A Constrained Multiobjective Evolutionary Algorithm Based on a Hybrid Constraint Handling Technique Using Population Trimming Strategy

You-xun KAN¹ and Xing-long JIANG²

¹The China University of Geosciences, Wuhan, China
²Shanghai Engineering Center for Microsatellites, Shanghai, China

Keywords: Constrained multiobjective optimization problems, Constraint handling, Adaptive penalty function, Multiobjective evolutionary algorithm, Population trimming strategy.

Abstract. For constrained multiobjective optimization problems (CMOPs), one of the fundamental issues faced by researchers is how to make comparisons between individuals within the population that results in a balanced selection of better individuals. The selection of better individuals need to be balanced between elitism, diversity and feasibility. In this paper, we propose a hybrid constraint handling technique of population trimming strategy and adaptive penalty function for multiobjective evolutionary algorithm NSGA-II to solve CMOPs. In our approach, a method of objectivization of constraint violations and proportional reduction is used to compare two individuals and trim the population, and as a result the new parent population consisted of the optimal feasible individuals and good infeasible individuals is obtained. To our knowledge, the distant matrix in proportional reduction procedure is firstly proposed for comparison and maintaining diversity in infeasible individuals. Furthermore, an adaptive penalty function method is utilized to give the fitness of individuals in parent population. Numerical simulations indicate that the proposed algorithm outperforms the current state-of-the-art algorithms, e.g. NSGA-II-CD, NAGA-II-WTY, in both convergence and diversity.

Introduction

Multiobjective optimization problems (MOPs), where more than one conflicting objectives have to minimized simultaneously [1], have extensive applications in the fields of science and engineering [2-3]. MOPs must satisfy a set of constraints in the meantime are called constrained multiobjective optimization problems (CMOPs), which is an important part of the optimization field. CMOPs should rise up the challenges from various limits on decision variables, the interference resulting from constraints, and the relationship between objective functions and constraints [4-5]. In the past twenty years, although a number of multiobjective evolutionary algorithms (MOEAs), e.g. NSGA-II [6], SPEA2 [7], PDE-PEDA [8], and FD-EA [9], have been suggested, less work has been done in the area of CMOPs. Most of constraint handling methods in CMOPs are extended from constrained single-objective optimization algorithm, and can be roughly classified into four categories:

Methods based on Constrained Dominance Principle

Method base on constrained dominance principle is the simplest and most commonly used constraint handling approach for CMOPs, and it was proposed to handle constraint for the improved version of Non-dominated Sorting Genetic Algorithm (NSGA-II) in literature [6]. When population fitness ranking is performed, feasible individuals are always considered better than infeasible ones. The main drawback of this method is that it may lose some information of infeasible regions, especially when large amount of feasible individuals are present. To deal with this drawback, a satisfaction level for the constraint is introduced to indicate how well an individual satisfies the constraint and the a level comparison is defined as an order relation that gives priority to the satisfaction level over objective values [10]. Furthermore, dynamic ε infeasible degree allowable constraint dominance relation was introduced [11], and those infeasible individuals with lower constraint violation and better objective values may constrained-dominate the feasible individuals with worth objective values.
Penalty Function Methods

Penalty function methods are the most common constraint handling technique in constrained single-objective optimization problems, and also widely used in CMOPs [4, 12]. In order to making the infeasible solutions have less probability to survive into the next generation comparing with feasible solutions, it uses the amount of constraint violation to punish the infeasible solutions by modifying their objective values which makes it easy to be implemented. However, a number of penalty factors which formulate the penalty function have to be chosen in order to keeping the balance between constraint violation and objective function, which is not a trivial task. These methods can be classified into four categories: Death penalty function methods [13], Static penalty function methods [14], Dynamic penalty function methods [15] and Adaptive penalty function methods [4, 16]. In adaptive penalty function methods, information gathered from the search process will be used to control the amount of penalty added to infeasible individuals.

Objectivization of Constraint Violations

Objectivization of constraint violations treats constraints as objectives, and CMOPs is transformed into an unconstrained one. In literature [17], the constraints are transformed into two new objectives: one is based on a penalty function and the other is made equal to the number of violated constraints. In literature [18-19] the original constrained minimization problem with \( k \) objectives is reformulated as an unconstrained minimization problem with \( k + 1 \) objectives, where an additional objective function is the number of constraint violations. Main drawback of these methods are the significant amount of time required for non-dominated sorting of the solutions and the risk of generating solutions with excellent objective function values and poor constraint satisfaction which may not be of any practical use [18].

Hybrid Method

The property of this category is to combine two or more constraint handing technique together to get better performance since different technique has its own advantages and fit for only a subset of problems [5]. In literature [20], an ensemble of constraint handling methods (ECHM) is proposed to tackle constrained multiobjective optimization problems. In literature [11], a hybrid constraint handling mechanism which combines the \( \varepsilon \)-comparison method and penalty method is present. In [21], a hybrid genetic algorithm combine the boundary simulation method with two specially designed trie-like data structures is proposed to solve CMOPs.

