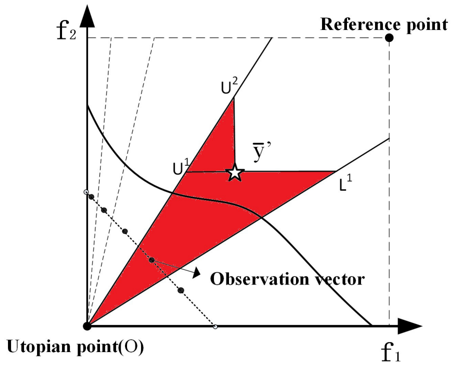

Fig. 2. Biased Cone Decomposition in Geometric Proportion.

It could be easily inferred that an observation vector has the following features: $v_i \geq 0$ and $v_1 + v_2 = 1$. Given a prescribed number of divisions N, a series of reference observation vectors $\mathbf{V}^k$ in geometric proportion can be defined as follows:

$$\mathbf{V}^k = (\frac{s(1-q^{k+1})}{1-q}, 1 - \frac{s(1-q^{k+1})}{1-q}), k = 0, 1, ...N-1. \tag{4}$$

where $V^0 = (s, 1-s)$ is the first reference observation vector in geometric proportion and $q > 1$ denotes the proportion. Furthermore, the region $C = \{\mathbf{y} =$

$(y_1, y_2)|y_1 \geq 0 \wedge y_2 \geq 0\}$ can be decomposed into N conical sub-regions and the k-th sub-region $\mathbf{C}^k = \{\mathbf{y} \in \mathbf{C}| \frac{s(1-q^{k-1}+1-q^k)}{2(1-q)} \leq v_1(\mathbf{y}) < \frac{s(1-q^k+1-q^{k+1})}{2(1-q)}\}$, $k = 0, ..., N-1$. It is evident form Fig. 1 that the individuals closer to the global optimum on the Pareto front can help the population to generate an offspring near the global optimum in a higher probability, compared with the individuals farther away from the global optimum. In view of this consideration, CADE employs the biased cone decomposition in geometric proportion, as shown in Fig. 2, rather than the uniform cone decomposition in CAEA. Similar to CAEA, the conical area is regarded as a significant indicator and CADE compares the conical area of individuals in the same sub-region and preserves the one with the smaller conical area. However, because of the biased cone decomposition in geometric proportion, the calculation equation of conical area is different from that in CAEA. The definition of conical area is as follows:

Definition 2 (Conical area). *Let $\overline{\mathbf{y'}} \in \mathbf{C}^k, 0 \leq k \leq N-1$, $\mathbf{y^r}$ denote the reference point which should be an approximate infinity point dominated by all the other solutions, then, the conical area for $\overline{\mathbf{y'}}$ is the area of the portion $C(\overline{\mathbf{y'}}) = \{\mathbf{y} \in \mathbf{C}^k | \neg(\mathbf{y} \prec \overline{\mathbf{y'}}) \wedge \mathbf{y} \prec \overline{\mathbf{y^r}}\}$, written as $S(\overline{\mathbf{y'}})$.*

It can simply be acquired from Fig. 2 that $S(\overline{\mathbf{y'}})$ represents the portion non-dominated by $\overline{\mathbf{y'}}$ in the conical sub-region $\mathbf{C}^k$ in which $\overline{\mathbf{y'}}$ lies. If $1 \leq k \leq N-2$, the portion $\mathbf{C}^k$ is concave quadrilateral $OL^1\overline{\mathbf{y'}}U^2$ where $O = (0,0)$, $L^1 = (b\overline{y'_2}, \overline{y'_2})$ is the intersection between line $f_2 = \overline{y'_2}$ and the lower boundary of $\mathbf{C}^k$, and $U^2 = (\overline{y'_1}, \frac{1}{a}\overline{y'_1})$ is the intersection between $f_1 = \overline{y'_1}$ and the upper boundary of $\mathbf{C}^k$. Specially, $a = t_a/(1-t_a)$ and $b = t_b/(1-t_b)$ where $t_a = 0.5s(2-q^k-q^{k-1})$ and $t_b = 0.5s(2-q^k-q^{k+1})$. Therefore, the conical area $S(\overline{\mathbf{y'}})$ is the sum of the areas of two triangles $\triangle OL^1U^1$ and $\triangle yU^2U^1$ and the area $S(\overline{\mathbf{y'}})$ can be calculated as follows:

