An Adaptive Multi-objective Immune Algorithm for Optimal Design of Truss Structures

Article *in* Journal of Asian Architecture and Building Engineering · September 2016

DOI: 10.3130/jaabe.15.557

CITATION
1

READS
60

4 authors, including:

Liyu Xie
Tongji University
14 PUBLICATIONS **63** CITATIONS
SEE PROFILE

He-Sheng Tang
Tongji University
73 PUBLICATIONS **348** CITATIONS
SEE PROFILE

Songtao Xue
Tohoku Institute of Technology
92 PUBLICATIONS **556** CITATIONS
SEE PROFILE

An Adaptive Multi-objective Immune Algorithm for Optimal Design of Truss Structures

Liyu Xie[1], Hesheng Tang*[1,2], Changyuan Hu[1] and Songtao Xue[1,3]

[1] Research Institute of Structural Engineering and Disaster Reduction, Tongji University, Shanghai 200092, China
[2] State Key Laboratory for Disaster Reduction in Civil Engineering, Tongji University, Shanghai 200092, China
[3] Department of Architecture, Tohoku Institute of Technology, Sendai, Japan

Abstract

In this paper, an adaptive immune clone selection algorithm for multi-objective optimization (AICSAMO) is proposed. A novel adaptive polynomial mutation operator with dynamic mutation probability is employed in AICSAMO. This adaptive mutation operator executes a rapid global search at the earlier stage of the algorithm and a fine-tuning search at the later stage of the algorithm, which adopts generation-dependent parameters to improve the convergence speed and global optimum searching ability. The effectiveness of AICSAMO is evaluated through the truss sizing and shape optimization problems of a 10-bar plane truss and a 25-bar space truss. According to the comparison of AICSAMO with various multi-objective optimization algorithms developed recently, the simulation results illustrate that AICSAMO has remarkable performance in finding a wider spread of optimal solutions and in maintaining better uniformity of the solutions with better convergence.

Keywords: immune algorithm; multi-objective optimization; adaptive mutation; sizing and shape optimization; truss structure

1. Introduction

Most real engineering problems involve multiple objectives, for example, minimizing cost and maximizing performance and reliability at the same time. These are difficult but realistic problems that engineers confront. For these multiple-objective problems (MOPs), the objectives generally conflict and one optimum objective is achieved at the price of sacrificing other objectives. In most cases, there is no single global optimal solution, and it is often necessary to determine a set of solutions that satisfy the predetermined constraints for an optimization problem. The dominant concept in defining an optimal point is that of the Pareto optimality, called the Pareto front.

In the last decade, evolutionary approaches have been the primary tools to solve real-world, multi-objective, optimization problems, such as Multi-objective Genetic Algorithm (MOGA) (Fonseca and Fleming 1993), Nondominated Sorting Genetic Algorithm (NSGA) (Srinivas and Deb 1994), Strength Pareto Evolutionary Algorithm (SPEA) (Zitzler and Thiele 1999), improved SPEA (SPEA2) (Zitzler *et al.* 2001), Pareto-Archived Evolution Strategy (PAES) (Knowles and Corne 2000), Fast Nondominated Sorting Genetic Algorithm (NSGA-II) (Deb *et al.* 2002), Multi-objective Evolutionary Algorithm (MEA) (Sarker *et al.* 2002), Dynamic Multi-objective Evolutionary Algorithm (DMOEA) (Yen and Lu 2003), and Differential Evolution for Multi-objective Optimization (DEMO) (Robič and Filipič 2005). All of these methods attempted to design effective and efficient algorithms to improve the abilities of the convergence and the diversity of the solution.

On the other hand, Artificial Immune Systems (AIS) have been developed since the 1990s as a new branch in computational intelligence (Farmer *et al.* 1986; Mori *et al.* 1997; De Castro and Timmis 2002; Dasgupta 2006; Castro and Von Zuben 2008). The AIS was inspired by the concept of the body's immune system: clonal selection, negative selection and immune network algorithms. The immune algorithm can prevent premature convergence without losing the diversity of solutions and can provide multiple suboptimal solutions (Mori *et al.* 1997). In contrast to GA, AIS has an affinity calculation function, which explains the relationship not only between the antigen and the antibody but also between antibodies. This function gives AIS the unique characteristic of guaranteeing the survival of the variant offspring that better match the antigen. In the literature, several authors (Chung *et al.* 1998; Tarakanov and Tarakanov 2005; de Castro and Von Zuben 2009) demonstrated that immune genetic algorithms were superior to GAs and ES in solving multimodal function optimization problems.

*Contact Author: Hesheng Tang,
Research Institute of Structural Engineering and Disaster Reduction, Tongji University, Shanghai 200092, China
E-mail; thstj@tongji.edu.cn
(Received April 8, 2015 ; accepted July 11, 2016)
DOI http://doi.org/10.3130/jaabe.15.557

