**Structural Engineering and Mechanics**, *Vol. 62, No. 3 (2017) 259-272*
DOI: https://doi.org/10.12989/sem.2017.62.3.259


# Optimum design of steel space structures using social spider optimization algorithm with spider jump technique

Ibrahim Aydogdu[*1], Perihan Efe[2a], Metin Yetkin[2b] and Alper Akin[3a]

[1]*Department of Civil Engineering, Akdeniz University, Dumlupinar Bvld., 07058, Antalya, Turkey*
[2]*Department of Civil Engineering, Balikesir University, Balikesir, Turkey*
[3]*Trinity Meyer Utility Structures, Memphis, TN, USA*

*(Received August 28, 2016, Revised March 11, 2017, Accepted March 15, 2017)*

**Abstract.** In this study, recently developed swarm intelligence algorithm called Social Spider Optimization (SSO) approach and its enhanced version of SSO algorithm with spider jump techniques is used to develop a structural optimization technique for steel space structures. The improved version of SSO uses adaptive randomness probability in generating new solutions. The objective function of the design optimization problem is taken as the weight of a steel space structure. Constraints' functions are implemented from American Institute of Steel Construction-Load Resistance factor design (AISC-LRFD) and Ad Hoc Committee report and practice which cover strength, serviceability and geometric requirements. Three steel space structures are optimized using both standard SSO and SSO with spider jump (SSO_SJ) algorithms and the results are compared with those available in the literature in order to investigate the performance of the proposed algorithms.

**Keywords:** optimization; swarm intelligence; metaheuristic; social spider optimization; space frame; space truss

## 1. Introduction

Structural optimization of steel structures deals with finding out the most appropriate steel sections that are to be assigned to member groups of the steel frame such that the structure satisfies the code requirements which cover strength, deflection, stability and geometric constraints while its weight is the minimum. Formulation of such optimization problem results in highly nonlinear discrete programming problem solution of which is not easy to attain. One of the techniques available within mathematical programming algorithms is called sequential linear discrete programming approach (Rao 2009). In this algorithm all nonlinear functions are linearized about a selected initial design point using a first-order Taylor's series expansion. Discrete variables are further re-defined by assigning zero-one variables to each of them and the problem is solved by using integer programming problem. The algorithm is cumbersome and its convergence depends on goodness of the initial design point. For large size optimization problems the algorithm is not efficient and most of the time it gives rise to convergence difficulties (Rao 2009). On the other hand metaheuristic algorithms provide an ease for finding the solution of such complex and large size discrete programming problems. This is why these methods became very common and popular in obtaining solution of discrete engineering design optimization problems problems (Saka 2003, Saka 2007, Aydogdu and Saka 2009, Hasançebi Çarbaş *et al*. 2009, Aydoğdu 2010, Aydoğdu *et al*. 2012, Aydoğdu *et al*. 2012, Degertekin 2012, Azad and Hasançebi 2013, Degertekin and Hayalioglu 2013, Saka and Geem 2013, Aydogdu and Akin 2014, Aydoğdu and Akın 2014, Azad and Hasançebi 2014, Azad *et al*. 2014, Saka 2014, Alqedra *et al*. 2015, Artar and Daloglu 2015, Artar and Daloglu 2015, Artar and Daloglu 2015, Azad and Hasançebi 2015, Gholizadeh and Poorhoseini 2015, Hasançebi and Azad 2015, Arafa *et al*. 2016, Artar 2016, Tunca and Çarbaş 2016, Aydogdu 2017, Carbas and Aydogdu 2017).

Swarm intelligence metaheuristic techniques mimic the collective behavior of swarms and the complex interaction between members in a swarm without surveillance. These techniques target producing high-quality solutions by focusing the population based on performance of the algorithm and using its advantages e.g. scalability, error toleration, adjustment, self-determination and similarity (Kassabalidis *et al*. 2001, Kennedy *et al*. 2001, Fister Jr, *et al*. 2013, Saka *et al*. 2013, Yang *et al*. 2013, Saka *et al*. 2016, Vardhini and Sitamahalakshmi 2016). Swarm intelligence models both learned and innate natural behaviors that emerged by transferring information among individuals within a population. The collective intelligent behavior, known as the aggregative conduct of individuals in a swarm, has attracted the attention of scientists over the last decades. Researchers have proposed many swarm based optimization algorithms such as ant colony optimization (ACO) (Dorigo *et al*. 1996), which simulates foraging behaviors of an ant colony, particle swarm optimization algorithm (PSO) (Shi and Eberhart 1998), which imitates the social behavior of bird flocking and fish schooling, artificial bee colony (ABC) (Karaboga 2005), which models the cooperative behavior of bee colonies, bacterial foraging

*Corresponding author, Ph.D.
E-mail: aydogdu@akdeniz.edu.tr
[a]Ph.D.
[b]M.sc.



ISSN: 1225-4568 (Print), 1598-6217 (Online)

optimization algorithm (Passino 2002), which emerged from the social foraging behavior of bacteria, cuckoo search (CS) algorithm (Yang and Deb 2009), which emulates lifestyle of cuckoo birds, and firefly (FF) algorithm (Yang 2010), which simulates the mating behavior of firefly insects etc.

Recently, a new swarm intelligence algorithm and an innovative approach called Social Spider Optimization (SSO) algorithm has been introduced by Cuevas *et al*. (Cuevas *et al*. 2013). The SSO algorithm inspired by the cooperative behavior of the social spiders and especially their foraging behavior as the cooperative movement of the spiders towards the food source position. In the SSO algorithm, individuals emulate a group of spiders which interact with each other based on the biological laws of the cooperative colony. The spiders move towards the food source by cooperating with each other. To determine the potential location of a prey, the vibrations propagated on the web are received and analyzed (Campón 2007). Contrary to most of the existing population based methods, SSO algorithm utilizes males and females as two different search agents (spiders) and a set of different evolutionary operators mimic different cooperative behaviors administrating each individual depending on gender. That approach helps to mimic the collaborative rules of the colony more realistically and avoid critical flaws in the computational mechanism. The SSO algorithm is described in Section 3 in detail.

Despite being a new optimization algorithm, the SSO algorithm has been applied to many optimization problems such as: optimal economic dispatch of thermal power unit problem (Esapour *et al*. 2015), design of plug-in electric vehicle (Kavousi-Fard *et al*. 2015), wind tribune systems (Khorramniah *et al*. 2015), feed forward neural networks learning (Mirjalili *et al*. 2015), optical flow methods parameters (Pereira *et al*. 2015), field weakening control of a separately excited DC motor (Hameed *et al*. 2016) and energy theft detection (Yu and Li 2016). However, no studies on the application of the SSO algorithm for optimum design of space structure problems has been found in the literature. Hence, this study is the first to apply the SSO algorithm to structural design optimization problems.

In this study, structural design optimization technique is presented for discrete optimum design of steel space structures based on the recent swarm intelligence technique of SSO algorithm. Firstly Standard SSO is used to design the steel space structures in order to investigate its efficiency and convergence capability. Later this algorithm is enhanced by utilizing the spider jump technique (SSO_SJ) which provides better exploration and exploitation capability to escape from local optimum. The explorability and escapability of the local minima of SSO algorithm as well as the diversity of the population are improved via this technique. The total structural weight of the frame is considered as the objective function which is subjected to constraints in the form of strength, displacement and serviceability requirements derived from the American Institute for Steel Construction (AISC) Load and Resistance Factor Design (LRFD) (LRFD 2000). The efficiencies of the proposed algorithms are numerically investigated using three steel space structures that are designed for minimum weight by using the existing metaheuristic algorithms: ABC, ACO, PSO, CS, FF, Artificial Bee Colony with Levy Flight (LFABC), Harmony Search (HS), Dynamic Harmony Search (DHS), Hybrid Teaching Learning Based Optimization-Harmony Search (hTLBO-HS) and Enhanced Firefly (EFF) algorithms. The solutions obtained in this study reveal that the SSO algorithm shows average performance, spider jump technique significantly improves the performance of SSO algorithm and the SSO_SJ performs better in terms of reporting optimum results over all other metaheuristic algorithms previously employed.

The rest of this paper is organized as follows. Section 2 presents the mathematical model for steel space frames and describes the objective function and constraints in the optimization problem. As mentioned before, in Section 3, a novel swarm algorithm SSO and its improved version SSO_SJ algorithm are both introduced. Section 4 describes the simulation results of the SSO algorithm on the numerical design examples and the comparison with other popular metaheuristics. Finally, conclusions are drawn in Section 5.

## 2. Mathematical model of structural optimization problem

Optimum design of steel space structures entails choosing optimum sections for its members from available steel sections list to satisfy the serviceability, strength and geometric requirements stated in the design codes. In the meantime, the structure is constructed economically.

Generally, the optimization problems are threefold which are objective functions, design variables and constraints. The mathematical model of the design optimization problems dependent on the steel design code is considered in the formulation and described in the following.

### *2.1 Objective function and design variables*

The structural optimization problems generally are intended to minimize the material cost of the structure. For steel structures, the material cost is directly related to the weight of the structure. Therefore, the objective function of the optimization problem is selected as the weight of the structure expressed as:

Find the steel sections of the optimum design

$$Minimize\ W(\boldsymbol{X}) = \sum_{i=1}^{NG} m_i \cdot \left( \sum_{j=1}^{n_i} l_j \right) \tag{1}$$

where: $W$ is the structure weight, $m_i$ is the unit weight of the steel section adopted for the $i^{th}$ group, $n_i$ is the total member number for the group $i$, $N_G$ is the total number of design groups, $l_j$ is the length of $j^{th}$ member for the group $i$, and $\boldsymbol{X}=[X_1,X_2,X_3\ldots X_{NG}]$ is the vector of the sequence number of steel sections selected for design groups from the standards treated as the design variable of the optimization problem.

