Expert Systems With Applications 125 (2019) 425–441

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

# Structural optimization using multi-objective modified adaptive symbiotic organisms search

Ghanshyam G. Tejani[a,*], Nantiwat Pholdee[b], Sujin Bureerat[b], Doddy Prayogo[c], Amir H. Gandomi[d]

[a] Department of Mechanical Engineering, School of Technology, GSFC University, Vadodara, Gujarat, India
[b] Department of Mechanical Engineering, Faculty of Engineering, Khon Kaen University, Thailand
[c] Department of Civil Engineering, Petra Christian University, Jalan Siwalankerto 121-131, Surabaya 60236, Indonesia
[d] School of Business, Stevens Institute of Technology, Hoboken, NJ 07030, USA

ARTICLE INFO

*Article history:*
Received 10 August 2018
Revised 17 January 2019
Accepted 18 January 2019
Available online 30 January 2019

*Keywords:*
Adaptive mechanism
Structural optimization
Meta-heuristics
Discrete variables
Constrained problems

ABSTRACT

Multiple objective structural optimization is a challenging problem in which suitable optimization methods are needed to find optimal solutions. Therefore, to answer such problems effectively, a multi-objective modified adaptive symbiotic organisms search (MOMASOS) with two modified phases is planned along with a normal line method as an archiving technique for designing of structures. The proposed algorithm consists of two separate improved phases including adaptive mutualism and modified parasitism phases. The probabilistic nature of mutualism phase of MOSOS lets design variables to have higher exploration and higher exploitation simultaneously. As search advances, a stability between the global search and a local search has a significant effect on the solutions. Therefore, an adaptive mutualism phase is added to the offer MOASOS. Also, the parasitism phase of MOSOS offers over exploration which is a major issue of this phase. The over exploration results in higher computational cost since the majority of the new solutions gets rejected due to inferior objective functional values. In consideration of this issue, the parasitism phase is upgraded to a modified parasitism phase to increase the possibility of getting improved solutions. In addition, the proposed changes are comparatively simple and do not need an extra parameter setting for MOSOS.

For the truss problems, mass minimization and maximization of nodal deflection are considered as objective functions, elemental stresses are considered as behavior constraints and (discrete) elemental sections are considered as side constraints. Five truss optimization problems validate the applicability of the considered meta-heuristics to solve complex engineering. Also, four constrained benchmark engineering design problems are solved to demonstrate the effectiveness of MOMASOS. The results confirmed that the proposed adaptive mutualism phase and modified parasitism phase with a normal line method as an archiving technique provide superior and competitive results than the former obtained results.



## 1. Introduction

Optimal truss design is among the hottest research challenge of structural engineering. Recently during each year, hundreds of papers related to the topic were published. Truss structures can be viewed as a set of 2-node links interconnected by spherical joints. They have been used in several engineering applications with the advantages in that they are simple to construct, low cost, easy to design and less difficult to construct in difficult-to-access regions (Pholdee & Bureerat, 2013b). The applications include a bridge, a tower, a transmission tower, and a billboard structure. In designing a truss structure, engineering will define its topology, shape, and elements' sizes. Usually, a trial-and-error approach can be applied. Nevertheless, for large trusses, such an approach is not efficient and effective.

Therefore, the application of an optimization technique is a better choice. The tradition truss optimization is used to get the best suitable topology, shape, and sizes to minimize weight or cost subject to structural safety constraints. Gradient-based optimizers can be used in cases of continuous design variables (Allwood & Chung, 1984; Fleury, 1980). However, over the last few decades, the use of meta-heuristics (MHs) are the main focus due to their simplicity to use, code, and implement. Unlike its gradient

* Corresponding author.
*E-mail addresses:* p.shyam23@gmail.com (G.G. Tejani), nantiwat@kku.ac.th (N. Pholdee), sujbur@kku.ac.th (S. Bureerat), prayogo@petra.ac.id (D. Prayogo), a.h.gandomi@stevens.edu (A.H. Gandomi).

https://doi.org/10.1016/j.eswa.2019.01.068
0957-4174/

counterpart, MHs can be applied to answer almost any type of design variables. The combination of several types of design variables for one optimization run is possible. This aid makes MHs more popular than gradient-based optimization methods for truss optimization. Moreover, some MHs can explore a Pareto front, in cases of multi-objective optimization (MO), within on simulation run.

The use of MHs for single-objective optimization has been commonplace. Over the years, there have been numerous MHs newly invented. Some of the popular techniques for truss design include a genetic algorithm (GA) (Lingyun, Mei, Guangming, & Guang, 2005; Wei et al., 2011; Zuo, Xu, Zhang, & Xu, 2011), particle swarm optimization (PSO) (Gomes, 2011), cuckoo search (Gandomi, Talatahari, Yang, & Deb, 2012), krill herd algorithm (Gandomi, Talatahari, Tadbiri, & Alavi, 2013), differential evolution (DE), teaching-learning based optimization (Camp & Farshchin, 2014; Degertekin & Hayalioglu, 2013; Savsani, Tejani, & Patel, 2016; Tejani, Savsani, & Patel, 2016b), Ray optimization (Kaveh & Khayatazad, 2013), colliding body algorithm (Kaveh & Mahdavi, 2014), Parameter-less population pyramid (Gandomi & Goldman, 2018), and grey wolf optimizer (GWO) (Kaveh & Zakian, 2017; Panagant & Bureerat, 2018). Later, some of those baseline algorithms have been modified or improved leading to more advanced versions e.g. adaptive DE (Bureerat & Pholdee, 2015), modified symbiotic organisms search (Kumar, Tejani, & Mirjalili, 2018; Tejani, Savsani, Patel, & Mirjalili, 2017; 2018c). The performance enhancement can also be achieved by means of hybridization such as hybridized passing vehicle search & simulated annealing (Tejani, Savsani, Bureerat, Patel, & Savsani, 2018b), and hybrid GWO & self-adaptive DE (Panagant & Bureerat, 2018). Recently, the performance test of a number of self-adaptive MHs on solving truss optimization has been investigated. It is found that most of CEC (Congress on Evolutionary Computation) competition winners are some of the top MHs for truss optimization (Pholdee & Bureerat, 2017).

Once more than one design objectives are posed, the optimization problem is called MO. It is furthermore called many-objective optimization in cases of a problem having more than three objective functions in order to state its difficulty to explore the entire Pareto front. The use of multi-objective meta-heuristics (MOMHs) for truss optimization has been studied for a decade. It is well recognized that a designer always needs for optimizing many objective functions at the same time and those objectives will always be conflicting with each other. The solutions for such a design problem are countless, and its solution set is termed a Pareto optimal set (or a Pareto front) if viewed as per the objective function domain. The main reason for MOMHs popularity in MO is that MOMHs is capable to get a Pareto front in a single run. The pioneering MOMHs were a multi-objective genetic algorithm (MOGA) (Fonseca & Fleming, 1993), a SPEA2 (Zitzler, Laumanns, & Thiele, 2001), and a NSGA-II (Deb, Pratap, Agarwal, & Meyarivan, 2002). Later there have been a great variety of improved versions of existing algorithms (Bureerat & Srisomporn, 2010; Kaveh & Laknejadi, 2011; Pholdee & Bureerat, 2012, 2013a, 2013b; Zitzler, Laumanns, & Thiele, 2002) and newly invented methods such as DE for MO (Robič & Filipič, 2005). Some of them were upgraded for solving many-objective optimization such as a non-dominated sorting genetic algorithm (NSGAIII) (Deb & Jain, 2013; Jain & Deb, 2013), Two-arch (Wang, Jiao, & Yao, 2015), and knee-point optimizer (Zhang, Tian, & Jin, 2015).

The use of MOMHs for truss optimization will provide benefit in that a designer can have many solutions for decision making (Kaveh & Mahdavi, 2018; Noilublao & Bureerat, 2011, 2013; Pholdee & Bureerat, 2012, 2013a, 2013b). Moreover, they can be used for reliability optimization of trusses (Ho-Huu, Duong-Gia, Vo-Duy, Le-Duc, & Nguyen-Thoi, 2018; Techasen et al., 2018). Several MOMHs were used to tackle multi-objective truss design in Noilublao and Bureerat (2011, 2013). The use of the so-called approximate gradient as a local search to enhance the performance of MOEAs was presented in Pholdee and Bureerat (2012, 2013a). Other work with MO of trusses can be found in Angelo, Barbosa, and Bernardino (2012, 2015), Greiner and Hajela (2012), Hosseini, Hamidi, Mansuri, and Ghoddosian (2015), Kaveh and Laknejadi (2013), Kaveh and Mahdavi (2018), Mousa, El-Shorbagy, and Abd-El-Wahed (2012), Richardson et al. (2012), Su, Wang, Gui, and Fan (2011) and Tejani, Bureerat, Pholdee, and Prayogo (2018c). It has been shown from the literature that a study on using MOMHs for truss design is much more advantageous.

As a result, this paper deals with modification and improvement of symbiotic organisms search for truss MO. Since it was first invented by Cheng and Prayogo (2014), the optimizer has been implemented on a number of applications while many modified versions have been additionally proposed (Ayala, Klein, Mariani, & Coelho, 2017; Çelik & Öztürk, 2017; Ezugwu & Adewumi, 2017; Ezugwu, Adewumi, & Frîncu, 2017; Guha, Roy, & Banerjee, 2017; Prayogo, Cheng, Wong, Tjandra, & Tran, 2018; Secui, 2016; Zhang, Sun, Yuan, Lv, & Ma, 2016; Çelik & Durgut, 2018). SOS was then being upgraded for MO (Tran, Cheng, & Prayogo, 2016; Duc-Hoc, 2017) leading to multi-objective symbiotic organisms search (MOSOS). Investigation on improving the performance of MOSOS for truss design is interesting since it is a new method that should be tested with this popular research topic. In this work, the main contribution is an incorporation of the random migration based search along with adaptive benefit factors (BFs) into MOSOS. These techniques are used to set better stability between exploration and to improve exploitation during mutualism phase, and to improve exploration during parasitism phase of MOSOS. A number of multi-objective truss design are used to validate the new algorithms while several state-of-the-art MOMHs are used to compare with the new MOSOS. The results show that our proposed method is powerful for truss optimization.

Because the optimizer has just been proposed, there is room for further development and investigation, as a consequence, this study is proposed to enhance the effectiveness of the MOSOS by incorporating a modified parasitism system. MOASOS and MOMASOS are employed to answer multi-objective truss design problems while the objectives comprise truss mass minimization and nodal deflection maximization. The solutions received from various optimizers are examined and presented.

## 2. The symbiotic organisms search (SOS) algorithm

Cheng and Prayogo (2014) developed SOS to serve as a continuous-based MH algorithm and a population-oriented searching technique; the technique finds global optimum solutions by retaining a set of possible ones called a population. SOS is focused upon symbiosis, which is the process by which organisms in an ecosystem possess biological interdependence with one another which allows them to grow and survive. Due to its excellent performance over the benchmark algorithms, SOS has been applied to numerous research fields since its introduction (Cheng, Prayogo, & Tran, 2015; Tran et al., 2016; Abdullahi, Ngadi, & Abdulhamid, 2016; Çelik & Öztürk, 2017; Guha et al., 2017; Panda & Pani, 2016; Prayogo & Susanto, 2018; Cheng et al., 2018; Prayogo et al., 2018; Tejani, Savsani, & Patel, 2016a, 2017, 2018a; Yu, Perwira Redi, Yang, Ruskartina, & Santosa, 2017; Çelik & Durgut, 2018).

The first step performed by the SOS algorithm is the initialization of the specific population in an ecosystem. Following this is a process by which the algorithm evaluates organisms' locations by computing the particular objectives', such that the organisms with the best solution is elected as '$X_{best}$'. This action takes place in iterations, finding the global best solution by updating to the most recently available solution until the solution is found. For this instance, three fundamentals of symbiosis, mutualism, commensalism, and parasitism inspired the principle rules used by the algorithm. These rules were used to update the positions of new organisms. Once the algorithm reaches the maximum number of function evaluations, termination of the loops is implemented. Below is an explanation of how mutualism, commensalism, and parasitism come into play in the MH.

### 2.1. Mutualism phase

This phase involves an association by which both parties positively benefit. In the relationship between a flower and a pollinator, the pollinator benefits from the food it can take from the flower, while the flower can turn into fruit from its contact with the pollinator. Due to this twofold positive benefit, the relationship can be deemed as a mutually beneficial symbiosis.