$$\begin{aligned} S(\overline{\mathbf{y'}}) &= 0.5(b-a)(\overline{y'_2})^2 + 0.5\tfrac{1}{a}(\overline{y'_1} - a\overline{y'_2})^2 \\ &= 0.5\tfrac{1}{a}\overline{y'_1}^2 + 0.5b\overline{y'_2}^2 - \overline{y'_1y'_2} \end{aligned} \tag{5}$$

where $\overline{\mathbf{y'}} = (\overline{y'_1}, \overline{y'_2}) \in \mathbf{C}^k$. In addition, the reference point $\mathbf{y^r}$ is required to calculate $S(\overline{\mathbf{y'}})$ if $\overline{\mathbf{y'}} \in \mathbf{C}^0$ or $\overline{\mathbf{y'}} \in \mathbf{C}^{N-1}$. The nadir point and the utopian point are used to calculate the reference point $(10^3(f_1^{max} - f_1^{min}), 10^3(f_2^{max} - f_2^{min}))$. If $\overline{\mathbf{y'}} \in \mathbf{C}^0$, the portion $C(\overline{\mathbf{y'}})$ is a concave pentagon $OL^1\overline{\mathbf{y'}}T^1T^2$ where T^1 is the intersection between $f_1 = \overline{y'_1}$ and $y_2 = \overline{y^r_2}$ and T^2 is the intersection between $f_2 = \overline{y^r_2}$ and $y_1 = 0$. Therefore, $S(\overline{\mathbf{y'}})$ is the sum of area of a triangle $\triangle OL^1U^1$ and a rectangle $\overline{\mathbf{y'}}T^1T^2U^1$: $S(\overline{\mathbf{y'}}) = 0.5b(\overline{y'_2})^2 + (\overline{y^r_2} - \overline{y'_2})\overline{y'_1}$. Similarly, if $\overline{\mathbf{y'}} \in \mathbf{C}^{N-1}$, the conical area $S(\overline{\mathbf{y'}}) = 0.5\frac{1}{a}(\overline{y'_1})^2 + (\overline{y^r_2} - \overline{y'_1})\overline{y'_2}$.

2.2 Feasible Subpopulation and Tolerance-based Sorting

In this paper, we define ω as a constraint tolerance value. Afterwards, the tolerance-based sorting is used for the feasible subpopulation (P1) so that one individual having both the lower objective value and the constraint violation degree in the range ω precedes the others having higher objective values or the violation degree out of the range ω. The tolerance-based dominance relationship, written as $\leq_\omega$, between a pair of solutions $\mathbf{x}$ and $\mathbf{x}'$ is defined as follows:

$$\mathbf{x} \leq_\omega \mathbf{x}' = \begin{cases} f_2(\mathbf{x}) \leq f_2(\mathbf{x}'), \text{if } f_1(\mathbf{x}) \leq \omega \text{ and } f_1(\mathbf{x}') \leq \omega \\ f_1(\mathbf{x}) \leq \omega \text{ and } f_1(\mathbf{x}') \geq \omega \\ f_1(\mathbf{x}) \leq f_1(\mathbf{x}'), \text{if } f_1(\mathbf{x}) > \omega \text{ and } f_1(\mathbf{x}') > \omega \end{cases} \tag{6}$$

After the individuals are sorted through the tolerance-based dominance relationship, the feasible subpopulation P1 in CADE is grouped into M levels in sequence. When one new child is used to update P1, it is easy to find out the level in which it lies. If the objective value of the child is better than that of the last individual in the k-th level, then this offspring belongs to this level.