### 2.2 Design constraints

#### 2.2.1 Strength constraints

In strength constraints, each frame member of the structure has sufficient strength to resist the internal forces developed due to factored external loading. The moment resisting frame and the pin-jointed structures are used in the study. The strength constraints of each member of these structures are described in Eqs. (2)-(3).

$$g_s(\boldsymbol{X}) = \frac{P_u}{\emptyset_c P_n} + \frac{8}{9}\left(\frac{M_{ux}}{\emptyset_b M_{nx}} + \frac{M_{uy}}{\emptyset_b M_{ny}}\right) \\ -1.0 \leq 0 \; for \; \frac{P_u}{\emptyset P_n} \geq 0.2 \tag{2}$$

$$g_s(\boldsymbol{X}) = \frac{P_u}{2\emptyset_c P_n} + \left(\frac{M_{ux}}{\emptyset_b M_{nx}} + \frac{M_{uy}}{\emptyset_b M_{ny}}\right) - 1.0 \leq 0 \\ for \; \frac{P_u}{\emptyset P_n} < 0.2 \tag{3}$$

where: $M_{nx}$ and $M_{ny}$ are the nominal moment capacities at primary and secondary axis respectively, $M_{ux}$ and $M_{uy}$ are the design moments exposing the member, $\emptyset_c$ and $\emptyset_b$ are the resistance factors for compression and bending. $P_n$ is the axial load capacity of the member, and $P_u$ is the ultimate axial force exposing to the member. $P_n$ and $P_u$ can be tension or compression. The moment capacities of the frame structures are calculated considering geometric non-linearity which is carried out as an alternative way expressed in Chapter C of the LRFD-AISC (LRFD 2000). In order to compare with literature studies, geometric non-linearity is not taken into account for the pin-jointed dome.

#### 2.2.2 Displacement donstraints

For the frame structures deflection, the inter-story drift and top-story drift limitations are contemplated which are given in following equations.

$$g_d(\boldsymbol{X}) = \frac{\delta_{j,l}}{\delta_{al}} - 1.0 \leq 0 \\ j = 1,2,\ldots,n_{sm} \quad l = 1,2,\ldots,n_{lc} \tag{4}$$

$$g_{td}(\boldsymbol{X}) = \frac{\Delta_{j,l}^{top}}{\Delta_{al}^{top}} - 1.0 \leq 0 \\ j = 1,2,\ldots,n_{jtop} \quad l = 1,2,\ldots,n_{lc} \tag{5}$$

$$g_{id}(\boldsymbol{X}) = \frac{\Delta_{j,l}^{is}}{\Delta_{al}^{is}} - 1.0 \leq 0 \\ j = 1,2,\ldots,n_{st} \quad l = 1,2,\ldots,n_{lc} \tag{6}$$

where, $\delta_{j,l}$ and $\delta_{al}$ are the computed and allowable deflections of the member $j$ for the load case $l$, $\Delta_{j,l}^{top}$ and $\Delta_{al}^{top}$ are the computed and permissible top story drifts (TSD) of the joint $j$ for the load case $l$, $\Delta_{j,l}^{is}$ and $\Delta_{al}^{is}$ are the occurred and allowable inter story drifts for the story $j$ and the load case $l$, $n_{sm}$ is the total number of critical members for the deflection, $n_{lc}$ is the number of load cases defined in the design problem, $n_{jtop}$ is the number of joints on the top story, and $n_{st}$ is the story number of the structure. Allowable displacement limits are determined from the ASCE Ad Hoc Committee report (Ellingwood 1986). In the study, the accepted range of drift limits are 1/750 to 1/250 times the building height for the Top Story Drift (TSD), 1/500 to 1/200 times the story height for the inter story drift (ISD) and 1/500 to 1/200 times the beam length for the deflection.

For the pin-jointed structures, only the displacement limits of nodes are considered which is expressed as

$$g_{di}(\boldsymbol{X}) = \frac{\delta i_{j,l}}{\delta i_{al}} - 1.0 \leq 0 \\ j = 1,2,\ldots,n_{nd} \quad l = 1,2,\ldots,n_{lc} \tag{7}$$

where, $\delta i_{j,l}$ and $\delta i_{al}$ are the computed and allowable displacements of the joint $j$ for the load case $l$ and $n_{sm}$ is the total joint number of the structure.

#### 2.2.3 Geometric constraints

Geometric constraints are defined for only the frame structures which are explained as follows

$$g_{cc}(\boldsymbol{X}) = \sum_{i=1}^{n_{ccj}}\left(\frac{D_i^a}{D_i^b} - 1.0\right) + \sum_{i=1}^{n_{ccj}}\left(\frac{m_i^a}{m_i^b} - 1.0\right) \leq 0 \tag{8}$$

$$g_{bc}(\boldsymbol{X}) = \sum_{i=1}^{n_{j1}}\left(\frac{B_f^{bi}}{D^{ci} - 2t_b^{ci}} - 1.0\right) \leq 0 \; or \\ \sum_{i=1}^{n_{j2}}\left(\frac{B_f^{bi}}{B_f^{ci}} - 1.0\right) \leq 0 \tag{9}$$

where: $n_{cc,j}$ is the number of column-column connection type for the optimization problem, $m_i^a$ and $m_i^b$ are the unit weights of the upper and lower columns for the $i^{th}$ column-column connection, $D_i^a$ and $D_i^b$ are the depths of upper and lower columns for the $i^{th}$ column-column connection, $n_{j1}$ is the number of a connection type 1: beams are connected to the web of a column, $n_{j2}$ is the number of a connection type 2: beams connected to the flange of a column, $D^{ci}$ is a depth of a column, $t_b^{ci}$ and $B_f^{bi}$ are the flange thickness and flange width of a beam for the connection type 1, and $B_f^{ci}$ and $B_f^{bi}$ are the flange widths of the column and beam for the connection type 2 (see Fig. 1)

### 2.3 Fitness evaluation

Metaheuristic algorithms provide solution of unconstrained optimization problems. The constrained optimization problems are required to be transformed into unconstrained one by using a penalty function as given in Eq.(10). If the structure design does not satisfy constraints described in section 2.2 the total penalty (*Pen*) of the structure design calculated as follows.

$$Pen = \sum_{i=1}^{NC} C_i \, ; C_i = \begin{Bmatrix} 0 \; for \; g_i(\boldsymbol{X}) \leq 0 \\ g_i(\boldsymbol{X}) \; for \; g_i(\boldsymbol{X}) > 0 \end{Bmatrix} \tag{10}$$

Where $C_i$ is constraint violation of the $i^{th}$ constraint function, $g_i$ represents $i^{th}$ constraint function. According to

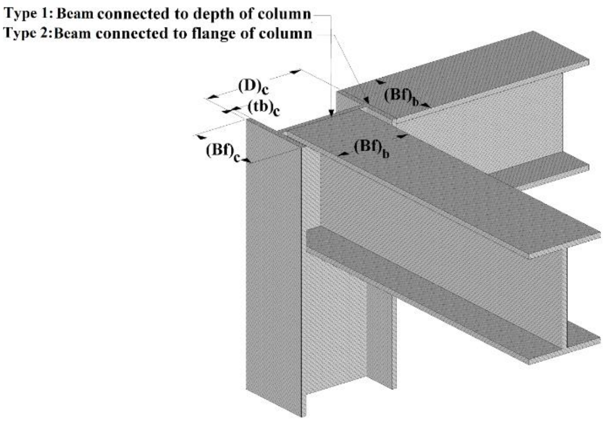


Fig. 1 Beam column connection of space frames

Pen value penalized weight of the structure ($W_p$) and fitness value ($J$) of the design are obtained using Eqs. (11)-(12) respectively.

$$W_p = W \cdot (1 + Pen)^2 \tag{11}$$

$$J = \frac{1}{W_p} \tag{12}$$

## 3. Social spider algorithm

### 3.1 Biological fundamentals of the social spider algorithm

Social insect societies are complex cooperative systems and they organize themselves within a set of constraints. In the nature, organized social colonies form as a result of interactions among the members such as constructing and utilizing their surroundings, protecting resources and allowing task specialization among society members (Oster and Wilson 1978; Hölldobler, Wilson *et al*. 1994).

The behavior of spiders is dependent upon their solidarity level that could be categorized into two groups as solitary and social (Aviles 1986, Lubin and Bilde 2007). Solitary spiders have poor relationships with the other group members and they maintain their own web. On the contrary, social spiders frequently gather in their spider web and contact with each other to form colonies (Burgess and Uetz 1982).

Social spider colony members are grouped into two categories: males and females. These members and the spider web are the two fundamental components of the colony. Social spider colonies have a female-dominant population and male members hardly reach the 30% of the total population (Aviles 1986). Different activities in the colony such as constructing and repairing the spider web, hunting and mating are fulfilled by each member depending on its gender (Yip *et al*. 2008). In order to perform these activities, the collective coordination among the members is required which is accomplished by small vibrations. The colony members decode these small vibrations to get information for their cooperative interactions and mating (Rypstra and Tirey 1991).

Cooperative interaction of the colony members is based on their gender. Female members are inclined to communicate with others through vibrations transferred from the spider web that represent the dominant members (Yip *et al*. 2008). Stronger vibrations are generated by heavier or closer colony members as the vibrations rely on their weight and distance. The female members are impacted by several factors such as procreation period, interest and randomization (Yip *et al*. 2008). The male population includes dominant and non-dominant members (Pasquet and Krafft 1992). Dominant males, which have better fitness characteristics than non-dominant males, are attracted to the nearest female while non-dominant males have a tendency to gather in the center of the male population in order to use their unutilized resources (Ulbrich and Henschel 1999).

Mating process carried out by the dominant males and females has a key role in sustaining the colony life. This process also allows for the information transfer between members (Jones and Riechert 2008). In order to produce off-springs, a dominant male mates with a number of females within a specific range (Elias *et al*. 2011).

### 3.2 Basic SSO algorithm

In this study, the basic collaborative rules of the social spider colony are simulated as part of the SSO algorithm. The spider web where the colony members communicate among each other is taken as the entire search space. The solution vector is defined as the member location in the colony. Vibrations which occur due to movement of spiders are used to transfer the information from one spider to another. The two main colony groups, males and females, are classified by their weights based on the fitness value of the objective function. Each member performs different evolutionary procedures that are dependent on coordinative behaviors of the colony members. In order to mimic this behavior, mathematical models for each operational process and proposed spider jump technique are presented in the following sections.