For mutualism phase, organism '*i*' is assigned as a solution ($X_i$) to interact with the secondary solution chosen via randomized selection ($X_k$) (in this instance, k ≠ i). This symbiotic relationship positively impacts both solutions. The BFs and a mutual vector *(MV)* dictate new solutions. $BF_1$ and $BF_2$ are determined via randomized selection between 1 or 2 (see Eqs. (4) and (5)). Because of this, either BF demonstrates an example of a solution experiencing positive benefits somewhat or entirely through symbiosis. The best solution ($X_{best}$) is an additional variable which solutions can be impacted by; it is selected through a random search from the Pareto set of non-dominated sorting. Meanwhile, a greedy selection is utilized in order to determine the fitter solutions. The following is the mathematical formulation behind the mutualism phase.

$$X_i' = X_i + rand(0,1) * (X_{best} - MV * BF_1) \tag{1}$$

$$X_k' = X_k + rand(0,1) * (X_{best} - MV * BF_2) \tag{2}$$

$$MV = \frac{X_i + X_k}{2} \tag{3}$$

$$\mathrm{BF}_1 = round[rand(0,1)] + 1 \tag{4}$$

$$\mathrm{BF}_2 = round[rand(0,1)] + 1 \tag{5}$$

where, $i, k \in (1, 2, \ldots, n)$; $i \neq k$

### 2.2. Commensalism phase

Commensalism involves a single organism receiving benefit from a symbiotic relationship while another is completely unaffected positively or negatively. One example of this includes the shark and remora fish, in which the fish suctions under the shark and gains access to the nutrients the shark does not eat. Meanwhile, the shark is not impacted by the exchange in any way. Commensalism is mimicked by the algorithm based on this fundamental.

For the commensalism phase, two solutions interact with one another ($X_i$ and $X_k$) (in this instance, $k \neq i$). While solution '*i*' is positively benefited from the other solution, solution '*k*' experiences no impact. The best solution ($X_{best}$) is an additional variable which solutions can be impacted by; it is selected through a random search from the Pareto set of non-dominated sorting. Meanwhile, the greedy selection is utilized in order to determine the fitter solutions. The following is the mathematical formulation behind the commensalism phase.

$$X_i' = X_i + rand(-1,\ 1) * (X_{best} - X_k) \tag{6}$$

where, $i, k \in (1, 2, \ldots, n)$; $i \neq k$

### 2.3. Parasitism phase

This parasitism phase requires one organism to be negatively affected to the benefit of a second organism. Humans and mosquitoes demonstrate this symbiotic relationship, in which the mosquito's bite releases a parasite into the human. Growing inside the body, the parasite can cause harm or kill the host if the situation becomes severe enough. One can identify the parasitic nature of this relationship in the fact that an organism is helped while the second is hurt.

The solution '$X_i$' takes queues from the Anopheles mosquito, constructing Parasite Vector (PV) which mimics the behaviors of the parasite. The formation of PV requires regeneration of parts of the solution '*i*' which are chosen via partially randomized selection using specific boundaries (*LB* and *UB*) as shown in Eq. (7). Meanwhile, '$X_k$' is derived from a solution chosen via randomized selection (note that $k \neq i$) and serves as the host of the parasite. Should the fitness value of solution '*k*' be surpassed by PV, the host will die, and the PV will take its place.

$$PV = \begin{cases} X_i^j & if\ rand(0,1) \leq rand(0,1) \\ LB + rand(0,1) * (UB - LB) & \text{Otherwise} \end{cases} \tag{7}$$

where, $j \in (1, 2, \ldots, m)$; $k \in (1, 2, \ldots, n)$; $j$ signifies design variables.

## 3. Modifications in multi-objective symbiotic organisms search (MOSOS)

Performance of MHs largely depends on the stability in the exploration & the exploitation. The exploration characterizes the global search capacity of the MHs and decides the accuracy of obtained solutions. The exploitation characterizes the local search capacity of the MHs and plays a significant part in the rapid convergence. As discussed earlier, the application of an adaptive controlling mechanism on the various MHs set a stability between the global search and a local search. Thus, adaptive BFs are proposed in the mutualism phase of MOSOS. Also, the parasitism phase of MOSOS is upgraded leading to a modified parasitism phase to address the issue regarding population diversity. The detailed discussion of the proposed improvements on the MOSOS algorithm is presented in the subsequent sections.

### 3.1. Multi-objective adaptive symbiotic organisms search (MOASOS)

In the mutualism phase of MOSOS, the two organisms of different species result from interactive learners into personal benefit of the symbiotic collaboration. Thus, the BFs ($BF_1$ and $BF_2$) are main components which defines the effect of MV. BFs are definite by a heuristically, and their values are one or two. This step outcomes in the state where populations/organisms '$X_i$' and '$X_k$' benefit partly or completely from MV. Therefore, in the mutualism phase the populations progress only with two possibilities. However, in the original mutualism, BF should not be at end positions only, but it can be in-between these limits also. Given this fact, Tejani et al. (2018c) upgraded this phase to adaptive mutualism phase by incorporating adaptive benefit factors ($ABF_1$ and $ABF_2$) to advance search capacity of the MHs, defined by the following equations:

$$\mathrm{ABF}_a = \begin{cases} f_a(X_i)/f_a(X_{best}), & if\ f_a(X_{best}) \neq 0 \\ 1 + round[rand(0,1)], & if\ f_a(X_{best}) = 0 \end{cases} \tag{8}$$

$$\mathrm{BF}_a = \begin{cases} 1, & if\ ABF_a < 1 \\ 2, & if\ ABF_a > 2 \\ ABF_a, & otherwise \end{cases} \tag{9}$$

where, $a = 1$ & 2.

The design variables ($X_i$) may get small and large displacement from their positions as various factors govern it during mutualism phase. These displacements of the design variables influence the exploration and the exploration. Hence, smaller value of BF lets the fine/local search in tiny moves but then results in faster convergence and bigger value of BF lets global search but then results in slower convergence. The '$ABF_1$' and '$ABF_2$' affects the exploration capability of the optimizer when a solution *('$X_i$' or '$X_k$')* is away from the best solution ('$X_{best}$'). The adaptive mutualism phase sets good exploitation when a solution is the neighbor of the resulting solution. Multi-objective adaptive SOS (MOASOS) purposes to efficiently incorporate the local and global search characteristic by using an adaptive mutualism phase.

### *3.2. Multi-objective modified adaptive symbiotic organisms search (MOMASOS)*

Furthermore, a parasitism phase is upgraded to a modified parasitism phase which leads MOASOS (Tejani et al., 2018c) to a new algorithm called multi-objective modified adaptive symbiotic organisms search (MOMASOS). The parasitism phase of MOSOS performances a significant role in upgrading the exploration ability of MOSOS. However, it is also observed that over exploration results in higher computational cost as a majority of the new solutions generated by the parasitism phase gets rejected due to inferior objective functional values compared to previous one (Do & Lee, 2017). Therefore, parasitism phase is improved with a modified parasitism phase of MOSOS.

In the original parasitism phase, a parasitism vector (PV), $X_i'$, is generated by mutating/altering values of few heuristically chosen design variables of the population '$X_j$', the Anopheles mosquito. Thus, the PV is a blend of '$X_i$' and random values within its bounds. The graphical representation of the parasitism phase is presented in Fig. 1. Let, '$X_i$' is the current solution with two design variables $(x_1, y_1)$ as shown in Fig. 1(a). Therefore, the updated solution ($X_i'$) or PV can either get a position within dotted lines (if single variable changes) as shown in Fig. 1(c and d) or it holds its position (see Fig. 1(b)) or it may move any random point within its bounds (if both variables change) with an equal probability. Hence, the original parasitism phase offers too explorative search which generates a large number of inferior solutions and consumes higher unnecessary computational cost.

In the modified parasitism phase, a modified parasitism vector (MPV), $X_i'$, is generated by migrating values of few heuristically chosen design variables of the population, '$X_j$', to the heuristically selected solution '$X_k$' (where $k \neq i$; selected randomly from non-dominated archive), or the Anopheles mosquito, to the current solution '$X_i$', a human host. Thus, MPV is a blend of design variables '$X_i$' and randomly selected solution '$X_k$'. The graphical representation of the modified parasitism phase is presented in Fig. 2. As discussed earlier randomly selected design variables of the solution ($X_k$) migrates to a current solution ($X_i$). Let, '$X_i$' is the current solution and '$X_k$' is randomly selected solution with two design variables ($x_1$, $y_1$) and ($x_2$, $y_2$). Thus, the updated solution ($X_i'$) or MPV can acquire a corner position of a dotted rectangle as shown in Fig. 2(b–e) with equal probability. Thus, this modification advances the exploration of search and also provides better the exploitation which offers a large number of acceptable solutions, and it also reduces computational cost. The following is the mathematical formulation of MPV behind the parasitism phase.

$$MPV,\ X_i' = \begin{cases} X_i^j & if\ rand(0,1) \le rand(0,1) \\ X_k^j & \text{otherwise} \end{cases} \tag{10}$$

where, $i,\ k \in (1,\ 2,\ \ldots,\ n)$; $j \in (1,\ 2,\ \ldots,\ m)$; $i \neq k$

Solutions are growing to a better form only if newer fitness is better than the previous one. Thus, the current solutions '$X_i$' and '$X_j$' are to be changed directly by the newer solutions '$X_i'$' and '$X_j'$', respectively. Else, the '$X_i'$' and '$X_j'$' will be incorporated to the advanced solution for choosing the next iteration ecosystem. Thus, these MHs are able to converge better by keeping good diversity among solutions. Since MHs may advance few significant data from dominated solutions in future update.

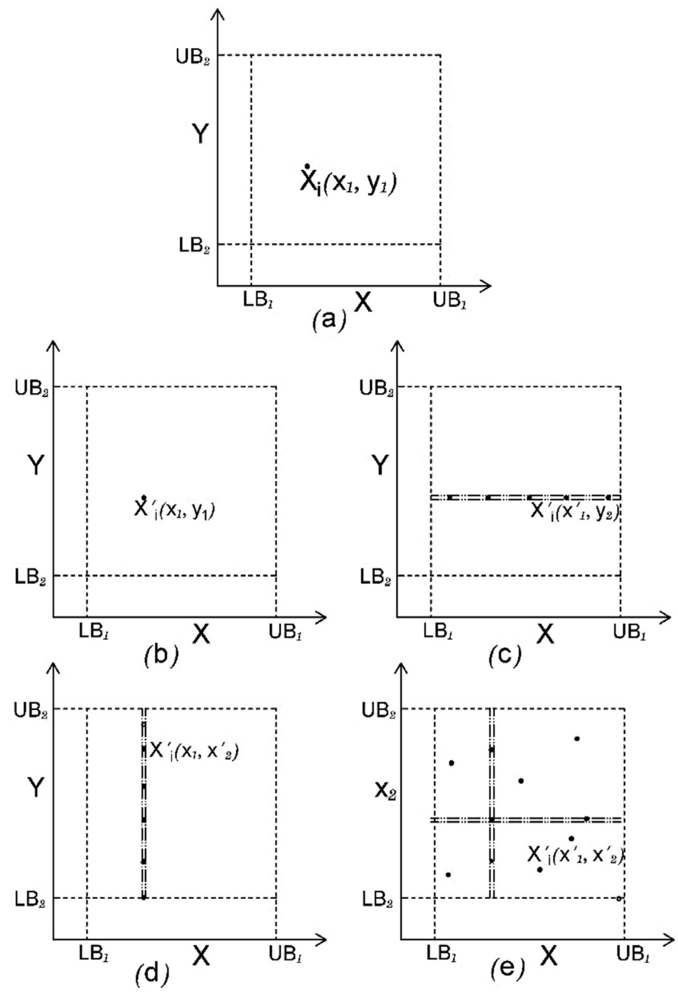


**Fig. 1.** The parasitism vectors.

The original version of MOSOS exploited the elitism strategy in combination with the crowd comparison for selection of the next generation population. This numerical strategy was successfully employed in NSGAII and some other MOEAs. The method works by using the dominance level of solutions being selected. Given a set of design solutions, non-dominated solutions are those who have dominance level being 1. If the non-dominated solutions are removed from the set, solutions having dominance level as 2 will be non-dominated solutions and so on. The idea is to choose solutions with lowest dominance levels for the next generation population. In cases that the number of solutions with lowest dominance levels exceeds the predefined population size, some of the solutions with the highest dominance level in the set who have lower cuboids are removed from the set. In this version of MOMASOS, a similar strategy is used but the normal line technique (Bureerat & Srisomporn, 2010) is used instead of the crowd comparison when some solutions are to be removed. The normal line method was originally proposed as an archiving technique for multiobjective population-based incremental learning. The method is illustrated in Fig. 3 where there are 5 solutions (circle markers)

