Nat Comput (2014) 13:421–445
DOI 10.1007/s11047-014-9415-z

Immune clonal coevolutionary algorithm for dynamic multiobjective optimization

Ronghua Shang · Licheng Jiao · Yujing Ren · Jia Wang · Yangyang Li

Published online: 20 February 2014

Abstract In this paper, a new evolutionary algorithm, called immune clonal coevolutionary algorithm (ICCoA) for dynamic multiobjective optimization (DMO) is proposed. On the basis of the basic principles of artificial immune system, the proposed algorithm adopts the immune clonal selection to solve DMO problems. In addition, the theory of coevolution is incorporated in ICCoA in global operation to preserve the diversity of Pareto-fronts. Moreover, coevolutionary competitive and cooperative operation is designed to enhance the uniformity and the diversity of the solutions. In comparison with NSGA-II, immune clonal algorithm for DMO and direction-based method, the simulation results obtained on 5 difficult test problems and on related performance metrics suggest that ICCoA can achieve better distributed solutions and be very effective in maintaining the uniformity of Pareto-fronts.

Keywords Dynamic multiobjective optimization · Immune clonal selection · Coevolution

1 Introduction

There are many multiobjective optimization problems existing in the real world and changing over time, which are called dynamic multiobjective optimization (DMO) problem (Jin and Branke 2005; Chen and Jiao 2010). Due to the important role of DMO problem in practical application, the study of algorithms for DMO problems is very necessary. Nevertheless, the designed algorithms to solve this kind of problems often have difficulties in tracking the optimum continuously (Farina et al. 2004; Wang et al. 2010). Although there are a lot of algorithms concentrating on dynamic simple-objective optimization (DSO) problems (Deb et al. 2002; Zitzler and Thiele 2005; Nusawardhana and Zak 2004) or static multiobjective optimization problems, there is few related research and promotion for DMO problems (Ursem et al. 2002; Shang et al. 2005; Lung and Dumitrescu 2010; Abido 2010). Farina et al. (2004) proposed a set of DMO test problems and the correlative solution: direction-based method (DBM). In 2005, an immune clonal algorithm for DMO was presented by Shang et al. (2005), which based on immune clonal mechanism. Goh and Tan proposed a dynamic competitive-cooperation coevolutionary algorithm (dCOEA) for DMO (Goh and Tan 2007), which focused on the competitive-cooperation strategy in co-evolution. Cámara et al. (2009) proposed some new measures for dynamic multi-objective problems, especially, some new measures for Pareto-fronts unknown problems. Huang et al. (2011) developed a dynamic multi-objective optimization algorithm, which is inspired by membrane computing. Helbig et al. (2013) investigated the effect of various approaches to manage boundary constraint violations on the performance of the dynamic Vector Evaluated Particle Swarm optimization algorithm. However, the research and promotion of DMO problems is still on the preliminary stage. Therefore, it is necessary to propose effective algorithms for DMO problems.

Artificial immune system (AIS) is an adaptive system enlightened by immunology, which can simulate the immunological functions, principles and models to solve complex problems (de Castro and Timmis 2002a; Gong et al. 2006). With the development in recent years, its algorithm is mainly concentrated on clonal selection

R. Shang (✉) · L. Jiao · Y. Ren · J. Wang · Y. Li
Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, China
e-mail: rhshang@mail.xidian.edu.cn

algorithm (de Castro and Timmis 2002b; Liu et al. 2010), negative selection algorithm (Forrest et al. 1994) and so forth. AIS has the potential to solve new problems as it can provide memory, unsupervised learning, self-organization and other evolutionary learning mechanisms. In addition, AIS can combine the advantages of machine reasoning and neural network (Jiao and Wang 2000).

The unawareness of the coexistence of competition and cooperation among different populations in most available traditional evolutionary algorithms resulted in a new kind of evolutionary algorithm, namely Coevolutionary Algorithm. Coevolutionary Algorithm is based on coevolution, which emphasizes that the evolution of some species is correlated with the one of other species. Its advantage lies in taking the coordinative relation among populations and the relation between population and environment into account. For this superiority, the study of coevolutionary algorithm has become one of the key issues of current evolutionary computation (Jiao et al. 2006; Goh and Tan 2009).

As a consequence, from the point of view of coevolution and inspired by the concept of immunodominance and the theory of immune clonal selection, in this paper, we address the DMO problems and propose a new algorithm ICCoA for DMO, in which the relation between competition and cooperation is considered to improve the uniformity and the diversity among different populations. On the basis of immune clonal algorithm for DMO (ICADMO) suggested by Shang et al. (2005), which is able to search the set of Pareto-optimal solution in the dynamic environment. ICCoA has improved the performance of the set by adopting the U-measure approach (Leung and Wang 2003; Wang and Dang 2008) as competitive operator and designing corresponding cooperative operator. On five test problems proposed by Farina et al. (2004), ICCoA has been compared with the famous genetic algorithm NSGA-II (Deb et al. 2002), direction-based method (DBM) (Farina et al. 2004) and ICADMO (Shang et al. 2005). The simulation results, which suggest that ICCoA has better performance, encourage the new algorithm to more complex and real-world DMO problems.

In the remainder of the paper, the paper is organized as follows. In Sect. 2, key concepts of DMO problem are introduced. Thereafter, the framework of the new algorithm ICCoA, is proposed in details in Sect. 3. Section 4 presents the test problems and the comparative results of ICCoA and the other three algorithms. The last section offers a brief conclusion and the future work.

2 Dynamic multiobjective optimization problem

In the real world, a number of world optimization problems involve optimization of several conflicting objectives and these optimization problems are called multiobjective optimization (MO) problems (Shang et al. 2012). The general model of an MO problem with M ($M > 1$) objectives is shown as follow:

$$\begin{cases} \min \quad \boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_M(\boldsymbol{x}))^T \\ s.t. \quad \boldsymbol{g}(\boldsymbol{x}) \le 0, \ \boldsymbol{h}(\boldsymbol{x}) = 0 \end{cases} \tag{1}$$

where $f_1(\boldsymbol{x})$, $f_2(\boldsymbol{x})$, …, and $f_M(\boldsymbol{x})$ are M objective functions that always cannot reach the minimal values simultaneously. $\mathbf{g}(\boldsymbol{x})$ is the inequality constraint and $\boldsymbol{h}(\boldsymbol{x})$ is the equality constraint. In order to make a clear notion of optimality in this scenario, three definitions based on Eq. (1) are given (Deb et al. 2002).

Definition 1 Pareto-dominance

Based on Eq. (1), given two points in the decision space $\boldsymbol{x}, \boldsymbol{x}^* \in \boldsymbol{X}$, $\boldsymbol{x}^*$ is said to dominate (Pareto-optimal) another solution $\boldsymbol{x}$ (denoted $\boldsymbol{x}^* \succ \boldsymbol{x}$) iff it satisfies the following conditions:

$$\begin{aligned} &(\forall \mathrm{i} \in \{1, \cdots, M\} : f_i(\boldsymbol{x}^*) \le f_i(\boldsymbol{x})) \\ &\quad \wedge (\exists k \in \{1, \cdots, M\} : f_k(\boldsymbol{x}^*) < f_k(\boldsymbol{x})) \end{aligned} \tag{2}$$

where "^" stands for "and", that is the conditions on both sides of the symbol "^" have to be satisfied. And "v" represents "or", that is any one of the two conditions on both sides of the symbol "v" can be satisfied.

Definition 2 Pareto-optimal and Pareto-optimal set

A solution $\boldsymbol{x}^* \in \boldsymbol{X}$ is said to be Pareto-optimal (non-dominated) iff

$$\neg \exists \boldsymbol{x} \in \boldsymbol{X} : \boldsymbol{x} \succ \boldsymbol{x}^* \tag{3}$$

The set $\boldsymbol{P}_s$ of all the Pareto-optimal solutions is called Pareto-optimal set.

Definition 3 Pareto-optimal front

The set $\boldsymbol{P}_F$ of all objective function values corresponding to the solutions in $\boldsymbol{P}_S$:

$$\boldsymbol{P}_F = \{ \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_M(\boldsymbol{x}))^T | \boldsymbol{x} \in \boldsymbol{P}_S \} \tag{4}$$

Based on the above definitions, a multi-objective optimization problem can be seen as looking for the Pareto-optimal solutions or approaching the Pareto-optimal front. So the solutions found by a good algorithm should approach the Pareto-optimal set (POS) and have a good diversity.

However, in the real world, there exist many problems which not only have many objectives but also change over time in the environment, and this kind of problems are called DMO problems (Farina et al. 2004). Without loss of generality, in this paper, we consider the following DMO problems:

Definition 4 DMO minimization problem for M objectives:

$$\begin{cases} \min \quad \boldsymbol{f} = \{f_1((\boldsymbol{x},t),f_2((\boldsymbol{x},t)\cdots,f_M(\mathbf{x},t)\} \\ s.t.\ \boldsymbol{g}(\boldsymbol{x},t) \le 0, \boldsymbol{h}(\boldsymbol{x},t) = 0 \end{cases} \tag{5}$$

where $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2,\ldots, \boldsymbol{x}_m) \in R^m$ denotes the decision variable vector, $t \in [t_0, t_s]$ is the time variable, $\boldsymbol{g}(\boldsymbol{x},t)$ and $\boldsymbol{h}(\boldsymbol{x},t)$ is the constraint condition, $\boldsymbol{X}(t) = \{\boldsymbol{x}|\boldsymbol{g}(\boldsymbol{x},t) \le 0$ and $\boldsymbol{h}(\boldsymbol{x},t) = 0\}$ stands for the feasible set at time t.

For a DMO problem, as the time progresses on, we want to find the Pareto-optimal solutions $\boldsymbol{x}^* \in \boldsymbol{X}(t)$ for some time step t. That is, At some moment t, let $\boldsymbol{x}^* \in \boldsymbol{X}(t)$, if there is no $\boldsymbol{x} \in \boldsymbol{X}(t)$ such that $f_i(\boldsymbol{x},t) \le f_i(\boldsymbol{x}^*,t)$ for every $i \in \{1,2,\ldots, M\}$, and $f_k(\boldsymbol{x},t) < f_k(\boldsymbol{x}^*,t)$ for some $k \in \{1,2,\ldots,M\}$, then $\boldsymbol{x}^*$ is defined as the Pareto-optimal solution or nondominated solution in $\boldsymbol{X}(t)$. In addition, POS is the set of optimal solutions in $\boldsymbol{X}(t)$. Pareto optimal front POF $= \{\boldsymbol{f}(\boldsymbol{x})|\boldsymbol{x} \in$ POS$\}$ is the mapping of POS in the objective space. Therefore, in a traditional MO problem, the solutions just a set of Pareto-optimal solutions that approximate the Pareto-front; however in DMO problem, the objective functions are related with time t, so the solutions are adjusted with the real time t. Therefore, there are generally four possible ways for a DMO problem to change with time (Farina et al. 2004):

Type 1 The POS changes, while the POF does not change.

Type 2 The POF changes, while the POS does not change.

Type 3 Both the POS and the POF change with time.

Type 4 Although the problem changes dynamically, neither the POS nor the POF changes.

In a dynamic system or for a dynamic problem, when the time changes, there are may be more types of above four types can occur simultaneously, which is a more complex situation. In this paper, we concentrate on the test problems of type 1, type 2 and type 3. Type 4 means that in the system it does not make any change in the POS or POF (Farina et al. 2004), which can be solved by static multi-objective optimization methods.

3 Immune clonal co-evolutionary algorithm for DMO

3.1 Immune clonal operation

3.1.1 Basic principles

In immune system, antigen is the substance which can induce the organism to have immune response and specifically react with corresponding antibody (Coello Coello and Cortes 2002; de Castro et al. 2010; Woolley and Milanović 2011). In AIS, which imitates the function of natural immune systems, the antigen denotes the problem and its constraints (de Castro and Timmis 2002b), while the antibody is the candidate solution to the problem. In this system, an antigen is any substance that causes the immune system to produce antibodies against it, and when an antigen is detected, those antibodies that best recognize an antigen will proliferate by cloning (Gong et al. 2008; Shang et al. 2012;Yang et al. 2009). In this paper, the antibody population is defined as $P = \{\boldsymbol{x}_1, \boldsymbol{x}_2,\ldots, \boldsymbol{x}_n\}$.

3.1.2 Immune clonal operators

Clonal operation is to proliferate the nondominated solutions, while clonal selection operation is to choose the more optimized offspring to form a new population (Shang et al. 2012). With the concept of Pareto-optimum, ICAD-MO (Shang et al. 2005) divides the antigen into dominated solutions and nondominated solutions and also enlarges the nondominated solutions with the immune clonal operators. This procedure realizes the enlargement of the search space and the competition among antibodies. Furthermore, this operation makes the various mutation and reconfiguration policy of some antigen possible and also maintains the diversity of the population.

The specific clonal proliferation operation is as follows:

$$\begin{aligned} \boldsymbol{P}' &= R_C^P\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n\} \\ &= R_C^P(\boldsymbol{x}_1) \cup R_C^P(\boldsymbol{x}_2) \cup \cdots \cup R_C^P(\boldsymbol{x}_n) \\ &= \{\boldsymbol{x}_1^1, \boldsymbol{x}_1^2, \cdots, \boldsymbol{x}_1^{q_1}\} \cup \{\boldsymbol{x}_2^1, \boldsymbol{x}_2^2, \cdots, \boldsymbol{x}_2^{q_1}\} \cup \cdots \cup \{\boldsymbol{x}_n^1, \boldsymbol{x}_n^2, \cdots, \boldsymbol{x}_n^{q_1}\} \\ &= \{\boldsymbol{x}_1^1, \boldsymbol{x}_1^2, \cdots, \boldsymbol{x}_1^{q_1}, \boldsymbol{x}_2^1, \boldsymbol{x}_2^2, \cdots, \boldsymbol{x}_2^{q_1}, \cdots, \boldsymbol{x}_n^1, \boldsymbol{x}_n^2, \cdots, \boldsymbol{x}_n^{q_1}\} \\ &= \{\boldsymbol{x}_1', \boldsymbol{x}_2', \cdots, \boldsymbol{x}_{q_1 * n}'\} \end{aligned} \tag{6}$$

R_c^p denotes the clonal proliferation operator. With this operation, we can enlarge the size of the population in global operation and attain better optimal ability. q_1 is the clonal proportion, which means the antibody is stimulated by the antigen and can realize the biological multiplication. In ICCoA, $q_1 = 5$.

The clonal selection operation is as follows:

$$\begin{aligned} \boldsymbol{P}'' &= R_C^S(\boldsymbol{P}') \\ &= R_C^S(\{\boldsymbol{x}_1', \boldsymbol{x}_2', \cdots, \boldsymbol{x}_{q_1^* n}'\}) \\ &= \{\boldsymbol{x}_1', \boldsymbol{x}_2', \cdots, \boldsymbol{x}_n'\} \end{aligned} \tag{7}$$

where R_c^S denotes the clonal selection operator.

In ICCoA, the antibodies such as $\boldsymbol{x}_1', \boldsymbol{x}_2', \cdots, \boldsymbol{x}_{q_1 * n}'$ in an antibody population $\boldsymbol{P}'$ are divided into dominated ones and nondominated ones. For any antibody $\boldsymbol{x}_i' \in \boldsymbol{P}', i \in \{1, 2,$

$\cdots, q_1 * n\}$, $\boldsymbol{x}_i^{'}$ is called a non-dominated antibody in the current iteration iff:

$$\neg\exists \boldsymbol{x}_j^{'} \neq \boldsymbol{x}_i^{'} \in \boldsymbol{P}^{'}, j \neq i, j \in \{1, 2, \cdots, q_1 * n\} : \boldsymbol{x}_j^{'} \succ \boldsymbol{x}_i^{'} \quad (8)$$

Otherwise, $\boldsymbol{x}_i^{'}$ is called a dominated antibody.