#### 3.2.1 Initialization

In the initialization phase, the SSO method determines the representation ratio of members from each gender. The majority of the spider colony is comprised of females. Therefore, the algorithm randomly determines the number of female members ($N_f$) within the range of 65-90% of the number of spiders ($N_s$) using Eq. (13). The algorithm assigns the remaining spiders as the male spiders as described in Eq. (14).

$$N_f = round[(0.9 - 0.25 \cdot rand) \cdot N_s] \tag{13}$$

$$N_m = N_s - N_f \tag{14}$$

where $N_m$ is total number of the male members, *rand* is a random number between [0, 1] and *round* is a function which rounds to integer number.

At the end of the process, the algorithm categorizes all spiders ($\boldsymbol{S}$) in two groups: female spiders ($\boldsymbol{f}$) and male spiders (***m***).

### *3.2.2 Fitness assignation*

In the SSO algorithm, the performances of all spiders are evaluated to assign them to different tasks. Therefore, the weight of each member ($w_i$) is calculated based on its fitness value (irrespective of gender) using following Eq. (15)

$$w_i = \frac{J(S_i) - worst_s}{best_s - worst_s} \quad i = 1, \ldots, N_s \tag{15}$$

where $J(S_i)$ is the fitness value of the $i^{th}$ spider with regard to the fitness function $J$ and the spider position. $worst_s$ and $best_s$ are the fitness values of the spiders which have the worst and the best fitness values in the colony, respectively. These values are determined according to the optimization problem type (maximization or minimization).

### *3.2.3 Female cooperative operator*

In a spider colony, female members change their position using the female cooperative operator. The position change of each female spider can be calculated using vibrations of two different spiders. The first one is the closest spider who is heavier and generates the vibration $Vibc_i$. The second spider is the best one in this colony whose vibration is symbolized as $Vibb_i$. In addition, a random movement parameter is added to the calculation of the movement. In this step, the final movement decision of inclination or revulsion is simulated as a stochastic process. If a randomly generated number is less than the threshold (PF), an inclination has occurred; or else, a revulsion has occurred. The female cooperative operator could be formulated as follows

$$\begin{aligned} f_i^{k+1} &= f_i^k + \alpha \cdot Vibc_i \cdot (S_c - f_i^k) + \beta \cdot Vibb_i \cdot \\ &\quad (S_b - f_i^k) + \delta\left(rand - \frac{1}{2}\right) \; for\; rand \le PF \\ f_i^{k+1} &= f_i^k - \alpha \cdot Vibc_i \cdot (S_c - f_i^k) - \beta \cdot Vibb_i \cdot \\ &\quad (S_b - f_i^k) + \delta\left(rand - \frac{1}{2}\right) \; for\; rand > PF \end{aligned} \tag{16}$$

where, $k$ represents the iteration number, $\alpha$, $\beta$ and $\delta$ are random numbers between [0,1]. The member $S_c$ and $S_b$ represent, respectively, the closest member to spider $i$ who is heavier and is the best individual in the colony.

The vibrations, $Vibc_i$ and $Vibb_i$, rely on the weight and distance of the related individuals. The perceived vibration, $Vibc_i$, by the member $i$ ($S_i$) is the result of the information transmitted by the individual $c$ ($S_c$). Individual $c$ must be the nearest member to $i$ and heavier than individual $i$ ($w_c > w_i$). The vibration, $Vibc_i$, is modelled according to the following Eq. (17)

$$Vibc_i = w_c \cdot e^{-d_{i,c}^2} \tag{17}$$

where the $d_{i,c}$ is the Euclidian norm of the members $i$ and $c$, such that $d_{i,c} = \| S_i - S_c \|$.

The perceived vibration, $Vibb_i$, by the individual $i$ ($S_i$) is the result of the information transmitted by the individual $b$ ($S_b$). Individual $b$ is holding the best weight (best fitness value) in the entire population $\boldsymbol{S}$. The vibration, $Vibb_i$, can be computed by the following Eq. (18)

$$\begin{aligned} Vibb_i &= w_b \cdot e^{-d_{i,b}^2} \\ with\; w_b &= max(w_k), \quad k \in \{1,2,..,N_s\} \end{aligned} \tag{18}$$

where the $d_{i,b}$ is the Euclidian norm of the members $i$ and $b$, such that $d_{i,b} = \| s_i - s_b \|$ and $max()$ is a mathematical function which finds a maximum value in the sequence.

In that female cooperative process, movement of each individual holds the inclination or revulsion of the local best individual $S_c$ and the global best $S_b$ seen so far. That inclination helps not only to avoid the quick concentration of members on one small region by preventing the members from moving towards the global best position, but also to encourage each individual to search around the local candidate in the region by exploring their specific neighborhood range ($S_c$). The use of this scheme makes the algorithm less susceptible to premature convergence by enhancing the exploitative behavior.

### *3.2.4 Male cooperative operator*

Based on the behavior of social spiders, male members move using the male cooperative operator. As previously stated, dominant males have better performance and are attracted to the nearest female spider in the spider web. On the other hand, non-dominant males are inclined to gather in the center of the male colony. To emulate that cooperative behavior, *D* and *ND* members are determined according to their weight (fitness). The males above the median weight, are assigned as the dominant male members and other males are labeled as non-dominant males. For such process, the male population is sorted in descending order by weight (fitness). Therefore, the weight of the member located in the middle is considered the median weight of males ($w_{median}$). By using that calculation process, movement of the male spiders can be computed by Eq. (19)

$$\begin{aligned} m_i^{k+1} &= m_i^k + \alpha \cdot Vibf_i \cdot (s_f - m_i^k) + \delta\left(rand - \frac{1}{2}\right) \\ &\quad if\; w_i > w_{median} \\ m_i^{k+1} &= m_i^k + \alpha \cdot \left(\frac{\sum_{h=1}^{N_m} m_h^k \cdot w_h}{\sum_{h=1}^{N_m} w_h} - m_i^k\right) \\ &\quad if\; w_i \le w_{median} \end{aligned} \tag{19}$$

where $S_f$ represents the nearest female to the male $S_i$. For non-dominant males, the part between parentheses corresponds to the median weight. The perceived vibration, $Vibf_i$, by the individual $i$ ($S_i$) is the result of the information transferred by the nearest female $S_f$ to male $i$. The value of vibration, $Vibf_i$, is calculated by Eq. (20)

$$Vibf_i = w_f \cdot e^{-d_{i,f}^2} \tag{20}$$

By using male cooperative operator, dominant and non-dominant male behaviors have been studied. Therefore, the dominant members are influenced by the others in order to provoke mating to create diversity into the population and *ND* members are influenced by the weighted mean of the male population $\boldsymbol{m}$ in order to avoid those very good members or extremely bad members.

#### 3.2.5 Mating operator

In the SSO algorithm, mating performed by $D$ members and the female spiders is an important operation for colony survival. A dominant male spider can mate with a set of females within a mating range ($r$) to produce an off-spring considering all the elements of the set. The mating operation is canceled if there is no female available in the range. The $r$ is defined as a radius which depends on the size of the search space and can be computed by the following model

$$r = \frac{\sum_{j=1}^{n} (p_j^{high} - p_j^{low})}{2n} \tag{21}$$

Where, $p_j^{low}$ and $p_j^{high}$ are lower and upper boundaries of the $j^{th}$ dimension. $n$ is the number of dimension of the optimization problem. During the mating process, each involved spider has the probability of influence on the new brood. New members are more likely to take on genetic qualities of the member having the best fitness.

The Roulette method is utilized based on the influence probability of each involved spider $i$ ($P_{si}$) that is defined as follows

$$P_{s_i} = \frac{w_i}{\sum_{j \in E} w_j} \tag{22}$$

where $E$ represents the set of spiders in the mating radius which are involved in the mating operation. Formed new spider $S_{new}$ is compared to the worst spider of the colony according to their weight (fitness) values. If the new member has better fitness than the worst member, the new member replaces the worst member. Otherwise, the new spider is not taken into consideration. In that process, the new spider assumes the gender and index of the replaced spider to ensure that the original ratio between female and male members is maintained. Because new individuals locally exploit the mating range, mating operation is an effective way to find better individuals.

### 3.3 Spider jump technique for improved SSO algorithm

This phase is not a part of the original SSO algorithm. This operation is proposed in this study to improve the performance of the SSO algorithm. The SSO algorithm is applied many times to the presented optimization problems given in section 4 with different search parameters described in section 4.1. According to obtained results, the basic version of the SSO algorithm does not show satisfactory performance for the presented optimization problem, especially in the exploration function of the algorithm. While utilizing the SSO algorithm, local convergence, stagnation and unbalance between the exploitation and exploration are observed for the undertaken optimization problems. The insufficient performance of the SSO algorithm does not allow the whole search space to be explored. In order to overcome this problem, the spider jump feature is added after the mating operation of the SSO algorithm.

In the spider jump technique, all members in the colony need to relocate (jump) instinctively because of curiosity, reproduction cycle and other random phenomena independently from cooperative behavior. In this phase, spiders in the colony randomly move to new locations to improve their fitness with a probability which depends on their weight. This probability is called the spider jump consideration probability ($P_{SJC}$) which is calculated as follows

$$\left(P_{SJC}\right)_i = 0.7 + 0.25 * w_i;\ i = 1,2,\ldots,N_S \tag{23}$$

If the $P_{SJC}$ of any spider is satisfied, the spider changes location randomly described as follows

$$\begin{gathered} S_{i,j} = p_j^{low} + \left(p_j^{high} - p_j^{low}\right) \\ if\ rand > \left(P_{SJC}\right)_i;\ i = 1,2,\ldots N_S;\ j = 1,2,\ldots,n \end{gathered} \tag{24}$$

where $S_{i,j}$ is the location of the spider $S_i$ in the $j^{th}$ dimension. If the fitness value of the $S_i$ at its new location is better than the fitness value of the worst spider, the new spider replaces the worst spider. Otherwise, the spider goes back to the old position. In that process, the spider at the new location assumes the gender and index from the worst spider to assure that the original rate between female and male spiders is maintained for the entire colony. Due to fact that new locations exploit the whole search space, this feature is an effective way to find better individuals.