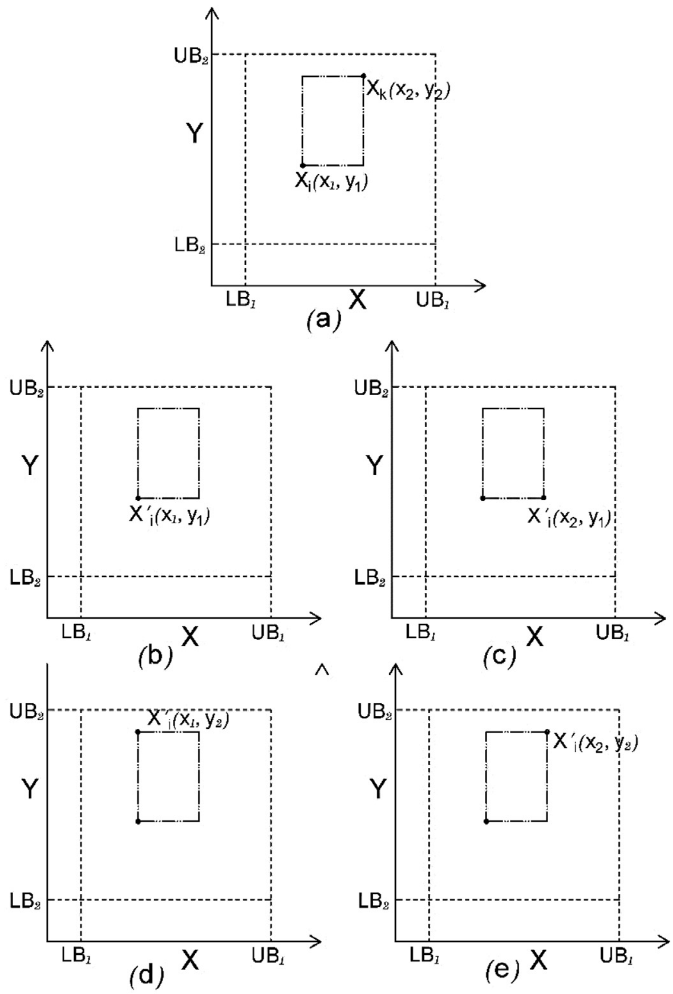


**Fig. 2.** The modified parasitism vectors.

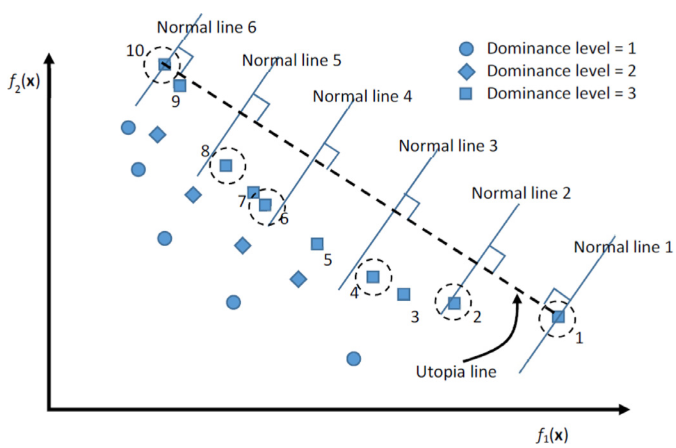


**Fig. 3.** Normal line method.

having dominance level as 1, 4 solutions (diamond markers) having dominance level as 2, and 10 solutions (square markers) having dominance level as 3. If the predefined population size is 15, it means the optimizer will keep all solutions with dominance levels 1 and 2; while, for the dominance level as 3, 4 solutions will be deleted from the population. The normal line method works, in cases of two objective functions, by identifying the anchor points who currently give the minimum values for $f_1$ and $f_2$. Then, the so-called Utopia line is drawn connecting the two anchors. The normal lines are those who are perpendicular to the Utopia line and equally placed along the line. The number of the normal lines is equal to the number of solutions require from those with the dominance level being 3, which for this example is 6 lines. The 6 selected solutions whose dominance level is 3 are those who are the closet solutions to their corresponding lines. In Fig. 3, the 6 selected solutions are inside the dashed circles.

The proposed MHs simulates initialization, mutualism phase (or adaptive mutualism phase), commensalism phase, parasitism phase (or modified parasitism phase), and stopping criteria. The combined flowchart of the proposed MHs is presented in Fig. 4.

## 4. Problem definition

A multi-objective truss design problem is defined to find discrete elemental cross-sections (design variables) to minimize truss mass and maximize deflection of nodes subject to elemental stress constraints. The truss optimum design problem is stated as:

$$Find,\ A = \{A_1,\ A_2, \ldots, A_m\} \tag{11}$$

to minimize mass and maximize nodal deflection of truss

$$f_1(A) = \sum_{i=1}^{m} A_i \rho_i\, L_i \ and\ f_2(A) = max\left(\left|\delta_j\right|\right)$$

*Subject to*:
Behavior constraints:
$g(A):\ Stress\ constraints,\ |\sigma_i| - \sigma_i^{max} \le 0$
Side constraints:
$Discrete\ cross-sectional\ areas,\ A_i^{min} \le A_i \le A_i^{max}$
$where,\ i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, n$

where, $A_{i,}\ \rho_{i,} L_i, E_i, and\ \sigma_i$ represent design variables (elements' cross-sections), density, elemental length, young's modules, and elemental stress of the '*i*th' the element respectively. '$\delta_j$' is a deflection of the '*j*th' node. The superscripts 'max' and 'min' stands for upper and lower allowable bounds respectively.

### 4.1. Dynamic penalty function

Considering both objective functions differently and it is to minimize objectives subject to 'p' limitations, the dynamic penalty function is stated as:

$$f(X) * (1 + \varepsilon_1 * \mathrm{C})^{\varepsilon_2},\ \mathrm{C} = \sum_{i=1}^{p} \mathrm{C}_i,\ \mathrm{C}_i = \left|1 - \frac{q_i}{q_i^*}\right| \tag{12}$$

where, $q_i$ signifies constraint violation with respect to the limit '$q_i^*$'. The parameter $p$ signifies a count of live constraints. The variables '$\varepsilon_1$' and '$\varepsilon_2$' are can be assumed by considering the problem characteristics. In this investigation, both '$\varepsilon_1$' and '$\varepsilon_2$' are assumed as 3, as per investigation of their effect on exploitation and exploration equilibrium (Tejani et al., 2016a, 2017, 2018a, 2018b).

## 5. Truss design problems and discussions

Five truss problems from Angelo et al. (2012, 2015) and Tejani et al. (2018c) are considered to test the effectiveness of the proposed MHs. For fair comparison, the similar parameters (Angelo, Bernardino, & Barbosa, 2015; Tejani et al., 2018c) are followed in this study. Thus, all the problems were performed with the population (organism) size of 100 and 50,000 functional evaluations. The proposed MHs are tested for 100 discrete runs. The front-hypervolume (HV) & front spacing-to-extent (STE) tests are considered for the assessment. The mean value of the HV of each MH is chosen to quantify the convergence rate of the MH and the standard deviation (STD) of HV is considered to quantify the search

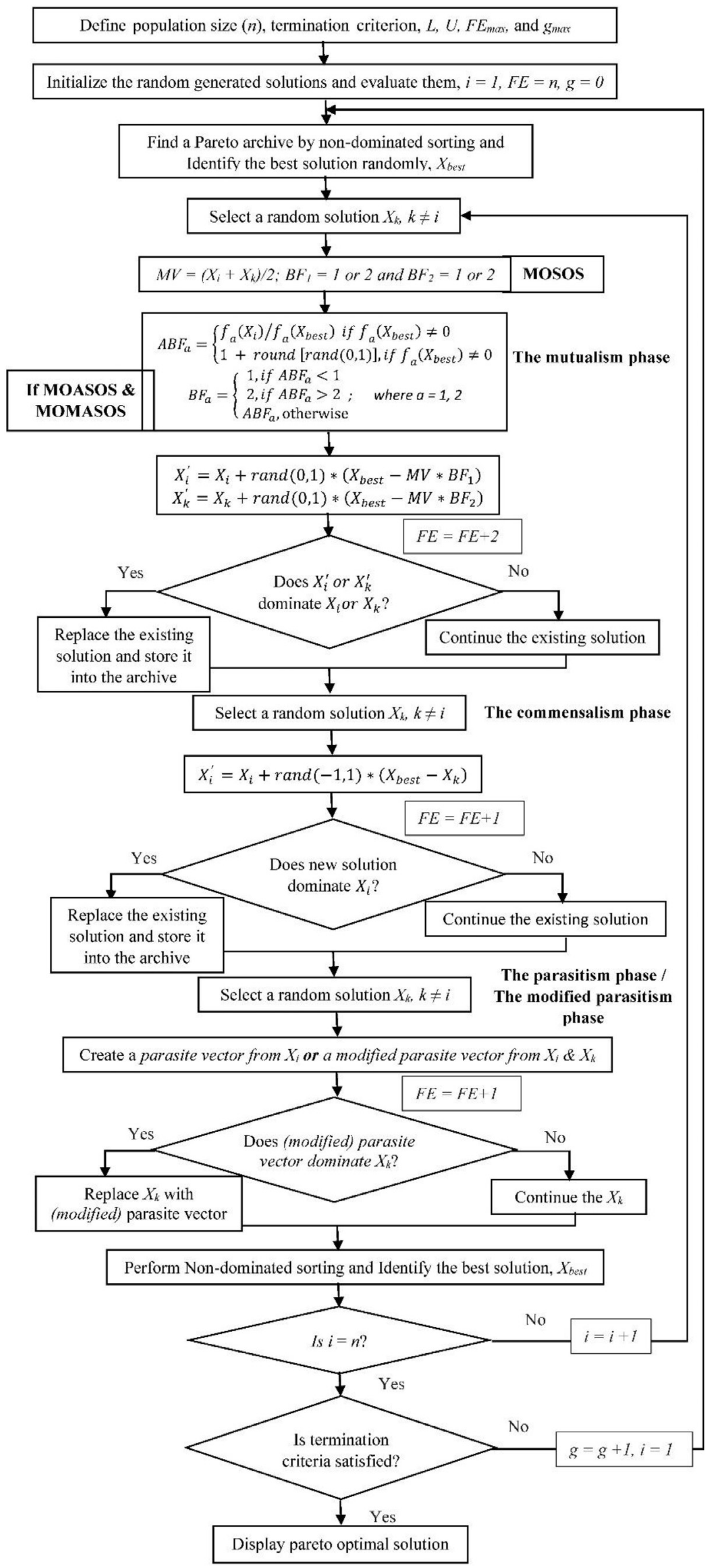


**Fig. 4.** Flowchart of the proposed algorithms.

**Table 1**
Design considerations of the truss problems.

| | The 10-bar truss | The 25-bar truss | The 60-bar truss | The 72-bar truss | The 942-bar truss |
|---|---|---|---|---|---|
| Design variables | $A_i, i = 1, 2, \ldots, 10$ | $A_i, i = 1, 2, \ldots, 8$ | $A_i, i = 1, 2, \ldots, 25$ | $A_i, i = 1, 2, \ldots, 16$ | $A_i, i = 1, 2, \ldots, 59$ |
| Constraints | $\sigma^{max} = 25$ ksi | $\sigma^{max} = 40$ ksi | $\sigma^{max} = 40$ ksi | $\sigma^{max} = 25$ ksi | $\sigma^{max} = 25$ ksi |
| Density | $\rho = 0.1$ lb/in$^3$ | $\rho = 0.1$ lb/in$^3$ | $\rho = 0.1$ lb/in$^3$ | $\rho = 0.1$ lb/in$^3$ | $\rho = 0.1$ lb/in$^3$ |
| Young modules | $E = 10^4$ ksi | $E = 10^4$ ksi | $E = 10^4$ ksi | $E = 10^4$ ksi | $E = 10^4$ ksi |

**Table 2**
The hypervolume values of results obtained for the 10-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 47,302.93 | 53,090.89 | 50,902.21 | 1294.12 | 100 | 5 |
| MOACS | 52,662.60 | 54,395.00 | 53,639.77 | 307.79 | 200 | 4 |
| MOSOS | 55,646.49 | 56,543.02 | 56,055.02 | 191.00 | 335 | 3 |
| MOASOS | 55,528.07 | 56,642.93 | 56,220.00 | 195.67 | 404 | 2 |
| MOMASOS | 55,890.60 | 56,873.05 | 56,389.83 | 166.10 | 461 | 1 |

reliability. Also, a front spacing (S) measure (Schott, 1995) is considered to test comparative distance between successive populations in the non-dominated set. The spacing of the front can be calculated as:

$$Spacing = \frac{1}{|P| - 1} \sum_{i=1}^{|P|} \left(d_i - \bar{d}\right)^2 \tag{13}$$

where $|P|$ is count of associates. $d_i$ is the Euclidian distance of objective '$i$' to its adjacent solution. $\bar{d}$ is the mean result of $d_i$'

The front extension is considered as:

$$Extent = \sum_{i=1}^{M} \left| f_i^{max} - f_i^{min} \right| \tag{14}$$

The smaller value of *Spacing* shows the superior Pareto front while, in contrast to the higher values of *Extent* is the superior. The simultaneous consideration of spacing and extent excels a new evaluation metric which simultaneously exams spacing and extent together, which is presented as the ratio of spacing to the extent,

$$STE = Spacing/Extent \tag{15}$$

where the smaller value of *STE* shows the superior non-dominated front.