According to the no free lunch theorem, different constraint handling methods can be effective during different stages of the search process, the hybrid method often has better performance with suitable assembly [22]. In this paper, we propose a hybrid constraint handling technique of population trimming strategy and adaptive penalty function on basis of NSGA-II. In our approach, a method of objectivization of constraint violations and proportional reduction is used to compare two individuals and trim the population, and as a result the new parent population consisted of the optimal feasible individuals and good infeasible individuals is obtained. To our knowledge, the distant matrix in proportional reduction procedure is firstly proposed for comparison and maintaining diversity of infeasible individuals. Furthermore, an adaptive penalty function method which can effectively use the information carried by infeasible individuals to solve CMOPs is utilized to give the fitness of individuals in parent population.

The paper is organized as follows. Section 2 gives a brief description of CMOPs. A detailed description of the proposed algorithm is given in Section 3. Numerical simulation results, analysis and discussion are given in Section 4. Conclusion and future work are given in Section 5.
Problem Description

A COMPs can be mathematically formulated as follow [4]:

Minimize \( f_i(x) = f_i(x_1, x_2, \ldots, x_n) \), \( i = 1, 2, \ldots, k \)

Subject to \( g_j(x) = g_j(x_1, x_2, \ldots, x_n) \leq 0 \), \( j = 1, 2, \ldots, p \)

\( h_j(x) = h_j(x_1, x_2, \ldots, x_n) = 0 \), \( j = p + 1, p + 2, \ldots, m \)

(1)

where \( x = (x_1, x_2, \ldots, x_n) \) is a \( n \)-dimensional decision variable vector, which is bounded between lower bound \( l = (l_1, l_2, \ldots, l_n) \) and upper bound \( u = (u_1, u_2, \ldots, u_n) \). \( f_i(x) \) is the \( i^{\text{th}} \) objective function, and \( k \) is the number of objective functions required to be simultaneously optimized. There are a total of \( m \) constraints to be satisfied with \( p \) inequality and \( m - p \) equality. The search space is hence divided correspondingly into the feasible regions and infeasible regions. Global optimal solutions are likely to be found by searching both from the feasible regions and infeasible regions.

Proposed Algorithm

The proposed algorithm consists of a hybrid constraint handling technique and a common MOEA, namely NSGA-II. The proposed algorithm is outlined in Algorithm 1.

**Algorithm 1**: The proposed algorithm

- **Input**: \( G_{\text{max}} \) (maximum number of generations)
  - \( N \) (population size)
- **Output**: \( F \) (feasible Pareto-optimal solutions in archive)

**Step1**: Initialization: Create a random parent population \( P_0 \) of size \( N \), then evaluate all individuals and select non-dominated feasible individuals into archive \( F \); Set \( t = 0 \).

**Step2**: Termination: If \( t = G_{\text{max}} \) is satisfied, export the archive \( F \) as the output of the algorithm, stop; Otherwise, \( t = t + 1 \).

**Step3**: Modified Objective Function and Assign Fitness: Calculate modified objective function values of each individuals in parent population \( P \) according to the adaptive penalty function, then assign fitness based on Pareto ranking and crowding distance.

**Step4**: Create Child Population: Binary tournament, selection, recombination, and mutation operators are used to create a child population \( Q \) of size \( N \).

**Step5**: Update Feasible Archive: Evaluate all individuals in \( Q \), then use the feasible individuals in \( Q \) to update archive \( F \).

**Step6**: Population Trimming: Get \( R \) by combining \( P \) and \( Q \); Identify Pareto ranking of individuals in \( R \) with objectivization constraint violations method; Trim the size of \( R \) to \( N \) using Pareto ranking and proportional reduction procedure; Let \( P_{t+1} = R \); go to step2.

Objectivization of Constraint Violations

In order to identify Pareto ranking of combined population, objectivization of constraint violations method which treats the constraints as an additional objective and constraint violation as the fitness is introduced. As a result, the search is focused around the constraint boundary [18]. The \( k+1 \) objective function of individual \( x \) is calculated as the summation of the normalized violations of each constraint divided by the total number of constraints [4]

\[
 f_{k+1}(x) = v(x) = \frac{1}{m} \sum_{j=1}^{m} \frac{c_j(x)}{c_{\text{max}}(x)}
\]

(2)

where

\[
 c_j(x) = \begin{cases} 
 \max(0, g_j(x)), & 1 \leq j \leq p \\
 \max(0, h_j(x) - \delta), & p + 1 \leq j \leq m 
\end{cases}
\]

(3a)
and
\[ c_j^\text{max}(x) = \max_x c_j(x) \]  \hspace{1cm} (3b)

\( \delta \) is the tolerance value for equality constraints (usually 0.001 or 0.0001). If the constraint violation \( c_j(x) \) is greater than zero, then the individual \( x \) violates the \( j^{\text{th}} \) constraint. It is clear that \( f_{k+1}(x) \) is set to zero if a solution is feasible [4-5].