Several AIS models have been applied in various fields, such as clustering, classification and pattern recognition (de Castro and Von Zuben 2002; Castro and Von Zuben 2010), robotics and control (Whitbrook *et al*. 2007), optimization (Miyamoto *et al*. 2004; Campelo *et al*. 2006; Xu *et al*. 2014), parameter estimation (Liu *et al.* 2009), and other similar machine learning problem domains (Harmer *et al*. 2002; Dasgupta *et al.* 2004). In recent years, AIS has been applied to solving MOPs, and studies show remarkable performances. Luh *et al.* (2003) proposed a novel scheme, named the multi-objective immune algorithm (MOIA), which searches unconstrained Pareto solutions. The results showed that the MOIA generally performs better than SPEA for six test functions and also better than MOGA, NPGA, and NSGA. Coello and Cortes (2005) presented a multi-objective immune system algorithm (MISA) based on the clonal selection principle. Freschi and Repetto (2005) proposed a vector artificial immune system (VAIS) based on the artificial immune network and claimed that their multi-objective immune algorithm was comparable to NSGA-II, which is the state-of-the-art algorithm for solving multi-objective optimization problems. Zhang (2007) also proposed an immune-algorithm for solving constrained nonlinear multi-objective optimization problems and reported that his algorithm performed well when compared to well-known evolutionary algorithms, such as MISA and SPEA. Gong *et al.* (2008) proposed a non-dominated neighbor-based immune algorithm (NNIA) for solving unconstrained MOPs by using a novel non-dominated neighbor-based selection technique, proportional cloning, heuristic search operators and elitism. In NNIA, the selection technique only performs on non-dominated individuals and selects minority-isolated individuals to clone proportionally to the crowding distance values, recombine and mutate. By using non-dominated neighbor-based selection and proportional cloning, the method enhances the local search results in the less-crowded regions of the current trade-off front. The experimental study showed that NNIA converges on the true Pareto optimal fronts when solving most test problems and is much better than the other algorithms, such as SPEA2, NSGA-II, PESA-II, and MISA (Gong and Jiao 2008). However, the probability of being trapped in the local optima of multimodal problems when using this algorithm is increased significantly. Therefore, how to maintain the balance between global optimization ability (i.e., exploration) and search efficiency (i.e., exploitation) has become an important issue in designing suitable mutation schemes for the multi-objective immune genetic algorithms.

Recently, several studies have employed different mutation operators into multi-objective optimization algorithms to improve their performance, such as Cauchy mutation operator (Yang and Fang 2011), polynomial mutation operator (Zitzler *et al.* 2001; Deb *et al.* 2002) and hybrid Gaussian and polynomial mutation operators (Chen and Lu 2008; Chen *et al.* 2010). Our previous work proposed an adaptive mutation regulation operator (Liu *et al.* 2009), which adopts a dynamic mutation probability with generation-dependent and affinity-adaptation parameters to improve the convergence speed and global optimum searching ability.

Polynomial mutation has better fine-grained search ability than Gaussian mutation in local regions, which is verified by multiple real-coded, multi-objective optimization algorithms. However, polynomial mutation with fixed distribution parameters is not efficient in searching the global Pareto optimal front (Chen *et al.* 2009). In this paper, an adaptive immune clone selection algorithm for multi-objective optimization (AICSAMO) is proposed, in which an adaptive polynomial mutation operator with dynamic mutation probability is employed. The main idea of this operator is to generate an adaptive mutation size according to the size of the current search space. This operator combines global exploration and local refinement efficiently and adopts a generation-dependent parameter to guarantee a good balance between global search and local search.

Optimization of truss and frame structures is a popular topic in mechanical, civil, and structural engineering due to the complexity of problems and the benefits to industry. This paper describes the proposed architecture of the constrained multi-objective immune algorithm and applies it to the optimal design of truss structures. In this article, the authors attempt to minimize the self-weight and displacement of trusses simultaneously with changing variables of the member size (i.e., cross-sectional areas of truss members) and geometry (i.e., loci of the truss nodes). Design constraints are imposed in terms of allowable material stresses and structural displacements. For each candidate design of a truss, the stresses, deflections, and other design constraints are evaluated to determine whether the design is feasible. If a design is infeasible, a penalty function is applied to the structural weight reflecting the degree of constraint violation. The penalized weight guides the algorithm to focus on designs with the smallest structural weight that satisfy the design constraints. Several standard truss-optimization problems from literature are illustrated using the proposed methods, and the comparative analysis with the other methods is discussed.

This paper is organized as follows. Section 2 briefly describes the definition of MOPs. The formulation of the considered truss optimization problem is provided in this section. In Section 3, the AICSAMO algorithm is described in detail. The effect of the proposed methodology is investigated in Section 4 using benchmark optimization examples of planar and spatial truss structures. Finally, conclusions are given.

2. Statement of the Multi-objective Truss Optimization Problem

In this study, an optimal design problem of a truss is considered. The total weight of the structure and the displacement at one node are minimized simultaneously using the cross-sectional areas of the truss numbers and the coordinates of joints as the design variables, and the design does not exceed allowable values for compressive and/or tensile stress in each member or deflection of any connection. Under these conditions, a multi-objective truss optimization problem can be expressed as

$$\begin{aligned} &\min_{\boldsymbol{x}} \left[f_1(\boldsymbol{x}), f_2(\boldsymbol{x}),..., f_n(\boldsymbol{x})\right]^T \\ s.t.\quad & g_i(\boldsymbol{x}) = \left|\frac{\sigma_i}{\sigma_{ai}}\right| - 1 \le 0 \quad i = 1,2,...,nM \\ & d_j(\boldsymbol{x}) = \left|\frac{\delta_j}{\delta_{aj}}\right| - 1 \le 0 \quad j = 1,2,...,nD \\ & \boldsymbol{x}_l \le \boldsymbol{x} \le \boldsymbol{x}_u \end{aligned} \tag{1}$$

where f_i is the i-th objective function; $g_i(x)$ and $d_j(x)$ are the stress and displacement constraints, respectively; σ_i and σ_{ai} are the computed axial stress in the i-th member and its allowable value, respectively; δ_i and δ_{aj} are the computed displacement in the direction of the j-th degree of freedom and its allowable value, respectively; nM is the total number of truss members; nD is the total number of active degrees of freedom; $\boldsymbol{x}$ is the vector of optimization or decision variables; $\boldsymbol{x}_l$ and $\boldsymbol{x}_u$ andare the lower and upper bounds of the design variable, respectively. The solution to the above problem is a set of Pareto points instead of a unique solution to a single-objective optimization problem.

