

Contents list available at JMCS

Journal of Mathematics and Computer Science

Journal Homepage: [www.tjmcs.com](http://www.tjmcs.com)



## Sizing optimization of truss structures under Frequency Constraints with Artificial Bee Colony Algorithm

M. Mashinchi Joubari<sup>1</sup>, M. H. Pashaei<sup>2</sup>, A. Fathi<sup>3</sup>

<sup>1, 2, 3</sup> Department of Mechanical Engineering, Babol University of Technology, Babol, Iran

<sup>1</sup>*mmmjouybari@gmail.com*

<sup>2</sup>*mpashaei@nit.ac.ir*

<sup>3</sup>*fathi@nit.ac.ir*

### Article history:

Received September 2013

Accepted October 2013

Available online October 2013

### Abstract

Due the wide applications of truss structures in industries including aviation, transportation, buildings and vital structures, design and optimization of these structures have become the most active research fields. Furthermore it turns out that in many of structures that have been exposed to wind, hurricanes or violent earthquake, the optimization according weight only is not an appropriate way. So recently, many efforts have been made according natural frequencies optimization and minimization of weight. Due to the importance of optimization and control of natural frequencies in structures and to avoid resonance, calculating of natural frequencies of the truss structures in different moods based on minimization of weight have been focused. To reach this goal method have been used in this research, optimization by Artificial Bee Colony algorithm (ABC). Optimized structures are a planar 10-bar truss and a space 72-bar truss. Conclusions show these method have better quality than other algorithms and they can use engineering complex structures optimization.

**Keywords:** Size optimization; Truss structures; Artificial Bee Colony algorithm; Frequency constraints; Engineering design

## 1. Introduction

It is well known that the natural frequencies are fundamental parameters affecting the dynamic behavior of the structures. Therefore, some limitations should be imposed on the natural frequency range to reduce the domain of vibration and also to prevent the response phenomenon in dynamic response of structures [1]. On the other hand, engineering structures are often supposed to be as light

as possible [2]. Thus a frequency constraint weight optimization process should be performed to obtain these two aims simultaneously.

Frequency constraints are highly nonlinear, non-convex and implicit with respect to the design variables [3]. Therefore mathematical programming approaches can be hard and time-consuming to apply to these optimization problems. Furthermore, a good starting point is vital for these methods to be executed successfully [4]. Under such circumstances, the population based meta-heuristic algorithms can serve as appropriate alternatives.

Several researchers have introduced and implemented different algorithms to address this problem. Grandhi and Venkayya [5] proposed an algorithm, where an optimality criterion based on uniform Lagrangian density was utilized for resizing and scaling procedure to locate the constraint boundary. Multiple equality and inequality frequency constraints were considered on truss structures.

Sedaghati et al. [6] employed an integrated finite element force method frequency analysis together with a mathematical programming technique to optimize both frame and truss structures with frequency constraints.

Wang et al. [7] used an optimality criterion based on the differentiation of the Lagrangian function, where an infeasible point with the minimum weight increment was considered as the starting point. Simultaneous shape and size optimization of three-dimensional truss structures with multiple natural frequency constraints was addressed.

Lingyun et al. [8] introduced a hybridization of the simplex search and genetic algorithms called niche genetic hybrid algorithm (NGHA) with a float point codification to tackle the problem of mass minimization of trusses with frequency constraints. This algorithm divides the initial population into groups called niches. Each niche is then explored by a simplex algorithm to obtain the optimum solutions. The procedure is continued by searching amongst the best niches using the simplex method in subsequent iterations.

Gomes [9] utilized the well-known Particle Swarm Optimization (PSO) algorithm to investigate simultaneous shape and size optimization of truss structures with multiple frequency constraints.

Lin et al. [10] proposed a bi-factor  $\alpha - \beta$  algorithm based on the Kuhn-Tucker criteria for the minimum weight design of structures under statical and dynamical constraints. Sizes of the elements and/or the coordinates of the nodes were considered as the design variables.

As a developed type of meta-heuristic algorithm, the Artificial Bee Colony algorithm (ABC) is introduced by Karaboga and Basturk for solving unconstrained and constrained function optimization problems [11-12].