In addition, Friedman's rank test, a statistical measure, is employed to rank the MHs based on the solutions found by the various optimizers. The five structural problems are addressed in the following units.

### *5.1. A 10-bar truss*

Fig. 5 presents the 10-bar truss which is a simplest and widely used truss problem compare to others. The truss properties and constraints are presented in Table 1. Fig. 5 also presents the length of each element, loading conditions, and support conditions of this truss. The discrete design variables (i.e. elemental cross-sectional areas) are assumed from forty-two discrete sections as 1.62, 1.8, 1.99, 2.13, 2.38, 2.62, 2.63, 2.88, 2.93, 3.09, 3.13, 3.38, 3.47, 3.55, 3.63, 3.84, 3.87, 3.88, 4.18, 4.22, 4.49, 4.59, 4.8, 4.97, 5.12, 5.74, 7.22, 7.97, 11.5, 13.5, 13.9, 14.2, 15.5, 16, 16.9, 18.8, 19.9, 22, 22.9, 26.5, 30, and 33.5 in$^2$ as per the previous studies (Angelo et al., 2012, 2015; Tejani et al., 2018c).

Table 2 presents the HV values for 100 optimization runs of this truss. The best, mean, and STD values of HV are considered to measure the effectiveness of the considered MHs statistically. The best mean solutions obtained by MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 50,902.21, 53,639.77, 56,055.02, 56,220.00, and 56,389.83 respectively. Also, the STD obtained using MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 1294.12, 307.79,

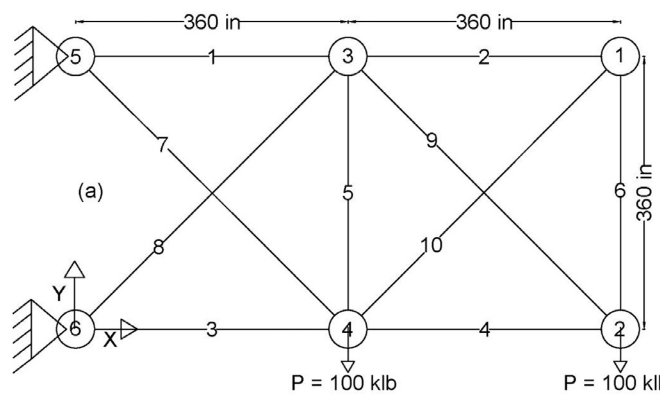


**Fig. 5.** The 10-bar truss.

191.00, 195.67, and 166.10 respectively. It is found from the results that MOMASOS performs the best followed by MOASOS and MOSOS as per the measure of search consistency. The Friedman's rank test is used to compare different MHs based on the ranks. According to the Friedman's rank test at 95% significant level, MOMASOS performs the best among the implemented MHs followed by MOASOS and MOSOS. Also, the results show that both versions of MOSOS are better than its basic version and previous studies. The results from the Friedman's rank test also indicates the significant difference among the considered MHs.

The front STE metric is considered, and the solutions are shown in Table 3. According to the Friedman's rank at 95% significant level, MOAS beats other MHs followed by MOMASOS and MOASOS, and similar results expressed as per mean of front STE. Also, MOMASOS and MOASOS perform superior than MOSOS.

Fig. 6 illustrates the best Pareto fronts of the considered MHs. It should be noted that best Pareto fronts obtained using MOAS and MOACS are discontinuous. On the contrary, Pareto fronts obtained using the proposed MHs are continuous, smooth, and have a wide range of diverse results, and the results are well distributed. Overall, these tests validate that MOMASOS is better performer followed by MOASOS and considered improvements upgrade the efficiency of MOSOS.

### *5.2. A 25-bar space truss*

The 25-bar truss is illustrated in Fig. 7. The truss properties and constraints are presented in Table 1. Loading is assumed as $P_{x1} = 1\ Klb$, $P_{y1} = P_{z1} = P_{y2} = P_{z2} = -10\ Klb$, $P_{x3} = 0.5\ Klb$, $P_{x6} = 0.6\ Klb$. Twenty-five elements are clubbed into eight groups

**Table 3**
The front Spacing-to-Extent values of results obtained for the 10-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 0.005387 | 0.024711 | 0.010590 | 0.003781 | 231 | 1 |
| MOACS | 0.007219 | 0.029625 | 0.014219 | 0.004558 | 385 | 5 |
| MOSOS | 0.007835 | 0.035081 | 0.011886 | 0.003791 | 318 | 4 |
| MOASOS | 0.007705 | 0.021266 | 0.011380 | 0.002189 | 305 | 3 |
| MOMASOS | 0.008818 | 0.018366 | 0.010721 | 0.001259 | 261 | 2 |

**Table 4**
The hypervolume values of results obtained for the 25-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 1848.04 | 1902.35 | 1878.74 | 9.77 | 123 | 5 |
| MOACS | 1850.64 | 1918.92 | 1890.61 | 14.39 | 177 | 4 |
| MOSOS | 1937.75 | 1940.43 | 1939.42 | 0.54 | 329 | 3 |
| MOASOS | 1938.75 | 1940.75 | 1939.84 | 0.51 | 371 | 2 |
| MOMASOS | 1944.13 | 1946.51 | 1945.61 | 0.45 | 500 | 1 |

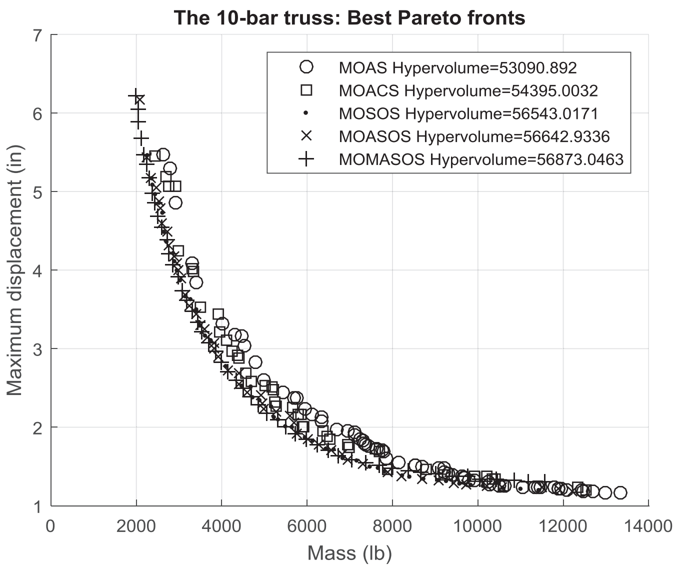


**Fig. 6.** Best Pareto fronts of the 10-bar truss.

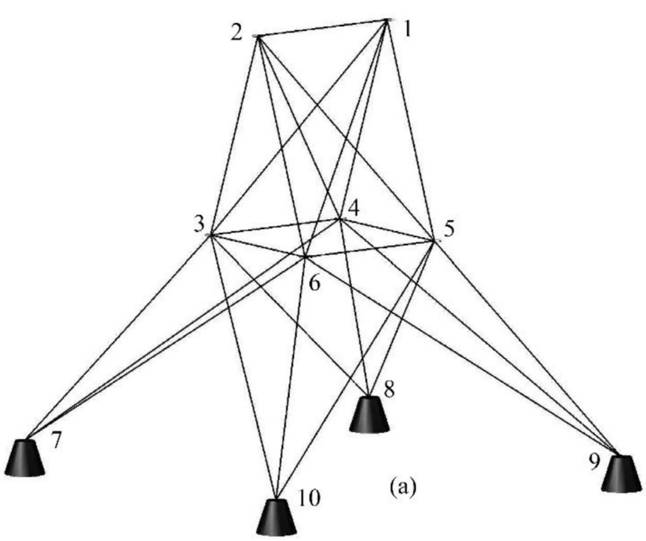


**Fig. 7.** The 25-bar space truss.

as per symmetry about x–z and y–z planes (Angelo et al., 2012, 2015; Tejani et al., 2018c). The discrete design variables (i.e. elemental cross-sections) are taken from thirty discrete sections as 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.8, 3, 3.2, and 3.4 in$^2$.

Table 4 compares the HV values for 100 independent optimization runs found from this work. The best, mean, and STD values of HV are given and will be considered to measure the performance of the considered MHs statistically. The best mean results reported by MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 1878.74, 1890.61, 1939.42, 1939.84, and 1945.61 respectively. Also, the STD obtained using MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 9.77, 14.39, 0.54, 0.51, and 0.45 respectively. It is found from the results that MOMASOS gives the finest convergence and consistency followed by MOASOS. Based on the Friedman's rank test at 95% significant measure, MOMASOS & MOASOS are the best & second-best players. Here also results show that both the versions of MOSOS are better than its basic version and previously used algorithms such as MOAS and MOACS; and MOMASOS variant is better than all the implemented MHs. The results from the Friedman's rank test also indicates the significant difference among the considered algorithms.

The front STE is tested for the truss and the findings are presented in Table 5. According to the Friedman's rank at 95% significant level, MOMASOS, MOSOS, and MOASOS rank first, second, and third respectively and mean of front STE values obtain similar results. Also, MOMASOS outperforms its basic version.

Fig. 8 presents the best Pareto fronts of the considered MHs. It should be noted that best Pareto fronts obtained using MOAS and MOACS are slightly discontinuous. On the contrary, Pareto fronts obtained using the proposed MHs are continuous, smooth, and have a wide range of diverse results, and the results are well distributed. Overall, these tests validate that MOMASOS is a fairly superior performer compare to others like the10-bar truss and these improvements elevate the efficacy of MOSOS.

### 5.3. A 60-bar ring truss

The 60-bar ring truss is illustrated in Fig. 9. The mechanical properties and limits are presented in Table 1. Sixty elements are grouped into twenty-five in view of symmetry similar to previous studies (Angelo et al., 2012, 2015; Tejani et al., 2018c). Multiple loading is assumed as load case 1: $P_{x1} = -10\ Klb$ *and* $P_{x7} = 9\ Klb$, load case 2: $P_{x15} = P_{x18} = -8\ Klb$ and $P_{y15} = P_{y18} = 3\ Klb$, and load case 3: $P_{x22} = -20\ Klb$ *and* $P_{y22} = 10\ Klb$. The discrete design variables (i.e. elemental cross-sections) are chosen from forty-five discrete sections as [0.5, 0.6, 0.7, ...,4.9] in$^2$.