3. Description of the Algorithm

3.1 The Basic Mechanism of Artificial Immune System

The immune system is a biological defense system for protecting the human body from the attack of foreign (harmful) organisms, which can recognize self-cells or non-self cells and exclude non-self cells (antigens). Artificial Immune Systems (AISs) are adaptive systems, inspired by the principles and processes of theoretical immunology. In AIS, clonal selection algorithms are a class of algorithms inspired by the clonal selection theory of an immune response. Roughly speaking, the clonal selection theory states that when the invasion of an antigen is detected, those antibodies that recognize the antigen proliferate by cloning. This process is called the clonal selection principle. During proliferation, the newly cloned cells undergo mutations or hypermutation with rates proportional to their affinity for the antigen. The highest affinity antibodies have the lowest mutation rates, and vice-versa. This process is an evolutionary process that led us to propose optimization algorithms to solve constrained MOPs. Throughout this paper, the **B** cell is treated as an antibody because it only includes identical antibodies. As associated with the problem defined in Section 2, an antibody is viewed as a feasible solution with real encoding. Given antibody **Ab**, affinity (i.e., *aff* (**Ab**)) is explained as the match capability between antibody **Ab** and antigen **Ag**, which can be depicted by means of their objective function values, i.e., $\boldsymbol{f}$(**Ab**) and $\boldsymbol{f}$(**Ag**).

Generally, the antigens correspond to the objective functions and constraint conditions, and the antibodies are associated with the feasible solutions in a constrained, multi-objective optimization problem. An antibody **Ab** is the coding of the variable (solution) $\boldsymbol{x}$, denoted by **Ab** = $e(\boldsymbol{x})$, and $\boldsymbol{x}$ is the decoding of antibody **Ab**, expressed as $\boldsymbol{x} = e^{-1}$(**Ab**). When real-valued presentation is adopted, that is, **Ab** = $e(\boldsymbol{x})$= $\boldsymbol{x}$, so, $\mathbf{Ab}_i = \boldsymbol{x}_i = (x_1,x_2,..., x_m)_i$, $1 \le i \le n$ where $\mathbf{Ab}_i$ represents the i-th antibody of the whole population or the i-th solution ($\boldsymbol{x}_i$), m is the number of decision variables, and n is the number of antibodies/solutions. An antibody population

$$\mathbf{B} = (\mathbf{Ab}_1, \mathbf{Ab}_2, ..., \mathbf{Ab}_n) \tag{2}$$

is an n-dimensional group of antibodies **Ab**, where n is the size of the antibody population **B**. The antibodies evolve continuously to search for the fittest ones, i.e., the most matched to specific antigens.

3.2 Non-dominated Clonal Selection and Proportional Cloning

The immune system response is a specific response to a particular 'non-self' material-antigen. When intrusion detection systems detect an antigen, the antibody recognizes the antigen and makes a decision whether to start proliferating by cloning; this process is called clonal selection. According to the concept of Pareto, an antibody can be classified as a non-dominated antibody or dominated antibody. In this paper, the set of dominant antibodies is denoted as **D**. These dominant antibodies are the non-dominated individuals in population **B**. For example, in the antibody population **B** = {$\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$}, if both $\mathbf{b}_2$ and $\mathbf{b}_3$ do not dominate $\mathbf{b}_1$, then $\mathbf{b}_1$ is a dominant antibody in the antibody population **B**; the non-dominated individuals are the Pareto optimal solutions of the problem. In the multi-objective algorithm, the dominant antibodies in **D** are ranked according to how much they contribute to the diversity of the objective function values. This can be measured by the crowding-distance (Deb *et al.* 2002). For an MOP, the crowding-distance of a dominant antibody $d \in \mathbf{D}$ is given by

$$\zeta(d, \mathbf{D}) = \sum_{i=1}^{n} \frac{\zeta_i(d, \mathbf{D})}{f_i^{\max} - f_i^{\min}} \tag{3}$$

where $f_i^{\max}$ and $f_i^{\min}$ are the maximum and minimum values of the ith objective and

$$\zeta_i(d,\mathbf{D})=\begin{cases}\infty, & \text{if } f_i(d)=\min\{f_i(d')|d'\in\mathbf{D}\}\\ & \text{or } f_i(d)=\max\{f_i(d')|d'\in\mathbf{D}\}\\ \text{otherwise} & \\ & \min\{f_i(d')-f_i(d'')|d''\in\mathbf{D}: f_i(d')<f_i(d)<f_i(d'')\}\end{cases} \quad (4)$$

The criteria of clonal selection are in descending order according to crowding distance; the first n_A individuals are selected as the active population **A**. A new population **C** is obtained by direct proportional cloning from **A**. Then, the values of clone number q_i are determined by the crowding-distance of the antibody individuals.

$$q_i=\left\lceil n_c\times\frac{\zeta(a_i,\mathbf{A})}{\sum_{j=1}^{|\mathbf{A}|}\zeta(a_j,\mathbf{A})}\right\rceil \quad (5)$$

where $\zeta(a_j, \mathbf{A})$ denotes the crowding-distance value of the active antibodies a_j, and n_c is the expected value of the size of the clone population.