## 2. Statement of the optimization problem

In truss sizing optimization problems, the structural topology is prescribed in advance and kept fixed in the solution process. Element cross-sectional areas are referred to as the design variables and are assumed to change continuously in this context. The natural frequencies are posed as constraints for the structure to avoid resonance with the external excitations.

In addition, each variable may be constrained within an acceptable region. Thus, the optimization problem can be defined mathematically as follows:

$$\text{Minimize } Mer(A) = F(A) \times f_{penalty} \quad (1)$$

$$F(A) = \sum_{e=1}^n L_e \rho_e A_e \quad (2)$$

subject to:

$$\omega_i \geq \omega_i^*, \quad i = 1, 2, \dots, q_1 \quad (3)$$

$$\omega_i \leq \omega_i^*, \quad i = q_1 + 1, \dots, q \quad (4)$$

$$A_j^{lower} \leq A_j \leq A_j^{upper} \quad (5)$$

where Mer is the merit function, F(A) is the weight of structure and is expressed in Eq.(2)  $L_e$ ,  $\rho_e$  and  $A_e$  are the length, material density, and cross-sectional area of the eth element, respectively. Here n is the total number of elements in the structure and k the number of the independent design variables.  $f_{penalty}$  is the penalty function which results from the violations of the constraints corresponding to the response of the structure. Inequality (2) represents that some natural frequencies  $\omega_i$ ,  $i = 1, 2, \dots, q_1$  numbering  $q_1$ , should exceed the prespecified lower limits. In inequality (3), other natural frequencies  $\omega_i$ ,  $i = 1, 2, \dots, q - q_1$  must be less than the corresponding upper limits, respectively. Inequality (4) indicates that the design variable  $A_j$ , including a sizing variable, should take a value between its lower bound  $A_j^{lower}$  and upper bound  $A_j^{upper}$ , respectively.

The penalty function is defined as [4]:

$$f_{penalty} = (1 + \varepsilon_1 \cdot v)^{\varepsilon_2}, \quad v = \sum_{i=1}^q v_i \quad (6)$$

Where q is the number of frequency constraints. If the ith constraint is satisfied  $v_i$  will be taken as zero, if not it will be taken as:

$$v_i = \left| 1 - \left( \frac{\omega_i}{\omega_i^*} \right) \right| \quad (7)$$

The parameters  $\varepsilon_1$  and  $\varepsilon_2$  are selected considering the exploration and the exploitation rate of the search space.

### 3. The behavior of real honeybees in their natural environment

Honeybees live in social units called colonies, depending on the time of year a typical colony includes a single queen, thousands of semi-sterile female workers and a few thousand males (drones).

Adult workers are responsible for executing all the tasks associated with colony living such as; processing and storing food, cleaning cells, feeding larvae (nursing behavior), secreting wax and constructing combs, and guarding the entrance [13]. When the female bees are about 3 weeks old, they begin foraging, cease performing most tasks within the hive and usually remain foragers for the rest of their lives [13]. Foragers are able to utilize a large number of flower nectar (food sources) in multiple directions up to 12km from the hive, but mostly they fly within a 3km radius [14].

In a colony, the female bees start the foraging process by randomly searching for the promising flower patches. After finding a food source, the bee loads up with nectar then returns to the hive and unloads her nectar. Then, she may inform her nest mates about her findings through the movements known as the “waggle dance.” This dance gives three pieces of information regarding the flower patch; the direction in which it can be found, distance from the hive and quality rating [15,16]. In a decentralized but intelligent fashion, some of the bees decide to follow their nest mates who have performed the waggle dance; others, to maximize their nectar intake, search for the food source without following the dancers. This means that each bee can follow one of three options after unloading the nectar: (a) abandon the food source and search for another promising flower patch, (b) continue to forage at the food source without recruiting nest mates, or (c) perform the waggle dance to recruit nest mates before returning to the food source.

Each bee follows one of the above options based upon the food level of that nectar source. If a bee finds a nectar source which is above a certain limit, she follows option (c). If the nectar source is average, the bee goes to forage at the food source without recruiting nest mates. Otherwise, the bee continues to search for promising nectar sources as option (a). The main goal of the foragers is to locate the most abundant nectar source [15, 16].