**Table 5**
The front Spacing-to-Extent values of results obtained for the 25-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 0.007937 | 0.058983 | 0.022595 | 0.008424 | 445 | 5 |
| MOACS | 0.005254 | 0.044937 | 0.017026 | 0.008361 | 354 | 4 |
| MOSOS | 0.011763 | 0.013790 | 0.013255 | 0.000364 | 294 | 2 |
| MOASOS | 0.012350 | 0.014755 | 0.013326 | 0.000332 | 306 | 3 |
| MOMASOS | 0.005622 | 0.006911 | 0.006569 | 0.000213 | 101 | 1 |

**Table 6**
The hypervolume values of results obtained for the 60-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 2465.08 | 3397.56 | 3179.88 | 166.65 | 173 | 4 |
| MOACS | 2905.27 | 3276.04 | 3106.68 | 74.18 | 127 | 5 |
| MOSOS | 4271.94 | 4304.66 | 4293.25 | 5.92 | 327 | 3 |
| MOASOS | 4290.28 | 4302.54 | 4297.03 | 2.81 | 373 | 2 |
| MOMASOS | 4303.67 | 4316.33 | 4311.69 | 2.30 | 500 | 1 |

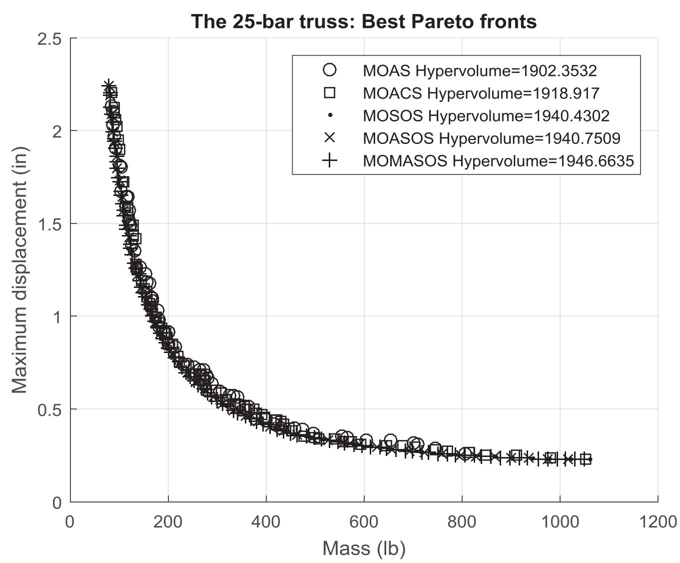


**Fig. 8.** Best Pareto fronts of the 25-bar truss.

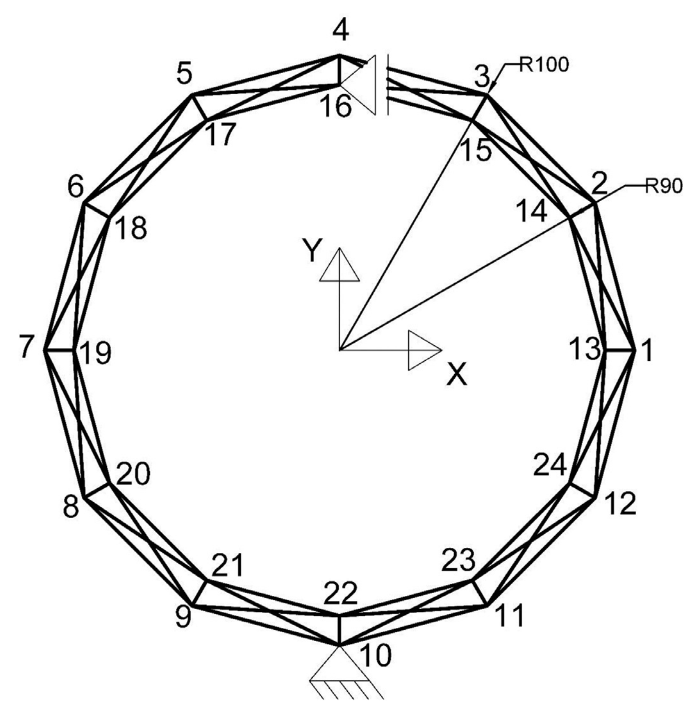


**Fig. 9.** The 60-bar ring truss.

Table 6 presents the HV obtained for the truss. The best, mean, and STD values of HV are specified and will be considered to measure the performance of the considered MHs statistically. The best mean results reported by MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 3179.88, 3106.68, 4293.25, 4297.03, and 4311.69 respectively. Also, the STD obtained using MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 166.65, 74.18, 5.92, 2.81, and 2.30 respectively. This is observed from the assessment that the best convergence and search consistency are obtained for MOMASOS while the second-best is MOASOS. The Friedman's rank test is used to compare different algorithms based on the ranks. According to the Friedman's rank test at 95% significant level, MOMASOS performs the best among the considered MHs followed by MOASOS and MOSOS. Moreover, the results show that both the versions of MOSOS are better than its basic version and previous studies such as MOAS and MOACS. MOMASOS still obtains the maximum HV for this truss. The results Friedman's rank test also indicates the significant difference among the considered algorithms. The conclusion based on the Friedman's rank test is that MOMASOS and MOASOS are again the top two performers.

The front STE is considered for the truss and the findings are illustrated in Table 7. According to the Friedman's rank test at 95% significant level, MOMASOS beats other MHs followed by MOSOS and MOASOS and similar outcomes are obtained as per mean of front STE. Also, MOMASOS performs the better compared to its basic version.

Fig. 10 shows the best Pareto fronts for all the proposed MHs. It is observed that the best Pareto fronts obtained using MOAS and MOACS are discontinuous and the results are distributed in a small region. On the contrary, Pareto fronts obtained using the proposed MHs are continuous, stable, and have a wide range of diverse results, and the results are well distributed. Overall, it is determined that MOMASOS is slightly better performer compare to the other MHs and considered improvements upsurges efficacy of MOSOS.

### 5.4. A 72-bar space truss

The 72-bar truss is illustrated in Fig. 11. The truss properties and constraints are presented in Table 1. Multiple loading is supposed as load case 1: $F_{1x} = F_{1y} = 5\ kips\ and\ F_{1z} = -5\ kips$ and load case 2: $F_{1z} = F_{2z} = F_{3z} = F_{4z} = -5\ kips$. Seventy-two elements are grouped into sixteen in view of symmetry similar to previous

**Table 7**
The front Spacing-to-Extent values of results obtained for the 60-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 0.009977 | 0.133920 | 0.034915 | 0.019500 | 451 | 5 |
| MOACS | 0.007890 | 0.074504 | 0.029912 | 0.013732 | 436 | 4 |
| MOSOS | 0.010137 | 0.012876 | 0.012025 | 0.000522 | 248 | 2 |
| MOASOS | 0.010783 | 0.012879 | 0.012147 | 0.000419 | 265 | 3 |
| MOMASOS | 0.005460 | 0.007324 | 0.006247 | 0.000400 | 100 | 1 |

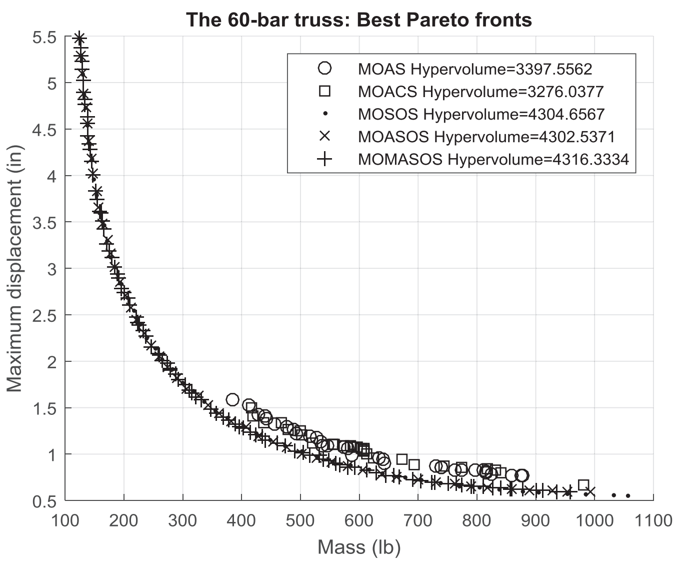


**Fig. 10.** Best Pareto fronts of the 60-bar truss.

studies (Angelo et al., 2012, 2015; Tejani et al., 2018c). The discrete design variables (i.e. elemental cross-sectional areas) are assumed from twenty-five discrete sections as [0.1, 0.2, 0.3,…, 2.5] in$^2$.

Table 8 shows the HV values obtained from this work. The best, mean, and STD values of HV are specified and will be considered to measure the effectiveness of the various MHs statistically. The best mean results reported by MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 2094.40, 2097.08, 2223.81, 2227.73, and 2233.05 respectively. Also, the STD obtained using MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 10.01, 18.78, 1.81, 1.38, and 1.05 respectively. It is found from the results that MOMASOS performs the best followed by MOASOS and MOSOS as per the measure of search consistency. The Friedman's rank test is used to compare different MHs based on the ranks. According to the Friedman's rank test at 95% significant level, MOMASOS performs the best among the considered MHs followed by MOASOS and MOSOS. Also, the results show that both the versions of MOSOS are better than its basic version and the previously used MHs such as MOAS, and MOACS; and MOMASOS variant is better than all the considered MHs. The results obtained from using the Friedman's rank test also indicates the significant difference among the considered MHs.

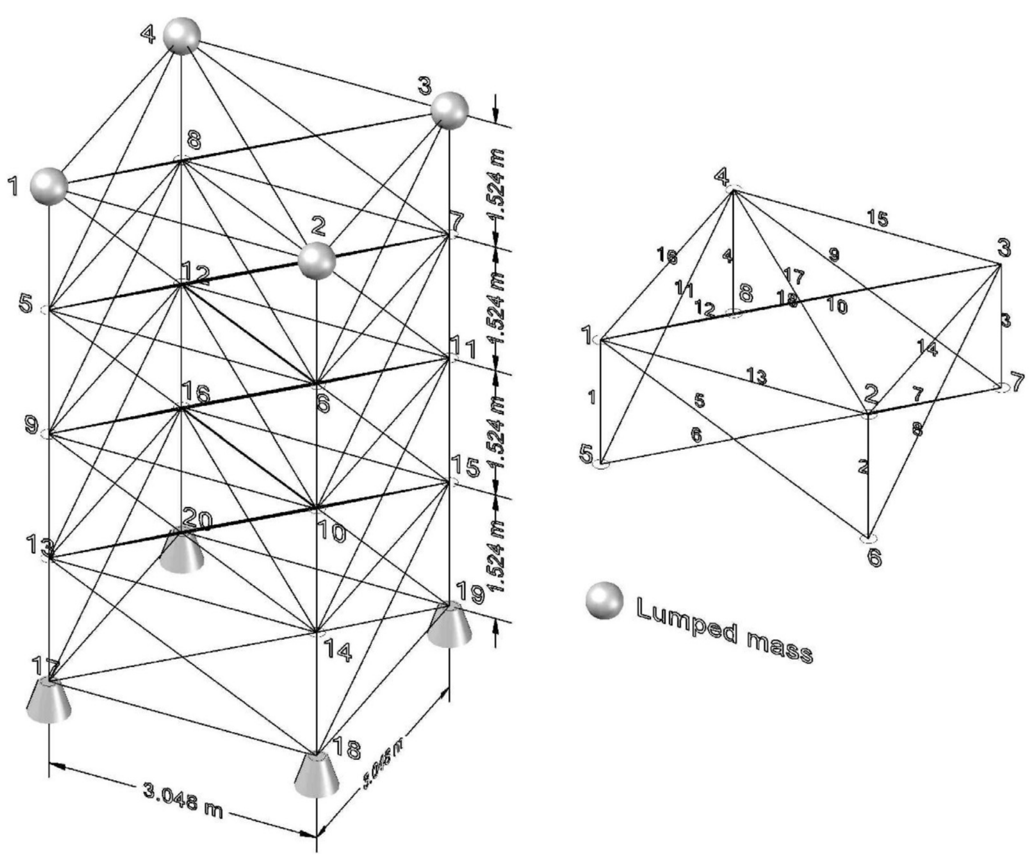


**Fig. 11.** The 72-bar space truss.

**Table 8**
The hypervolume values of results obtained for the 72-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 2066.98 | 2116.65 | 2094.40 | 10.01 | 142 | 5 |
| MOACS | 2039.80 | 2129.58 | 2097.08 | 18.78 | 158 | 4 |
| MOSOS | 2220.29 | 2227.53 | 2223.81 | 1.81 | 303 | 3 |
| MOASOS | 2223.28 | 2231.00 | 2227.73 | 1.38 | 397 | 2 |
| MOMASOS | 2229.91 | 2235.22 | 2233.05 | 1.05 | 500 | 1 |

**Table 9**
The front Spacing-to-Extent values of results obtained for the 72-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 0.010919 | 0.043568 | 0.022728 | 0.007183 | 422 | 4 |
| MOACS | 0.007918 | 0.076088 | 0.026837 | 0.013808 | 429 | 5 |
| MOSOS | 0.013350 | 0.015643 | 0.014393 | 0.000435 | 289 | 3 |
| MOASOS | 0.013026 | 0.015822 | 0.014224 | 0.000481 | 260 | 2 |
| MOMASOS | 0.006386 | 0.007519 | 0.006910 | 0.000202 | 100 | 1 |