3.3 Recombination Operator

The recombination operator has the ability to escape from a local optimum and share gene segments from parent chromosomes. Simulated binary crossover (SBX) is one of the main recombination operators used in various real-coded, multi-objective optimization algorithms (Deb *et al.* 2002; Gong *et al.* 2008). In this paper, the SBX operation is performed on the population before and after the application of the proportional clonal operation, which is similar to the operation used in Deb *et al.*, 2002 and Gong *et al.*, 2008. The operation can be described as follows:

$$\begin{aligned}&T^R(\mathbf{C},\mathbf{A})=T^R\left(\left\{c_1,...,c_{|\mathbf{C}|}\right\},\left\{a_1,...,a_{|\mathbf{A}|}\right\}\right)\\&=\left\{SBX\left(c_1,a_{r1}\right),...,SBX(c_i,a_{ri}),...,SBX(c_{|\mathbf{C}|},a_{r\mathbf{A}})\right\}\end{aligned} \quad (6)$$

where **C** and **A** are the populations after and before the clone operation, respectively; c_i is the i-th component of **C**; a_{ri} a is a random individual of **A**; and $SBX(c_i,a_{ri})$ (i=1,...,$|c|$) denotes selecting one individual with equal probability from the two offspring generated by a general crossover operator on clone c_i and an active antibody a_{ri} chosen randomly from **A**.

3.4 Adaptive Mutation Operator

To search for a more appropriate trade-off between exploration and exploitation, an adaptive polynomial mutation scheme is proposed to enhance the capability of population exploration. Given antibody $\mathbf{Ab}_i$, the adaptive polynomial mutation operator is defined as:

$$\mathbf{Ab}_i'\leftarrow\mathbf{Ab}_i+\delta_i\left(\max\{aff(\mathbf{Ab}_i)\}-\min\{aff(\mathbf{Ab}_i)\}\right) \quad (7)$$

where $\delta_i\in[-1,1]$ is a small variation obtained from the polynomial probability distribution.

$$p(\delta)=0.5(\eta_m+1)\left(1-|\delta|\right)^{\eta_m} \quad (8)$$

where η_m is the polynomial distribution index.

$$\delta_i=\begin{cases}\left(2r_i\right)^{1/(\eta_m+1)}-1, & r_i<0.5\\ 1-2\left[\left(1-r_i\right)\right]^{1/(\eta_m+1)}-1, & r_i\ge 0.5\end{cases} \quad (9)$$

where r_i is a sample random number that is uniformly-distributed between (0,1).

The property of η_m is changeable during the searching process, denoting that the degree of perturbation can be varied in the mutated solution. The value of η_m controls the magnitude of the expected mutation of the solution variable. The value of η_m is relatively small at the prophase of the running algorithm; then, a large perturbation of variables by mutation is achieved, which can be helpful for rapidly converging to the promising Pareto front in the initial stage. If the value of η_m is large, which means a small perturbation of variables during mutation, it will be beneficial to the fine-tuning search in the local neighbor region. To achieve gradually decreasing perturbation in the mutated solutions, the value of η_m is gradually increased. In this paper, a dynamic η_m is assigned according to the number of generations to achieve the above adaptation:

$$\eta_m=\eta_0+gen \quad (10)$$

where η_0 is a predefined constant with $50\le\eta_0\le100$ at the beginning of the mutation operator implementation, and *gen* is the number of iterations, which is used to make the mutation probability decrease along the searching process.

The mutation probability should be changed dynamically according to the adaptive ability of the immune system. For the later stages of evolution, the obtained solutions are close to the known Pareto optimal, and it is better to decrease the mutation probability to maintain good individuals. Taking these ideas into consideration, a dynamic mutation probability is used in this paper, which can be calculated as follows:

$$p_m=\frac{1}{m}+\lambda\left(1-\frac{gen}{gen_{\max}}\right)\frac{1}{m} \quad (11)$$

where m is the number of decision variables, λ<1 is a predefined parameter that adjusts the mutation scale, and $gen_{\max}$ is the maximum iteration number. The adaptive mutation operator executes global search at the earlier stage and local refinement at the later stage of the algorithm implementation, which can improve the convergence speed and global optimum searching ability.

In this study, to enhance the exploratory capability, the proposed algorithm uses a novel adaptive mutation operator. In addition, the main operators of AICSAMO include a proportional cloning operator and crossover operator. The authors adopt the same population selection method proposed by Gong *et al.* (2008), named non-dominated neighbor selection. Moreover, an elitism mechanism is used, and an archive is used to preserve the non-dominated solutions to prevent the loss of elitists.

The basic algorithm of the improved immune clone selection algorithm for multi-objective optimization used in this study is described in the next eight steps.

Step 1. Set the initial conditions and termination criterion. Set $gen = 0$.

Step 2. Randomly generate an antibody population, identify the non-dominated antibodies, and store them externally in a continuously updated archive.

Step 3. If the termination criterion is satisfied, go to step 8; otherwise, go to next step 4.

Step 4. Choose part of the antibodies with greater fitness values in the archive to perform proportional cloning to generate the child population.

Step 5. Perform the recombination operator on the child population.

Step 6. Perform the adaptive mutation operator on the child population.

Step 7. Identify the non-dominated antibodies in the mating pool by combining the archive and the child population. Then, antibodies with greater fitness values are selected to be preserved in the archive. Set $gen = gen + 1$. Go to step 3.

Step 8. Output the external archive as the approximate Pareto optimal set. Stop the algorithm.

4. Numerical Examples

In this study, two typical truss design problems, including size and shape optimizations, are used to validate the proposed algorithm. The performance of AICSAMO is compared with various optimization algorithms, i.e., NSGA-II and DEMO. The active population size of both examples is 20, the multiple cloning is 100, and the dominant population size is 100. The largest iteration $gen_{\max}$ is determined according to the specific optimization problem. The parameter settings of NSGA-II and DEMO are found in Deb *et al.* (2002) and Robič and Filipič (2005). With these parameters, AICSAMO has the same simulation conditions as other multi-objective optimization algorithms.