### 3.4 Computational procedure of the SSO and SSO_SJ algorithms for the optimization problem

In this study, the SSO and improved SSO with spider jump algorithms are used in order to nominate optimum steel sections to design groups of the structure with the purpose of minimizing the weight of the structure. Each structure design is assigned to one spider and the spider cannot represent more than one design at the same time. In the algorithms; each coordinate of the spider locations represents the sequence number of the steel sections, location of the spiders represents the design vector, the number of dimension ($n$) defined in the optimization problem represent the number of design groups defined in the structure ($N_G$), design vectors of all siders represent the design pool, the best spider represents the spider having the lightest weight in the design pool, and the worst spider represents the spider having the heaviest structure (the worst fitness) in the design pool. Fitness value of the spider $J(S)$ with regard to the penalized weight of the structure is described in Eq. (11).

The lower and upper boundaries of all design groups are equal to 1 and *NofSec* respectively, where *NofSec* is total number steel sections available in the section table list. In this study, *NofSec* is equal to 272 for frame structures (272 W-shape steel sections) and 37 for pin-jointed structures (37 steel pipe sections). One iteration process in the algorithms contains the following processes: analysis & design of the structure, calculation of its constraints violations and penalized weights. The computational procedure for the SSO and SSO_SJ algorithms for the steel space structures can be summarized as follows:

Step 1: Number of the female and male members in the spider colony calculated using Eqs. (13)-(14). Initial structure design is generated randomly and assigned to the female and male members as follows

$$f_{ij} = 1 + (NofSec - 1) \cdot rand \quad for\ i = 1, \dots, N_f;\ for\ j = 1, \dots, N_G \tag{25}$$

$$m_{ij} = 1 + (NofSec - 1) \cdot rand \quad for\ i = 1, \dots, N_m;\ for\ j = 1, \dots, N_G \tag{26}$$

Fitness values of the structures are calculated using Eq. (12) and stored into the design pool.

Step 2: Fitness values of the best and the worst spiders in the colony are determined as follows

$$best_s = max(J(S_1), J(S_2), \dots, J(S_{N_S})) \tag{27}$$

$$worst_s = min(J(S_1), J(S_2), \dots, J(S_{N_S})) \tag{28}$$

Where, min() is a mathematical function which finds a minimum value in the sequence. Then, the weight of each spider is calculated according to its fitness value by using Eq. (15).

Step 3: Female spiders change positions (modify their structure designs) described as follows

*For* $i=1,\dots,N_f$
  *Determine design vectors of nearest and best spiders* ($S_c$ *and* $S_b$) *(section 3.2.3)*
  *Calculate* $Vibc_i$ *and* $Vibb_i$ *using Eqs. (17) and (18)*
  *If rand<PF then*
    *For* $j=1,\dots,N_G$
$$f_{i,j}^{k+1}=f_{i,j}^{k}+\alpha\cdot Vibc_i\cdot\left(S_c-f_{i,j}^{k}\right)+\beta\cdot Vibb_i\cdot\left(S_b-f_i^k\right)+\delta\left(rand-\frac{1}{2}\right)$$
    *End*
  *Else*
    *For* $j=1,\dots,N_G$
$$f_{i,j}^{k+1}=f_{i,j}^{k}-\alpha\cdot Vibc_i\cdot\left(S_c-f_{i,j}^{k}\right)-\beta\cdot Vibb_i\cdot\left(S_b-f_i^k\right)+\delta\left(rand-\frac{1}{2}\right)$$
    *End*
  *End*
*End* (29)

Fitness values of the modified structures are calculated using Eq. (12) and colony memory is updated.

Step 4: Male spiders change positions (modify their structure designs) described as follows

*Find the median male spider (median) and determine its weight* ($w_{median}$)
*For* $i=N_{f+1},\dots,N_s$
  *Determine design vectors of nearest female spider* ($S_f$) *(section 3.2.4)*
  *Calculate* $Vibf_i$ *using Eq. (20)*
  *If* $w_i>w_{median}$ *then*
    *For* $j=1,\dots,N_G$
$$m_{i,j}^{k+1}=m_{i,j}^{k}+\alpha\cdot Vibf_i\cdot\left(S_f-m_{i,j}^{k}\right)+\delta\left(rand-\frac{1}{2}\right)$$
    *End*
  *Else*
    *For* $j=1,\dots,N_G$
$$m_{i,j}^{k+1}=m_{i,j}^{k}+\alpha\cdot\left(\frac{\sum_{h=1}^{N_m} m_{h,j}^{k}\cdot w_h}{\sum_{h=1}^{N_m} w_h}-m_{i,j}^{k}\right)$$
    *End*
  *End*
*End* (30)

Fitness values of the modified structures are calculated using Eq. (12) and colony memory is updated.

Step 5: New members (structure designs) are generated using mating process described as follows

*For* $i=N_{f+1},\dots,N_s$
  *If* $w_i>w_{median}$ *then*
    *For* $j=1,\dots,N_G$
      *Calculate the range of mating (r) using Eq. (21)*
      *Find ID of the female spiders in the range r* ($\mathbf{E}=E_1,E_2,\dots,E_k$)
      *Add male spider to group E* ($E_{k+1}=i$)
      *Sort spiders descending order of their weights*
      *If k>0 then*
        *For* $i1=1,\dots,k+1$
          $Ps_{i1}=\frac{w_{Ei1}}{\sum_{i2=1}^{k+1} w_{Ei2}}$
        *End*
        *index=1*
        $pr=Ps_{index}$
        *While rnd<pr*
          *index=index+1*
          $pr=pr+Ps_{index}$
        *End*
        *If* $E_{index}<=N_f$
          $S_j^{new}=f_{E_{index},j}$
        *Else*
          $S_j^{new}=m_{E_{index},j}$
        *End*
      *Else*
        $S_j^{new}=m_{i,j}$
      *End*
    *End*
    *Calculate fitness of* $S^{new}$ ($J(S^{new})$)
    *If* $J(S^{new})>worst_s$
      *Replace the* $S^{new}$ *with the worst spider*
      *Assign gender of the new spider as gender of the worst spider*
    *End*
  *End*
*End* (31)

Step 6 (additional step for the SSO_SJ algorithm): In this step, all spiders perform jump movement according to their jump probability described as follows

*For* $i=1,\dots,N_s$
  *Calculate spider jump consideration probability* $(P_{SJC})_i$ *using Eq. (22)*

*If i<=Nf then*
*For j=1,...,$N_G$*
*If rand>$(P_{SJC})_i$ then*
$S_j^{mod}=1+(NofSec-1)\cdot rand$
*Else*
$S_j^{mod}=f_{i,j}$
*End*
*End*
*Else*
*For j=1,...,$N_G$*
*If rand>$(P_{SJC})_i$ then*
$S_j^{mod}=1+(NofSec-1)\cdot rand$ (32)
*Else*
$S_j^{mod}=m_{i,j}$
*End*
*End*
*End*
*Calculate fitness of the modified location $J(S^{mod})$*
*If $J(S^{mod})>worst_s$*
*Replace the $S^{mod}$ with the worst spider*
*Assign gender of the new spider as gender of the worst spider*
*End*
*End*

Step 7: If the stopping criteria reaching the maximum iteration number is satisfied, the algorithm is completed. Otherwise, steps 2 to 5 are repeated for the SSO algorithm and steps 2 to 6 are repeated for the SSO_SJ algorithms.

## 4. Design examples

In this section, SSO and SSO_SJ are utilized for the optimum design of three different steel space structures which are taken from available literature (Hasançebi *et al*. 2009, Aydoğdu 2010, Aydoğdu and Akın 2014, Akın and Aydoğdu 2015, Aydoğdu *et al*. 2016, Çarbaş 2016, Saka *et al*. 2016). First two design examples are space frames having five story & 105 member and four story & 428 member respectively. For these examples, the steel wide flange I profile list consisting of 272 ready sections is used to size the member of the structures. The last example is a 354 member pin-jointed geodesic dome. For this example, the optimization algorithms try to find solutions considering 37 pipe (P and XP) sections available in the LRFD-AISC(LRFD 2000).

### *4.1 Parameter study of the optimization problems*

In order to increase the efficiency of the swarm intelligence optimization algorithms, a balance between exploitation and exploration should be constructed. This ability can be succeed by finding suitable values of a search parameters of the algorithm. The values can vary from one optimization problem to another optimization problem. Therefore, the parameter study is mostly required for the different optimization problems. In this study, five story 105 member steel space frame is used for the parameter study. The example is optimized with the SSO and the SSO_SJ algorithms using different values of the number of spider (NS) (25, 50 and 100) and threshold probability for the attraction movement of female spider (PF) (0.3, 0.4, 0.5, 0.6, 0.7). Results obtained from the runs are shown in Tables 1 and 2. According to the results, most the appropriate values of the internal parameters are determined in Table 3.