Based on Eq. (8), the antibodies in $\boldsymbol{P}^{'}$ are divided into two parts: non-dominated antibodies population $\boldsymbol{P}_{non}^{'}$ with N_{non} antibodies and dominated antibodies population $\boldsymbol{P}_{dom}^{'}$with N_{dom} antibodies. Where $N_{non} + N_{dom} = q_1 * n$. If $N_{non} > n$, then select n antibodies from $\boldsymbol{P}_{non}^{'}$ according to the APU strategy proposed by Shang et al. (2012). If $N_{non} < n$, then select $n - N_{non}$ antibodies in $\boldsymbol{P}_{non}^{'}$ randomly to compose a population denoted by $\boldsymbol{P}_{n-N_{non}}^{'}$ and $\boldsymbol{P}^{''} = \boldsymbol{P}_{non}^{'} \cup \boldsymbol{P}_{n-N_{non}}^{'}$, where $\boldsymbol{P}^{''}$ is consisted with $\boldsymbol{P}_{non}^{'}$(with N_{non} antibodies) and $\boldsymbol{P}_{n-N_{non}}^{'}$(with $n - N_{non}$ antibodies selected randomly from $\boldsymbol{P}_{non}^{'}$). Therefore, $\boldsymbol{P}^{''}$contains n $(n = N_{non} + n - N_{non})$ antibodies. With this operation, we can select the nondominated solutions as the next generation, which can preserve the performance of the population.

3.2 Coevolutionary operation

3.2.1 Basic principles

Coevolution emphasizes the interactions among different populations, which means that different individuals and different species are all mutually benefitted and mutually restricted, so we should not consider a single population. With the consideration of this situation, coevolution exists on the basis of the interdependent, mutual restrictive and mutual coordinated relationship of various populations (Jiao et al. 2006).

To avoid loss of diversity for only considering single population, ICCoA makes use of the competitive and cooperative relationship of multiple populations. Moreover, ICCoA uses the difference value of U-measure, which can measure the uniformity and spread for the optimal solutions, for populations as criterion to improve the uniformity, realize the co-existence of competition and cooperation and preserve the diversity (Leung and Wang 2003; Wang and Dang 2008).

3.2.2 Pareto neighborhood crossover operator

In the procedure of the algorithm, there may be better solution near the local POS. In order to avoid loss of the better solution, we use Pareto neighborhood crossover operator in Jiao et al. (2006) to implement local search operation on local Pareto-optimal solutions. The specific operator is presented in Algorithm 1.

Algorithm 1: Pareto Neighborhood Crossover Operator

Input: the current population $\boldsymbol{P}=\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$, the current Pareto optimal set $\boldsymbol{P}^{'}=\{\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_s\}$,

Output: New antibody $(y_1, y_2, \ldots, y_m)$, where m is the dimension number of the decision variable.

Begin

1. for an antibody $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im}) \in \boldsymbol{P}$ ($i \in \{1,2,\ldots,n\}$), randomly chosen an antibody $\boldsymbol{r}_j = (r_{j1}, r_{j2}, \ldots, r_{jm})$ from $\boldsymbol{P}^{'}$($j \in \{1,2,\ldots,s\}$).
2. $U(.,.)$: uniformly distributed random number
3. **For** $k = 1 : m$

   $y_i = r_{jk} + U(-1,1) \cdot (r_{jk} - x_{ik})$

   End for

End

This operation generates a new antibody $(y_1, y_2, \ldots, y_m)$, which replaces the antibody $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im})$ in the current population $\boldsymbol{P}$. This operator implements the exploitation of the Pareto neighborhood for better solutions and finds diversity solutions. Therefore, it can avoid loss of the diversity.

3.2.3 Co-evolutionary competitive and cooperative operation

In this paper, we use the U-measure proposed by Leung and Wang (2003) as criterion, which stands for the uniformity and the spread of the Pareto optimal solution distribution of the population in the objective space. The less the U-measure value is, the more uniform and broader the distribution will be.

For a two objective problem, U-measure is stated as follows:

$$d_{mean} = \frac{1}{N+1} \sum_{i=0}^{N} d_{i,i+1} \quad (9)$$

$$U_m = d_{std} = \sqrt{\frac{1}{N}\sum_{i=0}^{N}(d_{i,i+1} - d_{mean})^2} \tag{10}$$

where $d_{i,i+1}$ is the distance of every pair of adjacent points and d_{mean} is the mean distance.

For an M objective problem, U-measure is stated as follows:

$$d_{mean} = \frac{1}{2MN}\sum_{g\in T}\sum_{r=1}^{M}(d_{2r-1} - d_{2r}) \tag{11}$$

T. For a more detailed description, the interested reader is referred to Wang and Dang (2008).

In ICCoA, when the absolute value of the differential between the U-measure value for two populations is greater than a threshold value θ, the difference of the uniformity for the populations is very large. Thus, we adopt the competitive operation. Otherwise, it means that there is no obvious difference between the populations and the cooperative operation will be implemented. The U-measure value of $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ is U_{m1} and U_{m2} respectively. The specific operation is shown in Algorithm 2.

Algorithm 2: Co-evolutionary competitive and cooperative operation

Input: Initial independent populations $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$

Output: New population $\boldsymbol{P}_3$

Begin

1. POS_1, POS_2: Pareto optimal solutions of $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$
2. $\boldsymbol{x}_i^1$, $\boldsymbol{x}_i^2$: random antibodies of $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$
3. $\boldsymbol{r}_i^1$, $\boldsymbol{r}_i^2$: antibodies randomly chosen from POS_1 and POS_2
4. k: number of the antibodies randomly chosen for Pareto neighborhood crossover operation
5. $U(.,.)$: uniformly distributed radom number
6. Initialize an empty solution set $\boldsymbol{P}_3$
7. Calculate the U-measure value of $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$, denotes as U_{m1} and U_{m2};
8. $Dvalue = |U_{m1} - U_{m2}|$;
9. **If** $Dvalue >= \theta$
   - **If** $U_{m1} <= U_{m2}$
     - $\boldsymbol{P}_3 = \boldsymbol{P}_1$;
   - **Else** $\boldsymbol{P}_3 = \boldsymbol{P}_2$;
   - **End if**
   - **Else** **For** $i = 1 : k$
     - $\boldsymbol{y}_i^1 = \boldsymbol{r}_i^2 + U(-1, 1) \cdot (\boldsymbol{r}_i^2 - \boldsymbol{x}_i^1)$;
     - $\boldsymbol{y}_i^2 = \boldsymbol{r}_i^1 + U(-1, 1) \cdot (\boldsymbol{r}_i^1 - \boldsymbol{x}_i^2)$;
     - $\boldsymbol{y}_i = \frac{(\boldsymbol{y}_i^1 + \boldsymbol{y}_i^2)}{2}$
     - **End for**
     - $\boldsymbol{P}_3 = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_k\}$
   - **End if**

End

$$U_m = d_{std} = \sqrt{\frac{1}{2MN-1}\sum_{g\in T}\sum_{r=1}^{2M}(d_r - d_{mean})^2} \tag{12}$$

where d_{mean} is the mean distance, T is a set of Pareto solutions in the objective space, each point $g \in T$, $r \in [1, M]$. U_m can measure the uniformity and spread for the points in

As shown in Algorithm 2, when the differential between the U-measure value for two populations is greater than a threshold value θ, the algorithm implements coevolutionary competitive operation and the new population $\boldsymbol{P}_3$ is determined by the parent with less U-measure value. Forasmuch as the new population has better uniformity and

spread, the quality of the solutions in the objective space is improved. When the differential is less than θ, the algorithm will choose coevolutionary cooperative operation. Under this circumstance, the new population $\boldsymbol{P}_3 = \{\boldsymbol{y}_1, \boldsymbol{y}_2,\ldots, \boldsymbol{y}_k\}$ with a size of *Nondom* is chosen from half of the offspring $\boldsymbol{P}_{1a}$: $\boldsymbol{y}_i^1 = \boldsymbol{r}_i^2 + U(-1, 1).(\boldsymbol{r}_i^2 - \boldsymbol{x}_i^1)$ $(i = 1, 2,\ldots, k)$ and half of offspring $\boldsymbol{P}_{2b}$: $\boldsymbol{y}_i^2 = \boldsymbol{r}_i^1 + U(-1, 1).(\boldsymbol{r}_i^1 - \boldsymbol{x}_i^2)$ $(i = 1, 2,\ldots, k)$. As $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ are evolved absolutely independently, they can search in different areas in the objective space respectively. Therefore, with the cooperative operation the populations will exchange their information to expend the search area of the algorithm. Hence, making the most of the difference between different populations can improve the uniformity.

3.3 Uniformity maintenance operation

One of the most important performance metrics is the uniformity of the solutions in the objective space, therefore, in the DMO problems, we should make the distribution as uniform as possible. While according to Formula (7) and Formula (8), the nondominated solutions will be selected as the next generation, which will be denoted as the nondominated antibodies. The number of the nondominated antibodies may be larger and larger after the clonal proliferation operation and the speed of the algorithm will be seriously influenced in the iterative process.

In order to avoid this possibility, we implement the uniformity maintainance operator in Shang et al. (2012), namely antibody population updating (APU) strategy, which can maintain the uniformity of the population and guarantee the algorithm's convergent speed by deleting the solution in the crowded space. The most crowded antibodies in the POF will be deleted if the number of the selected nondominated antibodies is larger than a given number that we set before the iteration.

3.4 The proposed algorithm

Based on the algorithms above, the ICCoA for DMO can be presented in Algorithm 3, which is designed to improve both the uniformity and the diversity of the solutions in the objective space. ICCoA is different from ICADMO, as the new algorithm adopts the multi population strategy in coevolutionary algorithm and designing corresponding coevolutionary competitive cooperator and coevolutionary cooperative cooperator.

Algorithm 3: Immune Clonal Coevolutionary Algorithm for DMO

Step 1: Initialize antibody populations POP_1 and POP_2 with a size of N randomly, give the serial number of maximum iteration g_{max}, the serial number of initial iterations it:=0, the time step T:=0, the maxmum time step T_{max}.

Step 2: If $T < T_{\max}$, go to **Step 3**; otherwise, stop.

Step 3: Compute objective function value of each antibody and select *POF* of POP_1 and POP_2, denoted as $SMPOP_1$ and $SMPOP_2$ respectively.

Step 4: Implement immune clonal operation on $SMPOP_1$ and $SMPOP_2$ to generate $SMPOP_{1a}$ and $SMPOP_{2a}$ respectively.

Step 5: Implement nonuniform mutation operation on $SMPOP_{1a}$ and $SMPOP_{2a}$.

Step 6: Randomly select k antibodies of $SMPOP_{1a}$ to implement Pareto neighborhood crossover operation to generate a new population $SMPOP_{1b}$; Similarly, generate a new population $SMPOP_{2b}$ of $SMPOP_{2a}$.

Step 7: Select the non-dominated antibodies of each population and restore in $SMPOP_{1b}$ and $SMPOP_{2b}$ respectively with the number of $Nondom_1$ and $Nondom_2$.

Step 8: Use U-measure to test $SMPOP_{1b}$ and $SMPOP_{2b}$ with the measure values Um_1 and Um_2.

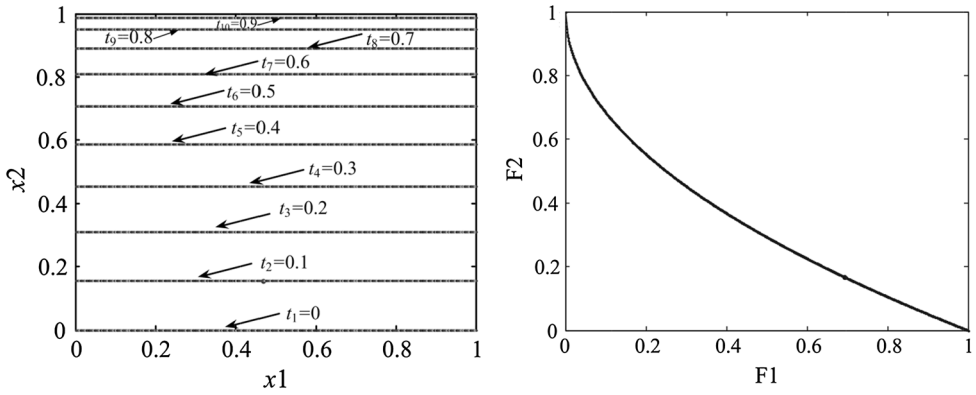
Step 9: If $|Um_1\text{-}Um_2| > \theta$, implement coevolutionary competitive operation to attain $SMPOP_3$ with the number of *Nondom*; Otherwise, implement coevolutionary cooperative operation to attain a new population $SMPOP_3$ with the number of *Nondom.*

Step 10: If $Nondom > N_1$, implement uniformity maintainance operation to make $Nondom = N_1$; Otherwise, store N_1 non-dominated antibodies in $SMPOP_1$.

Step 11: If $it < g_{max}$, it:=it+1, go to **Step3**; otherwise, output the results at time step T, and go to **Step 12.**

Step 12: T:=T+1; it:=0, go to **Step2.**

It can be seen from Algorithm 3, for every time step T, there will be g_{max} iterations for the Pareto-optimal solutions until $T \geq T_{\max}$.



(a) POS(t) for FDA1 at ten time steps. **(b)** POF(t) for FDA1 at different time t.

Fig. 1 POS(t) and POF(t) for FDA1 at different time steps

3.5 Performance metrics of ICCoA

A DMO problem can be seen as many standard multiobjective optimization problems, one at every moment. Generally, the multiobjective optimization problem is evaluated by the uniformity metric and the diversity metric. Therefore, we choose two metrics to test the performance of ICCoA.

3.5.1 Uniformity test: space metric (spacing, S (Van Veldhuizen and Lamont 2000))

The space metric measures the uniformity of Pareto solution distribution and determines how distributed the solutions in Pareto-front are. The smaller the metric is, the more uniform the distribution will be. Formally:

$$S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(\bar{d} - d_i)^2}/\bar{d} \tag{13}$$

where $d_i = \min_j(|f_1^i(\boldsymbol{x}) - f_1^j(\boldsymbol{x})| + |f_2^i(\boldsymbol{x}) - f_2^j(\boldsymbol{x})| + \ldots + |f_M^i(\boldsymbol{x}) - f_M^j(\boldsymbol{x})|)$, $i \neq j$, $j = 1,\ldots, n$, $\bar{d}$ denotes the mean value of d_i and n is the number of the antibodies on Pareto-front.

3.5.2 Diversity test: most spread metric (MS (Goh and Tan 2007))

Most spread metric measures the coverage degree of Pareto-front (PF_{known}) and the real Pareto-front (PF_{true}). The higher the metric is, the larger the coverage area will be, which means the diversity of the solutions in the objective space will be better. Formally:

$$MS = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\frac{\min\left((\overline{\mathrm{PF}_{\mathrm{true}}})_i, (\overline{\mathrm{PF}_{\mathrm{known}}})_i\right) - \max\left((\underline{\mathrm{PF}_{\mathrm{true}}})_i, (\underline{\mathrm{PF}_{\mathrm{known}}})_i\right)}{(\overline{PF_{true}})_i - (\underline{PF_{true}})_i}} \tag{14}$$

where $(\overline{PF_{known}})_i$ and $(\underline{PF_{known}})_i$ is the maximum and the minimum of the i-th objective function on PF_{known} respectively. Similarly, $(\overline{PF_{true}})_i$ and $(\underline{PF_{true}})_i$ is the maximum and the minimum of the i-th objective function on PF_{true} respectively.