**Adaptive Penalty Function**

The adaptive penalty function, combined with NSGA-II, are widely used to handle COMPs since it can exploit the information carried by infeasible individuals [4-5, 23]. To generate the modified objective functions, the normalized objective function for individual \( x \) in the \( i^{\text{th}} \)-objective function dimension is firstly calculated as:

\[ \tilde{f}_i(x) = \frac{f_i(x) - f_i^\text{min}}{f_i^\text{max} - f_i^\text{min}} \]  \hspace{1cm} (4)

where
\[
\begin{align*}
    f_i^\text{min} &= \min_x f_i(x) \\
    f_i^\text{max} &= \max_x f_i(x)
\end{align*}
\]

Then, using the summation of constraint violations \( v(x) \) as given in formula (2), the objective function for individual \( x \) in the \( i^{\text{th}} \)-objective function dimension is calculated as:

\[ F_i(x) = d_i(x) + [(1 - r_i)x_i(x) + r_i\gamma_i(x)] \]  \hspace{1cm} (6)

where
\[
\begin{align*}
    d_i(x) &= \begin{cases} 
        v(x) & \text{if } r_{i\delta} = 0 \\
        \sqrt{\tilde{f}_i(x)^2 + v(x)^2} & \text{otherwise}
    \end{cases} \quad (7a) \\
    x_i(x) &= \begin{cases} 
        0 & \text{if } r_{i\delta} = 0 \\
        v(x) & \text{otherwise}
    \end{cases} \quad (7b) \\
    \gamma_i(x) &= \begin{cases} 
        0 & \text{if } x \text{ a feasible individual} \\
        \tilde{f}_i(x) & \text{if } x \text{ an infeasible individual}
    \end{cases} \quad (7c)
\]

The parameter of \( \gamma_i \) is the feasibility ratio of parent population \( P_n \), it allows algorithm to switch between feasibility and optimality at any time during search process. If \( \gamma_i \) is low, infeasible individuals with higher constraint violations are penalized more than those with better objective value. Otherwise, those individuals with worse objective value penalized more than those with better values [4].

**Population Trimming Strategy**

After identifying Pareto ranking with \((k+1)\) objective functions, the combining population \( R_i \) of size \( 2N \) is sorted according to non-domination. In order to update the parent population of size \( N \) from \( R_i \), the population trimming strategy is proposed. The new parent population \( P_{r+1} \) is formed by adding solutions from the first front till the size exceeds \( N \). Thereafter, the exceeded individuals are deleted from the solutions of the last accepted front by proportional reduction.

The proportional reduction is applied in order to maintain the proportion of feasible individuals. For the feasible individuals, crowding-distance [6] values are used for reduction. The treatment of infeasible individuals on the other hand involves a proposed distance matrix, which is a square
matrix, where the element \( d_{ij} \) is the normalized Euclidean distance between individual \( x_i \) and individual \( x_j \) in objective space.

\[
d_{ij} = \sqrt{\sum_{t=1}^{T} \left( \frac{f^t_i(x_i) - f^t_j(x_j)}{f^t_{\text{max}} - f^t_{\text{min}}} \right)^2}
\]

(8)

where \( f^t_{\text{max}} \) and \( f^t_{\text{min}} \) are the maximum and minimum values of the \( t^{\text{th}} \) objective function in the infeasible solutions of the last accepted front. The individuals with minimum distance are selected in every iteration, in which the individual with maximum summation of constraint violations is deleted. The main purpose of this procedure is to preserve diversity among infeasible individuals and give pressure to search toward the feasible regions. An example of the reduction procedure with distant matrix is given in Figure 1. The elements with red dotted lines in the distance matrix are deleted accordingly, resulting in the removal, in order, of individual \( x_3 \) and \( x_5 \) from the infeasible dominant population.

At last, the number of feasible individuals should be calculated as

\[
n_D = \left\lceil \frac{n_F \times N_{\text{feas}}}{N_{\text{feas}} + N_{\text{inf}}} \right\rceil
\]

(9)

where \( N_{\text{feas}} \) and \( N_{\text{inf}} \) are the number of feasible and infeasible individuals in the last accepted front respectively. \( n_F \) is the number of individuals, that has exceeded the size of \( N \). Note that we use the ceil function to make the result an integer. The pseudo code for proportional reduction is given in algorithm 2.

As a result, the proposed algorithm obtain both the optimal feasible individuals, and good infeasible individu-als as parent population.

**Algorithm 2. Pseudo code for proportional reduction**

```
Begin
1) Divide the solutions of the last accepted front in \( P_{n+1} \) into feasible individuals \( B_{\text{feas}} \) and infeasible individuals \( B_{\text{inf}} \).
2) Sort \( B_{\text{feas}} \) in ascending order of crowding-distance values, and remove the first \( n_D \) individuals from \( P_{n+1} \) correspondingly.
3) For the infeasible individuals \( B_{\text{inf}} \), calculate the distance between individuals to form the distance matrix (denoted by \( M \)).
4) Do While ( the size of \( P_{n+1} \) is bigger than \( N \))
   4.1) Using distance matrix \( M \), sort the individuals with minimum distance in descending order according to the summation of constraint violations \( v(x) \), and delete the first individual \( x_i \) from \( B_{\text{inf}} \) and \( P_{n+1} \) correspondingly.
   4.2) Delete the row and column corresponding to \( x_j \) in \( M \).
End Do
End
```

412
Experimental Results and Analysis

To evaluate the performance of the proposed algorithm, comparison is done with two algorithms: NSGA-II-CD [6] and NSGA-II-WTY [4]. Both algorithms are considered state-of-the-art. Performance analysis and comparison of the proposed algorithm, NSGA-II-CD, and NSGA-II-WTY can be found in the following section.