Table 1 The best weights for the SSO algorithm with respect to different search parameters

| PF \ NS | 25 | 50 | 100 |
|---|---|---|---|
| 0 | 315.35 | 284.13 | 275.5 |
| 0.1 | 313.641 | 281.6 | 273.621 |
| 0.2 | 310.41 | 280.43 | 274.641 |
| 0.3 | 312.64 | 280.95 | 271.196 |
| 0.4 | 308.48 | 280.23 | 272.422 |
| 0.5 | 311.52 | 279.91 | 270.976 |
| 0.6 | 306.69 | 279.77 | 271.6 |
| 0.7 | 309.15 | 282.16 | 273.452 |
| 0.8 | 291.72 | 281.45 | 272.921 |
| 0.9 | 299.91 | 283.14 | 274.641 |
| 1 | 297.98 | 283.72 | 275 |

Table 2 The best weights for the SSO_SJ algorithm with respect to different search parameters

| PF \ NS | 25 | 50 | 100 |
|---|---|---|---|
| 0 | 281.42 | 263.90 | 267.42 |
| 0.1 | 281.47 | 260.28 | 269.85 |
| 0.2 | 284.52 | 261.66 | 269.70 |
| 0.3 | 280.52 | 259.26 | 268.93 |
| 0.4 | 280.92 | 258.42 | 269.98 |
| 0.5 | 278.55 | 259.43 | 260.40 |
| 0.6 | 282.56 | 262.73 | 261.82 |
| 0.7 | 286.45 | 262.84 | 261.78 |
| 0.8 | 284.70 | 263.62 | 262.86 |
| 0.9 | 288.73 | 262.92 | 263.42 |
| 1 | 291.42 | 264.41 | 265.43 |

Table 3 Search parameters of the optimization algorithms

| | NS | PF |
|---|---|---|
| SSO | 100 | 0.5 |
| SSO_SJ | 50 | 0.4 |

### *4.2 Five story 105 member steel space frame*

In the first design example taken from previous studies in the literature (Aydoğdu 2010; Akın and Aydoğdu 2015; Çarbaş 2016), the five story steel space frame has 105 members and 54 joints grouped into 11 independent design variables. 3-D and plan views of the frame are shown in Figs. 2-3. The frame is subjected to gravity and lateral loads computed per ASCE 7-10 (ASCE 2005). Three load combinations are defined in the example problem: 1.2D+1.6L+0.5S, 1.2D+0.5L+1.6S, 1.2D+1.6W+0.5L

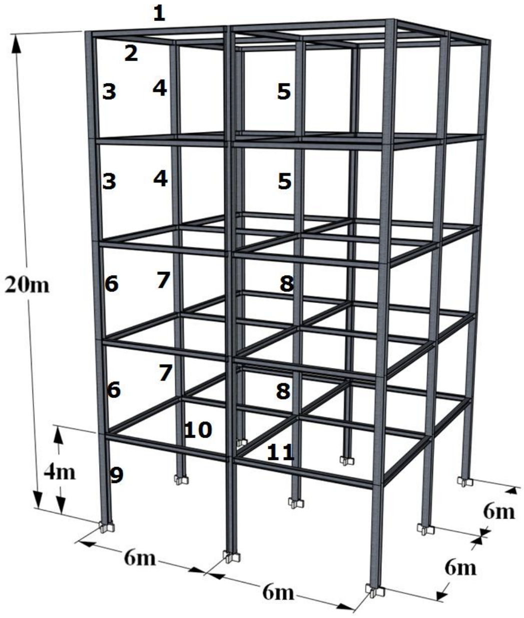


Fig. 2 3-D view of the five-story, 105member steel frame

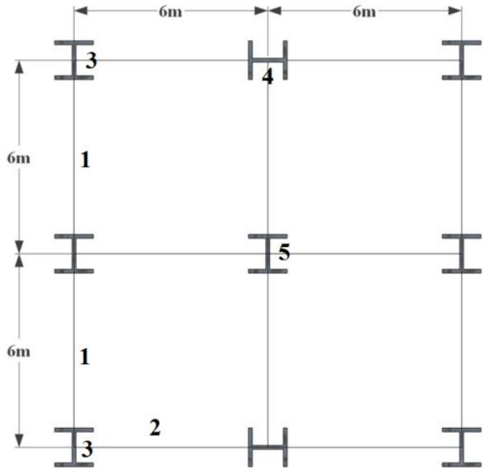


Fig. 3 Plan view of the five-story, 105member steel frame

where D, L, S and W represents dead, live, snow and wind loads respectively. The design loads, basic wind speed, drift and deflection limits of the frame are illustrated in Table 4.

The frame is optimized 50 times using the SSO and SSO_SJ algorithms with different seed values. The search parameters of the SSO and SSO_SJ algorithms are given in Table 3. The average weights and corresponding standard deviations on optimized weights are 281.14 kN and 7.37 kN respectively for the SSO algorithm and 261.91 kN and 2.72 kN respectively for the SSO_SJ algorithm. W-section designations, the lightest weights and the design details of the optimum designs are given in Table 5. Results of the literature studies are tabulated in Table 5 as well. Results indicate that the SSO_SJ algorithm finds the lightest weight among all algorithms (258.416 kN). The value is 2.69%, 7.65%, 4.17%, 4.86%, 9.92%, 30.04%, 30.92% and 51.33% lighter than the optimum weights of ACO, HS, hTLBO-HS, SSO, EFF, CS, PSO and FF algorithms respectively. The SSO algorithm shows the fourth best performance among

Table 4 Load details and displacement limitations of the first two examples

| | 105 member Frame | 428Member Frame |
|---|---|---|
| Dead L. | 2.88 kN/m$^2$ | 2.88 kN/m$^2$ |
| Live Load | 2.39 kN/m$^2$ | 2.39 kN/m$^2$ |
| Snow Load | 0.755 kN/m$^2$ | 0.755 kN/m$^2$ |
| Wind Speed | 65 m/s | 38 m/s |
| TSD | 6.67 cm | 3.5 cm |
| ISD | 1.33 cm | 0.875 cm |
| Def. Limit | 1.67 cm | 2 cm |

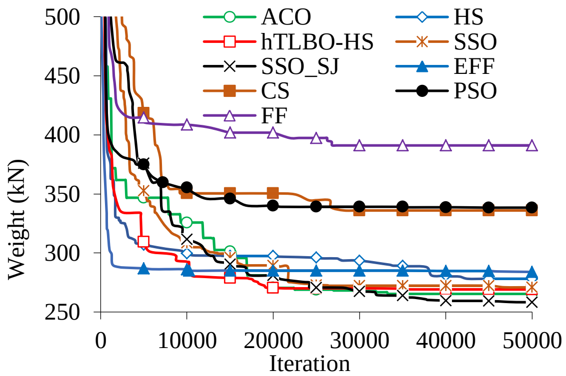


Fig. 4 Search histories of the best design for the first example

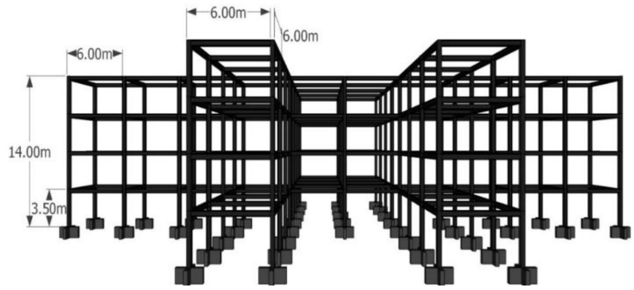


Fig. 5 Side view of the 428 member steel frame

nine algorithms whose weight is 270.98 kN. The search histories of the best design for the algorithms are shown in Fig. 4.

### *4.3 Four story 428 member irregular steel frame*

In the second design example taken from previous studies in the literature (Aydoğdu and Akın 2014, Aydoğdu *et al*. 2016, Çarbaş 2016), the four story steel space frame has 428 members and 172 joints grouped into 20 independent design variables. Side and plan views of the frame are shown in Figs. 5-6. The group details of the structure are given in Table 6. The load combinations of the study are described as (ASCE 2005): 1.2D+1.6L+0.5S, 1.2D+0.5L+1.6S, 1.2D+1.6W+L+0.5S. The design loads, basic wind speed, drift and deflection limits of the frame, computed according to ASCE 7-10 and Ad Hoc Committee on Serviceability, are illustrated in Table 4.

The frame is optimized 10 times using the SSO and SSO_SJ algorithms with different seed values. The obtained

Table 5 Design sections and limit values of the optimum designs for the 105 member space frame

| Member group | Type | ACO (Aydoğdu 2010) | HS (Aydoğdu 2010) | hTLBO-HS (Akın and Aydoğdu 2015) | SSO | SJ_ SSO | EFF (Çarbaş 2016) | CS (Çarbaş 2016) | PSO (Ç arbaş 2016) | FF (Çarbaş 2016) |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Beam | W460×52 | W530×66 | W460×52 | W46×52 | W460×52 | W410×46.1 | W460×52 | W460×52 | W310×52 |
| 2 | Beam | W200×35.9 | W310×38.7 | W200×35.9 | W360×44 | W200×35.9 | W200×41.7 | W250×44.8 | W200×41.7 | W410×53 |
| 3 | Column | W200×35.9 | W200×35.9 | W310×38.7 | W310×38.7 | W310×38.7 | W360×44 | W610×82 | W310×67 | W360×101 |
| 4 | Column | W310×38.7 | W200×35.9 | W200×46.1 | W200×41.7 | W200×35.9 | W250×49.1 | W200×35.9 | W460×113 | W610×101 |
| 5 | Column | W360×57.8 | W360×44 | W360×44 | W460×52 | W410×53 | W410×60 | W610×125 | W760×161 | W310×117 |
| 6 | Column | W460×52 | W310×38.7 | W310×74 | W310×38.7 | W410×53 | W410×60 | W610×82 | W310×67 | W530×101 |
| 7 | Column | W310×86 | W360×72 | W250×73 | W360×72 | W250×73 | W410×100 | W250×67 | W460×113 | W760×134 |
| 8 | Column | W610×101 | W610×92 | W610×101 | W760×134 | W690×125 | W530×123 | W760×185 | W760×161 | W690×125 |
| 9 | Column | W530×66 | W410×53 | W460×74 | W410×53 | W410×60 | W530×74 | W690×170 | W310×67 | W690×140 |
| 10 | Column | W460×89 | W360×72 | W250×73 | W460×74 | W310×74 | W460×106 | W760×147 | W460×113 | W760×134 |
| 11 | Column | W690×170 | W760×147 | W760×173 | W840×176 | W760×134 | W840×176 | W920×488 | W760×161 | W690×170 |
| Max. Str. Ratio | | 0.886 | 0.979 | 0.921 | 0.964 | 0.994 | 0.946 | 0.842 | 0.831 | 0.873 |
| TSD (cm) | | 4.983 | 4.837 | 4.708 | 4.945 | 5.083 | 4.822 | 3.95 | 3.663 | 4.588 |
| ISD (cm) | | 0.569 | 1.333 | 1.325 | 1.332 | 1.326 | 1.332 | 1.139 | 1.075 | 1.333 |
| Max. Def. (cm) | | 0.378 | 0.146 | - | 0.188 | 0.28 | - | - | - | - |
| Max. Iter. | | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 |
| Weight (kN) | | 265.38 | 278.196 | 269.184 | 270.976 | 258.416 | 284.04 | 336.05 | 338.38 | 391.06 |