#### 4. Modeling artificial bee colony behavior

In the ABC algorithm, each food source exploited by the bees represents a possible solution to given optimization problem. The location and amount of nectar from the flower patch correspond to the design variables and the fitness function, respectively. All the worker bees ( $N$ ) leave the hive to search for promising flower patches. After the workers bees return to hive with a certain amount of nectar, the first half ( $SN$ ) that found the best food sources become “employed bees.” The remainder of the bees watches the waggle dance to decide which of the employed bees followed. These bees, which watch the waggle dance, are called “unemployed bees” or “onlooker bees.”

Each food source has only one employed bee; that is, the number of food sources is equal to the number of employed bees. The number of unemployed bees which will fly to a food source depends on the amount of nectar at the source. The unemployed bees choose a food source according to the quality of the nectar. More unemployed bees prefer to visit an abundant nectar source while fewer or no unemployed bees choose the food source having less nectar than others. It means that unemployed bees select a food source according to a probability proportional to the amount of nectar to be found at the food source [17]. The probability  $p_i$  for that source  $i$  is calculated in the following way:

$$p_i = \frac{1/W_i(A)}{\sum_{j=1}^{sn} 1/W_j(A)} \quad (8)$$

A candidate food source is created from the neighborhood of the old food source. It means that the ABC algorithm uses the old food source  $(A_j^{old})$  to search for a candidate food source  $(A_j^{new})$ .

Numerically, the location of a candidate food source  $i$  is determined as:

$$A_j^{new} = A_j^{old} + \Phi(A_j^{old} - A_k^{old}) \quad (9)$$

Where  $\Phi$  is a random number between -1 and 1.  $A_j^{new}$  is an updated design variable. The left hand subscripts represent the solution number (food source,  $i=1, 2, 3, \dots, SN$ ) while the right hand script denote the design variable number ( $j=1, 2, 3, \dots, D$ ).  $k$  is a randomly chosen integer number but cannot be equal to  $i$ .  $A_k^{old}$  plays an important role in the ABC convergence behavior since it is employed to control the exploration abilities of the bees. It directly influences the location of the new food source, which is based on the previous location of other food sources in the regions of the design space. Every employed bee determines a new food source in the neighborhood of its currently associated food source and evaluates the new amount of food as shown in Eq. (1). If the food level in the new location is better than the old one, the new position becomes the food source; otherwise, the old location is maintained as the best food source.

As mentioned above, the ABC algorithm is iterative. If there is no improvement in the amount of nectar from a food source after a predefined iteration (LIMIT), this food source is discarded by its employed bee. These employed bees become "scout bees" acting as the colony's explorers [20]. Concerned primarily with finding any kind of nectar source and they do not have any guidance as to where to look for food. The scouts may accidentally discover rich, entirely unknown food sources and when this happens the scout bee becomes an employed bee. A new location found by a scout bee  $i$  is calculated as:

$$A_j^{new} = A_j^l + \lambda(A_j^u - A_j^l) \quad (10)$$

Where  $\lambda$  is a random number between 0 and 1.  $A_j^l$  and  $A_j^u$  are the lower and upper bounds of the  $j$ th variable, respectively.

## 5. Numerical Examples

### 5.1 A 10- bar truss

Size optimization of the 10-bar planar truss shown in Fig. 1 is considered. This is a well-known problem in the field of weight optimization of the structures with frequency constraints. The cross-sectional area of each of the members is considered to be an independent variable. A non-structural mass of 454.0 kg is attached to the free nodes. Table 1 shows the material properties, variable bounds, and frequency constraints for this example. This problem has been investigated by Grandhi and Venkayya [5] using the optimality algorithm. Sedaghati et al. [6] have solved it by sequential quadratic programming and the finite element force method. Wang et al. [7] have used an evolutionary node shift method and Lingyun et al. [8] have used a niche hybrid genetic algorithm to optimize this structure. Gomes has analyzed this problem using the particle swarm algorithm [9]. Kaveh and Zolghadr [2] have investigated the problem using the Standard and an enhanced CSS.