Specifically, the constraint tolerance value ω in the tolerance-based sorting needs to be controlled over the function evaluations so that the algorithms can finally obtain high quality solutions with lower constraints violations. In this paper, ω is controlled as follows:

$$\omega(t) = 0.5 \times (1 - \frac{t}{T}) \times f_1(x_\lambda), 0 \leq t \leq T \tag{7}$$

where x_λ is the individual in the last conical sub-region, T is the allowed maximum number of generations, and t means the number of current generations.

3 Proposed Algorithm: CADE

3.1 Framework of CADE

The framework of CADE is given in Algorithm 1. First of all, the population P' is randomly generated in the decision space. After the initialization of population, the biased cone decomposition in geometric proportion is used to divide the objective space to $popsize_2$ conical sub-regions and only one individual is preserved for every conical sub-regions as described in Line 3. Specifically, the index of each individual is calculated and for each sub-region, the one with smallest conical area is preserved among the individuals having the same index. The rest individuals are chosen as the feasible subpopulation P1. Subsequently, in every generation, we use an adaptive selection parameter to control the operators of generating offerspring. Thereafter, the objective value is used to decide whether to update the utopian point z^{ide}. Then, we calculate the index of the offspring and need to compare the conical area between the offspring and the individual with the same index in P2 and we retain the one with the smaller conical area as described in Algorithm 2. Afterwards, P1 is updated as presented in Line 10-14. Besides, the ω and Ada_rate are updated when $FES \ mod \ N = 0$. Finally, the best solution in the final population acquired by CADE, is outputted.

Algorithm 1: the Framework of CADE

Input: $popsize_1$: the size of P1; $size_{p1}$: the size of one level in P1; $popsize_2$: the size of P2; s: the first item of the biased cone decomposition in P2; q: the proportion of the biased cone decomposition in P2; Max_Fes: the maximum number of function evaluations; Ada_rate: the adaptive rate to choose the operation to generate a child; ω: the constraint tolerance degree; $stage$: the early stage of the evolution process. y: the generated offspring.

Output: $x^{\#}$:the best solution in the final population.

Create N initial solutions $P' = \{\mathbf{y}^1, \mathbf{y}^2, \cdots, \mathbf{y}^N\}$ by randomly sampling from the decision space Ω, where $N = popsize_1 + popsize_2$;

Initialize and the ideal point $\mathbf{z}^{ide} = (z_1^{ide}, z_2^{ide})$ where $z_j^{ide} = \texttt{Min}_{\ \mathbf{y} \in P'} f_j(\mathbf{y})$;

$P2 \leftarrow$For each conical sub-region, a individual with smallest conical area is associate with. For the rest individual, each is assigned to one conical region which is closet to its observation vector respectively;

$P1 \leftarrow$ Rank the rest individuals through the tolerance-based sorting, and group $P1$ into $M = \frac{popsize_1}{size_{p1}}$ levels in sequence;

while $Fes \leq Max\_Fes$ **do**

- Generate offspring y by Adaptive DE operators and Update z^{ide};
- **if** z^{ide} *is updated and* $Fes \leq stage$ **then**
  - $P1 \leftarrow$ Rank the rest individuals through the tolerance-based sorting;
- $P2 \leftarrow$ `ConeUpdatePopTwo`$(P2,y)$;
- **if** $Fes \leq stage$ **then**
  - $P1 \leftarrow$ update the first individual worse than y in the level y associated with by ω comparsion.
- **else**
  - $P1 \leftarrow$ update the individual randomly chosen according to feasibility rule.
- **end**
- **if** Fes *mod* N *is* 0 **then**
  - Update Ada_rate, ω;

end

return $x^{\#}$

Algorithm 2: Cone Update of the Conical Subpopulation (ConeUpdatePopTwo)

Input: $\mathbf{y}$: an offspring for update; z: the current ideal point;

Output: $P2$: the updated conical subpopulation.