4 Discussion of experimental results

4.1 Test problems

ICCoA borrows five test problems from Farina et al. (2004). The DMO Problems are stated as follows:

4.1.1 FDA1

The FDA1 are defined by Eq. (15):

$$\begin{cases} \min F_1(\boldsymbol{x}_{\mathrm{I}}) = x_1 \\ \min F_2(\boldsymbol{x}_{\mathrm{I}}, \boldsymbol{x}_{\mathrm{II}}) = g(\boldsymbol{x}_{\mathrm{II}}) \cdot h(F_1, g) \\ where, g(\mathbf{x}_{\mathrm{II}}) = 1 + \sum_{x_i \in \mathbf{x}_{\mathrm{II}}} (x_i - \mathrm{G(t)})^2 \\ h(F_1, \mathrm{g}) = 1 - \sqrt{\frac{F_1}{g}}, \quad G(t) = \sin(0.5\pi t), \quad t = \frac{1}{n_\tau}\left\lfloor \frac{\tau}{\tau_T} \right\rfloor \end{cases} \tag{15}$$

where $\boldsymbol{x}_{\mathrm{I}} = (x_1) \in [0,1], \boldsymbol{x}_{\mathrm{II}} = (x_2, \cdots, x_n) \in [-1,1], n = 20$, $\tau_T = 5, n_t = 10$.

In FDA1, POS at time t is $x_i = G(t)$ ($i = 2,3, \ldots, n$). $G(t)$ changes with time t, and hence POS will change with time t as well. Figure 1 gives the POS(t) and POF(t) for FDA1 at different time steps.

Figure 1a gives the POS(t) for FDA1, where variations on only the first and the second decision variables are shown for ten time steps when $t_1 = 0$, $t_2 = 0.1$, $t_3 = 0.2$, $t_4 = 0.3$, $t_5 = 0.4$, $t_6 = 0.5$, $t_7 = 0.6$, $t_8 = 0.7$, $t_9 = 0.8$, and $t_{10} = 0.9$. However, in FDA1, the resulting POF does

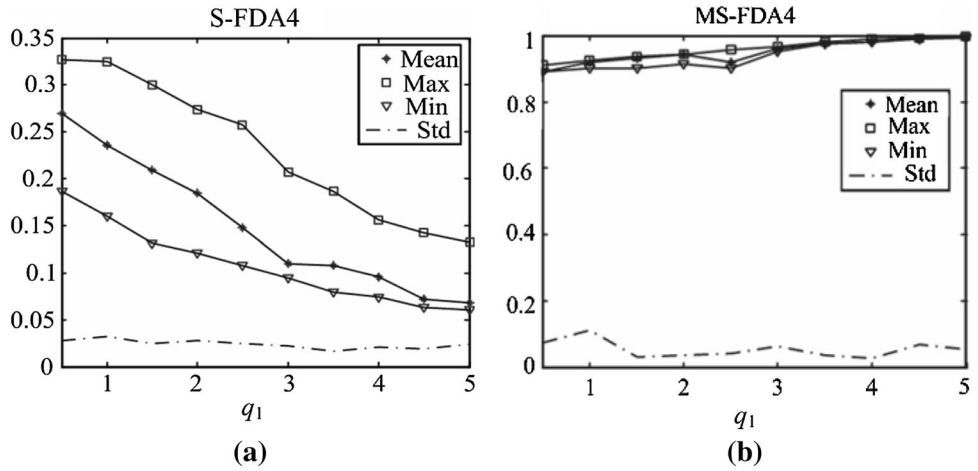


Fig. 2 Influence of the performance for ICCoA with different values of q_1

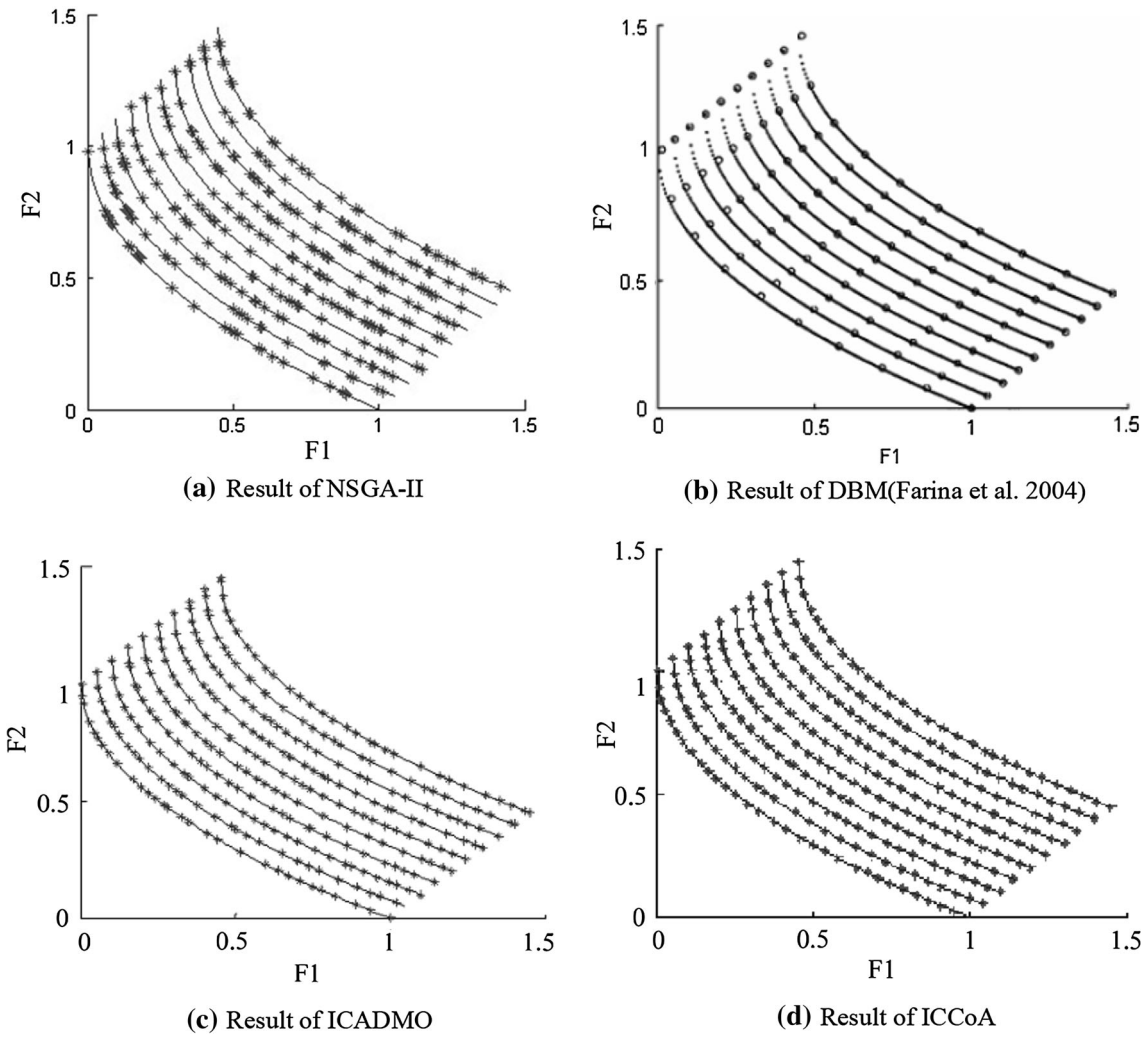
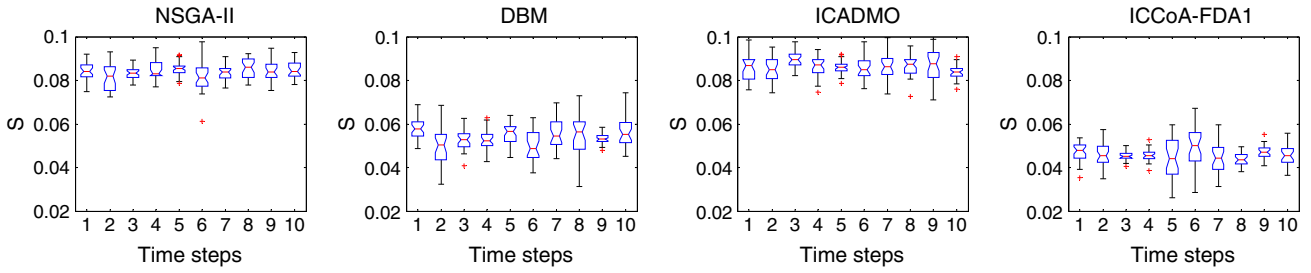
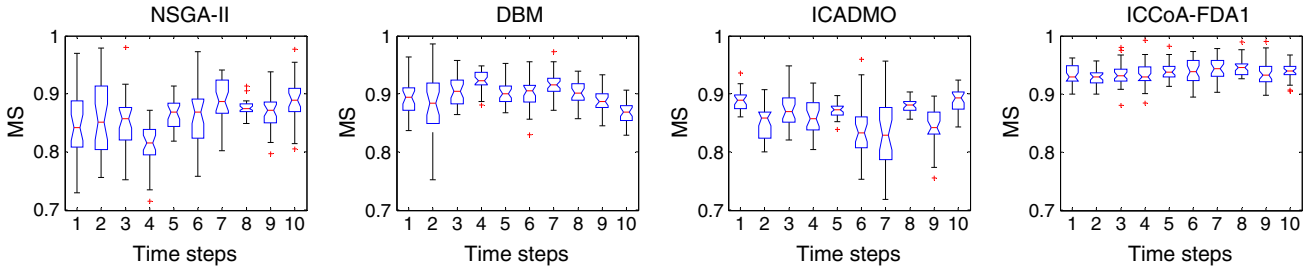


Fig. 3 Experimental results of FDA1



(a) Box plots on the measure S



(b) Box plots on the measure MS

Fig. 4 *Box plots* on the measure S and MS at ten time steps for FDA1

not change and In any time, POF(t) is always $F_2 = 1 - \sqrt{F_1}$. Figure 1b gives the POF(t) for FDA1 and it remains the same.

4.1.2 FDA2

$$\begin{cases} \min F(\boldsymbol{x}_{\mathrm{I}}) = x_1 \\ \min F(\boldsymbol{x}_{\mathrm{I}}, \boldsymbol{x}_{\mathrm{II}}) = g(\boldsymbol{x}_{\mathrm{II}}).h(\boldsymbol{x}_{\mathrm{III}}, F_1, g) \\ where, g(\boldsymbol{x}_{\mathrm{II}}) = 1 + \sum_{x_i \in \mathbf{x}_{\mathrm{II}}} x_i^2 \, h(\boldsymbol{x}_{\mathrm{III}}, F_1, g) \\ = 1 - \left(\frac{F_1}{g}\right)^{(H(t) + \sum_{x_i \in \mathbf{x}_{\mathrm{II}}} (x_i - H(t))^2)} \\ H(t) = 0.75 + 0.7 \sin(0.5\pi t), \\ G(t) = \sin(0.5\pi t), t = \frac{1}{n_\tau} \left\lfloor \frac{\tau}{\tau_T} \right\rfloor \end{cases} . \tag{16}$$

where, $\boldsymbol{x}_{\mathrm{I}} = (x_1) \in [0,1], \boldsymbol{x}_{\mathrm{II}}, \boldsymbol{x}_{\mathrm{III}} \in [-1,1], |\boldsymbol{x}_{\mathrm{II}}| = |\boldsymbol{x}_{\mathrm{III}}| = 15, n = 31, \tau_T = 5, n_t = 10$.

For FDA2, POF swings from a convex shape to a non-convex shape with the change of $H(t)$, while the corresponding POS in the variable space remains unchanged. In any time, POF is $F_2 = 1 - F_1^{H(t)+15*(1+H(t))^2}$.

4.1.3 FDA3

In FDA3, both POS and POF change with time, and the density of the solutions on POF varies with time. In any time, POF is $F_2 = (1 + G(t)) \times (1 - \sqrt{F_1})$.

$$\begin{cases} \min F_1(\boldsymbol{x}_{\mathrm{I}}) = \sum_{x_i \in \mathbf{x}_{\mathrm{I}}} x_i^{F(t)} \\ \min F_2(\boldsymbol{x}_{\mathrm{I}}, \boldsymbol{x}_{\mathrm{II}}) = g(\mathbf{x}_{\mathrm{II}}) \cdot h(F_1, g) \\ \text{where}, g(\boldsymbol{x}_{\mathrm{II}}) = 1 + G(x) + \sum_{x_i \in \mathbf{x}_{\mathrm{II}}} (x_i - G(t))^2 \\ h(F_1, g) = 1 - \sqrt{\frac{F_1}{g}}, G(t) = \sin(0.5\pi t), \\ F(t) = 10^{2\sin(0.5\pi t)}, t = \frac{1}{n_\tau} \left\lfloor \frac{\tau}{\tau_T} \right\rfloor \end{cases} \tag{17}$$

where, $\boldsymbol{x}_{\mathrm{I}} \in [0,1], \boldsymbol{x}_{\mathrm{II}} \in [-1,1], n = 31, \tau_T = 5, n_t = 10, |\boldsymbol{x}_{\mathrm{I}}| = 5, |\boldsymbol{x}_{\mathrm{II}}| = 25$.

4.1.4 FDA4

In this problem, POF is three-dimensional. POS changes with time, while POF maintains unchanged. In any time, POF is $\sum_{i=1}^{\mathrm{M}} \left(\mathrm{F}_i^*\right) = 1 + G(t)$.

$$\begin{cases} \min\limits_x F_1(\boldsymbol{x}) = (1 + g(\boldsymbol{x}_{\mathrm{II}})) \prod_{i=1}^{M-1} \cos\left(\frac{x_i \pi}{2}\right) \min\limits_x F_2(\boldsymbol{x}) \\ = (1 + g(\mathbf{x}_{\mathrm{II}})) \left(\prod_{i=1}^{M-2} \cos\left(\frac{x_i \pi}{2}\right)\right) \sin(\frac{x_{M-1}\pi}{2}) \min\limits_x F_M(\boldsymbol{x}) \\ = (1 + g(\mathbf{x}_{\mathrm{II}})) \sin\left(\frac{x_1 \pi}{2}\right) where, g(\mathbf{x}_{\mathrm{II}}) \\ = \sum_{x_i \in \mathbf{x}_{\mathrm{II}}} (x_i - G(t))^2, k = 2 : M - 1 \\ G(t) = |\sin(0.5\pi t)|, t = \frac{1}{n_\tau} \left\lfloor \frac{\tau}{\tau_T} \right\rfloor \end{cases} \tag{18}$$

Table 1 The *t*-test results of ICCoA with the compared algorithm on S and MS for FDA1 at ten time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	1.5982e−023	1	4.3230e−022	1	2.9506e−010	1	4.5098e−008	1	2.1198e−010	1	1.0880e−007	1
t_2	9.5159e−023	1	7.5250e−023	1	0.0167	1	5.8244e−006	1	7.6999e−014	1	1.3667e−004	1
t_3	2.2250e−030	1	5.1674e−030	1	1.1197e−008	1	1.9739e−009	1	2.1278e−009	1	1.6656e−004	1
t_4	1.5818e−026	1	6.1446e−025	1	9.9005e−008	1	1.4330e−013	1	3.1998e−013	1	0.0480	1
t_5	8.4783e−022	1	1.2068e−020	1	3.3391e−006	1	3.2695e−012	1	2.5556e−018	1	2.2760e−008	1
t_6	1.2477e−015	1	2.9130e−018	1	0.5305	0	1.6306e−009	1	4.8897e−011	1	4.8474e−008	1
t_7	7.3299e−023	1	9.9778e−021	1	4.5930e−006	1	4.6614e−008	1	2.4251e−011	1	2.8840e−006	1
t_8	1.9463e−027	1	1.7157e−029	1	2.6488e−006	1	5.1846e−018	1	9.4233e−017	1	2.7733e−010	1
t_9	1.7437e−024	1	4.8559e−023	1	4.2423e−009	1	2.8887e−009	1	1.4121e−011	1	5.9719e−009	1
t_{10}	2.4410e−023	1	1.9374e−025	1	1.7258e−007	1	3.2005e−008	1	9.3734e−012	1	5.2933e−017	1

where $\boldsymbol{x}_{\mathrm{II}} = (x_M, \cdots, x_n), x_i \in [0,1](i = 1:n), n = 12$, $\tau_T = 5, n_t = 10, |\boldsymbol{x}_{\mathrm{II}}| = 10$.