**Table 10**
The hypervolume values of results obtained for the 942-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 44,994,042.34 | 56,957,732.05 | 51,655,929.40 | 3,380,287.34 | 150 | 4 |
| MOACS | 49,107,320.86 | 54,406,191.49 | 52,288,426.39 | 939,066.55 | 150 | 4 |
| MOSOS | 60,523,090.07 | 63,407,711.68 | 62,031,040.61 | 575,341.58 | 330 | 3 |
| MOASOS | 60,620,317.17 | 63,708,998.98 | 62,540,452.51 | 655,217.19 | 370 | 2 |
| MOMASOS | 64,805,943.66 | 112,438,733.19 | 80,006,236.39 | 7,903,607.97 | 500 | 1 |

**Table 11**
The front Spacing-to-Extent values of results obtained for the 942-bar truss.

| Algorithms | Min | Max | Mean | STD | Friedman test | Friedman rank |
|---|---|---|---|---|---|---|
| MOAS | 0.014273 | 0.120749 | 0.042659 | 0.020981 | 469 | 5 |
| MOACS | 0.010403 | 0.079907 | 0.029013 | 0.014678 | 401 | 4 |
| MOSOS | 0.012192 | 0.016924 | 0.014856 | 0.000761 | 257 | 2 |
| MOASOS | 0.013294 | 0.016218 | 0.015050 | 0.000637 | 273 | 3 |
| MOMASOS | 0.004880 | 0.008056 | 0.006423 | 0.000566 | 100 | 1 |

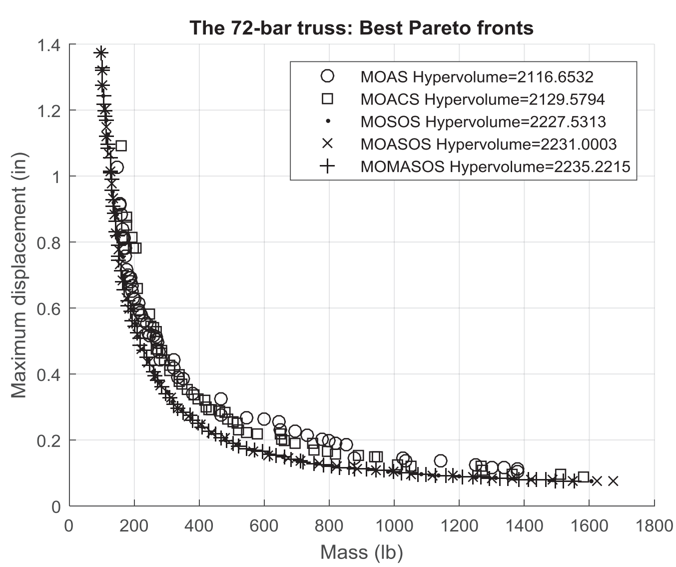


**Fig. 12.** Best Pareto fronts of the 72-bar truss.

The front STE is considered for the truss and the findings are shown in Table 9. It is noticed as per the Friedman's rank at 95% significant level, the MOMASOS, MOASOS, and MOSOS are top three performer in this order and mean values of front STE also obtain similar results. Also, the MOMASOS and MOASOS perform superior compare to MOSOS.

Fig. 12 shows the best Pareto fronts obtained all MHs. It is observed that the best Pareto fronts obtained using MOAS and MOACS are discontinuous and the results are distributed in a small region. On the contrary, Pareto fronts obtained using the proposed MHs are continuous, stable, and have a wide range of diverse results, and the results are well distributed. Overall, it is noticed that MOMASOS is slightly better performer compare to the other MHs and considered improvements upsurges efficacy of MOSOS.

### 5.5. A 942-bar tower truss

The 942-bar truss is illustrated in Fig. 13. The mechanical properties and limits are presented in Table 1. Vertical loading along Z-axis is −3, −6, and −9 kips at each node in the 1st, 2nd, and 3rd sections, respectively; lateral loading along X-axis is 1.5 and 1.0 kips at each node on the right and left sides of the tower truss; and Lateral loading along Y-axis is 1 kips at each node, respectively. The discrete design variables (i.e. cross-sections) are selected from two-hundred discrete sections as [1, 2, 3,..., 200] in$^2$. The bars are clustered into fifty-nine clusters to see structural similarity (Angelo et al., 2012, 2015; Tejani et al., 2018c).

Table 10 presents the HV values calculated for the truss. The best, mean, and STD values of HV are shown and will be considered to measure the effectiveness of the considered MHs. The best mean results obtained by MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 51,655,929.40, 52,288,426.39, 62,031,040.61, 62,540,452.51, and 80,006,236.39 respectively. Also, STD obtained using MOAS, MOACS, MOSOS, MOASOS, and MOMASOS are 3,380,287.34, 939,066.55, 575,341.58, 655,217.19, and 7,903,607.97

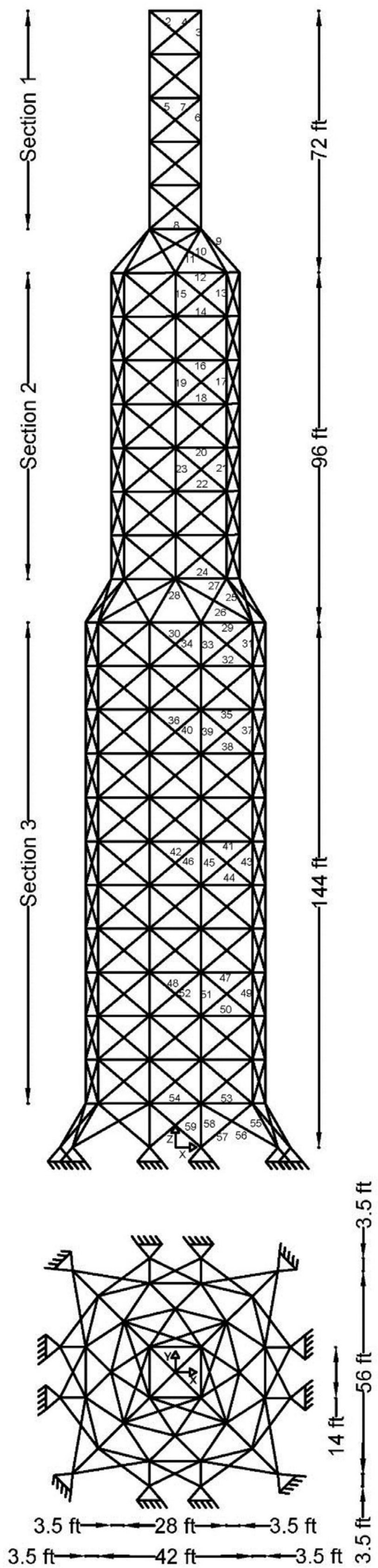


Fig. 13. The 942-bar tower truss.

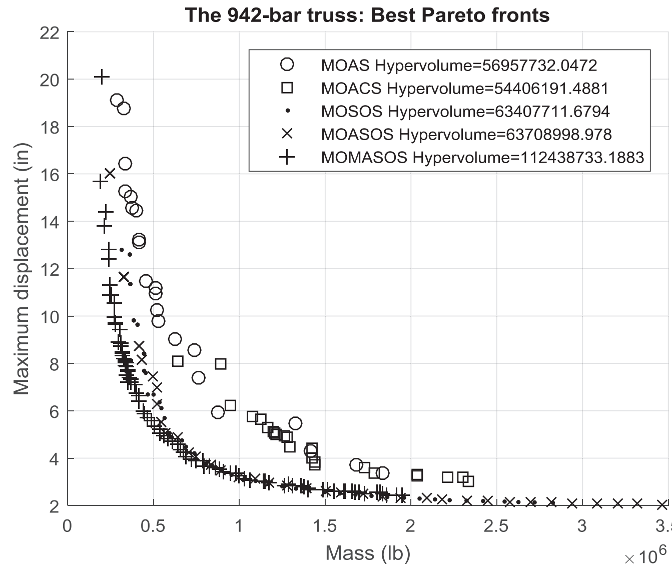


**Fig. 14.** Best Pareto fronts of the 942-bar truss.

respectively. It is found from the results that MOMASOS performs the best followed by MOASOS and MOSOS as per the measure of search consistency. In this large-scale problem case, it is shown that, with the use of the normal line method, MOMASOS is considerably superior to those using the crowd comparison as with NSGAII. The Friedman's rank test is used to compare different MHs based on the ranks. According to the Friedman's rank test at 95% significant level, MOMASOS performs the best among the considered MHs followed by MOASOS and MOSOS. Here also results demonstrate that both the versions of MOSOS are better than its basic version and previous studies such as MOAS, and MOACS; and MOMASOS variant is better optimizer compare to other MHs. The results obtained from the Friedman's rank test also indicates the significant difference among the considered MHs.

The front STE are tested for the truss and the findings are presented in Table 11. According to the Friedman's rank at 95% significant level, MOMASOS, MOSOS, and MOASOS are top three performer in this order and mean values of front STE also obtain similar results. Also, MOMASOS performs better compared to its basic version.

Fig. 14 shows the best Pareto fronts found for considered MHs. It is observed that the best Pareto fronts obtained using MOAS and MOACS are discontinuous and the results are distributed in a small region. On the contrary, Pareto fronts obtained using the proposed MOMASOS, MOASOS, and MOSOS are continuous, stable, and have a wide range of diverse results, and the results are well distributed. The Pareto fronts found using the proposed MHs are superior compare to MOAS and MOACS. Fig. 15 presents the convergence history of the truss using MOSOS and MOMASOS. The higher hypervolume value the better the Pareto front. It has been found that MOMASOS has very good convergence rate compared to MOSOS and thus owns better convergence characteristics for the truss problems.

Overall, it can be observed that MOMASOS performs superior than the other considered MHs and proposed improvements lead to better efficacy of MOSOS.

## 6. Engineering benchmark problems

In this section, four well-known engineering design optimization problems (Mirjalili, Jangir, & Saremi, 2017) are employed to validate the efficiency of MOMASOS. The first design problem is speed reducer design to minimize weight and stress of the speed

**Table 12**
The results obtained for the speed reducer design optimization problem (the previous results are adopted from Mirjalili et al., 2017).

| | GD | | MS | | S | | IGD | | Hypervolume | |
|---|---|---|---|---|---|---|---|---|---|---|
| Algorithms | Mean | STD | Mean | STD | Mean | STD | Mean | STD | Mean | STD |
| MOPSO | 0.9883 | 0.1789 | – | – | 16.6850 | 2.6960 | – | – | – | – |
| NSGA-II | 9.8437 | 7.0810 | – | – | 2.7654 | 3.5340 | – | – | – | – |
| MOALO | 1.1767 | 0.2327 | 0.8390 | 0.1267 | 1.7706 | 2.7690 | 0.8672 | 0.1490 | – | – |
| MOSOS | 1.6794 | 0.1358 | 0.4977 | 0.0082 | 39.3635 | 4.2992 | 0.0005 | 3.63E-05 | 1,864,894 | 19,141 |
| MOMASOS | 0.8501 | 0.1152 | 0.4953 | 0.0111 | 14.8755 | 1.9790 | 0.0003 | 3.08E-05 | 1,893,295 | 18,865 |

**Table 13**
The results obtained for the disk brake design optimization problem (the previous results are adopted from Mirjalili et al., 2017).

| | GD | | MS | | S | | IGD | | Hypervolume | |
|---|---|---|---|---|---|---|---|---|---|---|
| Algorithms | Mean | STD | Mean | STD | Mean | STD | Mean | STD | Mean | STD |
| MOPSO | 0.0244 | 0.1231 | 0.4604 | 0.1096 | – | – | – | – | – | – |
| NSGA-II | 3.0771 | 0.1078 | 0.7972 | 0.0661 | – | – | – | – | – | – |
| MOALO | 0.0011 | 0.0025 | 0.4496 | 0.0543 | 0.0421 | 0.0058 | 0.0194 | 0.0008 | – | – |
| MOSOS | 0.0055 | 0.0005 | 0.5649 | 0.0066 | 0.1687 | 0.0121 | 0.0006 | 0.0001 | 5.75 | 0.04 |
| MOMASOS | 0.0029 | 0.0003 | 0.5121 | 0.0041 | 0.0403 | 0.0032 | 0.0003 | 3.65E-05 | 5.94 | 0.05 |