Set $temp = \frac{2(f_1(y)-z_1^{ide})(1-q)}{s(1+1/q)(f_1(y)-z^(ide)_1+f_2(y)-z^ide_2)}$;

Let $k_1 = \left\lfloor \log_q temp \right\rfloor$;

Let $k_2 = \frac{2(f_1(y^{k_1})-z_1^{ide})(1-q)}{s(1+1/q)(f_1(y^{k_1})-z^(ide)_1+f_2(y^{k_1})-z^ide_2)}$;

if $k1 \neq k2 \vee S_{k_1}(f(y) - f(z^{ide})) < S_{k_1}(f(y^{k_1}) - f(z^{ide}))$ **then**

- $y^{k_1} = y$

return $P2$

3.2 DE Operators with the Adaptive Selection Parameter

For stability, CADE employs two DE operators, $/DE/current-to-rand/exp$ and $/DE/rand/exp$, to generate offspring. Both of them exhibit great performances to solve COPs. Moreover, CADE uses an adaptive selection parameter Ada_rate to control the probability in which the first DE operator is selected. In each generation, Ada_rate is updated as follows:

$$Ada\_rate = c \times Ada\_rate + (1-c) \times \frac{a}{(a+b)} \tag{8}$$

where a is the number of using $/DE/current-to-rand/exp$ to update the individuals successfully while b is the number of using $/DE/rand/exp$ and c is a control parameter, usually set as 0.5. To avoid only one operators is used, the Ada_rate is set as 0.95 when $Ada\_rate > 0.95$ and as 0.05 when $Ada\_rate < 0.05$. In addition, it is very important for multiobjectives optimization to utilize neighbors to produce a child and CADE selects parents from a whole population consisting of P1 and P2 for DE operators. The first parent employed in both DE operators is chosen in probability 0.75 according to the conical area between two individuals selected randomly from the whole population and is picked randomly in probability 0.25. However, The rest parents are picked from the first one's neighbors with a certain probability 0.5 and are selected randomly from the whole population in probability 0.5.

3.3 the Update of Subpopulations

When an offspring is generated, both P1 and P2 need to be updated. Algorithm 2 presents the update procedure of P2. When updating P2, the sub-region where the offspring lies is firstly located according to the biased cone decomposition and the index k_1 of the corresponding conical sub-region is calculated, as described in Line 2 of Algorithm 2. Then, the index k_2 of the solution y' associated with this conical sub-region in the current P2 is acquired based in Line 3 of Algorithm 2. If $k_1 \neq k_2$, then the offspring is saved and associated with this conical sub-region. Otherwise, if $k_1 = k_2$, the conical area of the offspring and the solution y' are compared and the one with the smaller conical area is preserved.

However, the procedure of updating P1 is different as shown in Line 10-14 of Algorithm 1. When $Fes \leq stage = 1 \times 10^5$, the child y is used to update the first individual $y^{'}$ worse than y in the level where y lies according to the tolerance-based dominance rule for the reason it can make P1 keep diversity in the range ω of constraint violation. In the later stage, i.e., $Fes > stage$, CADE picks an individual randomly from P1 and uses the feasibility rule to update it. In such a situation, the feasibility rule can help P1 converge fast and find a global optimum more quickly.

4 Empirical Results and Discussion

In this section, the general performance of CADE is first validated on 13 widely used benchmark constrained optimization problems collected in [9]. Then, the capability of CADE are compared against with those of the three existing popular algorithms, namely, SR, SaDE[10], and CMODE, on these test instances.

In our experiments, the sizes of the feasible subpopulation P1 and the conical subpopulation P2 are set as 120 and 60, respectively, thereby the total population size equals to 180. In CADE, the number M of levels for P1 and the proportion q for P2 are respectively set to be 2 and 1.1. In addition, the scaling factor F and the crossover control parameter C_r are randomly chosen, respectively, from [0.5, 0.6] and [0.9, 0.95] for both $DE/current - to - rand/exp$ and $DE/rand/exp$ in the adaptive hybrid DE operators. The termination criterion is satisfied when the number of function evaluations (FES) reaches 5×10^5 for every algorithm on every test instance. The other parameters of SR, SaDE and CMODE use the corresponding recommended values provided, respectively, by [4], [10] and [7]. All the four algorithms are implemented in C++ and run on an Intel Core I5-3470 3.20 GHz PC with 4GB RAM. To assess the overall performances of algorithms, a total of 25 statistically independent runs of each algorithm have been executed for each test case.