Test Problems

The proposed algorithm is tested on a set of 8 constrained multiobjective test problems CTP1-CTP8 [24]. The test problems pose various challenges to multiobjective algorithms, such as disconnected Pareto fronts, narrow regions of feasibility near constraint boundary, discontinuous feasible regions, etc. The test problems are defined as follows:

$$\begin{align*}
\min f_i(x) &= x_i \\
\min f_2(x) &= c(x) \left[ 1 - \frac{f_i(x)}{c(x)} \right] \\
\text{subject to} \quad &\cos(\theta) \left[ f_2(x) - e \right] - \sin(\theta) f_i(x) \geq 0 \\
&\left| a \left[ \sin \left( b \pi \left[ \sin(\theta) \left[ f_2(x) - e \right] + \cos(\theta) f_i(x) \right] \right) \right] \right|^p
\end{align*}$$

(10)

The six constraint parameters for CTP2-CTP8 are listed in Table 1.

The complexity of CTP test problems changes according to the complexity function $c(x)$. In this study, $c(x)$ is the Rastrigin’s function as defined in equation (13).

$$c(x) = 1 + \sum_{i=2}^{n} \left[ x_i^2 - 10 \cos(2\pi x_i) + 10 \right]$$

(11)

where $x_i \in [0,1]$, $x \in [-5,5]$ for $i=2,3,\cdots,n$, $n=5$.

| Table 1. Parameters for the test problems CTP2-CTP8. |
|------------------|-----------|-----|-----|-----|-----|
| $\theta$ | $A$ | $b$ | $c$ | $d$ | $E$ |
| CTP2 | -0.20$\pi$ | 0.20 | 10.0 | 1 | 6 | 1 |
| CTP3 | -0.20$\pi$ | 0.10 | 10.0 | 1 | 0.5 | 1 |
| CTP5 | -0.20$\pi$ | 0.10 | 10.0 | 2 | 0.5 | 1 |
| CTP6 | 0.10$\pi$ | 40.00 | 0.5 | 1 | 2.0 | -2 |
| CTP7 | -0.05$\pi$ | 40.00 | 5.0 | 1 | 6.0 | 0 |
| CTP8 | 0.10$\pi$ | 40.00 | 0.05 | 1 | 2.0 | -2 |
| | -0.05$\pi$ | 40.00 | 2.0 | 1 | 6.0 | 0 |

Experimental Setup

All the algorithms run 50 times on a set of test problems with a population of 100, crossover rate of 0.8, mutation rate of 0.2, and maximum generation number of 100. These parameters are consistent with what used in literature [4]. In addition, we use SBX crossover [25] with distribution index 15 and polynomial mutation with distribution index 20. To ensure a fair comparison, an archive of size 100 is used to store Pareto-optimal solutions for each algorithm respectively.

Performance Metrics

It is suggested that for a $k$-objective optimization problem, at least $k$ performances are needed to compare two or more solutions [26]. To compare the results, the four performance metrics following are used [27].

**Coverage of two set (Q)** [28]. Let $X_1, X_2$ be two approximate Pareto-optimal set. The function of equation (12) maps the ordered pair $(X_1, X_2)$ to the interval $[0, 1]$.
\[ \zeta(X_1, X_2) = \left| \left\{ a_i \in X_2 : \exists a_i \in X_1 : a_i \succ a_i \right\} \right| / |X_2| \]  

(12)

where the symbol \( \succ \) means dominate. When \( \zeta(X_1, X_2) = 1 \), it means all the solutions in \( X_2 \) can be dominated by some solutions in \( X_1 \).

**Generational distance (GD)** [29]. The concept of generational distance is defined as

\[ GD = \frac{\sum_{i=1}^{n} d_i}{n} \]  

(13)

where \( n \) is the number of vector in the set of nondominated solutions found so far, and \( d_i \) is the Euclidean distance between each of these and nearest member of the true Pareto-optimal fronts (PFs). A lower value of GD corresponds to better convergence with respect to the Pareto optimal front.

**Hypervolume (In)** [30]. Hypervolume indicator for a set \( S \) is defined as the volume occupied by the set in function space with respect to a reference point. Mathematically, it is defined as

\[ I_n = \{ \omega a_i | v_i \in S \} \]  

(14)

where \( a_i \) is the volume occupied by the solution \( v_i \). A high value of Hypervolume indicates nondominated optimal solutions dominate a larger region in the objective space. It measures both convergence and diversity of the nondominated fronts. We choose \( (x_0, y_0) \) as our reference point, where \( x_0, y_0 \) corresponding to the maximum value of the true PFs in the \( x \) and \( y \) dimension respectively.