4.1 10-bar Plane Truss Size Optimization with Continuous Design Variables

The 10-bar plane truss with the node and element numbering is shown in Fig.1., where L=360 in and P=100 ksi. The elastic modulus of the material is $E=10^7$ psi, and the density is ρ=0.1 lb·in^{-3}. The upper and lower boundaries of each truss element are 0.1 and 30 in^2. The axial stress for all members is less than 25 ksi. The objective is to minimize the volume of the structure and the vertical displacement at node 2 simultaneously using the cross-sectional areas of the 10-bar truss as the design variables.

Graphically, Fig.2. displays the Pareto optimal solutions obtained by AICSAMO after 50, 100 and 500 iterations. AICSAMO can obtain the approximation set that is distributed uniformly and close to the Pareto optimal front after 100 runs. In addition, the comparisons with the Pareto optimal solutions derived by Fadel and Li (2002) employing the weighting, Tchebycheff and ε-constraint methods are presented in Fig.3. Numerous simulation results of the two extreme objective values utilizing single-objective methods (Fadel and Li, 2002) and the constrained multi-objective immune algorithm (CMOIA) (Luh and Chueh, 2004) are adopted for comparison, as listed in Table 1. Fig.3. and Table 1. show that AICSAMO is capable of finding more uniformly distributed satisfactory solutions and the two extreme objective values.

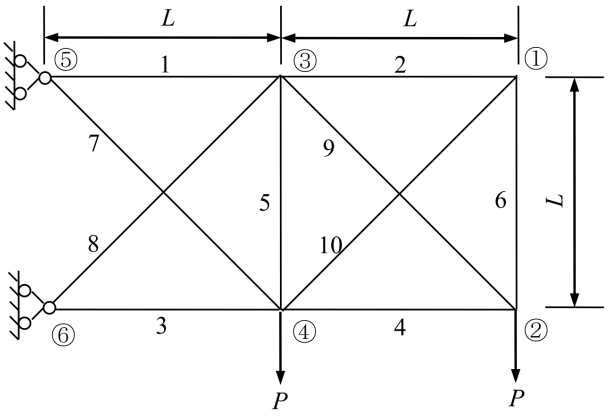


Fig.1. 10-bar Plane Truss Structure

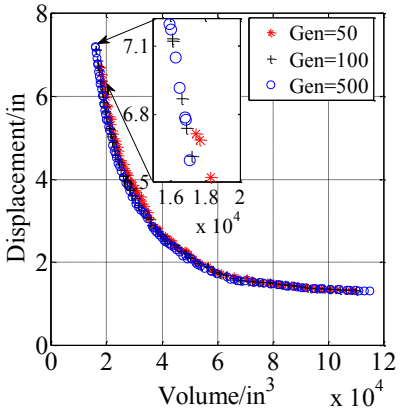


Fig.2. Feasible Pareto Solutions after 50, 100 and 500 Iterations

Fig.4. illustrates the comparison with the Pareto solutions derived by other multi-objective methods (NSGA-II and DEMO). In Fig.4., AICSAMO has the best spread of solutions for the 10-bar plane truss problem compared with NSGA-II and DEMO. Based on the non-dominated neighbor-based selection mechanism with the adaptive polynomial mutation

Table 1. Comparison of the Two Extreme Values for a 10-bar Plane Truss

Methods	Extreme values	
AICSAMO	[115130.0225, 1.3034]	[15960.2700, 7.1977]
CMOIA	[108413.542, 1.3611]	[17935.1162, 6.3562]
Weighting Method	[115114.7524, 1.3034]	[15936.5626, 7.1969]
Tchebycheff Method	[114858.8405, 1.3037]	[15945.5321, 7.1913]
ε-constraint Method	[115114.7513, 1.3034]	[15930.3384, 7.1995]

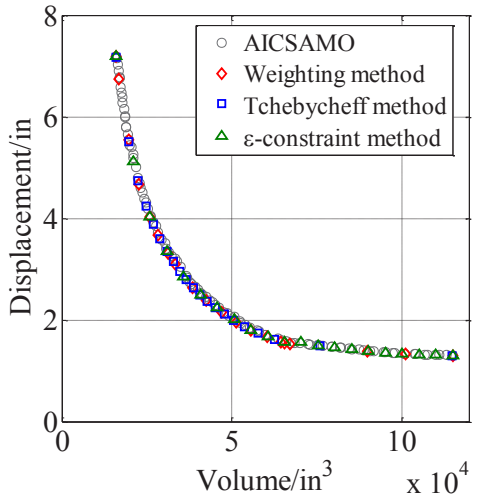


Fig.3. Comparisons of the Feasible Pareto Solutions for the 10-bar Truss

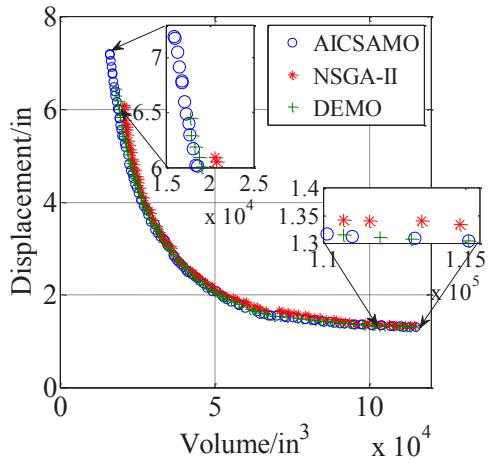


Fig.4. Feasible Pareto Solutions Comparison with NSGA-II and DEMO for the 10-bar Truss

operator, AICSAMO obtains desirable results in terms of the diversity metric. In addition, it can be observed from Fig.4. that in the vertical direction, AICSAMO can expand further than the other two methods. In the horizontal direction, the extreme point of AICSAMO is as far as DEMO and further than NSGA-II. Moreover, AICSAMO needs the least amount of time to solve the 10-bar plane truss problem. The execution times of DEMO, NSGA-II and AICSAMO are 86 s, 43.2 s and 22.4 s, respectively.