**Table 14**
The results obtained for the welded beam design optimization problem (the previous results are adopted from Mirjalili et al., 2017).

| | GD | | MS | | S | | IGD | | Hypervolume | |
|---|---|---|---|---|---|---|---|---|---|---|
| Algorithms | Mean | STD | Mean | STD | Mean | STD | Mean | STD | Mean | STD |
| MOPSO | 0.3741 | 0.0422 | – | – | 2.5303 | 0.2270 | – | – | – | – |
| NSGA-II | 0.3601 | 0.0470 | – | – | 2.3635 | 0.2550 | – | – | – | – |
| MOALO | 0.1264 | 0.0327 | 0.3700 | 0.0025 | 1.1805 | 0.1440 | 0.1062 | 0.0152 | – | – |
| MOALO | 0.0240 | 0.0060 | 0.4694 | 0.0506 | 0.3469 | 0.0554 | 0.0012 | 0.0012 | 1.31 | 0.23 |
| MOMASOS | 0.0123 | 0.0036 | 0.4474 | 0.0625 | 0.1424 | 0.0156 | 0.0010 | 0.0012 | 1.54 | 0.32 |

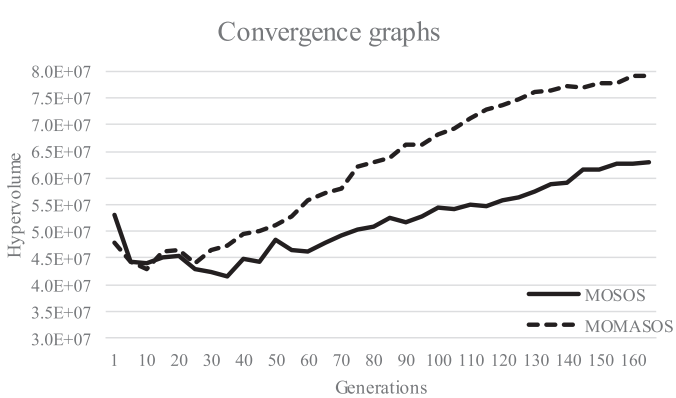


**Fig. 15.** Convergence graphs of the 942-bar truss.

reducer subject to eleven behavior constraints and seven side constraints. The second design problem is disk brake design to minimize stopping time and mass of the brake subject to five behavior constraints and four side constraints. The third design problem is welded beam design to minimize fabrication cost and deflection of the beam subject to four behavior constraints and four side constraints. And the last design problem is cantilever beam design to minimize weight and end deflection of a cantilever beam subject to two behavior constraints and two side constraints. The details of the considered problems can be studies from Ray and Liew (2002) and Deb, Zhu, and Kulkarni (2015). For fair comparison, the similar parameters (Mirjalili et al., 2017) and constrained handling techniques are followed in this study. Thus, all the problems were performed with the population size of 100, functional evaluations size of 10,000, and an archive size of 100. The proposed algorithms are tested for 30 independent runs. The Generational Distance (GD) (Veldhuizen & Lamont, 2000), Maximum Spread (MS), and metric of spacing (S) (shown in Eq. (13); Schott, 1995), Inverted Generational Distance (IGD) (Sierra & Coello, 2005), and Hypervolume. Smaller values of these measure show the superior Pareto front except for hypervolume.

$$GD = \frac{\sqrt{\sum_{i=1}^{|P|} (d_i)^2}}{|P|} \tag{16}$$

where '$|P|$' is count of obtained Pareto optimal solutions and $d_i$ is the Euclidean distance between the $i$th Pareto optimal solution and the adjacent neighbor true Pareto optimal solution.

$$MS = \sqrt{\sum_{i=1}^{f} max(d(a_i, b_i))} \tag{17}$$

where $f$ is count of objective functions, and $d$ counts the Euclidean distance, $a_i$ and $b_i$ are the highest and lowest values of $i$th objective function respectively.

$$IGD = \frac{\sqrt{\sum_{i=1}^{|P'|} (d'_i)^2}}{|P|} \tag{18}$$

where '$|P'|$' is the true Pareto optimal solutions and $d'_i$ is the Euclidean distance between the i-th true Pareto optimal obtained solution and the adjacent neighbor true Pareto optimal solution. The engineering design problems (i.e. speed reducer design, disk brake design, welded beam design, and cantilever beam design) used in this study are stated Ray & Liew, 2002 and Deb et al., 2015.

Tables 12–15 compare results obtained for the considered engineering design problems using multiple objective particle swarm

**Table 15**
The results obtained for the cantilever beam design optimization problem (the previous results are adopted from Mirjalili et al., 2017).

| | GD | | MS | | S | | IGD | | Hypervolume | |
|---|---|---|---|---|---|---|---|---|---|---|
| Algorithms | Mean | STD | Mean | STD | Mean | STD | Mean | STD | Mean | STD |
| MOPSO | – | – | – | – | – | – | – | – | – | – |
| NSGA-II | – | – | – | – | – | – | – | – | – | – |
| MOALO | 0.0002 | 1.62E-05 | 0.7673 | 0.1685 | 0.0083 | 0.0029 | 0.0002 | 0.0001 | – | – |
| MOSOS | 0.0003 | 1.71E-05 | 0.3755 | 0.0009 | 0.0359 | 0.0036 | 4.80E-05 | 6.50E-06 | 0.5595 | 0.0005 |
| MOMASOS | 0.0002 | 3.60E-06 | 0.3748 | 0.0002 | 0.0170 | 0.0010 | 3.32E-05 | 2.82E-06 | 0.5597 | 0.0001 |

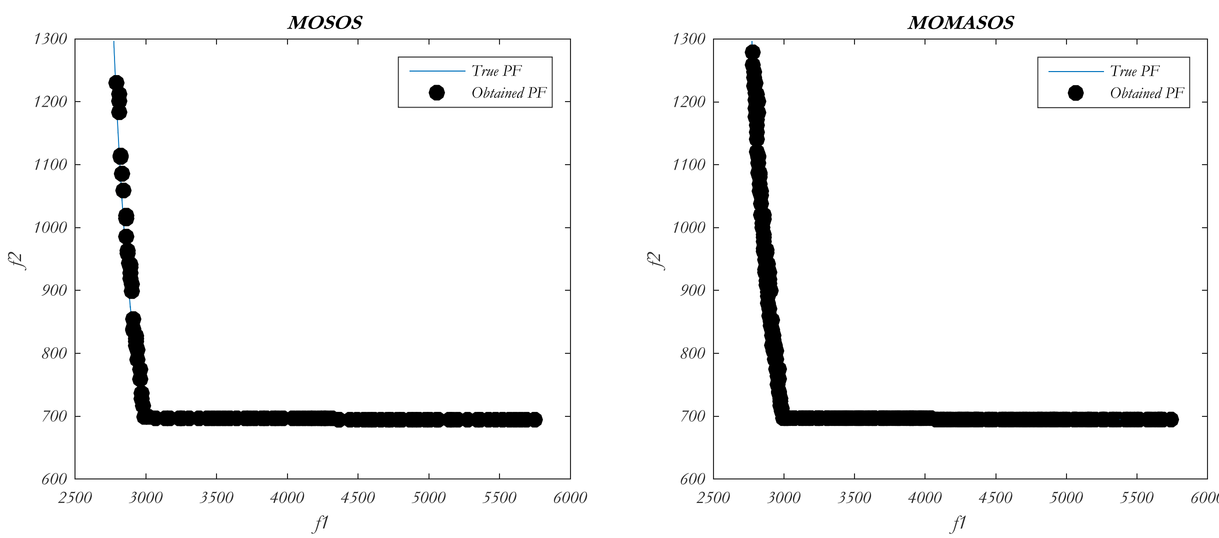


**Fig. 16.** Best Pareto optimal front obtained by the MOSOS and MOMASOS on the speed reduced design problem.

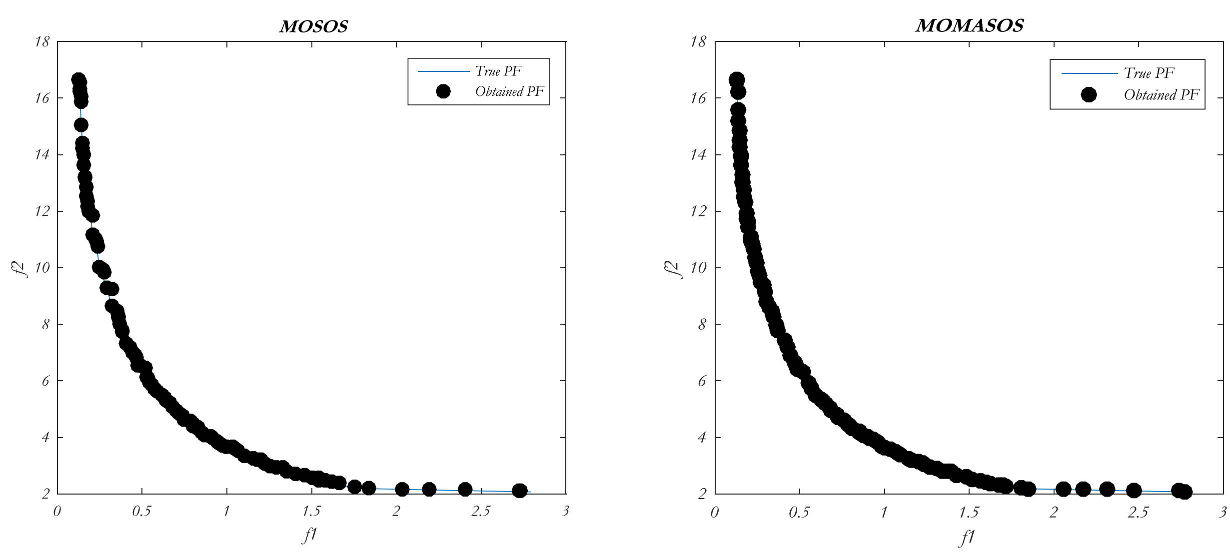


**Fig. 17.** Best Pareto optimal front obtained by the MOSOS and MOMASOS on the disk brake design problem.

optimization (MOPSO) (Coello Coello & Lechuga, 2002), NSGA-II (Deb et al., 2002), Multi-Objective Ant Lion Optimizer (MOALO) (Mirjalili et al., 2017), MOSOS, and MOMASOS with the various measures such as GD, MS, S, IGD, and hypervolume obtained. As per the results shown in result tables, it can be concluded that MOMASOS gives the best results compare to the true Pareto front and the minimum values of D, MS, S, and IGD. Also, the hypervolume of MOMASOS seems better compare to MOSOS.

The best Pareto optimal fronts obtained by MOSOS and MOMASOS for speed reducer design problem is shown in Fig. 16. It is noted that Pareto optimal front obtained using MOMASOS is uniformly distributed and near to true Pareto front compare to MOSOS.

The best Pareto optimal fronts attained by MOSOS and MOMASOS for disk brake design problem is shown in Fig. 17. It can be observed that Pareto optimal front obtained using MOMASOS is widely spread compare to MOSOS and also have majority part on the true Pareto front.