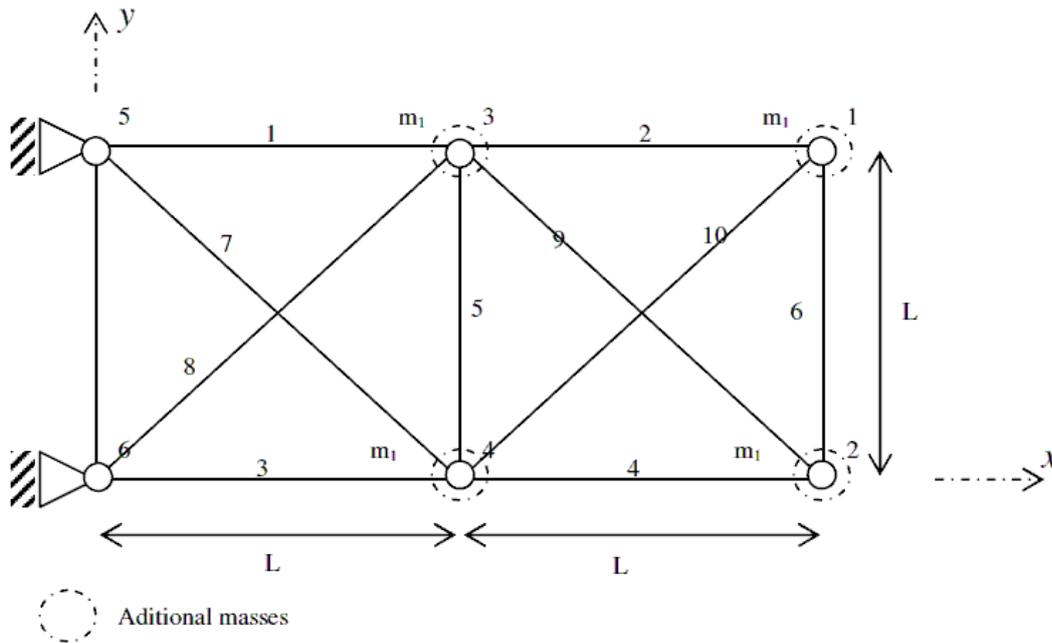


Fig 1. A 10-bar planar truss[8]

Table 1. Material properties, variable bounds and frequency constraints for 10-bar truss structure

Property/unit	Value
$E$ (modulus of elasticity)/ $N/m^2$	$6.98 \times 10^{10}$
$\rho$ (Material density)/ $kg/m^3$	2770.0
Added mass/kg	454.0
Design variable lower bound/ $m^2$	$0.645 \times 10^{-4}$
$L$ (Main bar's dimension)/m	9.144
Constraints on first three frequencies/Hz	$\omega_1 \geq 7, \omega_2 \geq 15, \omega_3 \geq 20$

Table 2 represents the design vectors and the mass of the corresponding structures obtained by different researchers.

**Table 2: Optimal design cross sections ( $cm^2$ ) for several methods for the ten bar planar truss (weight does not include added masses)**

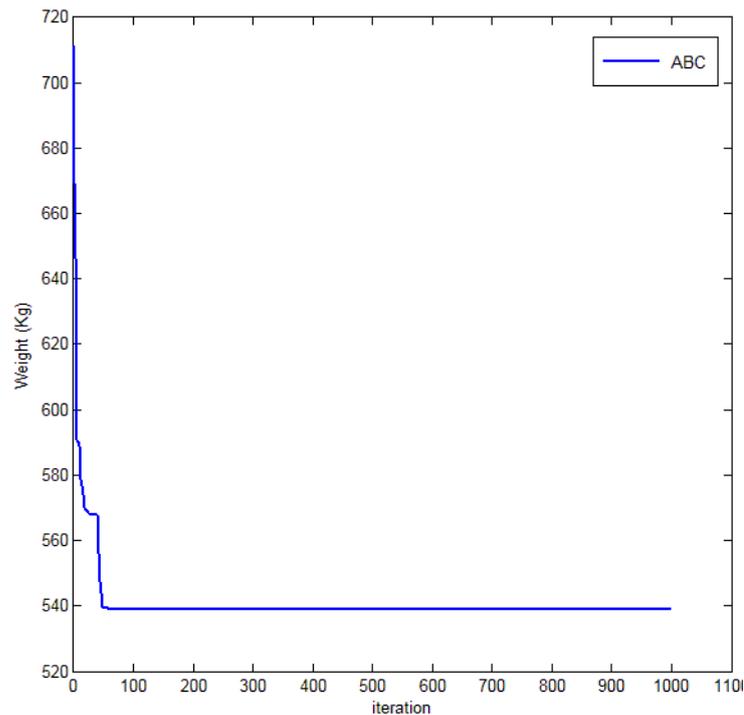
Element Number	Wang [7]	Gomes [9]	Sedaghati [6]	This Work
$A_1$	32.456	37.712	38.245	31.053
$A_2$	16.577	9.959	9.916	8.101
$A_3$	32.456	40.265	38.619	31.055
$A_4$	16.577	16.788	18.232	10.190
$A_5$	2.115	11.576	4.419	1.019
$A_6$	4.467	3.955	4.419	6.007
$A_7$	22.810	25.308	20.097	20.164
$A_8$	22.810	21.613	24.097	20.169
$A_9$	17.490	11.576	13.890	10.138
$A_{10}$	17.490	11.186	11.452	10.135
Weight (Kg)	553.8	537.98	535.8	529.376