4.2 25-bar Space Truss Shape Optimization with Discrete Design Variables

In this section, the performance of AICSAMO is studied on a 25-bar space truss, as shown in Fig.5. The problem is to find the cross-sectional area of each member and the coordinates of some joints such that the total structural weight and the vertical displacement of node 1 are minimized concurrently. The material properties are taken as density ρ=0.1 lb·in^{-3}, elastic modulus E=10^7 psi, and L=25 in. The limit of the principal stress in each truss element is below the maximum allowable stress of ±40 ksi. The truss members are divided into eight groups, as shown in Table 2. The discrete values considered for this example are selected from the set {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0, 3.1, 3.2, 3.3, 3.4} (in^2). In addition, the loading conditions given in Table 3. are applied to the truss structure. The node coordinates of the 25-bar space truss are listed in Table 4.

Table 2. Group Members of the 25-bar Space Truss

Group number	Bar number (number of end joints)			
1	1(1,2)			
2	2(1,4)	3(2,3)	4(1,5)	5(2,6)
3	6(2,5)	7(2,4)	8(1,3)	9(1,6)
4	10(3,6)	11(4,5)		
5	12(3,4)	13(5,6)		
6	14(3,10)	15(6,7)	16(4,9)	17(5,8)
7	18(3,8)	19(4,7)	20(6,9)	21(5,10)
8	22(3,7)	23(4,8)	24(5,9)	25(6,10)

Table 3. Loading Conditions of the 25-bar Space Truss

Node	F_x(kips)	F_y(kips)	F_z(kips)
1	1.0	10.0	-10
2	0	10.0	-10
3	0.5	0	0
6	0.6	0	0

Table 4. 25-bar Space Truss Node Coordinates

Node	x(in)	y(in)	z(in)
1	-37.5	0	200
2	37.5	0	200
3	-37.5	37.5	100
4	37.5	37.5	100
5	37.5	-37.5	100
6	-37.5	-37.5	100
7	-100	100	0
8	100	100	0
9	100	-100	0
10	-100	-100	0

The x, y and z coordinates of joints 3, 4, 5 and 6 are allowed to vary, whereas the positions of joints 1 and 2 remain unchanged. The configurations are selected as x_4, y_4, z_4, x_8 and y_8, with double symmetry required in both the x-z and y-z planes. Therefore, the problem includes 8 sizing variables and 5 configuration

variables. The shape constraints for the configuration variables are 20 in$\leq x_4 \leq$60 in, 40 in$\leq y_4 \leq$80 in, 90 in$\leq z_4 \leq$130 in, 40 in$\leq x_8 \leq$80 in, and 100 in$\leq y_8 \leq$140 in.

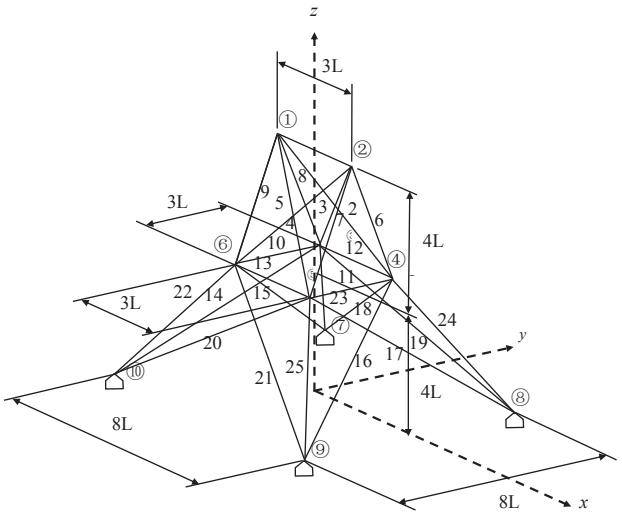


Fig.5. The 25-bar Space Truss Structure

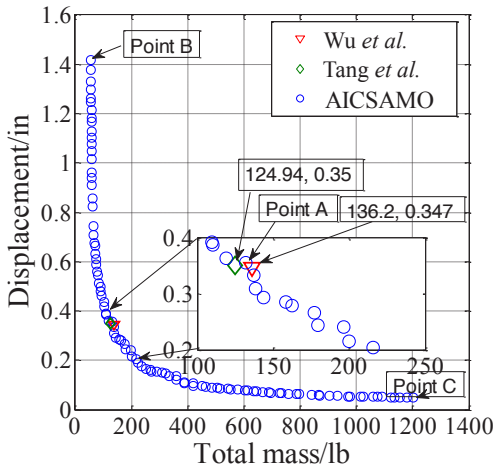
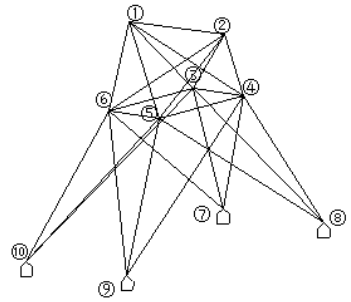
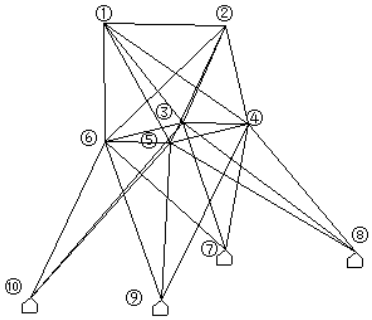