4.1.5 FDA5

For FDA5, POF is three-dimensional. Both POS and POF change with time, and the density of the solutions on POF varies with time. In any time, POF is $\sum_{i=1}^{M}\left(F_i^*\right)^2 = 1 + G(t)$.

$$\begin{cases} \min\limits_x F_1(\boldsymbol{x}) = (1+g(\mathbf{x}_{\mathrm{II}}))\prod\limits_{i=1}^{M-1}\cos\left(\frac{y_i\pi}{2}\right)\min\limits_x F_2(\boldsymbol{x}) \\ = (1+g(\mathbf{x}_{\mathrm{II}}))\left(\prod\limits_{i=1}^{M-2}\cos\left(\frac{y_i\pi}{2}\right)\right)\sin\left(\frac{y_1\pi}{2}\right)\cdots\cdots\min\limits_x F_M(\mathbf{x}) \\ = (1+g(\mathbf{x}_{\mathrm{II}}))\sin\left(\frac{y_1\pi}{2}\right) \\ where, g(\mathbf{x}_{\mathrm{II}}) = G(t) + \sum\limits_{x_i\in\mathbf{x}_{\mathrm{II}}}(x_i - G(t))^2; \\ G(t) = |\sin(0.5\pi t)|; F(t) = 1 + 100\sin^4(0.5\pi t) \\ y_i = x_i^{F(t)}, for \quad i = 1:M-1; t = \frac{1}{n_\tau}\left\lfloor\frac{\tau}{\tau_T}\right\rfloor \end{cases} \tag{19}$$

where $\boldsymbol{x}_{\mathrm{II}} = (x_M, \cdots, x_n), x_i \in [0,1](i = 1:n), n = 12$, $\tau_T = 5, n_t = 10, |\boldsymbol{x}_{\mathrm{II}}| = 10$.

4.2 Parameter settings

On all test problems, the parameters of ICCoA are as follows: the size of antibody population N, the iteration number g_{max}, the size of nondominated antibody population N_{non}, the expected number of nondominated antibody population N_1 and the proportion of cloning q_1. In every test DMO problem, we give a fix maximum runtime T. The influence of N, g_{max}, and N_1 to the performance of the algorithm is very clear, and if we don't take the complexity of the algorithm into consideration, the greater the values of these parameters are, the better the results of the algorithm will be. In general, the larger the N is, the greater the value of N_{non} will be. Taking the effectiveness and the complexity of the proposed algorithm, we set $N = 300$, at the first time step, $g_{max} = 150$ and at other time steps, $g_{max} = 100$. The size of nondominated antibody population N_{non}: the value is related to the test problem and the size of antibody population. The expected number of nondominated antibody population N_1: the value is related to the test problem. The influences of the parameters q_1 and p_m to the algorithm's performance are more complex. In ICCoA, Let the Inconsistent mutation probability: $p_m = 1/n$ (n is the dimension of the decision variable vector space).

The influence of the clonal proportion q_1 to the algorithm is quite complex. To illustrate the effect of q_1 with different values, we take FDA4 as an example by empirical analysis.

Set $N = 300$, stop iteration number $g_{max} = 100$ and inconsistent mutation probability $p_m = 1/n$. The clonal proportion q_1 is equal-length sampled from 0.5 to 5 with an interval of 0.5. With running independently for 20 times, the maximum value, the minimum value, the mean value and the variance of the performance metrics are shown in Fig. 2.

It can be seen from Fig. 2a that the change of metric S is very clear with the variation of q_1. The result indicates that the spread of the solutions on Pareto-fronts is more and more uniform with the enlargement of q_1.

Figure 2b shows the change of metric MS with different value of q_1. It is indicated that the change is not quite obvious, which means that with the amplification of q_1, the spread of solutions is more and more extensive but with little change.

Table 2 The Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA1 at ten time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	1.7344e−006	1	1.7344e−006	1	2.3534e−006	1	1.0246e−005	1	2.3534e−006	1	1.2381e−005	1
t_2	1.7344e−006	1	1.7344e−006	1	0.0218	1	1.0570e−004	1	1.7344e−006	1	3.3173e−004	1
t_3	1.7344e−006	1	1.7344e−006	1	4.7292e−006	1	2.1266e−006	1	5.7517e−006	1	4.1955e−004	1
t_4	1.7344e−006	1	1.7344e−006	1	6.3391e−006	1	1.7344e−006	1	1.7344e−006	1	0.0598	0
t_5	1.7344e−006	1	1.7344e−006	1	3.7243e−005	1	1.7344e−006	1	1.7344e−006	1	8.4661e−006	1
t_6	1.7344e−006	1	1.7344e−006	1	0.5999	0	2.8786e−006	1	2.6033e−006	1	4.2857e−006	1
t_7	1.7344e−006	1	1.7344e−006	1	5.3070e−005	1	6.9838e−006	1	1.9209e−006	1	5.3070e−005	1
t_8	1.7344e−006	1	1.7344e−006	1	5.3070e−005	1	1.7344e−006	1	1.7344e−006	1	1.9209e−006	1
t_9	1.7344e−006	1	1.7344e−006	1	6.3391e−006	1	2.3534e−006	1	1.7344e−006	1	3.1817e−006	1
t_{10}	1.7344e−006	1	1.7344e−006	1	3.5152e−006	1	5.2165e−006	1	2.1266e−006	1	1.7344e−006	1

As a result of the factors above, the clonal proportion q_1 of the algorithm is set as 5 in this paper.

4.3 Experimental results and comparisons

In this section, the simulation results of NSGA-II, DBM, ICADMO and ICCoA on five test problems are given. In all the problems, the POF of NSGA-II, ICADMO and ICCoA are represented as '*', while the POF of DBM is represented as 'o', which are borrowed from the original reference (Farina et al. 2004).

Moreover, we will compare following performance of ICCoA and ICADMO:

(1) The uniformity of Pareto solution distribution;
(2) The diversity of Pareto solutions.

where (1) indicates the uniformity of the algorithms, and (2) illustrates the diversity of POF. The uniformity and diversity are both the important metrics to measure a multi-objective optimization algorithm. In this paper, the simulation box plots (Chambers et al. 1983) on the measure S and MS on 30 independent runs for ICCoA and ICADMO are given.

4.3.1 FDA1

POF of FDA1 is convex. The POFs of NSGA-II, DBM, ICADMO and ICCoA are illustrated in Fig. 3.

Figure 3 shows the obtained solutions of four algorithms at ten steps when $t_1 = 0$, $t_2 = 0.1$, $t_3 = 0.2$, $t_4 = 0.3$, $t_5 = 0.4$, $t_6 = 0.5$, $t_7 = 0.6$, $t_8 = 0.7$, $t_9 = 0.8$, and $t_{10} = 0.9$ from the lower-left to the upper-right respectively. Figure 3a shows that NSGA-II cannot maintain a good uniformity and diversity of solutions on the Pareto-optimal front. Figure 3b indicates that when F2 is approaching 1, DBM fails to find the POF, thus, it has difficulty in preserving diversity. Figure 3c shows that the POF of ICADMO has better diversity than DBM, but it is not uniform. However, the results depicted in Fig. 3d prove that ICCoA does better than DBM and ICADMO at each time step. It reaches a better distribution in the objective space and it is able to maintain uniformity. Therefore, ICCoA is the best to keep the diversity and uniformity of the solutions in the objective space for FDA1.

The box plots on the measure S and MS on FDA1 for four algorithms are illustrated in Fig. 4.

As the box plots in Fig. 4a show, the upper quartile, the median and the lower quartile of the S values for ICCoA are all smaller than the corresponding values of the compared algorithms, which indicates that for FDA1, ICCoA is able to preserve the uniformity of the solutions. Figure 4b shows the box plots on the measure MS for FDA1. Although the results on FDA1 of ICCoA are similar with DBM and ICADMO at ten time steps, the median and the lower quartile of the MS values for ICCoA are all greater than the corresponding values of the other three algorithms. Thus, we can figure that ICCoA improves the diversity of FDA1.

For a thorough comparison the *t* test (Sun et al. 2011) and the Wilcoxon signed rank test (Gibbons 1985) have also been carried out. For the *t*-test, function "$[h, p] = t$-test(x, y)" in matlab statistic toolbox is used, which performs a paired *t*-test of the hypothesis that two matched (or paired) samples in the vectors x and y come from distributions with equal means" (Gibbons 1985). If there is a significant difference, $h = 1$; otherwise, $h = 0$. And p is the p-value, which shows the significance level. For the Wilcoxon signed rank test, h is the result of the test, and p is the probability of a hypothesis of equal median for two paired samples. If the median of the difference between

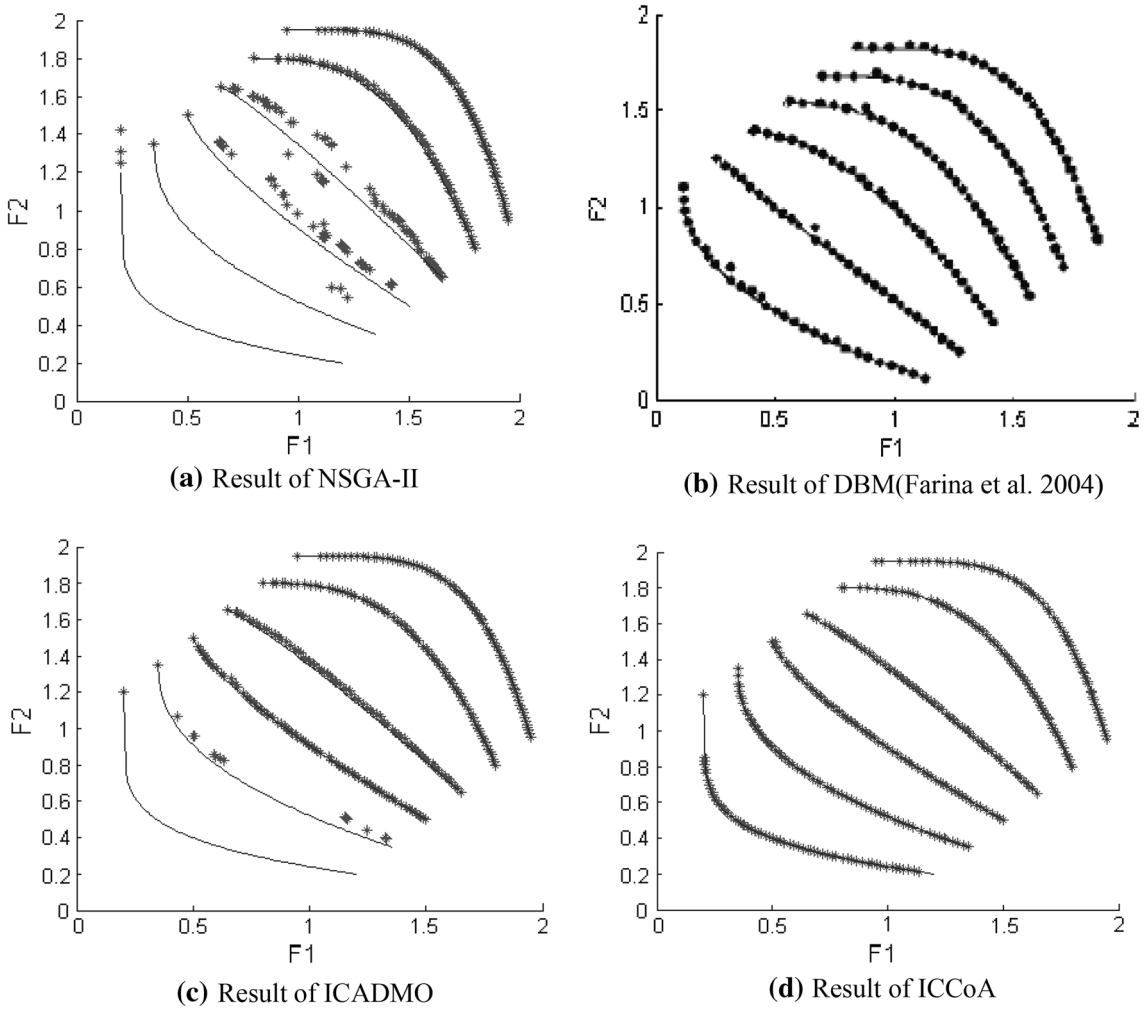


Fig. 5 Experimental results of FDA2

ICCoA and another compared algorithm is zero, $h = 0$; Otherwise, there is a significant difference, then $h = 1$ (Gibbons 1985).

Table 1 shows the *t*-test results of ICCoA with the compared algorithm on S and MS for FDA1 at ten time steps over 30 independent runs.

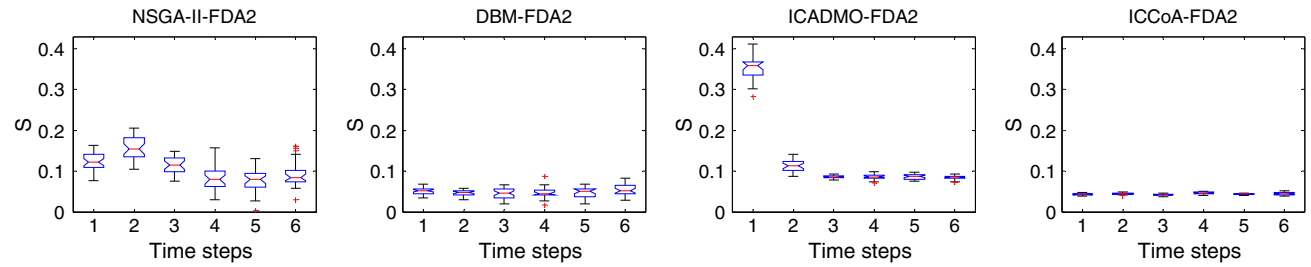
It can be seen from Table 1 that, for test problem FDA1, at most of the time steps, for the metric S, the differences between ICCoA and the compared algorithms are significant, except for compared with algorithm DBM at time step t_6, which means the solutions obtained by ICCoA have a significant improvement on the uniform distribution at most of the time steps. At all the time steps, for the metric MS, the differences between ICCoA and the compared algorithms are significant, which indicates the solutions obtained by ICCoA can obtain a significant improvement in covering the large area in the Pareto-front and in maintaining the best diversity.

Table 2 gives the Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA1 at ten time steps over 30 independent runs.