4.1 General Performance of CADE

As suggested by Liang [9], if a solution $\mathbf{x}$ is feasible and $f(\mathbf{x}) - f(\mathbf{x}^*) \leq 0.0001$ where $\mathbf{x}^*$ represents the optimal feasible solution, the solution x could be considered as a solution satisfying the success condition. Consequently, the difference between $f(\mathbf{x})$ and $f(\mathbf{x}^*)$ is referred as the function error value for the best-so-far solution $\mathbf{x}$ in our experiments.

In our experiment, for each of 13 instances, the best-so-far solution $x^{\#}$ acquired by CADE is feasible at each run in 5×10^5 function evaluations. And the convergence graphs of function error values (in log scale) obtained by CADE in the median run over FES are displayed in Fig. 3. It is worth noting that all the points whose function error values are less than or equal to 0 are not plotted in Fig. 3. It can be gained from Fig. 3 that CADE is able to find a global optimal solution for g04, g06, g08, g11 and g12 by only using about 1.5×10^5 FES. In addition, Figure 3 also clearly indicates that CADE requires about 3×10^5 FES for g01, g05 and g09. Furthermore, it can be inferred from Fig. 3 that for 9 out of 13 test instances, i.e., except g02, g06, g09 and g13, CADE can finally find the best known optimal solutions at 5×10^5 FES, which demonstrates that the feasibility-based rule of updating P1 in the later stage makes CADE able to search a global optimum quickly.

4.2 Comparison with Some Other Popular DE-based Approaches

In order to clearly discover the advantages of CADE for constrained optimization, CADE is further compared against with three popular constrainted optimization approaches: SR, SaDE and CMODE. Table 1 presents the best, worst and mean of function error values achieved, respectively, by SR, SaDE, CMODE and CADE when FES $= 5 \times 10^5$ for test instances g02-g10 and g13. Since all four approaches can find the nearly same exact optimal solution in each of their

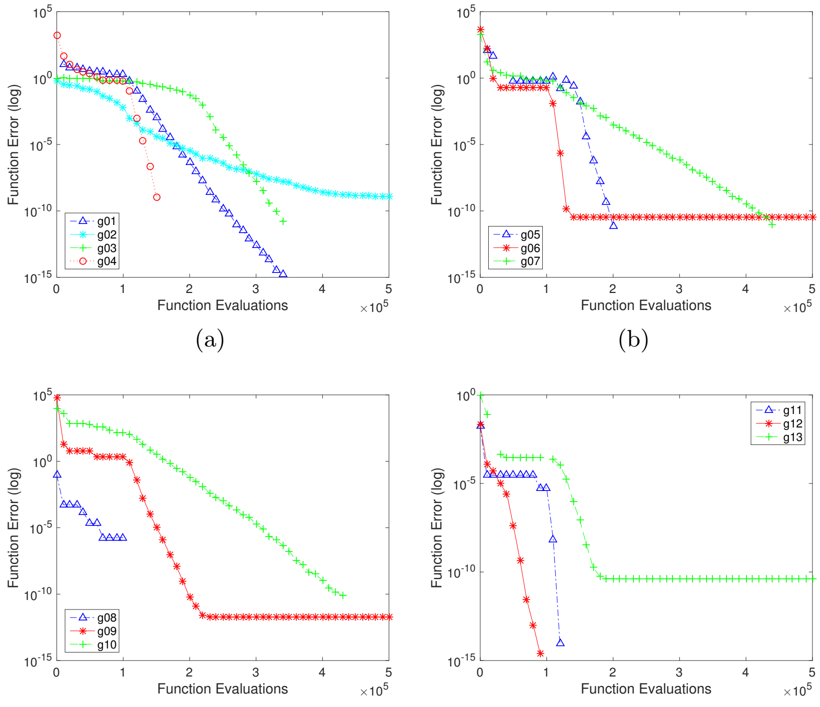

Fig. 3. Convergence graphs of function error values obtained by CADE for g01-g04 (a), g05-g07 (b), g08-g10 (c) and g11-g13 (d).