Table 6 Group definition of the 428 member steel frame

| Story | Side beam | Inner Beam | Corner Column | Side Column | Inner Column |
|---|---|---|---|---|---|
| 1 | 1 | 2 | 9 | 10 | 11 |
| 2 | 3 | 4 | 12 | 13 | 14 |
| 3 | 5 | 6 | 15 | 16 | 17 |
| 4 | 7 | 8 | 18 | 19 | 20 |

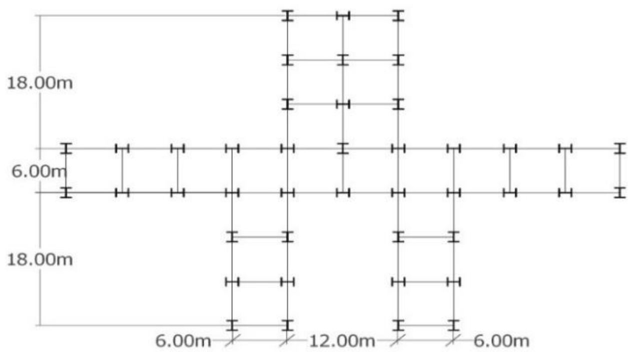


Fig. 6 Plan view of the 428 member steel frame

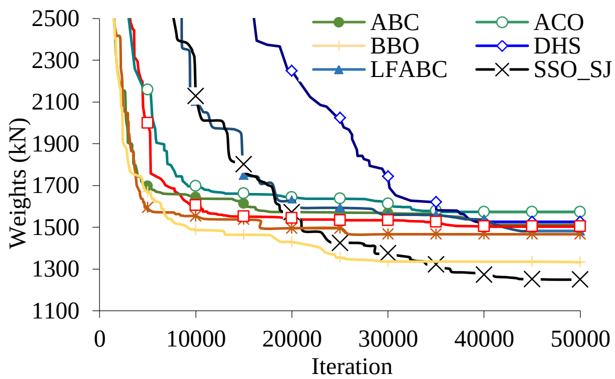


Fig. 7 Search histories of the best designs for the 428 member steel frame

results are compared to the literature results (Aydoğdu and Akın 2014, Aydoğdu *et al*. 2016, Çarbaş 2016). The average weights and corresponding standard deviations on optimized weights are 1924.1 kN and 55.3 kN respectively for the SSO algorithm and 1265.6 kN and 12.7 kN respectively for the SSO_SJ algorithm. The lightest weights, the design details and W-section designations of the optimum designs are given in the Table 7. It is clearly illustrated in the table that the lightest weight is obtained as 1249.11 kN by using the SSO_SJ algorithm. This weight is 6.66%, 17.41%, 18.62%, 20.40%, 21.05% and 25.95% lighter than the optimum weights of BBO, SSO, LFABC, TLBO, ABC, DHS, hTLBO-HS and ACO algorithms respectively. The SSO algorithm shows the third best performance whose weight is 1466.57 kN. The design histories of these algorithms for the best solutions are also plotted in Fig. 7.

### *4.4 354 member steel geodesic dome*

In the third design example taken from previous studies in the literature (Aydoğdu 2010, Akın and Aydoğdu 2015), the steel geodesic dome has 354 members with a 40 m diameter and 8.28m height and 127 joints grouped into 22 independent design variables. Plan, side and 3-D views of the structure are shown in Figs. 8-10 respectively. Three load combinations consisting of dead (D), snow (S) and wind (W) loads are considered for loading the dome which are detailed Fig. 11. The displacement limitation of the all nodes is taken as 11.1 cm.

Table 7 Design sections and limit values of the optimum designs for the 428 member steel frame

| # | | ACO (Aydoğdu *et al.* 2016) | ABC (Aydoğdu *et al.* 2016) | BBO (Çarbaş 2016) | DHS (Aydoğdu *et al.* 2016) | LFABC (Aydoğdu *et al.* 2016) | SSO | SJSSO | TLBO (Aydoğdu and Akın 2014) |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Beam | W250×25.3 | W310×38.7 | W360×32.9 | W250×32.7 | W410×46.1 | W250×25 | W200×41.7 | W310×38.7 |
| 2 | Beam | W310×28.3 | W360×32.9 | W250×32.7 | W360×39 | W310×28.3 | W310×32.7 | W360×39 | W360×32.9 |
| 3 | Beam | W460×52 | W460×52 | W460×52 | W360×44 | W360×44 | W310×44.5 | W360×44 | W460×52 |
| 4 | Beam | W460×52 | W460×52 | W310×32.7 | W410×46.1 | W460×52 | W460×74 | W310×44.5 | W460×52 |
| 5 | Beam | W360×44 | W310×32.7 | W530×66 | W530×74 | W410×60 | W360×32.9 | W200×41.7 | W310×32.7 |
| 6 | Beam | W310×28.3 | W310×32.7 | W460×52 | W460×68 | W410×38.8 | W310×32.7 | W310×38.7 | W310×32.7 |
| 7 | Beam | W200×41.7 | W310×38.7 | W360×32.9 | W250×38.5 | W360×32.9 | W360×32.9 | W200×41.7 | W310×38.7 |
| 8 | Beam | W310×28.3 | W360×39 | W460×52 | W250×44.8 | W410×60 | W360×32.9 | W310×28.3 | W360×39 |
| 9 | Column | W360×57.8 | W200×46.1 | W410×53 | W200×59 | W460×144 | W360×51 | W200×46.1 | W200×46.1 |
| 10 | Column | W360×44 | W200×46.1 | W250×49.1 | W200×59 | W310×38.7 | W310×38.7 | W250×58 | W200×46.1 |
| 11 | Column | W310×79 | W200×46.1 | W200×46.1 | W360×51 | W200×46.1 | W310×97 | W200×31.3 | W200×46.1 |
| 12 | Column | W360×110 | W840×210 | W410×100 | W760×134 | W460×144 | W760×134 | W200×46.1 | W840×210 |
| 13 | Column | W760×134 | W460×74 | W250×80 | W250×89 | W310×143 | W460×74 | W360×64 | W460×74 |
| 14 | Column | W360×101 | W690×140 | W360×134 | W360×122 | W200×46.1 | W310×143 | W410×60 | W690×140 |
| 15 | Column | W1000×296 | W1000×321 | W460×113 | W920×271 | W460×144 | W1000×296 | W200×46.1 | W1000×321 |
| 16 | Column | W920×201 | W920×201 | W310×97 | W610×92 | W310×143 | W920×201 | W610×101 | W920×201 |
| 17 | Column | W920×201 | W760×147 | W360×147 | W840×193 | W360×147 | W840×193 | W460×113 | W760×147 |
| 18 | Column | W1100×499 | W1100×390 | W920×201 | W1100×343 | W760×173 | W1100×343 | W200×46.1 | W1100×390 |
| 19 | Column | W920×201 | W920×201 | W840×193 | W690×170 | W690×217 | W920×201 | W1100×343 | W920×201 |
| 20 | Column | W1000×249 | W760×147 | W530×150 | W840×226 | W360×216 | W840×193 | W610×113 | W760×147 |
| Max. Str. Ratio | | 1 | 1 | 0.978 | 0.847 | 0.883 | 0.997 | 0.994 | 0.98 |
| TSD (cm) | | 2.856 | 2.91 | 2.867 | 2.663 | 3.01 | 2.833 | 2.27 | 2.68 |
| ISD (cm) | | 0.87346 | 0.875 | 0.875 | 0.87353 | 0.535 | 0.875 | 0.812 | 0.869 |
| Max. Def. (cm) | | 0.288 | 0.49 | - | 0.296 | 0.512 | 0.354 | 0.669 | 0.357 |
| Maximum Iter. | | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 |
| Weight (kN) | | 1573.21 | 1512.11 | 1332.29 | 1526.01 | 1481.73 | 1466.57 | 1249.11 | 1503.91 |

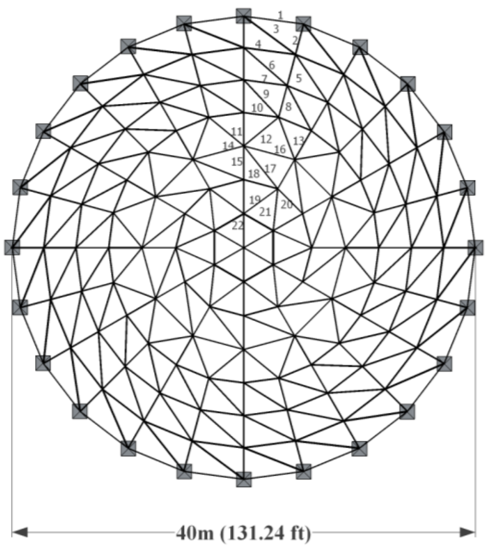


Fig. 8 Plan view of the 354 member geodesic dome

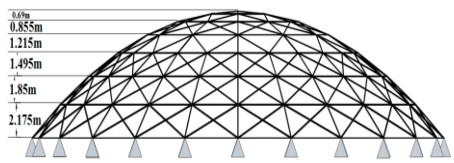


Fig. 9 Side view of the 354 member geodesic dome

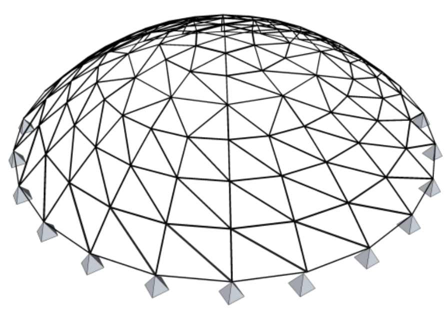

Fig. 10 3-D view of the 354 member geodesic dome

The lightest weights, maximum constraints values and section selections of the optimum designs obtained by the SSO and SSO_SJ algorithm are shown in the Table 8. The results are also compared to the ACO, ABC, PSO, CS), and FF algorithm results which are also illustrated in Table 8. According to the table, the SSO_SJ algorithm obtains the lightest weight among all results which is 140.22kN. This weight is 1.02% lighter than the ABC's lightest weight,