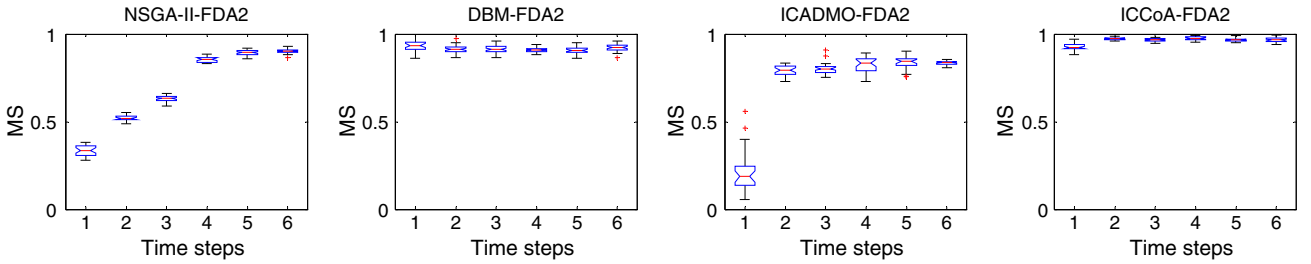
As for the Wilcoxon signed rank test, it can be seen from Table 2 that, for test problem FDA1, at most of the time steps, for the metric S, ICCoA gets significant differences compared with the other three algorithms, except for compared with algorithm DBM at time step t_6. For the metric S, ICCoA gets significant differences compared with the other three algorithms, except for compared with algorithm DBM at time step t_4. The Wilcoxon signed rank test results indicate that, at most of the time steps, ICCoA constitutes a significant improvement on both uniformity and diversity of POF for FDA1.

4.3.2 FDA2

POF of FDA2 swings from a convex shape to a non-convex shape. The POFs at six time steps when $t_1 = 3.4$, $t_2 = 3.7$,



(a) Box plots on the measure S



(b) Box plots on the measure MS

Fig. 6 *Box plots* on the measure S and MS at six time steps for FDA2

Table 3 The t-test results of ICCoA with the compared algorithm on S and MS for FDA2 at six time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	2.3641e−019	1	8.4037e−031	1	7.3556e−005	1	1.1450e−037	1	2.4082e−024	1	0.3891	0
t_2	8.3634e−020	1	3.6193e−022	1	0.0783	0	8.7419e−042	1	1.5094e−024	1	3.7287e−014	1
t_3	4.1280e−019	1	3.5640e−033	1	0.1251	0	6.1063e−040	1	1.0165e−021	1	1.3753e−013	1
t_4	6.6070e−008	1	1.2019e−023	1	0.6151	0	7.4292e−025	1	1.4679e−017	1	1.2702e−018	1
t_5	2.4520e−006	1	9.0506e−025	1	0.1426	0	7.8866e−018	1	3.0942e−019	1	1.2063e−014	1
t_6	1.4869e−008	1	1.1109e−025	1	0.0032	1	1.6534e−017	1	1.8579e−028	1	3.1349e−013	1

$t_3 = 4$, $t_4 = 4.3$, $t_5 = 4.6$, and $t_6 = 4.9$ from the lower-left to the upper-right of NSGA-II, DBM, ICADMO and ICCoA are illustrated in Fig. 5.

Figure 5a shows that NSGA-II fails to find its POFs at time steps: $t_1 = 3.4$ and $t_2 = 3.7$ and the solutions at $t_3 = 4$ and $t_4 = 4.3$ are not uniformly distributed and cannot converge to the true Pareto-front. From Fig. 5b, we can see that DBM preserves good diversity, while at every time step, it is unable to preserve the convergence, and moreover, it tends to have difficulties in preserving uniformity.

Figure 5c shows that ICADMO has better convergence, but at the first time step $t_1 = 3.4$ and the second time step $t_2 = 3.7$, it appears to fail to maintain uniformity and diversity. At other time steps, the POF of ICADMO also has better distribution. However, ICCoA behaves very well at every time step for the POF uniformly distribution and reaches a broader spread. Therefore, ICCoA is better than NSGA-II, DBM and ICADMO to preserve the diversity and uniformity of Pareto optimal solutions for FDA2.

The box plots on the measure S and MS on FDA2 for four algorithms are illustrated in Fig. 6.

As the box plots depicted in Fig. 6a, the upper quartile, the median and the lower quartile of the S values for ICCoA are all smaller than the corresponding values of the compared algorithm, and ICADMO is incapable of efficient distribution on any time, thus ICCoA is able to preserve the uniformity of the solutions in the objective space for FDA2. Figure 6b shows the box plots on the measure MS for FDA2. Although the results on FDA1 of ICCoA are similar with DBM at six time steps, except for time step 1 obtained by DBM, the median of the MS values for ICCoA

Table 4 The Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA2 at six time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	1.7344e−006	1	1.7344e−006	1	2.8308e−004	1	1.7344e−006	1	1.7344e−006	1	0.3933	0
t_2	1.7344e−006	1	1.7344e−006	1	0.0495	1	1.7344e−006	1	1.7344e−006	1	1.9209e−006	1
t_3	1.7344e−006	1	1.7344e−006	1	0.1779	0	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1
t_4	5.7517e−006	1	1.7344e−006	1	0.6435	0	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1
t_5	3.7243e−005	1	1.7344e−006	1	0.1254	0	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1
t_6	2.3534e−006	1	1.7344e−006	1	0.0082	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1

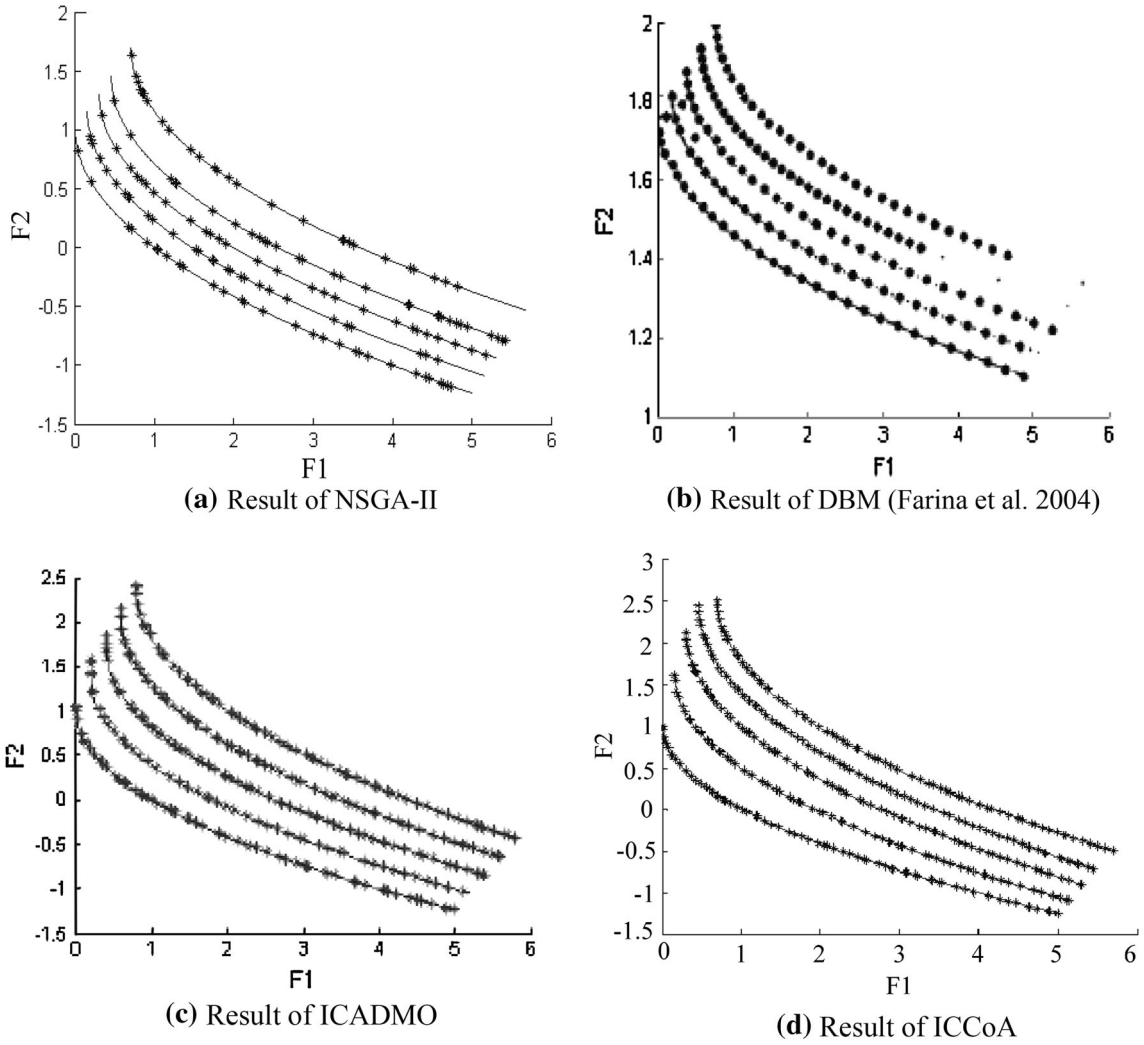
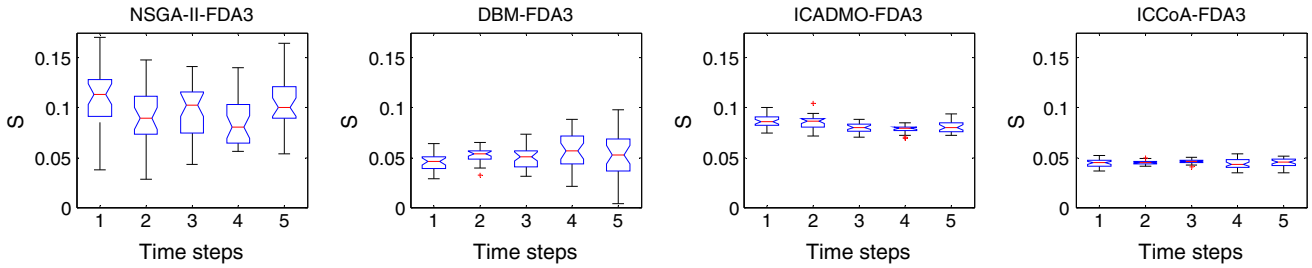


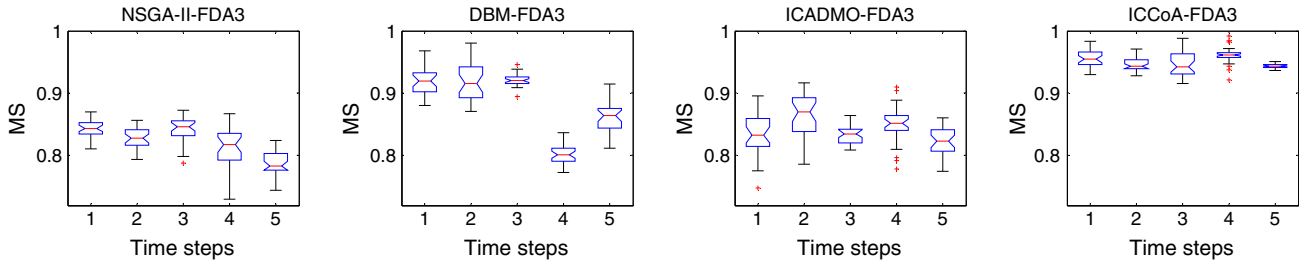
Fig. 7 Experimental results of FDA3

are greater than the corresponding values of the other three algorithms. So ICCoA has better diversity than the compared algorithm at most of the time steps.

The *t*-test results of ICCoA with the compared algorithm on S and MS for FDA2 at six time steps over 30 independent runs are shown in Table 3.



(a) Box plots on the measure S



(b) Box plots on the measure MS

Fig. 8 *Box plots* on the measure S and MS at five time steps for FDA3

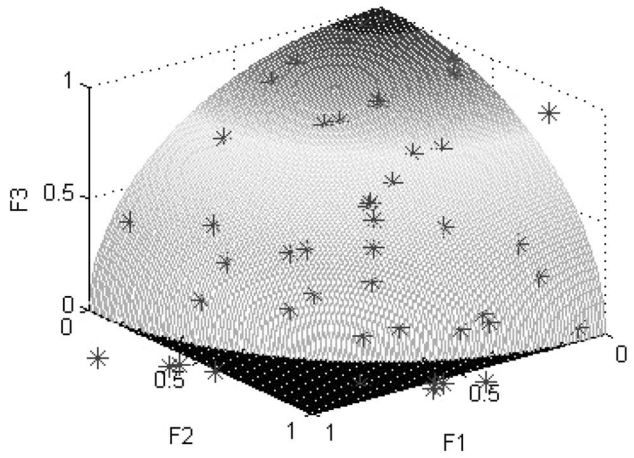
Table 5 The t-test results of ICCoA with the compared algorithm on S and MS for FDA3 at five time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	1.3992e−012	1	5.9054e−024	1	0.3128	0	1.5064e−023	1	8.4560e−018	1	2.3246e−010	1
t_2	1.7674e−009	1	3.9860e−024	1	1.0385e−006	1	4.7750e−025	1	5.9870e−014	1	7.2959e−005	1
t_3	4.2761e−011	1	2.3749e−024	1	0.0325	1	4.4086e−019	1	1.9405e−021	1	3.3764e−007	1
t_4	1.1115e−010	1	5.3013e−024	1	2.9464e−004	1	1.3910e−020	1	2.3268e−017	1	5.8991e−028	1
t_5	1.1891e−013	1	6.9741e−023	1	0.0451	1	4.7823e−027	1	2.0735e−023	1	3.1287e−017	1

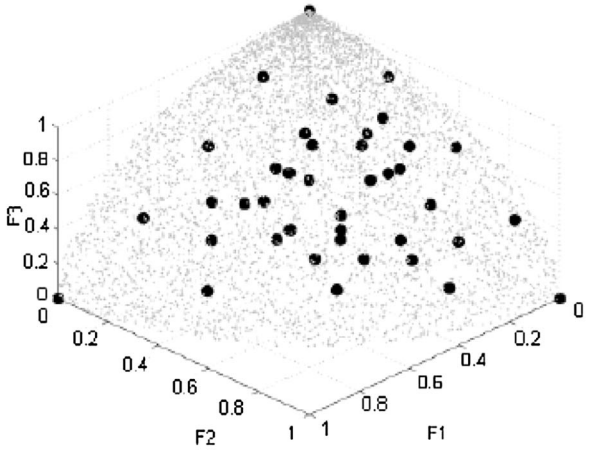
Table 6 The Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA3 at five time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	1.9209e−006	1	1.7344e−006	1	0.4165	0	1.7344e−006	1	1.7344e−006	1	2.1266e−006	1
t_2	3.1817e−006	1	1.7344e−006	1	4.0715e−005	1	1.7344e−006	1	1.7344e−006	1	2.4118e−004	1
t_3	2.1266e−006	1	1.7344e−006	1	0.0316	1	1.7344e−006	1	1.7344e−006	1	1.9729e−005	1
t_4	1.7344e−006	1	1.7344e−006	1	0.0012	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1
t_5	1.7344e−006	1	1.7344e−006	1	0.0270	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1

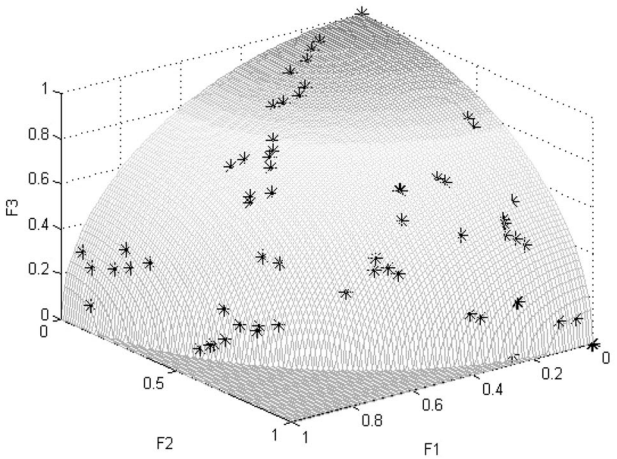
It can be seen from Table 3 that, for test problem FDA2, for metric S, compared with algorithm DBM, ICCoA fails to get significant differences at time step t_2, t_3, t_4 and t_5 for metric S and at time step t_1 for metric MS, which indicates that at these time steps ICCoA can not get the significant improvement on the uniform distribution or the coverage for Pareto-front of the solutions. However, at all of the time steps, for the metrics S and MS, the differences between ICCoA and NSGA-II, ICADMO algorithms are significant, which indicates that the solutions obtained by ICCoA have a significant improvement on uniform distribution and coverage area in the Pareto-front.