**Spacing (S)** [31]. Let \( A \) be the final approximate Pareto-optimal set. The function \( S \)

\[ S = \sqrt{\frac{1}{|A| - 1} \sum_{i=1}^{4} (\bar{d} - d_i)^2} \]  

(15)

where

\[ d_i = \min_j \left( \sum_{m=1}^{k} |f_m(a_i) - f_m(a_j)| \right) \]  

(16)

and \( a, a_i \in A; i, j = 1, 2, \ldots, 4 \); \( \bar{d} \) is the average value of all \( d_i \); and \( k \) is the number of objective functions. A value of zero for this metric indicates all members of the PFs currently available are equidistantly spaced. It is a performance metrics that measures uniformity.

In this paper, we use the first three performance metrics to measure the algorithms for every test problem. The Spacing metrics is only used in CTP1 and CTP6, as the true PFs of the other test functions involve disconnected regions or discrete points or have nonuniform distribution.

**Experiment Results Comparison**

The CTP problems are classified into 3 groups according to the characteristics of their PFs [5].

**Group1**: CTP1 and CTP6, both with continuous PFs;

**Group2**: CTP2, CTP7 and CTP8, all PFs with a finite number of disconnected regions;

**Group3**: CTP3, CTP4, and CTP5, all PFs consisting of a finite number of discrete points.

Both qualitative and quantitative comparisons are made to validate the proposed algorithm. For qualitative comparison, the finally feasible nondominated fronts which were obtained for same initial population are show in the Figure 2, 4, and 6, where true PFs are marked as blue points, and the Pareto-optimal solutions obtained from the algorithms are marked as red circles. The infeasible objective spaces of CTP problems are filled with green. The quantitative comparison results are show in Figs. 3, 5, 7 and 9. The box plots marked with ‘\( \zeta \), GD, In, S’ give the comparison of the proposed algorithm, NSGA-II-CD and NSGA-II-WTY in Coverage, GD, Hypervolume and
Spacing metric. The labels ‘1, 2, 3’ indicates the proposed algorithm, NSGA-II-CD and NSGA-II-WTY respectively.

Figure 2. Simulation results of three algorithms on Group1.

Figure 3. Box plots of performance metrics on Group1.

For Group 1 test problems, the experimental results are given in Figure 2. It could be observed that the proposed algorithm is the only algorithm among the three able to attain Pareto-optimal solutions over all true PFs. Figure 3 shows the box plots of performance metrics of the three algorithms on Group1 test problems. For CTP1, we can observe that the proposed algorithm is generally dominated by NSGA-II-CD and NSGA-II-WTY as revealed by metrics \( \zeta \). Furthermore, comparatively big \( GD \), large \( I_H \) and \( S \) indicate that the proposed algorithm has weak disadvantage on convergence and uniformity but better diversity of optimal solutions than the other algorithms. Especially, the proposed algorithm has a small number of outliers on \( I_H \) compared with the other algorithms which indicates it has best performance of stability on CTP1. The result of CTP1 in Figure 2 indicates that the other two algorithms due to their focused search on partial PFs, tend to get locally better approximate Pareto-optimal solutions, while the proposed algorithm has a globally better performance. For CTP6, comparable \( \zeta \), small \( GD \), large \( I_H \), and comparable \( S \) indicate that the proposed algorithm has better convergence, better diversity, and comparable uniformity than the other algorithms. Meantime, the values of \( GD \) and \( I_H \) get from proposed algorithm distribute intensively, and there is no outlier in them, which demonstrates the strong stability of the proposed algorithm on CTP6. In short, the proposed algorithm has comparable convergence, better diversity, and comparable uniformity than the other algorithms for the test problem with continuous PFs.
For Group 2 test problems, the experimental results are given in Figure 4, from which we can see Group 2 test problems have a disconnected regions of PFs. Comparing three algorithms on CTP2 and CTP8, it could be observed that the proposed algorithm has the best performance among the three algorithms as it can attain more disconnected regions where true PFs distribute intensively than the other algorithms. For CTP7, the proposed algorithm and NSGA-II-WTY get comparative advantage, while NSGA-II-CD loses a part of disconnected regions. For quantitative comparison, the number of disconnected region which is a suitable metrics to measure diversity performance of an algorithm on Group2 test problems according to [5] is given in Table 2. Results demonstrate the superiority of the proposed algorithm, with the largest mean and smallest standard deviations number of disconnected regions on all three test problems among the tested algorithms. In addition, Figure 5 gives the box plots of performance metrics of the three algorithms on Group 2 test problems. For CTP2, large $I_H$ with intensive distribution, comparable $\zeta$ and $GD$ indicate that the proposed algorithm has better diversity, better stability and comparable convergence than the other two algorithms. For CTP7 and CTP8, comparatively better $\zeta$, small $GD$ and large $I_H$ indicate that the proposed algorithm has better diversity and convergence than the other algorithms. To sum up, the proposed algorithm has obvious advantage in convergence and diversity than the other two algorithms for the test problems with PFs distributed on disconnected regions.
Table 2. Statistics of number of disconnected regions found by three algorithms on Group2 test problems.