Table 8 Design sections and limit values of the optimum designs for the 354 member geodesic dome

| Group Number | ABC (Saka *et al.* 2016) | PSO (Saka *et al.* 2016) | ACO (Saka *et al.* 2016) | CS (Saka *et al.* 2016) | FF (Saka *et al.* 2016) | SSO | SSO_SJ |
|---|---|---|---|---|---|---|---|
| 1 | P2 | P2 | P2 | P2 | P2 | XP2 | P2 |
| 2 | P3 | P3 | P3 | P4 | P3.5 | P3 | P3 |
| 3 | P4 | P3.5 | P4 | P3.5 | P4 | P4 | P3.5 |
| 4 | P3 | P3.5 | P3.5 | P3.5 | P | P3.5 | P3.5 |
| 5 | P3 | P3 | P3 | P3.5 | P3.5 | P3 | P3 |
| 6 | P3 | P3 | P3 | P3 | P3.5 | P3 | P3 |
| 7 | P3 | P3 | P3 | P3 | P3 | P3 | P3 |
| 8 | P2.5 | P3 | P2.5 | P3 | P3 | P2.5 | P2.5 |
| 9 | P3 | P3 | P3 | P3 | P3 | P3 | P3 |
| 10 | P3 | P3 | P3 | P3 | P3 | P3 | P3 |
| 11 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 |
| 12 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 |
| 13 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 |
| 14 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 |
| 15 | P2.5 | P2.5 | XP2.5 | P2.5 | P2.5 | XP2.5 | P2.5 |
| 16 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 | P2.5 |
| 17 | XP2 | XP2 | XP2 | XP2 | XP2 | P2.5 | P2 |
| 18 | P2 | P2 | XP2 | P2.5 | P2.5 | XP2 | XP2 |
| 19 | XP2 | XP2 | P2 | XP2 | XP2 | P2 | P2 |
| 20 | P2 | P2.5 | XP2 | P2.5 | XP2.5 | P2 | XP2 |
| 21 | P2 | P2 | P2 | P2 | P2 | P2 | P2 |
| 22 | P2 | P2 | P2 | P2 | P2 | P2 | P2 |
| Weight (kN) | 142.87 | 144.53 | 146.65 | 150.78 | 158.32 | 149.15 | 141.42 |
| Max. Str. ratio | 0.997 | 0.844 | 0.896 | 0.801 | 0.8 | 0.998 | 0.9 |
| Max. Disp. (cm) | 1.73 | 1.67 | 1.71 | 1.61 | 1.59 | 1.71 | 1.72 |
| Max. Iter. | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 | 50000 |

6.62% lighter than CS's best weight, 11.95% lighter than the FFA's lightest weight, 3.69% lighter the ACO best weight, 2.20% lighter than the PSO's lightest weight and 5.46% lighter than the SSO's lightest weight. Weight of the fourth best design obtained by using the SSO algorithm (149.15 kN). The search histories of the algorithms for the best design are also illustrated in Fig. 12.

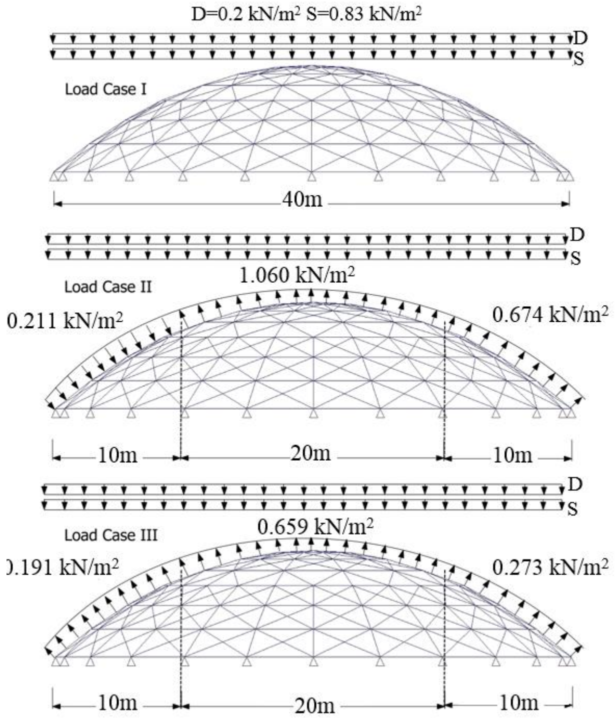


Fig. 11 Loading of the 354 member geodesic dome

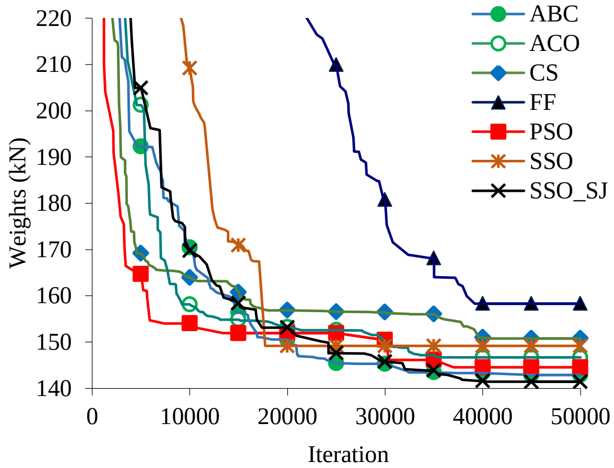


Fig. 12 Search histories of the 354 member geodesic dome

## 5. Conclusions

In this paper, SSO algorithm and an improved version of the SSO algorithm with spider jump approach are utilized for the optimum design of the steel space structures in order to investigate their robustness and efficiencies for the optimization problem. For these purposes, the presented algorithms are numerically examined using the three examples. According to results, the SSO shows the 4$^{th}$ best performance in the first problem, 3$^{rd}$ best performance in the second problem and 5$^{th}$ best performance in the last example which is evaluated as the average performance. However, the SSO_SJ algorithm returns the best performance in the all design examples. The design search history graphs generated for these examples using SSO_SJ algorithm clearly evince a significant performance improvement achieved compared to the SSO algorithm. Respectively, SSO_SJ algorithm found 2.69% and 1.62% lighter weights than the second best algorithms in example 1 and 3 and this performance improvement reached approximately 6.66% in example 2. Also, a comparison of optimum designs of the problems attained with SSO and SSO_SJ algorithms verifies that SSO_SJ has better performance between 4.86% and 17.41% for the presented examples. Even if SSO algorithm has better convergence rate than SSO_SJ in early stages of the iterations, SSO_SJ performs better to improve the solutions in the following stages. SSO_SJ algorithm is more effective and stable in obtaining high quality solutions and has higher success rates under the same conditions.

The sensitivity analysis of algorithm parameters showed that initial choice of control parameters affects the convergence rate of the SSO and SSO_SJ algorithms, and the solution accuracy of the final design obtained. But this effect is minimized by the spider jump technique throughout the evolution progresses. As a result, improved SSO algorithm with spider jump technique has remarkable performance in optimum design of the steel space structures by increasing the SSO algorithm's ability to find the global optima. SSO and SSO_SJ algorithms have algorithm parameters which reduces the sensitivity analysis procedure. This feature is another advantage of the presented algorithms than ACO and other metaheuristics having many algorithm parameters. Future research on SSO_SJ algorithm can be performed for other structural engineering problems such as nonlinear domes, reinforced concrete structures, and retaining walls.

## References


Akın, A. and Aydoğdu, İ. (2015), "Optimum design of steel space frames by hybrid teaching-learning based optimization and harmony search algorithms", *Int. J. Mech. Aerosp. Indus. Mech. Manuf. Eng.*, **9**(7), 1367-1374.

Alqedra, M., Khalifa, A. and Arafa, M. (2015), "An intelligent tuned harmony search algorithm for optimum design of steel framed structures to AISC-LRFD", *Adv. Res.*, **4**(6), 421-440.

Arafa, M., Khalifa, A. and Alqedra, M. (2016), "Design optimization of semi-rigidly connected steel frames using harmony search algorithm", *J. Eng. Res. Technol.*, **2**(2), 95-104.

Artar, M. (2016), "Optimum design of steel space frames under earthquake effect using harmony search", *Struct. Eng. Mech.*, **58**(3), 597-612.

Artar, M. and Daloglu, A.T. (2015), "The optimization of multi-storey composite steel frames with genetic algorithm including dynamic constraints", *Tek Dergi*, **26**(2), 7077-7098.

Artar, M. and Daloglu, A.T. (2015), "Optimum design of steel frames with semi-rigid connections and composite beams", *Struct. Eng. Mech.*,**55**(2), 299-313.

Artar, M. and Daloglu, A.T. (2015), "Optimum design of steel space frames with composite beams using genetic algorithm", *Steel Compos. Struct.*, **19**(2), 503-519.

ASCE (2005), ASCE 7-05, Minimum Design Loads for Buildings and Other Structures, American Society of Civil Engineers, Reston, Virginia, ABD.

Aviles, L. (1986), "Sex-ratio bias and possible group selection in the social spider Anelosimus eximius", *Am. Natural.*, 1-12.

Aydogdu, I. (2017), "Cost optimization of reinforced concrete cantilever retaining walls under seismic loading using a biogeography-based optimization algorithm with Levy flights", *Eng. Optim.*, **49**(3), 381-400.

Aydogdu, I. and Akin, A. (2014). "Optimum design of geodesic aluminum domes using firefly algorithm", *Proceedings of the ACE 2014 11th International Congress on Advances in Civil Engineering*, İstanbul, Turkey.

Aydogdu, I. and Saka, M.P. (2009). "Ant colony optimization of irregular steel frames including effect of warping", *Proceedings of the Twelfth International Conference on Civil, Structural and Environmental Engineering Computing*, Madeira, Portugal.

Aydoğdu, İ. (2010), "Optimum design of 3-D irregular steel frames using ant colony optimization and harmony search algorithms", PhD Thesis, Middle East Technical University, August, Ankara, Turkey

Aydoğdu, İ. and Akın, A. (2014), "Teaching and learning-based optimization algorithm for optimum design of steel buildings", *Comput. Civil Build. Eng*., 2167-2175.

Aydoğdu, İ., Akın, A. and Saka, M. (2012), *Discrete Design Optimization of Space Steel Frames using the Adaptive Firefly Algorithm*, Civil Comp, Dubrovnik, Crotia.