runs on g01, g11 and g12, Table 1 doesn't show the results for these three test instances. In Table 1, the best result among four compared algorithms for each test instance is specially highlighted in boldface. It is evident from Table 1 that CADE performs best, followed by CMODE and SaDE, while SR behaves worst. Specifically, for 8 out of the 10 test cases, CADE obtains obviously lower error value than the other three algorithms. In particular, it is worth noting that, for g02 and g03, only CADE and CMODE can discover the solutions satisfying the success condition and CADE performs better than CMODE there. In conclusion, CADE could produce competitive results due to the dual-population scheme than other popular constraint optimization techniques.

5 Conclusion

In this paper, we propose a multiobjective approach, CADE, to solve constrained optimization problem. CADE employs the dual population scheme so that the information in the Pareto front can be utilized to achieve the global optimum in a more systematic way. Specially, more promising individuals can be preserved in

Table 1. Function error values achieved, respectively, by SR, SaDE, CMODE and CADE when FES=5×10^5 for test instances g02-g10 and g13.

Prob.	Item	SR	SaDE	CMODE	CADE
g02	Best	8.7256e-03	8.0719e-10	4.1726e-09	**3.0904e-10**
	Wrost	9.0653e-02	1.8353e-02	1.1836e-07	**2.7070e-08**
	Mean	4.1432e-02	2.0560e-03	2.0387e-08	**4.8107e-09**
g03	Best	5.0010e-04	1.3749e-10	2.3964e-10	**-1.0002e-11**
	Wrost	5.0010e-04	1.3389e-04	2.5794e-09	**2.3970e-09**
	Mean	5.0010e-04	1.3532e-05	1.1665e-09	**2.3070e-10**
g04	Best	-9.0502e-08	2.1667e-07	7.6398e-11	**-3.2742e-11**
	Wrost	-9.0502e-08	2.1667e-07	7.6398e-11	**-2.1828e-11**
	Mean	-9.0502e-08	2.1667e-07	7.6398e-11	**-2.5466e-11**
g05	Best	1.3956e-03	0.0000e+00	-1.8190e-12	**-2.7285e-12**
	Wrost	1.3956e-03	0.0000e+00	-1.8190e-12	**-2.7285e-12**
	Mean	1.3956e-03	0.0000e+00	-1.8190e-12	**-2.7285e-12**
g06	Best	1.3924e+04	4.5475e-11	**3.3651e-11**	3.4561e-11
	Wrost	1.3924e+04	4.5475e-11	**3.3651e-11**	3.4561e-11
	Mean	1.3924e+04	4.5475e-11	**3.3651e-11**	3.4561e-11
g07	Best	1.2586e-04	6.8180e-08	7.9783e-11	**-2.0151e-11**
	Wrost	2.5699e-02	6.3431e-05	7.9783e-11	**-2.0091e-11**
	Mean	5.1985e-03	4.7432e-06	7.9783e-11	**-2.0130e-11**
g08	Best	2.3700e-10	8.1964e-11	8.1964e-11	**-1.8036e-11**
	Wrost	2.3700e-10	8.1964e-11	8.1964e-11	**-1.8036e-11**
	Mean	2.3700e-10	8.1964e-11	8.1964e-11	**-1.8036e-11**
g09	Best	-2.1794e-10	3.7440e-07	**-9.8225e-11**	1.7053e-12
	Wrost	1.7727e-05	3.7440e-07	**-9.8111e-11**	1.7053e-12
	Mean	1.8240e-06	3.7440e-07	**-9.8198e-11**	1.7053e-12
g10	Best	2.4606e-05	6.6393e-11	6.2755e-11	**-3.5470e-11**
	Wrost	4.3716e+00	7.8300e-06	6.3664e-11	**-3.5470e-11**
	Mean	1.4789e+00	1.6907e-06	6.2827e-11	**-3.5470e-11**
g13	Best	8.3338e-06	5.3102e-006	**4.1897e-11**	**4.1897e-11**
	Wrost	3.8491e-01	3.8491e-001	**4.1897e-11**	**4.1897e-11**
	Mean	7.6989e-02	1.0778e-001	**4.1897e-11**	**4.1897e-11**