<table>
<thead>
<tr>
<th>Test problems</th>
<th>Algorithms</th>
<th>Disconnected regions</th>
<th>Mean</th>
<th>S.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Proposed</td>
<td></td>
<td>12.96</td>
<td>0.20</td>
</tr>
<tr>
<td>CTP2</td>
<td>NSGA-II-CD</td>
<td>12.10</td>
<td>1.67</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NSGA-II-WTY</td>
<td>12.46</td>
<td>0.91</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>6.98</td>
<td>0.14</td>
<td></td>
</tr>
<tr>
<td>CTP7</td>
<td>NSGA-II-CD</td>
<td>6.00</td>
<td>1.40</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NSGA-II-WTY</td>
<td>6.28</td>
<td>1.09</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>2.86</td>
<td>0.35</td>
<td></td>
</tr>
<tr>
<td>CTP8</td>
<td>NSGA-II-CD</td>
<td>1.34</td>
<td>0.92</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NSGA-II-WTY</td>
<td>1.76</td>
<td>0.82</td>
<td></td>
</tr>
</tbody>
</table>

The best results are marked in bold.

For Group 3 test problems, Figure 6 gives the experimental results of the three algorithms. For CTP3, the proposed algorithm and NSGA-II-WTY get comparative advantage, while NSGA-II-CD loses a part of discrete points. Comparing three algorithm on CTP4, it could be observed that the proposed algorithm has the best performance among the three algorithms as it can attain more discrete points, the distance between which and the true Pareto-optimal solutions tend to jeopardize their discovery by the comparison algorithms. For CTP5, we can see that the proposed algorithm has the best performance among the three algorithms as it can get the nearer and extended optimal Pareto fronts. For quantitative comparison, the comparison over the number of discrete points (Table 3) reveals that the proposed algorithm has significant advantage over the other two algorithms, indicated by a much larger mean of the number of discrete points.

Figure 6. Simulation results of three algorithms on Group3.

Figure 7. Box plots of performance metrics on Group3.
Table 3. Statistics of number of discrete point found by three algorithms on Group3 test problems.

<table>
<thead>
<tr>
<th>Test problems</th>
<th>Algorithms</th>
<th>Discrete points</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>9.40</td>
</tr>
<tr>
<td></td>
<td>NSGA-II-CD</td>
<td>4.74</td>
</tr>
<tr>
<td></td>
<td>NSGA-II-WTY</td>
<td>5.30</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>1.24</td>
</tr>
<tr>
<td>CTP3</td>
<td>NSGA-II-CD</td>
<td>0.26</td>
</tr>
<tr>
<td></td>
<td>NSGA-II-WTY</td>
<td>0.36</td>
</tr>
<tr>
<td></td>
<td>Proposed</td>
<td>7.52</td>
</tr>
<tr>
<td>CTP4</td>
<td>NSGA-II-CD</td>
<td>3.06</td>
</tr>
<tr>
<td></td>
<td>NSGA-II-WTY</td>
<td>3.24</td>
</tr>
</tbody>
</table>

The best results are marked in bold.

Furthermore, Figure 7 gives the box plots of performance metrics of the three algorithms on Group 3 test problems. The proposed algorithm generally dominates NSGA-II-CD and NSGA-II-WTY as revealed by metrics $\zeta$. In addition, comparatively small $GD$ and large $I_H$ indicate that the proposed algorithm has better convergence and diversity than the other two algorithms on Group 3 test problems. In summary, proposed algorithm has obvious superiority than the other algorithms for the test problem with PFs distributed on discrete points.

To have a clear view of the performance of the proposed algorithm as compared to NSGA-II-CD and NSGA-II-WTY, the performance metrics of all tested algorithms on CTP1–CTP8 are presented in Table 4, 5, 6.

The comparison of Coverage metric (Table 4) shows that the proposed algorithm has better Coverage than the other two algorithms on all test problems, save CTP1, CTP2 and CTP7. Even on CTP2 and CTP7, the proposed algorithm has comparable performance with the best algorithm of the three. The results indicate that the proposed algorithm has generally good performance with ability to locate well-spread and near-optimal PFs.

Table 4. Comparison of Coverage metric.