Table 3 represents the natural frequencies of the optimized structures obtained by different researchers. It can be seen that all of the constraints are satisfied with an exception of the structure obtained by Sedaghati et al. [6].

**Table 3. Natural frequency (HZ) of the optimized structures (the ten- bar planar truss)**

Frequency number	Wang [7]	Gomes [9]	Sedaghati [6]	This work
1	7.011	7.000	6.992	7.008
2	17.302	17.786	17.599	17.147
3	20.001	20.000	19.973	20.001
4	20.100	20.063	19.977	20.225
5	30.869	27.776	28.173	28.4248
6	32.666	30.939	31.029	31.420
7	48.282	47.297	47.628	48.090
8	53.306	52.286	52.292	52.432

Fig. 2 shows the convergence curve of the best result obtained by ABC algorithm for the 10-bar planar truss.



**Figure 2. The convergence curve for ABC algorithm (the ten- bar planar truss)**

### 5.2 A 72- bar space truss

Topology and element numbering of a 72-bar space truss is depicted in Fig. 3. The elements are classified in 16 design groups according to Table 4. Four non-structural masses of 2270 kg are attached to the nodes 1–4.

This is a mere size optimization. This example has been solved by Sedaghati [19] using the force method (FM). Gomes [9] has investigated the problem using the Particle Swarm Optimization. Kaveh and Zolghadr [2] have investigated the problem using the standard and enhanced CSS. Material properties, variable bounds, frequency constrains and added masses are listed in Table 5. The optimal cross-sectional areas for the 72-bar space truss, obtained by different researchers, together with the results gained by the proposed algorithm are shown in Table 6. It can be seen that optimization by the proposed algorithm obtained the best result so far.

**Table 4. Member linking detail for the 72 bar space truss**

Group number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Members	1-4	5-12	13-16	17-18	19-22	23-30	31-34	35-36	37-40	41-48	49-52	53-54	55-58	59-66	67-70	71-72