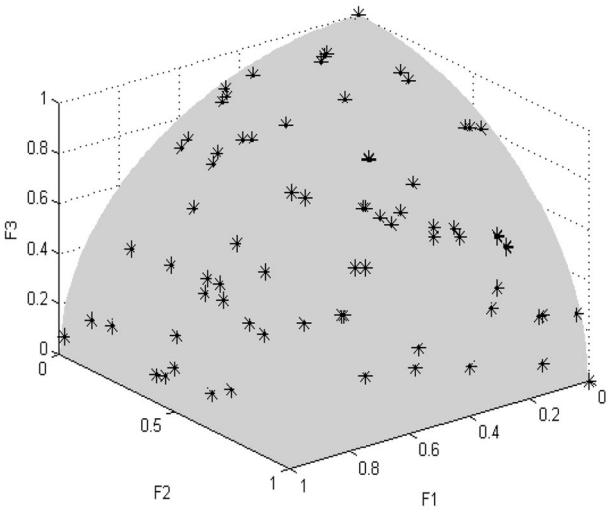
(a-1) Result of NSGA-II (t_1=0)



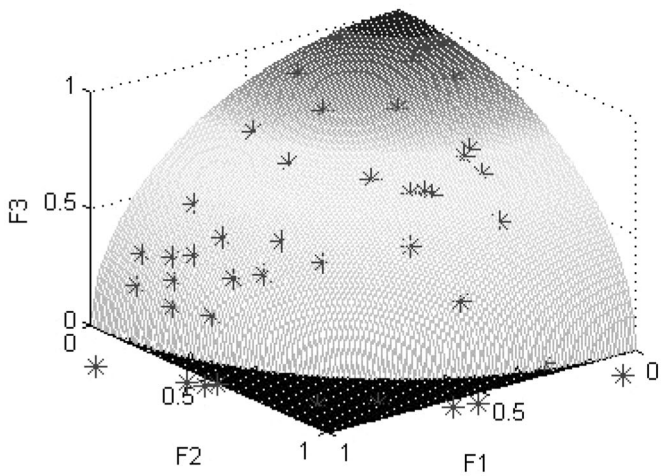
(b-1) Result of DBM (t_1=0) (Farina et al. 2004)



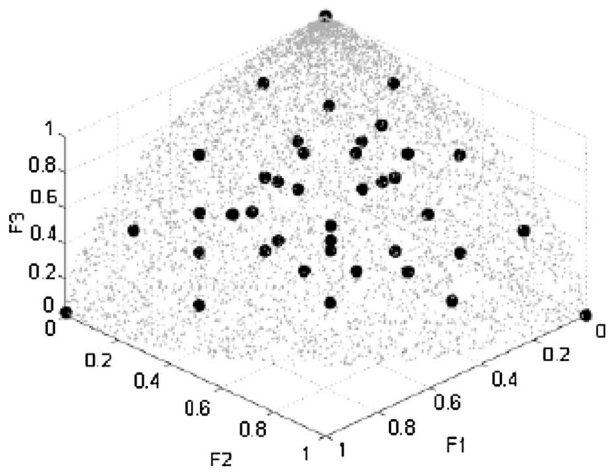
(c-1) Result of ICADMO (t_1=0)



(d-1) Result of ICCoA (t_1=0)

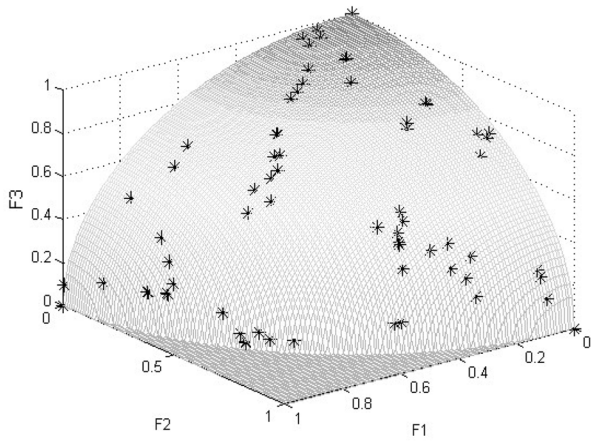


(a-2) Result of NSGA-II (t_2=0.5)

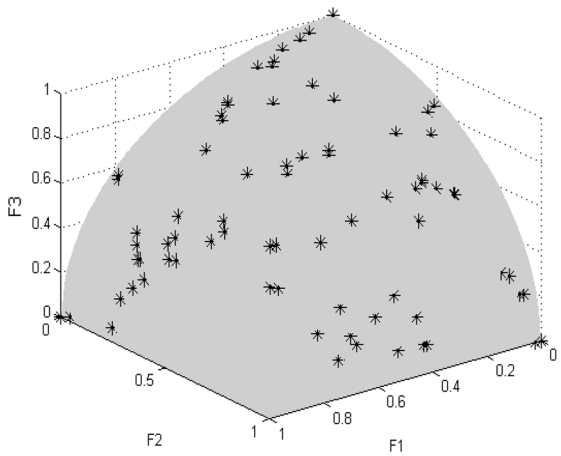


(b-2) Result of DBM (t_2=0.5) (Farina et al. 2004)

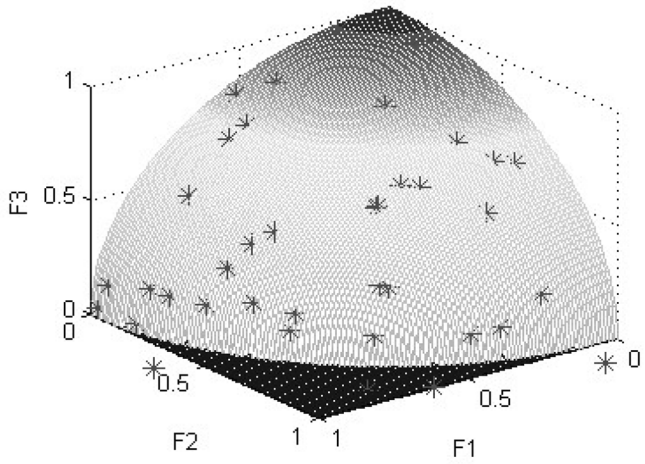
Fig. 9 Experimental results of FDA4



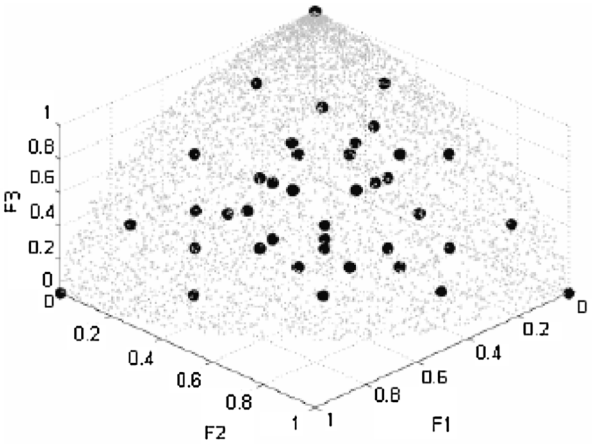
(c-2) Result of ICADMO (t_2=0.5)



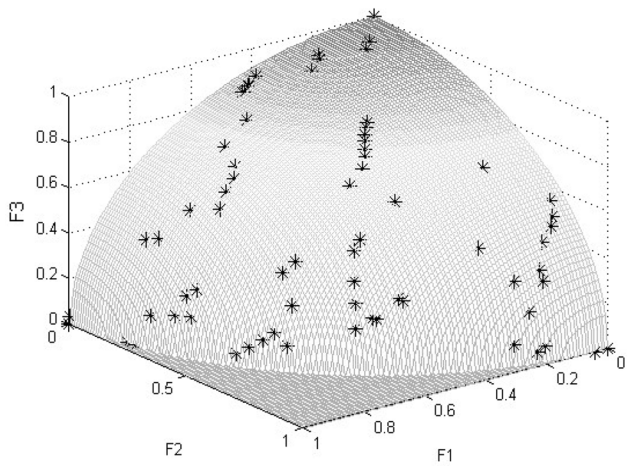
(d-2) Result of ICCoA (t_2=0.5)



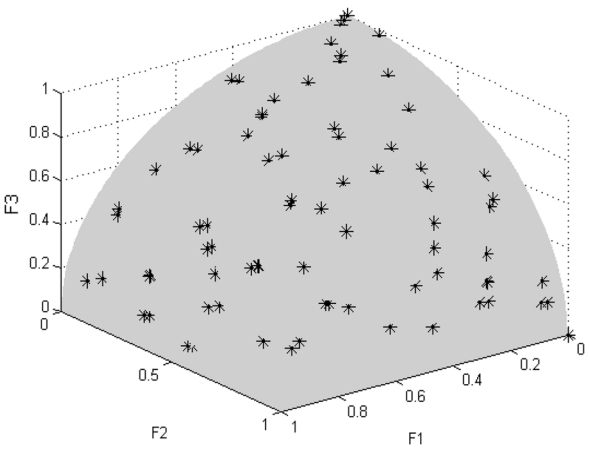
(a-3) Result of NSGA-II (t_3=1)



(b-3) Result of DBM (t_3=1) (Farina et al. 2004)

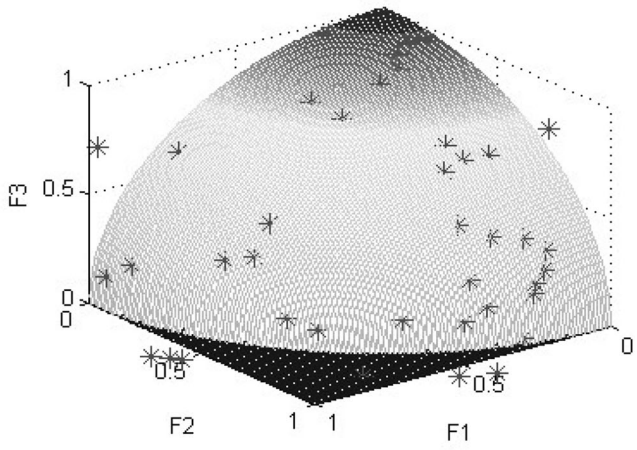


(c-3) Result of ICADMO (t_3=1)

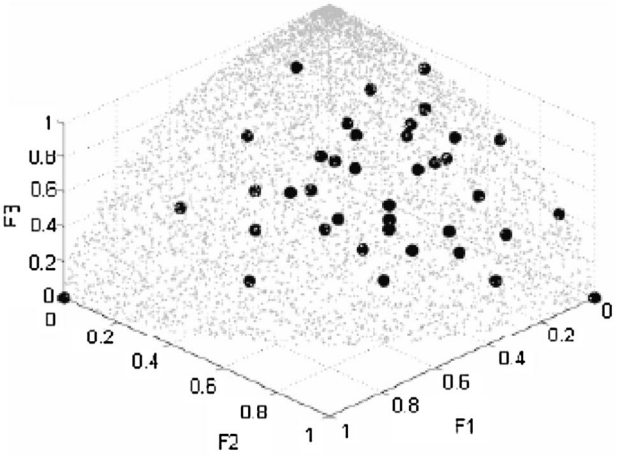


(d-3) Result of ICCoA (t_3=1)

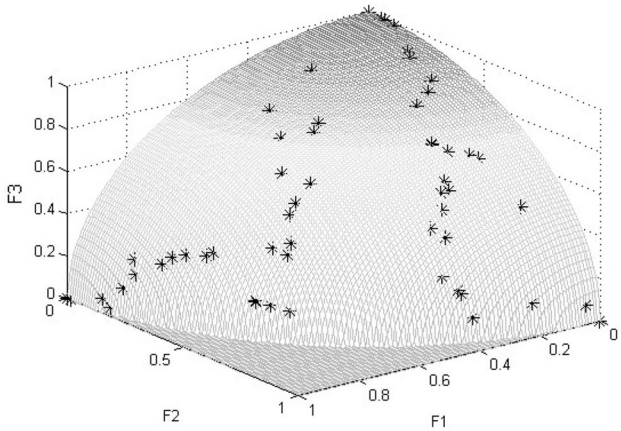
Fig. 9 continued



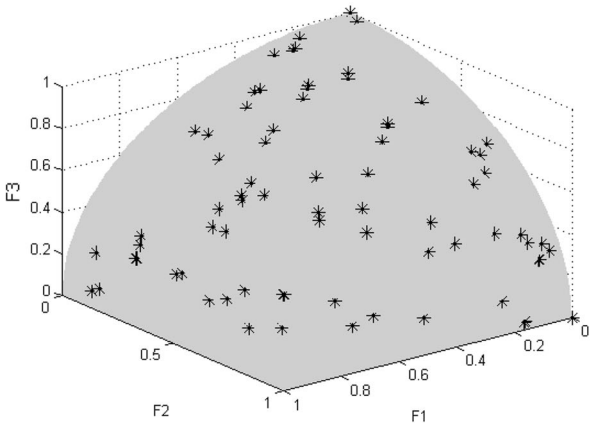
(a-4) Result of NSGA-II (t_4=1.5)



(b-4) Result of DBM (t_4=1.5) (Farina et al. 2004)



(c-4) Result of ICADMO (t_4=1.5)



(d-4) Result of ICCoA (t_4=1.5)

Fig. 9 continued

Table 4 gives the Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA2 at six time steps over 30 independent runs.

Table 4 shows that compared with NSGA-II and ICADMO, at all the time steps, for both S and MS, ICCoA gets significant differences. The median differences between ICCoA and DBM are significant at time steps t_1, t_2 and t_6 for metric S and at time steps t_2, t_3, t_4, t_5 and t_6 for metric MS. The statistic results show that at most of the time steps, ICCoA gets significant improvement of the obtained solutions for test problem FDA2.

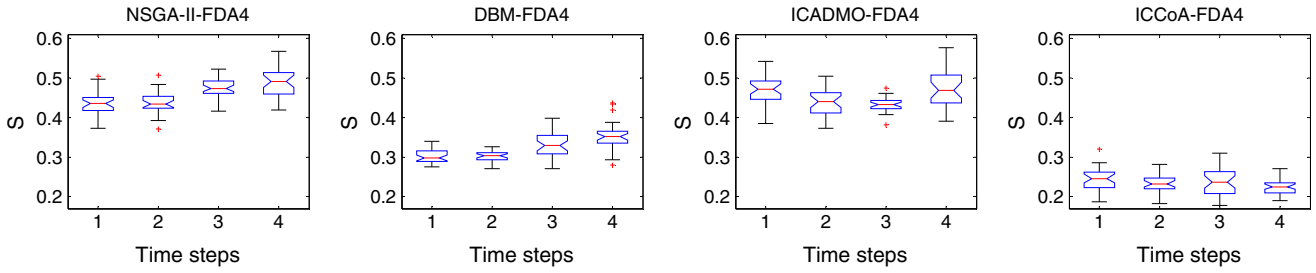
4.3.3 FDA3

POF of FDA3 changes over time. The POFs at five time steps when $t_1 = 0.1$, $t_2 = 0.4$, $t_3 = 0.7$, $t_4 = 1$, $t_5 = 1.4$, from the lower-left to the upper-right of NSGA-II, DBM, ICADMO and ICCoA are illustrated in Fig. 7.