Fig.6. Feasible Pareto Solutions and Comparisons of the 25-bar Truss

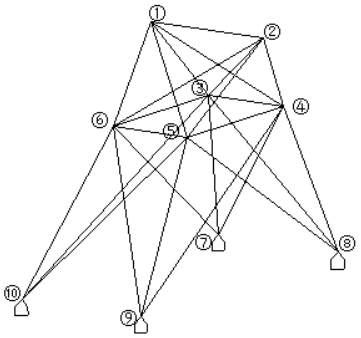
Fig.6. shows the Pareto front of the feasible non-dominated solutions using AICSAMO after 500 iterations. In addition, numerous simulation results using the single-objective from the literature (Wu and Chow 1995, Tang *et al.*, 2005) are adopted for comparison, as depicted in Fig.6. The results from Fig.6. indicate that there is no meaningful difference in performance of the other single-objective methods for the extreme objective values. The optimum geometry of point A, point B and point C is shown in Fig.7. The comparison of feasible Pareto solutions between AICSAMO, NSGA-II and DEMO is shown in Fig.8. From Fig.8., it's showing that in two directions (extreme ends of the displacement and volume), AICSAMO can expand further than the other two methods. Moreover, all Pareto optimal solutions for the problem are scattered uniformly. In addition, the execution time of AICSAMO is the shortest. For the 25-bar truss problem, the execution times of DEMO, NSGA-II and AICSAMO are 110 s, 54.4 s and 31 s, respectively.



(a) point A



(b) point B



(c) point C

Fig.7. Optimum Geometry

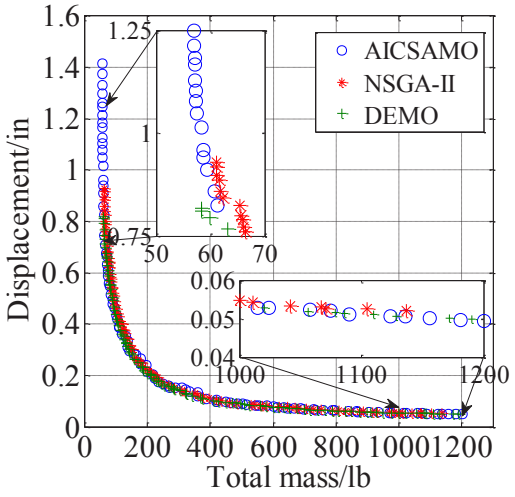


Fig.8. Feasible Pareto Solutions Comparison with NSGA-II and DEMO for the 25-bar Truss

The results from the 10-bar and 25-bar truss optimization problems show that AICSAMO achieves better results, which are comparable to the results of the algorithms NSGA-II and DEMO. The simulation results validate the good exploration ability of AICSAMO in solving multi-objective optimization problems.

5. Conclusion

In this paper, an effective clonal algorithm for MOPs is presented, in which an adaptive polynomial mutation operator with dynamic mutation probability is employed. Two classical optimization problems of a 10-bar plane truss and 25-bar space truss are solved using the proposed method, and the results are compared with other optimization methods. The simulation results show that the proposed algorithm has better performance in finding a better spread of solutions and in maintaining better uniformity of the solutions with better convergence. This paper demonstrated the feasibility of these simple adaptive approaches. Further experimental and theoretical analysis should be conducted.

Acknowledgement

This study was supported by the Ministry of Science and Technology of China, Grant No. SLDRCE14-B-03, and the National Natural Science Foundation of China, Grant No. 51178337 and 51478356.