<table>
<thead>
<tr>
<th>Test problems</th>
<th>$\zeta$(Proposed, NSGA-II-CD) Mean</th>
<th>S.D.</th>
<th>$\zeta$(NSGA-II-CD, Proposed) Mean</th>
<th>S.D.</th>
<th>$\zeta$(Proposed, NSGA-II-WTY) Mean</th>
<th>S.D.</th>
<th>$\zeta$(NSGA-II-WTY, Proposed) Mean</th>
<th>S.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.207</td>
<td>0.114</td>
<td>0.354</td>
<td>0.170</td>
<td>0.168</td>
<td>0.100</td>
<td>0.376</td>
<td>0.151</td>
</tr>
<tr>
<td>CTP1</td>
<td>0.279</td>
<td>0.108</td>
<td>0.320</td>
<td>0.096</td>
<td>0.309</td>
<td>0.134</td>
<td>0.328</td>
<td>0.123</td>
</tr>
<tr>
<td>CTP2</td>
<td>0.746</td>
<td>0.138</td>
<td>0.308</td>
<td>0.161</td>
<td>0.707</td>
<td>0.133</td>
<td>0.329</td>
<td>0.158</td>
</tr>
<tr>
<td>CTP3</td>
<td>0.817</td>
<td>0.193</td>
<td>0.128</td>
<td>0.164</td>
<td>0.791</td>
<td>0.201</td>
<td>0.133</td>
<td>0.165</td>
</tr>
<tr>
<td>CTP4</td>
<td>0.433</td>
<td>0.164</td>
<td>0.325</td>
<td>0.132</td>
<td>0.464</td>
<td>0.159</td>
<td>0.318</td>
<td>0.119</td>
</tr>
<tr>
<td>CTP5</td>
<td>0.262</td>
<td>0.134</td>
<td>0.235</td>
<td>0.082</td>
<td>0.240</td>
<td>0.064</td>
<td>0.264</td>
<td>0.073</td>
</tr>
<tr>
<td>CTP6</td>
<td>0.450</td>
<td>0.370</td>
<td>0.299</td>
<td>0.338</td>
<td>0.417</td>
<td>0.376</td>
<td>0.283</td>
<td>0.301</td>
</tr>
<tr>
<td>CTP7</td>
<td>0.462</td>
<td>0.354</td>
<td>0.158</td>
<td>0.123</td>
<td>0.284</td>
<td>0.265</td>
<td>0.237</td>
<td>0.125</td>
</tr>
</tbody>
</table>

The better results are marked in bold.

Table 5. Comparison of GD metric.

<table>
<thead>
<tr>
<th>Test problems</th>
<th>Proposed Mean</th>
<th>S.D.</th>
<th>NSGA-II-CD Mean</th>
<th>S.D.</th>
<th>NSGA-II-WTY Mean</th>
<th>S.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6.60e-04</td>
<td>3.13e-04</td>
<td>5.10e-04</td>
<td>2.12e-04</td>
<td>5.08e-04</td>
<td>2.87e-04</td>
</tr>
<tr>
<td>CTP1</td>
<td>1.22e-03</td>
<td>3.92e-04</td>
<td>1.05e-03</td>
<td>4.25e-04</td>
<td>1.27e-03</td>
<td>7.67e-04</td>
</tr>
<tr>
<td>CTP2</td>
<td>7.28e-03</td>
<td>1.56e-03</td>
<td>7.73e-03</td>
<td>1.54e-03</td>
<td>7.65e-03</td>
<td>1.52e-03</td>
</tr>
<tr>
<td>CTP3</td>
<td>3.83e-02</td>
<td>1.01e-02</td>
<td>6.32e-02</td>
<td>2.17e-02</td>
<td>5.81e-02</td>
<td>1.50e-02</td>
</tr>
<tr>
<td>CTP4</td>
<td>3.92e-03</td>
<td>9.80e-04</td>
<td>4.09e-03</td>
<td>1.78e-03</td>
<td>4.13e-03</td>
<td>1.42e-03</td>
</tr>
<tr>
<td>CTP5</td>
<td>1.30e-03</td>
<td>2.41e-04</td>
<td>1.27e-02</td>
<td>7.30e-02</td>
<td>1.84e-03</td>
<td>1.38e-03</td>
</tr>
<tr>
<td>CTP6</td>
<td>4.24e-04</td>
<td>2.64e-04</td>
<td>5.98e-03</td>
<td>2.23e-02</td>
<td>7.59e-03</td>
<td>3.54e-02</td>
</tr>
<tr>
<td>CTP7</td>
<td>1.23e-03</td>
<td>4.03e-04</td>
<td>1.74e-01</td>
<td>3.10e-01</td>
<td>6.73e-02</td>
<td>2.08e-01</td>
</tr>
</tbody>
</table>

The best results are marked in bold.
The comparison of GD metric (Table 5) shows that the proposed algorithm outperforms the other NSGA-II-CD and NSGA-II-WTY in term of GD for all problems except CTP1 and CTP2. Even on CTP1 and CTP2, the proposed algorithm has comparable performance with the best algorithm of the three. The results indicate that the proposed algorithm has in general a significant advantage on convergence compared with the other algorithms.

The comparison of Hypervolume metric (Table 6) shows that the proposed algorithm has the highest Hypervolume among the three tested algorithms regardless of test problems, which indicates the proposed algorithm enjoys the best extended Pareto-optimal solutions that dominate the largest region in the objective space. In addition, the standard deviations of Hypervolume metrics for the proposed algorithm are consistently low, which indicates a more reliable performance for the proposed algorithm than the other algorithms.