Aydoğdu, İ., Akın, A. and Saka, M. (2012), *Optimum Design of Steel Space Frames by Artificial Bee Colony Algorithm*, Ankara, Turkey.

Aydoğdu, İ., Akın, A. and Saka, M. (2016), "Design optimization of real world steel space frames using artificial bee colony algorithm with Levy flight distribution", *Adv. Eng. Softw.*, **92**, 1-14.

Azad, S.K. and Hasançebi, O. (2013), "Upper bound strategy for metaheuristic based design optimization of steel frames", *Adv. Eng. Softw.*, **57**, 19-32.

Azad, S.K. and Hasançebi, O. (2014), "An elitist self-adaptive step-size search for structural design optimization", *Appl. Soft Comput.*, **19**, 226-235.

Azad, S.K. and Hasançebi, O. (2015), "Computationally efficient discrete sizing of steel frames via guided stochastic search heuristic", *Comput. Struct.*, **156**, 12-28.

Azad, S.K., Hasançebi, O. and Saka, M. (2014), "Guided stochastic search technique for discrete sizing optimization of steel trusses: A design-driven heuristic approach", *Comput. Struct.*, **134**, 62-74.

Burgess, J. and Uetz, G. (1982), "Social spacing strategies in spiders", *Spider Communication: Mechanisms and Ecological Significance*, 317-351.

Campón, F.F. (2007), "Group foraging in the colonial spider Parawixia bistriata (Araneidae): effect of resource levels and prey size", *Animal Behav.*, **74**(5), 1551-1562.

Carbas, S. and Aydogdu, I. (2017). "Utilization of harmony Search algorithm in optimal structural design of cold-formed steel structures", *International Conference on Harmony Search Algorithm*.

Cuevas, E., Cienfuegos, M., Zaldivar, D. and Perez-Cisneros, M. (2013), "A swarm optimization algorithm inspired in the behavior of the social-spider", *Exp. Syst. Appl.*, **40**(16), 6374-6384.

Çarbaş, S. (2016), "Design optimization of steel frames using an enhanced firefly algorithm", *Eng. Optim.*, **48**(12), 2007-2025.

Çarbaş, S. (2016), "Optimum structural design of spatial steel frames via biogeography-based optimization", *Neural Comput. Appl.*, 1-15.

Degertekin, S.O. (2012), "Optimum design of geometrically non-linear steel frames using artificial bee colony algorithm", *Steel Compos. Struct.*, **12**(6), 505-522.

Degertekin, S.O. and Hayalioglu, M.S. (2013), "Sizing truss structures using teaching-learning-based optimization", *Comput. Struct.*, **119**, 177-188.

Dorigo, M., Maniezzo, V. and Colorni, A. (1996), "Ant system: optimization by a colony of cooperating agents", *Syst. Man Cyber. Part B: Cyber. IEEE Tran.*, **26**(1), 29-41.

Elias, D.O., Andrade, M.C. and Kasumovic, M.M. (2011), "Dynamic population structure and the evolution of spider mating systems", *Adv. Insect Phys.*, **41**, 65.

Ellingwood, B. (1986), "Structural serviceability: a critical appraisal and research needs", *J. Struct. Eng.*, **112**(12), 2646-2664.

Esapour, K., Hoseinzadeh, R., Akbari-Zadeh, M.R. and Zare, J. (2015), "A new sufficent method based on levy-social spider technique for optimal economic dispatch of thermal power unit", *J. Intell. Fuzzy Syst.*, **28**(3), 1137-1143.

Fister, Jr, I., Yang, X.S., Fister, I., Brest, J. and Fister, D. (2013), "A brief review of nature-inspired algorithms for optimization", arXiv preprint arXiv:1307.4186.

Gholizadeh, S. and Poorhoseini, H. (2015), “Optimum design of steel frame structures by a modified dolphin echolocation algorithm”, *Struct. Eng. Mech.*, **55**(3), 535-554.

Hameed, W.I., Kadhim, A.S. and Al-Thuwaynee, A.A.K. (2016), “Field weakening control of a separately excited DC motor using neural network optemized by social spider algorithm”, *Eng*, **8**(1), 1.

Hasançebi, O. and Azad, S.K. (2015), “Adaptive dimensional search: a new metaheuristic algorithm for discrete truss sizing optimization”, *Comput. Struct.*, **154**, 1-16.

Hasançebi, O., Çarbaş, S., Doğan, E., Erdal, F. and Saka, M.P. (2009), “Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures”, *Comput. Struct.*, **87**(5-6), 284-302.

Hölldobler, B., Wilson, E.O., Keller, L., Gordon, E., Bisseleua, B., Vidal, S., Bos, M., Tylianakis, J., Steffan-Dewenter, I. and Tscharntke, T. (1994), *Journey to the Ants: a Story of Scientific Exploration*, CATIE, Turrialba. (Costa Rica)

Jones, T.C. and Riechert, S.E. (2008), “Patterns of reproductive success associated with social structure and microclimate in a spider system”, *Animal Behav.*, **76**(6), 2011-2019.

Karaboga, D. (2005), “An Idea based on honey bee swarm for numerical optimization”, Technical Report-tr06, Engineering Faculty, Computer Engineering Department, Erciyes University.

Kassabalidis, I., El-Sharkawi, M.A., Marks, R.J., Arabshahi, P. and Gray, A.A. (2001). “Swarm intelligence for routing in communication networks”, *Global Telecommunications Conference, GLOBECOM'01, IEEE*.

Kavousi-Fard, A., Abbasi, A., Rostami, M.A. and Khosravi, A. (2015), “Optimal distribution feeder reconfiguration for increasing the penetration of plug-in electric vehicles and minimizing network costs”, *Energy*, **93**, 1693-1703.

Kennedy, J., Kennedy, J.F., Eberhart, R.C. and Shi, Y. (2001), *Swarm Intelligence*, Morgan Kaufmann.

Khorramnia, R., Akbarizadeh, M.R., Jahromi, M.K., Khorrami, S.K. and Kavusifard, F. (2015), “A new unscented transform for considering wind turbine uncertainty in ED problem based on SSO algorithm”, *J. Intell. Fuzzy Syst.*, **29**(4), 1479-1491.

LRFD, A. (2000), “Load & Resistance Factor Design Specification”, American Institute of Steel Construction, Chicago, Illinois.

Lubin, Y. and Bilde, T. (2007), “The evolution of sociality in spiders”, *Adv. Study Behav.*, **37**, 83-145.

Mirjalili, S.Z., Saremi, S. and Mirjalili, S.M. (2015), “Designing evolutionary feedforward neural networks using social spider optimization algorithm”, *Neural Comput. Appl.*, **26**(8), 1919-1928.

Oster, G.F. and Wilson, E.O. (1978), *Caste and Ecology in the Social Insects*, Princeton University Press

Pasquet, A. and Krafft, B. (1992), “Cooperation and prey capture efficiency in a social spider, Anelosimus eximius (Araneae, Theridiidae)”, *Ethology*, **90**(2), 121-133.

Passino, K.M. (2002), “Biomimicry of bacterial foraging for distributed optimization and control”, *Control Syst., IEEE*, **22**(3), 52-67.

Pereira, D.R., Delpiano, J. and Papa, J.P. (2015), “On the optical flow model selection through metaheuristics”, *Eurasip. J. Image Vide*, **2015**(1), 11.

Rao, S.S. (2009), *Engineering Optimization: Theory and Practice*, John Wiley & Sons, Canada.

Rypstra, A.L. and Tirey, R.S. (1991), “Prey size, prey perishability and group foraging in a social spider”, *Oecologia*, **86**(1), 25-30.

Saka, M.P. (2003), *Optimum Design of Skeletal Structures: A Review*, Saxe-Coburg Publications, Stirlingshire, UK.

Saka, M.P. (2007), *Optimum Design of Steel Frames using Stochastic Search Techniques Based on Natural Phenomena: A Review*

Saka, M.P. (2014), “Shape and topology optimization design of skeletal structures using metaheuristic algorithms: a review”, *Comput. Tech. Rev.*, **9**, 31-68.

Saka, M.P., Çarbaş, S., Aydoğdu, İ. and Akın, A. (2016), *Use of Swarm Intelligence in Structural Steel Design Optimization*, Springer

Saka, M.P., Doğan, E. and Aydoğdu, İ. (2013), “Review and analysis of swarm-intelligence based algorithms”, *Swarm Intelligence and Bio-Inspired Computation*, Edited by Yang, Cui, Xiao and Gandomi, Elsevier.

Saka, M.P. and Geem, Z.W. (2013), “Mathematical and metaheuristic applications in design optimization of steel frame structures: an extensive review”, *Math. Prob. Eng.*, **2013**, 1-33.

Shi, Y. and Eberhart, R. (1998). “A modified particle swarm optimizer”, *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*.

Tunca, O. and Çarbaş, S. (2016), “Biogeography-based optimization algorithm for designing of planar steel frames”, *IJISAE*, **4**, 53-57.

Ulbrich, K. and Henschel, J. (1999), “Intraspecific competition in a social spider”, *Ecolog. Model.*, **115**(2), 243-251.

Vardhini, K.K. and Sitamahalakshmi, T. (2016), “A review on nature-based swarm intelligence optimization techniques and its current research directions”, *Ind. J. Sci. Technol.*, **9**(10), DOI: 10.17485/ijst/2016/v9i10/81634.

Yang, X.S. (2010), *Firefly Algorithm, Levy Flights and Global Optimization*, Springer

Yang, X.S., Cui, Z., Xiao, R., Gandomi, A.H. and Karamanoglu, M. (2013), *Swarm Intelligence and Bio-inspired Computation: Theory and Applications*, Newnes

Yang, X.S. and Deb, S. (2009). “Cuckoo search via Lévy flights”, *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*.

Yip, E.C., Powers, K.S. and Avilés, L. (2008), “Cooperative capture of large prey solves scaling challenge faced by spider societies”, *Proceedings of the National Academy of Sciences*, **105**(33), 11818-11822.

Yu, J.J.Q. and Li, V.O.K. (2016), “A social spider algorithm for solving the non-convex economic load dispatch problem”, *Neurocomput.*, **171**, 955-965.


*CC*