References

1) Campelo, F., Guimares, F.G., Igarashi, H., Ramirez, J.A., and Noguchi, S., 2006. A modified immune network algorithm for multimodal electromagnetic problems. *IEEE Transactions on Magnetics*, 42 (4), pp.1111-1114.
2) Castro, P.A.D. and Von Zuben, F.J., 2008. MOBAIS: A Bayesian Artificial Immune System for Multi-objective Optimization. *7th International Conference on Artificial Immune System (ICARIS)*, Thailand, August 10-13.
3) Castro, P.A.D. and Von Zuben, F.J., 2010. Multi-objective feature selection using a Bayesian Artificial Immune System. *Journal of Intelligent Computing and Cybernetics*, 3(2), pp.235-256.
4) Castro, P.A.D. and Von Zuben, F.J., 2009. Multi-objective Bayesian Artificial Immune System: Empirical Evaluation and Comparative Analyses. *Journal of Mathematical Modelling and Algorithms*, 8, pp.151-173.
5) Chen, M.R. and Lu, Y.Z., 2008. A novel elitist multiobjective optimization algorithm: multiobjective extremal optimization. *European Journal of Operational Research*, 188 (3), pp.637-651.
6) Chen, J., Lin, Q., and Ji, Z., 2010. A hybrid immune multiobjective optimization algorithm. *European Journal of Operational Research*, 204 (2), pp.294-302.
7) Chen, J., Lin, Q., and Hu, Q., 2009. An Improved Clonal Algorithm in Multiobjective Optimization. *Journal of Software*, 4 (9), 976-983.
8) Chung, J.S., Jung, H.K., and Hahn, S.Y., 1998. A study on comparison of optimization performances between immune algorithm and other algorithms. *IEEE Transactions on Magnetics*, 34 (5), pp.2972–2975.
9) Coello Coello, C.A. and Cortes, N.C., 2005. Solving multiobjective optimization problems using an artificial immune system. *Genetic Programming and Evolvable Machines*, 6 (2), pp.163-190.
10) Dasgupta, D., 2006. Advances in Artificial Immune Systems. *IEEE Computational Intelligence Magazine*, 1(4), pp.40-49.
11) Dasgupta, D., Krishna Kumar, K., Wong, D., and Berry, M., 2004. Negative selection algorithm for aircraft fault detection. *Third International Conference*, ICARIS 2004, Catania, Sicily, Italy, September 13-16.
12) Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., 2002. A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6 (2), pp.182-197.
13) de Castro, L.N. and Von Zuben, F.J., 2002. Learning and Optimization Using the Clonal Selection Principle. *IEEE Transactions on Evolutionary Computation*, 6 (3), pp.239-251.
14) de Castro, L. N. and Timmis, J., 2002. *Artificial Immune Systems: A New Computational Intelligence Approach*. Berlin, Germany: Springer.
15) Fadel, G. and Li, Y., 2002. Approximating the Pareto curve to help solve bi-objective design problems. *Structural and Multidisciplinary Optimization*, 23 (4), pp.280-296.
16) Farmer, J.D., Packard, N.H., and Perelson, A.S., 1986. The immune system, adaptation, and machine learning. *Physica D: Nonlinear Phenomena*, 22(1-3), pp.187-204.
17) Fonseca, C.M. and Fleming, P.J., 1983. Multiobjective genetic algorithms. *IEE Colloquium on Genetic Algorithms for Control Systems Engineering*, London, UK, May 1-5.
18) Freschi, F. and Repetto, M., 2005. Multi-objective optimization by a modified artificial immune system algorithm. *4th International Conference, ICARIS on artificial immune systems*, Banff, Alberta, Canada, August 14-17.
19) Gong, M., Jiao, L., Du, H., and Bo, L., 2008. Multi-objective Immune Algorithm with Nondominated Neighbor-based Selection. *Evolutionary Computation*, 16 (2), pp.225-255.
20) Harmer, P.K., Williams, P.D., Gunsch, G.H., and Lamont, G.B., 2002. An Artificial Immune System Architecture for Computer Security Applications. *IEEE Transactions on Evolutionary Computation*, 6(3), pp.252-280.
21) Knowles, J.D. and Corne, D.W., 2000. Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 8(2), pp.149-72.
22) Liu, Z., Tang, H., and Fan, C., 2009. Parameter estimation using an adaptive immune clone selection algorithm. *2009 Global Congress on Intelligent Systems*, Xiamen, China, May 19-21.
23) Luh, G. and Chueh, C., 2004. Multi-objective optimal design of truss structure with immune algorithm. *Computers & Structures*, 82(11-12), pp.829-844.
24) Luh, G., Chueh, C., and Liu, W., 2003. MOIA: multi-objective immune algorithm. *Engineering. Optimization*, 35(2), pp.143-64.
25) Miyamoto, A., Nakamura, H., and Kruszka, L., 2004. Application of the Improved Immune Algorithm to Structural Design Support System. *Journal of Structural Engineering*, 130(1), pp.108-119.
26) Mori, K., Tsukiyama, M., and Fukuda, T., 1997. Application of an immune algorithm to multi-optimization problems. *Electrical Engineering in Japan*, C117 (5), pp.593-597.
27) Robič, T. and Filipič, B., 2005. DEMO: Differential Evolution for Multiobjective Optimization. *In Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization*, Guanajuato, Mexico, March 9-11.
28) Sarker, R., Liang, K. H., and Newton, C., 2002. A new multiobjective evolutionary algorithm. *European Journal of Operational Research*, 140(1), pp.12-23.
29) Srinivas, N. and Deb, K., 1994. Multiobjective optimization using nondominated sorting in genetic algorithms. *Journal of Evolutionary Computation*, 2(3), pp.221-48.
30) Tang, W., Tong L., and Gu, Y., 2005. Improved genetic algorithm for design optimization of truss structures with sizing, shape and topology variable. *International Journal for Numerical Methods in Engineering*, 62, pp.1737-1762.
31) Tarakanov, A.O. and Tarakanov, Y.A., 2005. A comparison of immune and genetic algorithms for two real-life tasks of pattern recognition. *International Journal of Unconventional Computing*, 1(4), pp.357-374.
32) Whitbrook, A.M., Aickelin, U., and Garibaldi, J.M., 2007. Idiotypic Immune Networks in Mobile-Robot Control. *IEEE Transactions on Systems, Man, and Cybernetics*, 37(6), pp.1581-1598.
33) Wu S.-J. and Chow P.-T., 1995. Integrated discrete and configuration optimization of trusses using genetic algorithms. *Computers & Structures*, 55, pp.695-702.
34) Xu, Y., Wang, L., Wang, S. and Liu, M., 2014. An effective hybrid immune algorithm for solving the distributed permutation flow-shop scheduling problem. *Engineering Optimization*, 46(9), pp.1269-1283.
35) Yang, Y. and Fang, H., 2011. Multi-objective Immune Algorithm With Dynamic Memetic Cauchy Mutation. *2011 IEEE Workshop on Memetic Computing (MC)*, Paris, France, April 11-15.
36) Yen, G.G. and Lu, H., 2003. Dynamic multiobjective evolutionary algorithm: adaptive cell-based rank and density estimation. *IEEE Transaction Evolutionary Computation*, 7(3), pp.253-74.
37) Zhang, Z., 2007. Immune optimization algorithm for constrained nonlinear multiobjective optimization problems. *Applied Soft Computing*, 7(3), pp.840-857.
38) Zitzler, E. and Thiele, L., 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Transaction Evolutionary Computation*, 3(4), pp.257-71.
39) Zitzler, E., Laumanns, M., and Thiele, L., 2001. SPEA2: improving the strength Pareto evolutionary algorithm. *In proceeding of: Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*. Athens. Greece, September 19-21.