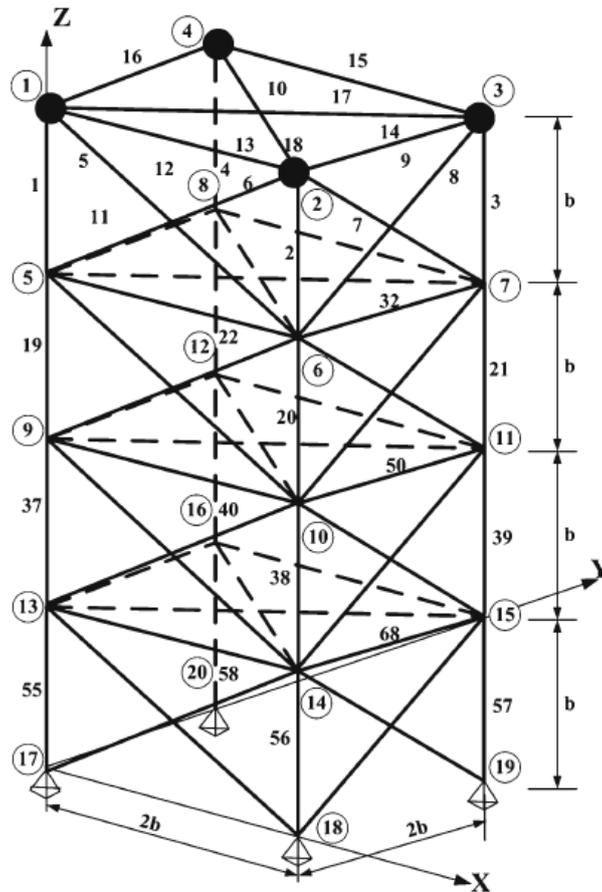


Fig 3. A 72-bar space truss

Table 5. Material properties, variable bounds and frequency constraints  
for 72-bar space truss

Property/unit	Value
$E$ (modulus of elasticity)/ $N/m^2$	$6.98 \times 10^{10}$
$\rho$ (Material density)/ $kg/m^3$	2770.0
Added mass/kg	2270.0
Design variable lower bound/ $m^2$	$0.645 \times 10^{-4}$
Constraints on first three frequencies/Hz	$\omega_1 = 4, \omega_3 \geq 6$

**Table 6: Optimal design cross sections ( $cm^2$ ) for several methods for the 72 bar space truss (weight does not include added masses)**

Group number	Sedaghati [19]	Kaveh [2]	Gomes [9]	This Work
1	3.499	2.528	2.987	3.315
2	7.932	8.704	7.849	8.229
3	0.645	0.645	0.645	0.645
4	0.645	0.645	0.645	0.645
5	8.056	8.283	8.765	8.186
6	8.011	7.888	8.153	7.976
7	0.645	0.645	0.645	0.645
8	0.645	0.645	0.645	0.645
9	12.812	14.666	13.450	13.367
10	8.061	6.793	8.073	7.880
11	0.645	0.645	0.645	0.645
12	0.645	0.645	0.645	0.645
13	17.279	16.464	16.684	17.351
14	8.088	8.809	8.159	8.061
15	0.645	0.645	0.645	0.645
16	0.645	0.645	0.645	0.645
Weight (Kg)	327.605	328.814	328.823	327.381

Table 7 represents the natural frequencies obtained by various methods for the 72- bar space truss.

Fig. 4 shows the convergence curve of the best result obtained by ABC algorithm for the 72-bar space truss.

**Table 7. Natural frequency (HZ) of the optimized structures (the 72 bar space truss)**

Frequency number	Sedaghati [19]	Kaveh [2]	Gomes [9]	This Work
1	4.000	4.000	4.000	4.000
2	4.000	4.000	4.000	4.000
3	6.000	6.006	6.000	6.001
4	6.247	6.210	6.219	6.247
5	9.074	8.684	8.976	9.047

## 6. Conclusions

In this paper we optimize two samples of structures include a planar 10-member truss and a space 72-member truss. The trusses were optimized under frequency constrains with Artificial Bee Colony algorithm (ABC). Conclusions show that ABC has better qualities than other algorithms, also the accuracy of this algorithm is too high and it prevent the converge of algorithm to local optima. In other words this algorithm has high potential to solve optimized problems under limited conditions.