the conical sub-region closer to the global optimum by the biased cone decomposition in geometric proportion and the conical area indicator is used to guide the conical subpopulation to approximate the PF. In addition, the tolerance-based sorting for the feasible subpopulation in the early stage can keep a good diversity of population. Experimental results demonstrate that CADE has a great capability for constrained optimization by employing the dual population scheme and biased cone decomposition. Our ongoing research is focused on applications of CADE for various engineering problems such as the design problem of pressure vessel.

Acknowledgments. This work was supported in part by the Natural Science Foundation of Guangdong Province, China, under Grant 2015A030313204, in part by the Pearl River S&T Nova Program of Guangzhou under Grant 2014J2200052, in part by the National Natural Science Foundation of China under Grant 61203310 and Grant 61503087, in part by the Fundamental Research Funds for the Central Universities, SCUT, under Grant 2017MS043, in part by the Guangdong Province Science and Technology Project under Grant 2015B010131003 and in part by the China Scholarship Council (CSC) under Grant 201406155076 and Grant 201408440193.

References

1. Wang, Y., Cai, Z.: A Constrained Optimization Evolutionary Algorithm Based on Multiobjective Optimization Techniques. The 2005 IEEE Congress on Evolutionary Computation 2, pp. 1081-1087. IEEE (2008).
2. Fan, L., Liang, Y., Liu, X., Gu, X.: A Hybrid Evolutionary Algorithm Based on Dominated Clustering for Constrained Bi-Objective Optimization Problems. International Conference on Computational Intelligence and Security, pp. 543-546. IEEE (2017).
3. Deb, K.: An Efficient Constraint Handling Method for Genetic Algorithms. Computer Methods in Applied Mechanics and Engineering 186(2-4), 311-338 (2000).
4. Runarsson, T. P., Yao, X.: Stochastic Ranking for Constrained Evolutionary Optimization. IEEE Trans. on Evolutionary Computation 4(3), 284-294 (2000).
5. Liu, J., Fan, Z., Goodman, E.: SRDE: An Improved Differential Evolution Based on Stochastic Ranking. ACM/SIGEVO Summit on Genetic and Evolutionary Computation, pp. 345-352. ACM (2009).
6. Ying, W., Peng, D., Xie, Y., Wu, Y.: An Annealing Stochastic Ranking Mechanism for Constrained Evolutionary Optimization. International Conference on Information System and Artificial Intelligence, pp. 576-580. IEEE (2017).
7. Wang, Y., Cai, Z.: Combining Multiobjective Optimization with Differential Evolution to Solve Constrained Optimization Problems. IEEE Trans. on Evolutionary Computation 16(1), 117-134 (2012).
8. Ying, W., Xu, X., Feng, Y., Wu, Y.: An Efficient Conical Area Evolutionary Algorithm for Bi-objective Optimization. IEICE Trans. on Fundamentals of Electronics Communications and Computer Sciences E95.A(8), 1420-1425 (2012).
9. Liang, J., Runarsson, T. P., Mezura-Montes, E., Clerc, M., Suganthan, P. N., Coello, C. A., Deb, K.: Problem Definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization. Int. J. of Computer Assisted Radiology and Surgery(2), (2005).
10. Huang, V., Qin, A., Suganthan, P. N.:Self-adaptive Differential Evolution Algorithm for Constrained Real-Parameter Optimization. IEEE Congress on Evolutionary Computation, pp. 17-24. IEEE (2006).