<table>
<thead>
<tr>
<th>Test problems</th>
<th>Proposed</th>
<th>NSGA-II-CD</th>
<th>NSGA-II-WTY</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>S.D.</td>
<td>Mean</td>
</tr>
<tr>
<td>CTP1</td>
<td>0.300</td>
<td>0.002</td>
<td>0.296</td>
</tr>
<tr>
<td>CTP2</td>
<td>0.337</td>
<td>0.004</td>
<td>0.323</td>
</tr>
<tr>
<td>CTP3</td>
<td>0.411</td>
<td>0.007</td>
<td>0.391</td>
</tr>
<tr>
<td>CTP4</td>
<td>0.322</td>
<td>0.037</td>
<td>0.148</td>
</tr>
<tr>
<td>CTP5</td>
<td>0.296</td>
<td>0.004</td>
<td>0.278</td>
</tr>
<tr>
<td>CTP6</td>
<td>1.359</td>
<td>0.012</td>
<td>1.082</td>
</tr>
<tr>
<td>CTP7</td>
<td>0.493</td>
<td>0.004</td>
<td>0.445</td>
</tr>
<tr>
<td>CTP8</td>
<td>1.186</td>
<td>0.083</td>
<td>-0.37</td>
</tr>
</tbody>
</table>

The best results are marked in bold.

In summary, experimental results of all test problems indicate that the proposed algorithm has decent performance for CMOPs, generally better than the current state-of-art algorithms in both convergence and diversity.

**Further Discussion**

In order to maximize the extent of the obtained non-dominated front, various techniques are available for density estimation and maintaining diversity in a MOEA, such as crowding-distance method [6] and the kth nearest neighbor method [7], less work has done in the area of maintaining diversity technique for infeasible individuals in a CMOEA, which is important for taking information carried by infeasible individuals. In our approach, objectivization of constraint violations method firstly maintains infeasible individuals with better objective values, then the proposed distance matrix which is a density estimation technique for infeasible individual is used to maintain diversity. An example of the comparison of three density estimation techniques for infeasible individuals on CTP2 test problem is given in Figure 8. The feasible region of CTP2 problems are filled with green, and the infeasible individuals and the infeasible individuals been deleted are marked as blue plus and red circle respectively. We can see that the distance matrix
shows the superiority in maintaining diversity comparing with crowding-distance method and the $k^{th}$ nearest neighbour method, as it not only retains the individuals closed to feasible region but also retains the individuals which distributes widely in infeasible region that are always helpful to get stuck from local Pareto-optimal fronts. As a result, the proposed algorithm obtains a better spread of nondominated solutions compared with the other methods on CTP2 in Figure 4. In addition, the superiority of the proposed algorithm is obvious on CTP8 in Figure 4, as it has a number of local Pareto-optimal fronts divided by infeasible regions.

Furthermore, the adaptive penalty function method balances penalty values between the objective function and constraint violation by the feasibility ratio, and makes the algorithm search from both the feasible regions and the infeasible regions. Different from NSGA-II-WTY, the proposed algorithm trims the combining population before the adaptive penalty function procedure, which improves the quality of the population. Meantime, the feasibility ratio of the population after trimming is also a typical parameter to measure the solutions of search process, as it uses the approximate nondominated population to determine search direction. As observed from the experimental results above, the proposed algorithm yields more evenly distributed, well extended and near-optimal Pareto fronts, and has better convergence and diversity performance compared with NSGA-II-CD and NSGA-II-WTY.

Conclusions
In this paper, the proposed CMOEA involves a MOEA of NSGA-II and a new hybrid constraint handling technique of population trimming strategy and adaptive penalty function. In our approach, a method of objectivization of constraint violations and proportional reduction is used to compare two individuals and trim the population. To our knowledge, the distant matrix in proportional reduction procedure is firstly proposed for comparison and maintaining diversity in infeasible individuals, and the results show its superiority compare with the other density estimation techniques. Furthermore, the adaptive penalty function method is utilized to give the fitness of individuals in the high-quality parent population after trimming, and results the solutions with well extended and near-optimal Pareto fronts.

The proposed algorithm is tested against NSGA-II-CD and NSGA-II-WTY, over a problem set of 8 (CTP1-CTP8). Experimental results demonstrate that the proposed algorithm is generally better than NSGA-II-CD and NSGA-II-WTY in terms of Coverage, GD, Hypervolume, and Spacing. In addition, we observe consistent low deviation in Hypervolume over the duration of multiple runs, which indicates strong stability of the proposed algorithm. We thus conclude the proposed algorithm shows great promise for solving constrained multiobjective problems.

For future works, we intend to apply the proposed algorithm to more realistic problems, such as the planning of satellite network etc. A feasibility study of the proposed algorithm in those scenario, including more extensive and application-oriented test cases would be done and expected to reveal the application value of the proposed algorithm. We also intend to investigate the possibility of combining the constraint handling technique proposed with other search algorithms.

Acknowledgements
The presented work was supported by the National Natural Science Foundation of China, under Grants 61401278.

References