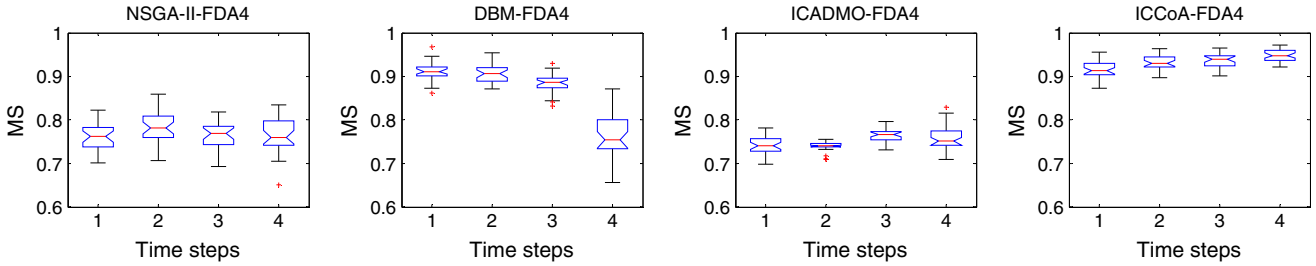
Figure 7a shows that NSGA-II has difficulties in preserving the uniformity and the diversity of the solutions. From Fig. 7b, it can be seen that DBM fails to find a widespread POF at the fourth and the fifth time steps. While in Fig. 7c, ICADMO has improved the diversity a lot, but it does not guarantee the uniformity. Figure 7d shows that ICCoA does better at any time step for it reaches both better distribution and uniformity. Therefore, ICCoA is the best to keep the diversity and uniformity of the solutions in the objective space on FDA3.

The box plots on the measure S and MS on FDA3 for four algorithms are illustrated in Fig. 8.

Figure 8a shows that at most of the time steps, the median of S values obtained by ICCoA are all much smaller than the corresponding values of the compared algorithms except for time step t_1 of DBM, which shows that for FDA3, ICCoA is able to preserve the solutions uniform at most of the time steps. The box plots on the



(a) Box plots on the measure S



(b) Box plots on the measure MS

Fig. 10 *Box plots* on the measure S and MS at four time steps for FDA4

Table 7 The t-test results of ICCoA with the compared algorithm on S and MS for FDA4 at four time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	2.1765e−020	1	5.8597e−024	1	5.5940e−011	1	2.1631e−019	1	3.6343e−023	1	0.4239	0
t_2	8.1604e−023	1	1.9208e−023	1	1.0176e−013	1	4.1203e−018	1	2.6357e−032	1	2.7505e−006	1
t_3	1.7824e−023	1	8.0186e−023	1	3.5885e−012	1	9.6886e−022	1	1.1240e−026	1	1.0543e−009	1
t_4	8.7315e−027	1	3.3989e−022	1	1.3067e−017	1	7.0440e−020	1	2.1008e−024	1	1.6234e−019	1

measure MS for FDA3 are depicted in Fig. 8b. The values at five time steps of ICCoA are all greater than the corresponding values of the compared algorithm. Thus, ICCoA has improved the diversity of FDA3 very well.

Table 5 gives the t-test results of ICCoA with the compared algorithm on S and MS for FDA3 at five time steps over 30 independent runs.

It can be seen from Table 5 that, for test problem FDA3, for metric S, compared with the other algorithms, at most of the time steps, ICCoA can get significant differences except for at time step t_1 for DBM, which indicates that at most of the time steps ICCoA can get the significant improvement on the uniform distribution of the obtained solutions. Moreover, at all of the time steps, for metric MS, the differences between ICCoA and the compared algorithms are significant, which indicates that the solutions obtained by ICCoA have a significant improvement on the coverage area in the Pareto-front.

Table 6 gives the Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA3 at five time steps over 30 independent runs.

It can be seen from Table 6 that, the results of the Wilcoxon signed rank test also demonstrates the advantage of ICCoA. ICCoA is significantly better at most of the time steps for test problem FDA3 except at time step t_1 for DBM, which shows that at most of the time steps, ICCoA achieves significantly improvement on both uniformity and diversity of the obtained solutions.

4.3.4 FDA4

The POF of FDA4 remains unchanged. The POFs at four time steps when $t_1 = 0$, $t_2 = 0.5$, $t_3 = 1$, $t_4 = 1.5$ of NSGA-II, DBM, ICADMO and ICCoA on FDA4 are illustrated in Fig. 9.

Table 8 The Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA4 at four time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	*p*	h	*p*	h	*p*	h	*p*	h	*p*	h
t_1	1.7344e−006	1	1.7344e−006	1	1.9209e−006	1	1.7344e−006	1	1.7344e−006	1	0.2536	0
t_2	1.7344e−006	1	1.7344e−006	1	1.9209e−006	1	1.7344e−006	1	1.7344e−006	1	2.1630e−005	1
t_3	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	1.9209e−006	1
t_4	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1

It can be seen from Fig. 9 that for FDA4, in spite of the changing time, all of the four algorithms have found the same spherical surface with a radius of 1. From Fig. 9 a-1 to a-4, b-1 to b-4, c-1 to c-4, we can see that ICADMO has broader spread than NSGA-II and DBM, which means it improves the diversity. Nevertheless, ICADMO is unable to maintain the uniformity. From the test results in Fig. 9d-1 to d-4, it has shown that ICCoA achieves uniform distribution and better diversity at every time step.

The box plots on the measure S and MS on FDA4 for four algorithms are illustrated in Fig. 10. From the results of the box plots in Fig. 10a, it can be seen that the results on FDA4 of ICCoA are much smaller than the compared algorithms at fou time steps. The upper quartile, the median and the lower quartile of the S values for ICCoA are all smaller than the corresponding values of compared algorithms. The larger values indicate that for FDA4, the POF of ICADMO and NSGA-II do not have uniform distribution.

Figure 10b shows the box plots on the measure MS for FDA4 of four algorithms. Except for time step t_1 for DBM, at most time steps, the median and the lower quartile of the MS values for ICCoA are all greater than the corresponding values of the compared algorithms, which means ICCoA improves the diversity of FDA4.

Table 7 gives the *t*-test results of ICCoA with the compared algorithm on S and MS for FDA4 at four time steps over 30 independent runs.

It can be seen from Table 7 that, for test problem FDA4, at all of the time steps, for the metric S, the differences between ICCoA and the compared algorithms are significant, which means the solutions obtained by ICCoA have a significant improvement on the uniform distribution of the obtained solutions. At most of all the time steps, except for compared with algorithm DBM at time step t_1, for the metric MS, the differences between ICCoA and the compared algorithms are significant, which indicates the solutions obtained by ICCoA have a significant improvement in covering the area in the Pareto-front and in maintaining the best diversity.

Table 8 gives the Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA4 at four time steps over 30 independent runs.

As for the Wilcoxon signed rank test, it can be seen from Table 8 that, for test problem FDA4, at all of the time steps, for the metric S, ICCoA gets significant differences compared with the other three algorithms. For the metric S, ICCoA gets significant differences compared with the other three algorithms, except for compared with algorithm DBM at time step t_1. The Wilcoxon signed rank test results indicate that, on most of the time steps, ICCoA constitutes a significant improvement on both uniformity and diversity of POF for FDA4.

4.3.5 FDA5

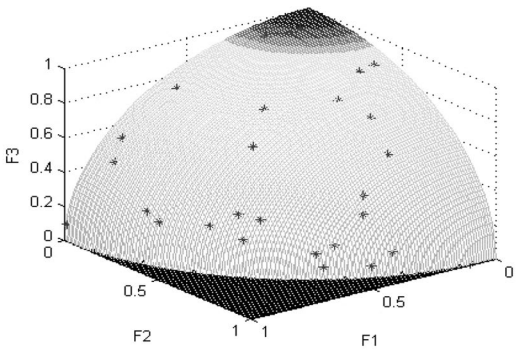
The POF of FDA5 changes over time. The POFs at four time steps when $t_1 = 0$, $t_2 = 0.125$, $t_3 = 0.25$, and $t_4 = 0.375$ obtained by NSGA-II, DBM, ICADMO and ICCoA are illustrated in Fig. 11.

Figure 11 shows that for test problem FDA5, the acquired POF of NSGA-II fail to converge to its real POF at some time steps and have difficulties to keep its uniformity. Moreover, the attained POF of DBM is worse than both ICADMO and ICCoA, which indicates that the other two algorithms have better diversity and uniformity. However, from Fig. 11c-1 to c-4 and Fig. 11d-1 to d-4, we can see that ICCoA shows an improvement of uniformity over ICADMO, and it performs well at every time step.

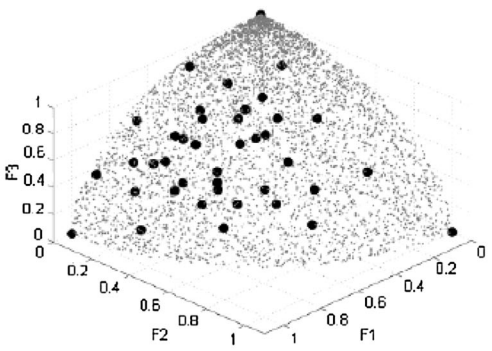
The box plots on the measure S and MS on FDA5 for four algorithms are illustrated in Fig. 12. Figure 12a show that the upper quartile, the median and the lower quartile of the S values for ICCoA are all smaller than the corresponding values of the compared algorithms, which indicates that ICCoA is able to find more uniform solutions for FDA5.

With the box plots in Fig. 12b, we can see that the median and the lower quartile of the MS values at four time steps for ICCoA are all greater than the corresponding values of the compared algorithms, which shows that ICCoA is able to improve the diversity of POF for FDA5.

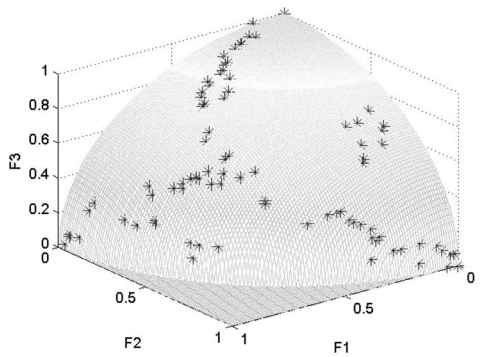
The t-test results of ICCoA with the compared algorithm on S and MS for FDA5 at four time steps over 30 independent runs are shown in Table 9.



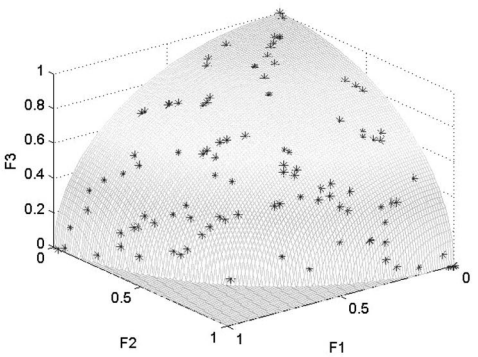
(a-1) Result of NSGA-II (t_1=0)



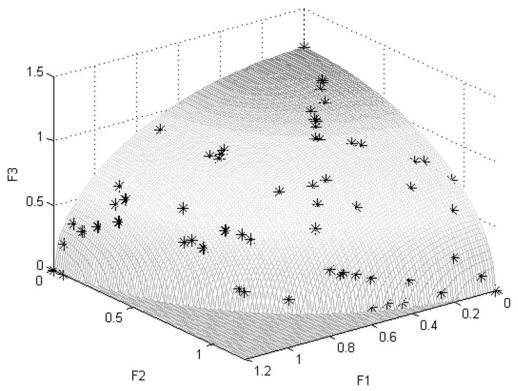
(b-1) Result of DBM (t_1=0) (Farina et al. 2004)



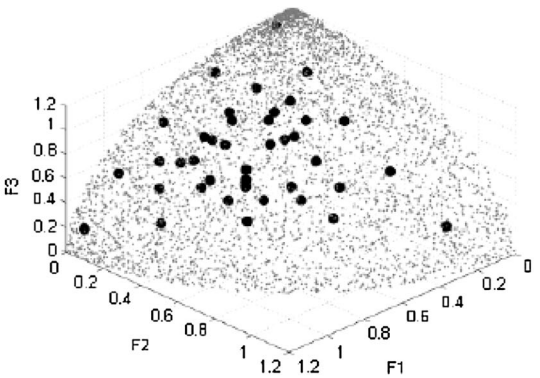
(c-1) Result of ICADMO (t_1=0)



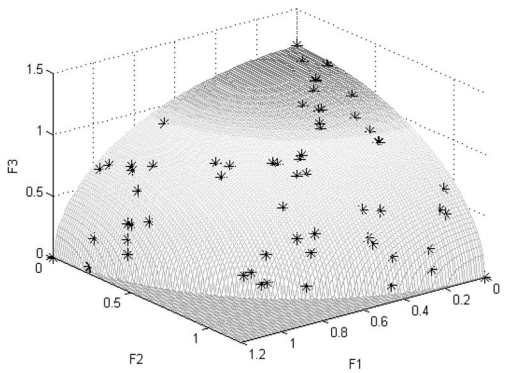
(d-1) Result of ICCoA(t_1=0)



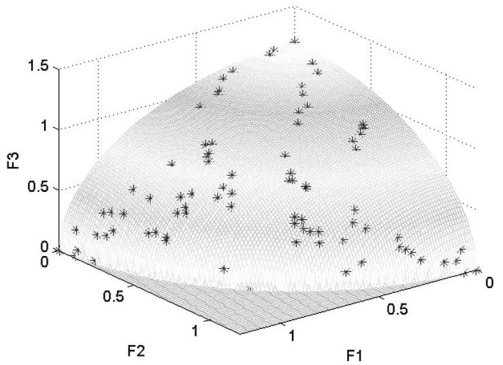
(a-2) Result of NSGA-II (t_2=0.125)



(b-2) Result of DBM (t_2=0.125) (Farina et al. 2004)

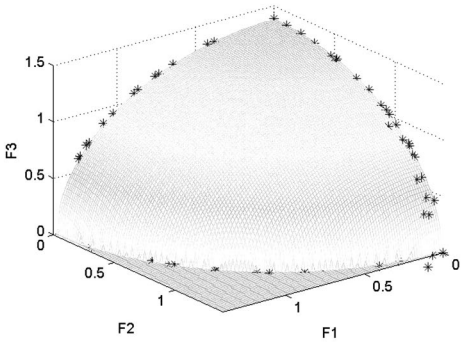


(c-2) Result of ICADMO (t_2=0.125)

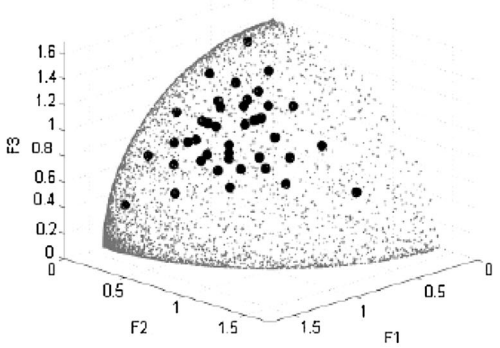


(d-2) Result of ICCoA(t_2=0.125)

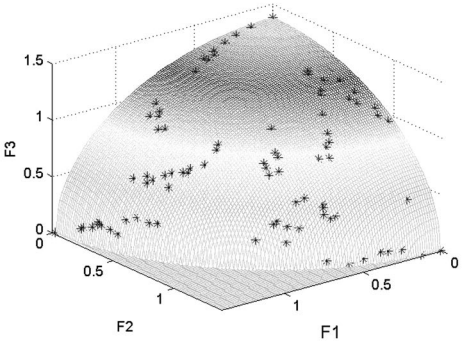
Fig. 11 Experimental results of FDA5



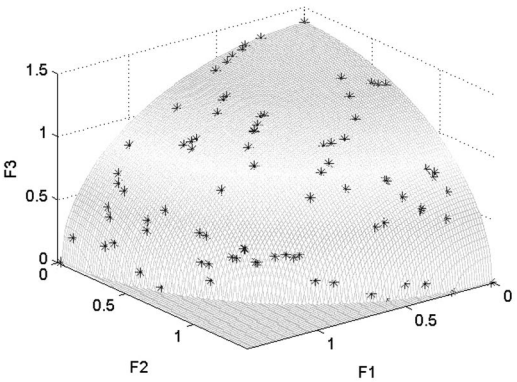
(a-3) Result of NSGA-II (t_3=0.25)