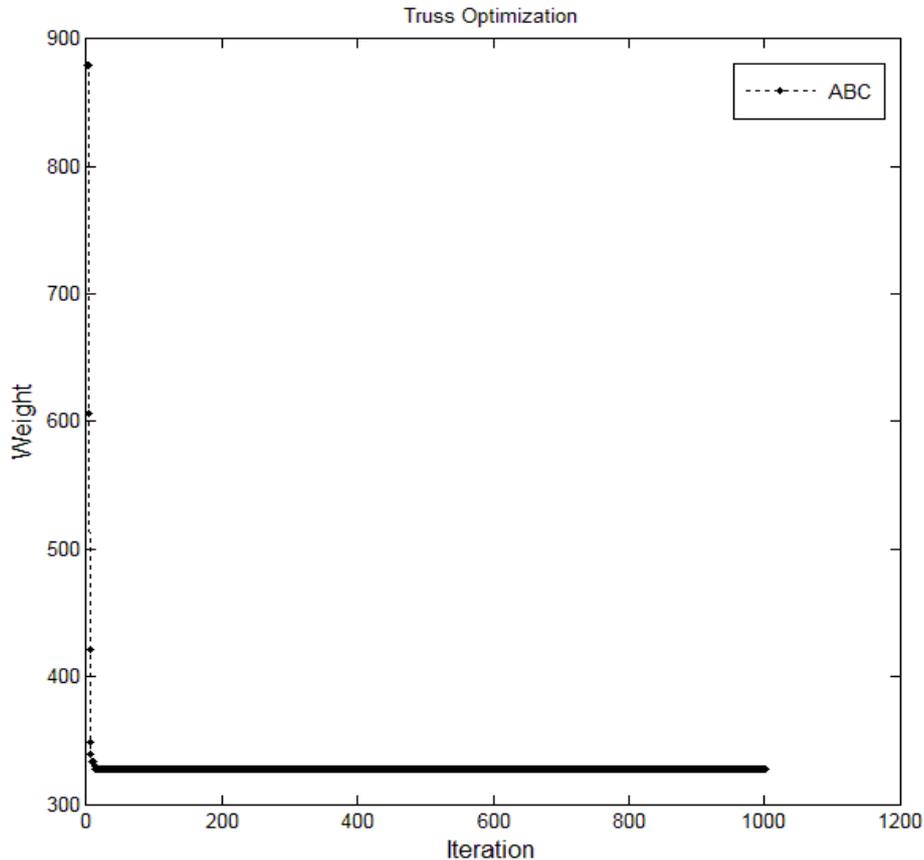


Figure 4. The convergence curve for ABC algorithm (the 72 bar space truss)

## References

- [1] S. Gholizadeh, E. Salajagheh, and P. Torkazadeh, *Journal of Sound and Vibration*. 312, 316-331 (2008).
- [2] A. Kaveh and A. Zolghadr, *Asian J Civil Eng.* 12, 487-509 (2011).
- [3] R. V. Grandhi, *AIAA J.* 31, 12 2296-2303 (1993).
- [4] A. Kaveh and S. Talatahari, *Acta Mech.* 213, 267-89 (2010).
- [5] R. V. Grandhi and VB. Venkayya, *AIAA J.* 26, 858-66 (1988).
- [6] R. Sedaghati, A. Suleman, Tabarrok B, *AIAA J.* 40, 382-8 (2002).
- [7] D. Wang, WH. Zha and JS. Jiang, *AIAA J.* 42, 1452-6 (2004).
- [8] W. Lingyun, Z. Mei, W. Guangming and M. Guang, *J Comput Mech.* 25, 361-368 (2005).
- [9] MH. Gomes, *Expert Sys Appl.* 38, 957-968 (2011).

- [10] Lin. JH, Chen. WY and Yu. YS, *Comput Struct.* 15, 5 507–515 (1982).
- [11] D. Karaboga and B. Basturk, *Applied Soft Computing.* 8, 1, 687–697 (2008).
- [12] D. Karaboga and B. Basturk, *J Glob Optim.* 39, 459–471 (2007).
- [13] M. Sonmez, *Applied Soft Computing.* 11, 2406-2418 (2011).
- [14] D.T. Pham, A. Granbarzadeh, E. Koc, , S. Otri, S. Rahim and M. Zaidi, Editors. *Innovation Production Machines and System Virtual Conference*, (2006).
- [15] D. Teodorovic and M.D. Orco, *Advanced or an IA Methods in Transportation.* 51–60 (2005).
- [16] D.T. Pham and E. Koc, A. Granbarzadeh and S. Otri, Editors. *Proceedings of 5th International Symposium of Intelligence Manufacturing Systems.* (2006), Sakarya, Turkey.
- [17] A. Singh, *Applied Soft Computing.* 9, 2, 625– 631 (2009).
- [18] C. Camp, B. Bichon and S.P. Stovall, *Journal of Structural Engineering.* 131, 3, 369–379 (2005).
- [19] R. Sedaghati, *Int J Solids Struct.* 42, 5848–5871 (2005).