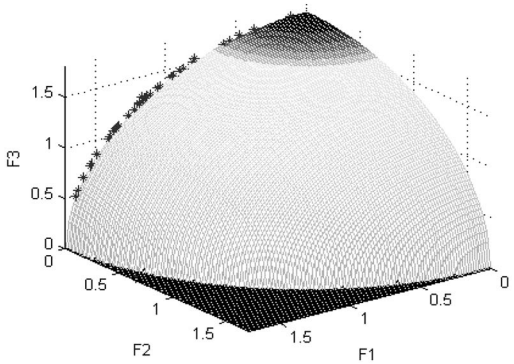
(b-3) Result of DBM (t_3=0.25) (Farina et al. 2004)



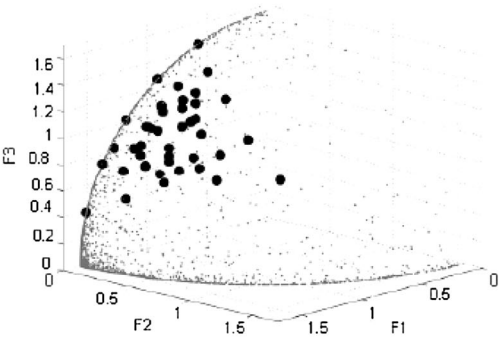
(c-3) Result of ICADMO (t_3=0.25)



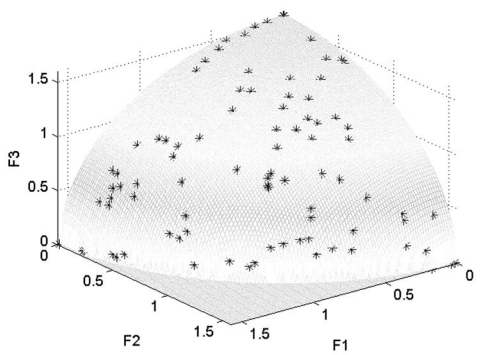
(d-3) Result of ICCoA(t_3=0.25)



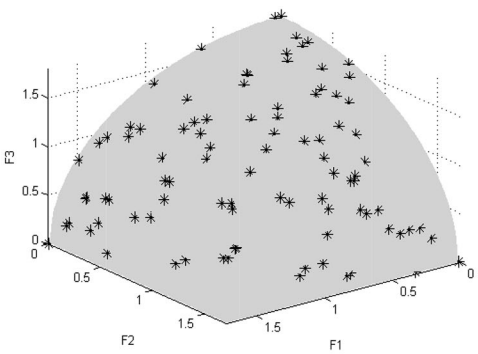
(a-4) Result of NSGA-II (t_4=0.375)



(b-4) Result of DBM (t_4=0.375) (Farina et al. 2004)

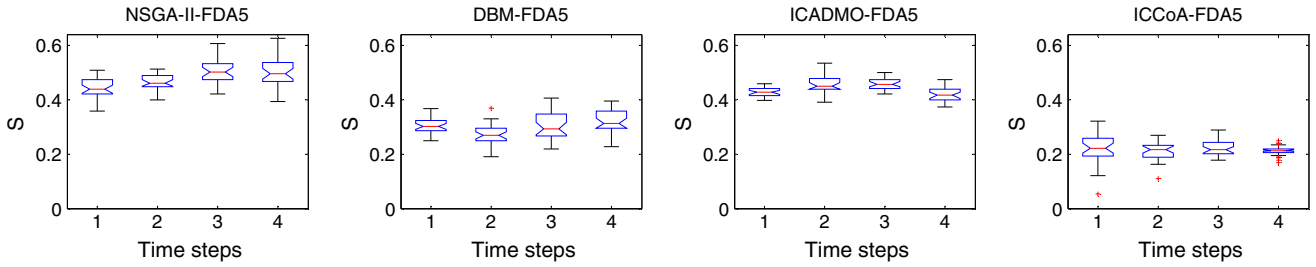


(c-4) Result of ICADMO (t_4=0.375)

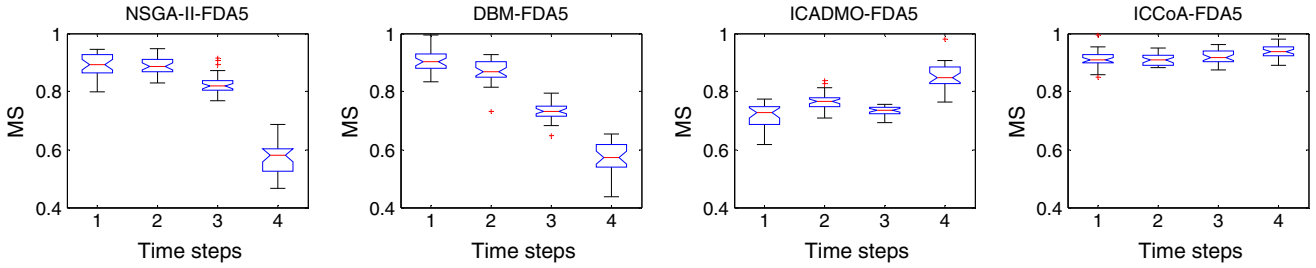


(d-4) Result of ICCoA(t_4=0.375)

Fig. 11 continued



(a) Box plots on the measure S



(b) Box plots on the measure MS

Fig. 12 *Box plots* on the measure S and MS at four time steps for FDA5

Table 9 The t-test results of ICCoA with the compared algorithm on S and MS for FDA5 at four time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	p	h	p	h	p	h	p	h	p	h
t_1	5.1658e−018	1	3.9873e−018	1	1.3172e−008	1	0.0100	1	1.4099e−018	1	0.6108	0
t_2	3.5199e−022	1	7.7033e−025	1	1.1889e−006	1	0.0034	1	2.4621e−017	1	3.5278e−006	1
t_3	1.3501e−023	1	2.5984e−023	1	7.0937e−008	1	4.5462e−012	1	2.6914e−024	1	3.1192e−020	1
t_4	1.2702e−022	1	3.2628e−025	1	4.2988e−013	1	2.7415e−023	1	4.0759e−009	1	6.6519e−025	1

It can be seen from Table 9 that, for test problem FDA5, for metric S, at all of the time steps, ICCoA is able to get significant differences, which indicates that ICCoA can get the significant improvement on the uniform distribution of the solutions. Compared with algorithm DBM, ICCoA fails to get significant differences at time step t_1 for metric MS, which indicates ICCoA can not get the significant improvement on the coverage for Pareto Front of the solutions. However, at the other three time steps, for the metrics MS, the differences between ICCoA and the compared algorithms are significant, which indicates that the solutions obtained by ICCoA have a significant improvement on the coverage area in the Pareto-front.

Table 10 gives the Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA5 at four time steps over 30 independent runs.

Table 10 shows that compared with NSGA-II and ICADMO, at all the time steps, for both S and MS, ICCoA gets significant differences. The median differences between ICCoA and DBM are significant at all the time steps for metric S and at time steps t_2, t_3 and t_4 for metric MS. The statistic results show that at most of the time steps, ICCoA gets significant improvement of the obtained solutions for test problem FDA5.

5 Conclusions

In this paper, in order to solve DMO problems, an ICCoA for DMO has been presented. On the basis of the thought of immune clone in artificial immune system, ICCoA employs the theory of coevolution to propose an improving strategy for the uniformity and diversity of Pareto-optimal solutions. On five different difficult test problems borrowed from the literature, it is shown that the experimental results are in accordance with the theory of ICCoA. By being

Table 10 The Wilcoxon signed rank test results of ICCoA with the compared algorithm on S and MS for FDA5 at four time steps over 30 independent runs

Time steps	NSGA-II(S)		ICADMO(S)		DBM		NSGA-II(MS)		ICADMO(MS)		DBM(MS)	
	p	h	p	h	p	h	p	h	p	h	p	h
t_1	1.7344e−006	1	1.7344e−006	1	3.5152e−006	1	0.0166	1	1.7344e−006	1	0.7343	0
t_2	1.7344e−006	1	1.7344e−006	1	2.1630e−005	1	0.0041	0	1.7344e−006	1	1.6394e−005	1
t_3	1.7344e−006	1	1.7344e−006	1	3.8822e−006	1	1.9209e−006	1	1.7344e−006	1	1.7344e−006	1
t_4	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	1.7344e−006	1	8.4661e−006	1	1.7344e−006	1

compared with the results of the classic genetic algorithm NSGA-II, DBM and ICADMO and being measured on two performance metrics, ICCoA is proved to improve both the uniformity and the diversity and reach better distribution on Pareto-fronts. However, with regard to future perspectives, it is worthwhile to improve the speed of convergence and find a better way to evaluate a DMO algorithm.

Acknowledgments We would like to express our sincere appreciation to the anonymous reviewers for their insightful and valuable comments, which have greatly helped us in improving the quality of the paper. This work was partially supported by the National Basic Research Program (973 Program) of China under Grant 2013CB329402, the National Natural Science Foundation of China, under Grants 61371201, 61001202, 61203303, and 61272279, the National Research Foundation for the Doctoral Program of Higher Education of China, under Grants 20100203120008, the Fundamental Research Funds for the Central Universities, under Grant K5051302028, the Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) under Grant B07048, and the Program for Cheung Kong Scholars and Innovative Research Team in University under Grant IRT1170.

References

Abido MA (2010) Multiobjective particle swarm optimization with nondominated local and global sets. Nat Comput 9(1):747–766. doi:10.1007/s11047-009-9171-7

Cámara M, Ortega J, de Toro F (2009) Performance measures for dynamic multi-objective optimization. Bio-inspired systems: computational and ambient intelligence. Springer, Berlin, pp 760–767

Chambers M, Cleveland WS, Kleiner B et al (1983) Graphical methods for data analysis. Wadsworth Brooks/Cole, Pacific Grover

Chen WS, Jiao LC (2010) Adaptive tracking for periodically time-varying and nonlinearly parameterized systems using multilayer neural networks. IEEE Trans Neural Netw 21(2):345–351

Coello Coello AC, Cortes NC (2002) An approach to solve multiobjective optimization problems based on an artificial immune system. in: Proceedings of 1st international conference artificial immune system, Int. Center Adv. Res. Identif. Sci. (ICARIS), pp 212-221

de Castro LN, Timmis J (2002a) Artificial immune systems: a new computational intelligence approach. Springer, Berlin, pp 1–357

de Castro LN, Timmis J (2002b) Learning and optimization using the clonal selection principle. IEEE Trans Evolut Comput 6(3): 239–251

de Castro LN, Timmis J, Knidel H, Von Zuben F (2010) Artificial immune systems: structure, function, diversity and an application to biclustering. Nat Comput 9(3):575–577. doi:10.1007/s11047-009-9145-9

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evolut Comput 6(2):182–197

Farina M, Deb K, Amato P (2004) Dynamic multiobjective optimization problems: test cases, approximations, and applications. IEEE Trans Evolut Comput 8(5):425–442

Forrest S, Perelson AS, Allen L, Cherukuri R (1994) Self–nonself discrimination in a computer. In: Proceedings of the 1994 IEEE symposium on research in security and privacy, IEEE Computer Society Press, Los Alamitos, 1994, pp 202–212

Gibbons JD (1985) Nonparametric statistical inference, 2nd edn. Marcel Dekker, New York

Goh CK, Tan KC (2007) An investigation on noisy environments in evolutionary multiobjective optimization. IEEE Trans Evolut Comput 11(3):354–381

Goh CK, Tan KC (2009) A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. IEEE Trans Evolut Comput 13(1):103–127

Gong MG, Du HF, Jiao LC (2006) Optimal approximation of linear systems by artificial immune response. Sci China Ser F 49(1): 63–79

Gong MG, Jiao LC, Du HF, Bo LF (2008) Multiobjective immune algorithm with nondominated neighbor-based selection. Evolutionary computation. MIT Press, Cambridge, pp 225–255. doi:10.1162/evco.2008.16.2.225

Helbig M, Engelbrecht AP et al (2013) Dynamic multi-objective optimization using PSO. Metaheuristics for Dynamic Optimization Studies in Computational Intelligence 433:147–188

Huang L, Suh IH, Abraham A (2011) Dynamic multi-objective optimization based on membrane computing for control of time-varying unstable plants. Inf Sci 181(11):2370–2391

Jiao LC, Wang L (2000) A novel genetic algorithm based on immunity. IEEE Trans Syst Man Cybern 30(5):552–561

Jiao LC, Liu J, Zhong WC (2006) An organizational co-evolutionary algorithm for classification. IEEE Trans Evolut Comput 10(1):67–80

Jin Y, Branke J (2005) Evolutionary optimization in uncertain environments—a survey. IEEE Trans Evolut Comput 9(3): 303–317

Leung Y-W, Wang YP (2003) U-measure: a quality measure for multiobjective programming. IEEE Trans Syst Man Cybern 33(3):337–343

Liu RC, Sheng ZC, Jiao LC, Zhang W (2010) Immunodomaince based clonal selection clustering algorithm. IEEE Congr Evolut Comput 2010:1–7

Lung RI, Dumitrescu D (2010) Evolutionary swarm cooperative optimization in dynamic environments. Nat Comput 9(1):83–94. doi:10.1007/s11047-009-9129-9

Nusawardhana, Zak SH (2004) Simultaneous perturbation extremum seeking method for dynamic optimization problems. In: Proceedings of the 2004 American control conference. IEEE Press, Piseataway, 2004, pp 2805–2810

Shang RH, Jiao LC, Gong MG, Lu B (2005) Clonal selection algorithm for dynamic muitiobjective optimization. In: Hao Y et al (Eds) Proceedings of the 2005 international conference on computational intelligence and security LNCS, vol 3801. Springer, Berlin, 2005, pp 846–851

Shang RH, Jiao LC, Liu F, Ma WP (2012) A novel immune clonal algorithm for MO problems. IEEE Trans Evolut Comput 16(1):35–50

Sun J, Fang W, Palade V, Wu XJ, Xu WB (2011) Quantum-behaved particle swarm optimization with Gaussian distributed local attractor point. Appl Math Comput 218:3763–3775

Ursem RK, Krink T, Jensen MT, Michalewicz Z (2002) Analysis and modeling of control tasks in dynamic systems. IEEE Trans Evolut Comput 6(4):378–389

Van Veldhuizen DA, Lamont GB (2000) On measuring multiobjective evolutionary algorithm performance. Congress on evolutionary computation (CEC 2000). IEEE Press, Piscataway, pp 204–211

Wang YP, Dang CY (2008) An evolutionary algorithm for dynamic multi-objective optimization. Appl Math Comput 205(1):6–18

Wang HF, Yang SX, Ip WH, Wang DW (2010) A particle swarm optimization based memetic algorithm for dynamic optimization problems. Nat Comput 9(3):703–725. doi:10.1007/s11047-009-9176-2

Woolley NC, Milanović JV (2011) An immune system inspired clustering and classification method to detect critical areas in electrical power networks. Nat Comput 10(1):305–333. doi:10.1007/s11047-010-9204-2

Yang DD, Jiao LC, Gong MG (2009) Adaptive multiobjective optimization based on nondominated solutions. Comput Intell 25(2):84–108. doi:10.1111/j.1467-8640.2009.00332.x

Zitzler E, Thiele L (2005) A simple mulf-membered evolution strategy to solve constraint optimization problems. IEEE Trans Evolut Comput 9(1):1–17