The best Pareto optimal fronts obtained by MOSOS and MOMASOS for disk brake design problem is shown in Fig. 18. It is noted

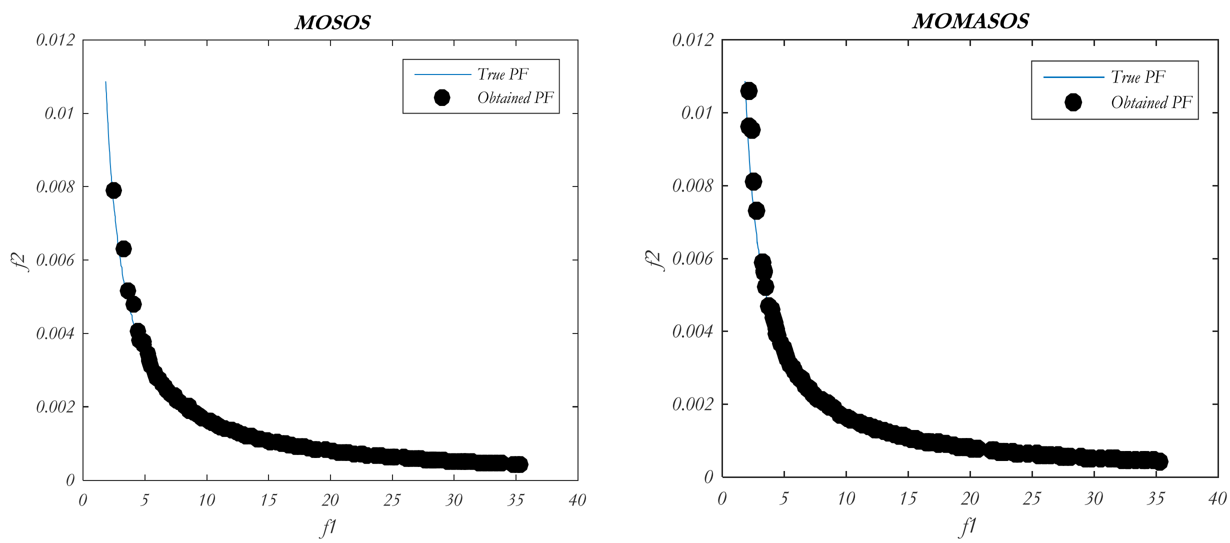


**Fig. 18.** Best Pareto optimal front obtained by the MOSOS and MOMASOS on the welded beam design problem.

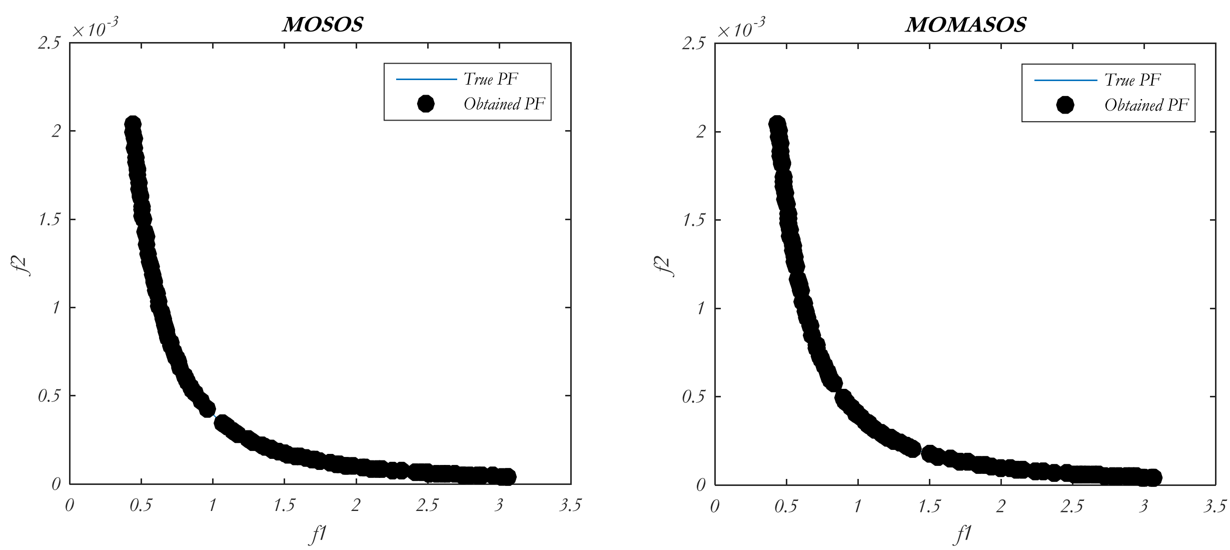


**Fig. 19.** Best Pareto optimal front obtained by the MOSOS and MOMASOS on the cantilever beam design problem.

that Pareto optimal front obtained using MOMASOS is broadly distributed compare to MOSOS and also have majority part on the true Pareto front.

The best Pareto optimal fronts obtained by MOSOS and MOMASOS for cantilever beam design problem is shown in Fig. 19. It is noted that both Pareto optimal fronts are nearly identical and on the true Pareto front.

Overall, it can be observed that MOMASOS performs superior than the methods that available in the previous literature. It can be seen from the results that the proposed improvements lead to better efficacy of MOSOS.

## 7. Conclusions

Due to the high performance and reasonable quality of obtained results in complex problems, nature-inspired meta-heuristics become an important field in expert and intelligent systems. Many decision-making problems in engineering are highly nonlinear and challenging to be solved using traditional methods. Truss design optimization problems constitute a large number of design variables and complex objective and constraints including weight, displacements, stresses, and geometrical configurations. Although multiple improvements have been reported in past literature, the problems are becoming more complex and challenging to be solved using the existing meta-heuristic algorithms. Thus, the meta-heuristic algorithms need to be improved in terms of efficiency and fitting specific problems.

A modified version of MOSOS is proposed in this paper. The new algorithm is modified in such a way to improve both the exploration and exploitation of MOSOS in reproducing design solutions for multi-objective optimization. The comparative performance studies, based on the hypervolume and spacing-to-extent indicators, reveal that the proposed MOMASOS outperforms its original MOSOS and other multi-objective meta-heuristics implemented in multi-objective truss optimization and engineering design optimization problems. The proposed adaptive mutualism and modified parasitism phases significantly improve the performance of MOSOS.

The contribution of this paper is twofold. First, we proposed two new modifications along with a normal line method as an archiving technique on an existing MOSOS to improve the quality of obtained results for multi-objective optimization in complex truss structure and engineering design problems. Two modifications are introduced including the addition of adaptive parameters in mutualism and an improved parasitism phase in order to significantly improve both the exploration and exploitation of MOSOS. Second, the successful improvement and applications of meta-heuristic algorithms are in greater need in the background of expert and intelligent systems and it provides a potential alternative for solving more complex and challenging problems that cannot be solved using the existing meta-heuristic algorithms.

Our future work is to extend the proposed MOMASOS for solving reliability optimization of trusses. Once uncertainties or random variables are taken into consideration, the truss design problem is considered robust or reliability optimization which is more complex than deterministic optimization. The powerful MOMASOS should positively respond to these difficulties.

## Credit authorship contribution statement

**Nantiwat Pholdee:** Data curation. **Sujin Bureerat:** Data curation. **Doddy Prayogo:** Formal analysis. **Amir H. Gandomi:** Conceptualization, Data curation, Formal analysis.

## Acknowledgment

The authors are grateful for the support from the Thailand Research Fund (TRF), grant number RTA6180010.

## References

Abdullahi, M., Ngadi, M. A., & Abdulhamid, S. M. (2016). Symbiotic organism search optimization based task scheduling in cloud computing environment. *Future Generation Computer Systems, 56*, 640–650. doi:10.1016/j.future.2015.08.006.

Allwood, R. J., & Chung, Y. S. (1984). Minimum-weight design of trusses by an optimality criteria method. *International Journal for Numerical Methods in Engineering, 20*(4), 697–713 doi:oRg/10.1002/nme.1620200409.

Angelo, J. S., Barbosa, H. J. C., & Bernardino, H. S. (2012). Multi-objective ant colony approaches for structural optimization problems. In *Proc. of the eleventh international conference on computational structures technology*. doi:10.4203/ccp.99.66.

Angelo, J. S., Bernardino, H. S., & Barbosa, H. J. C. (2015). Ant colony approaches for multiobjective structural optimization problems with a cardinality constraint. *Advances in Engineering Software, 80*(C), 101–115. doi:10.1016/j.advengsoft.2014.09.015.

Ayala, H. V. H., Klein, C. E., Mariani, V. C., & Coelho, L. D. S. (2017). Multiobjective symbiotic search algorithm approaches for electromagnetic optimization. *IEEE Transactions on Magnetics, 53*(6), 2015–2018 doi:oRg/10.1109/TMAG.2017.2665350.

Bureerat, S., & Pholdee, N. (2015). Optimal truss sizing using an adaptive differential evolution algorithm. *Journal of Computing in Civil Engineering*, 4015019 doi:oRg/10.1061/(ASCE)CP.1943-5487.0000487.

Bureerat, S., & Srisomporn, S. (2010). Optimum plate-fin heat sinks by using a multi-objective evolutionary algorithm. *Engineering Optimization, 42*(4), 305–323 doi:oRg/10.1080/03052150903143935.

Camp, C. V., & Farshchin, M. (2014). Design of space trusses using modified teaching–learning based optimization. *Engineering Structures, 62–63*, 87–97 doi:oRg/10.1016/j.engstruct.2014.01.020.

Cheng, M.-Y., Prayogo, D., & Wu, Y.-W. (2018). Prediction of permanent deformation in asphalt pavements using a novel symbiotic organisms search–least squares support vector regression. *Neural Computing and Applications*.

Cheng, M.-Y., & Prayogo, D. (2014). Symbiotic organisms search: A new metaheuristic optimization algorithm. In *Computers and structures: 139* (pp. 98–112). Elsevier Ltd. doi:10.1016/j.compstruc.2014.03.007.

Cheng, M.-Y., Prayogo, D., & Tran, D.-H. (2015). Optimizing multiple-resources leveling in multiple projects using discrete symbiotic organisms search. *Journal of Computing in Civil Engineering*. doi:10.1061/(ASCE)CP.1943-5487.0000512.

Çelik, E., & Durgut, R. (2018). Performance enhancement of automatic voltage regulator by modified cost function and symbiotic organisms search algorithm. *Engineering Science and Technology, an International Journal, 21*(5), 1104–1111. https://doi.org/10.1016/j.jestch.2018.08.006.

Çelik, E., & Öztürk, N. (2017). First application of symbiotic organisms search algorithm to off-line optimization of PI parameters for DSP-based DC motor drives. *Neural Computing and Applications, 30*(5), 1689–1699. https://doi.org/10.1007/s00521-017-3256-5.

Coello Coello, C. A., & Lechuga, M. S. (2002). MOPSO: a proposal for multiple objective particle swarm optimization. In Proceedings of the 2002 congress on evolutionary computation, *CEC 2002, 2* (pp. 1051–1056). https://doi.org/10.1109/CEC.2002.1004388.

Deb, K., & Jain, H. (2013). An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, Part I: Solving Problems with Box Constraints. *IEEE Transactions on Evolutionary Computation, 18*(4), 577–601 doi:oRg/10.1109/TEVC.2013.2281534.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*(2), 182–197 doi:oRg/10.1109/4235.996017.

Deb, K., Zhu, L., & Kulkarni, S. (2015). Multi-scenario, multi-objective optimization using evolutionary algorithms: Initial Results. In *2015* IEEE congress on evolutionary computation *(CEC'2015)* (pp. 1877–1884).

Degertekin, S. O., & Hayalioglu, M. S. (2013). Sizing truss structures using teaching-learning-based optimization. *Computers and Structures, 119*, 177–188 doi:oRg/10.1016/j.compstruc.2012.12.011.

Do, D. T. T., & Lee, J. (2017). A modified symbiotic organisms search (mSOS) algorithm for optimization of pin-jointed structures. *Applied soft computing*: 61. Elsevier B.V. doi:10.1016/j.asoc.2017.08.002.

Ezugwu, A. E. S., & Adewumi, A. O. (2017). Discrete symbiotic organisms search algorithm for travelling salesman problem. *Expert Systems with Applications, 87*, 70–78 doi:oRg/10.1016/j.eswa.2017.06.007.

Ezugwu, A. E. S., Adewumi, A. O., & Frîncu, M. E. (2017). Simulated annealing based symbiotic organisms search optimization algorithm for traveling salesman problem. *Expert Systems with Applications, 77*, 189–210. https://doi.org/10.1016/j.eswa.2017.01.053.

Fleury, C. (1980). An efficient optimality criteria approach to the minimum weight design of elastic structures. *Computers and Structures, 11*(3), 163–173 doi:oRg/10.1016/0045-7949(80)90155-8.

Fonseca, C. M., & Fleming, P. J. (1993). Multiobjective genetic algorithms. *IEE Colloquium on Genetic Algorithms for Control Systems Engineering*, 1–6.

Gomes, H. M. (2011). Truss optimization with dynamic constraints using a particle swarm algorithm. *Expert Systems with Applications, 38*(1), 957–968 doi:oRg/10.1016/j.eswa.2010.07.086.

Greiner, D., & Hajela, P. (2012). Truss topology optimization for mass and reliability considerations — Co-evolutionary multiobjective formulations. *Structural and Multidisciplinary Optimization, 45*, 589–613 doi:oRg/10.1007/s00158-011-0709-9.

Guha, D., Roy, P., & Banerjee, S. (2017). Quasi-oppositional symbiotic organism search algorithm applied to load frequency control. *Swarm and Evolutionary Computation, 33*, 46–67. https://doi.org/10.1016/j.swevo.2016.10.001.

Gandomi, A. H., Talatahari, S., Yang, X.-S., & Deb, S. (2012). Design optimization of truss structures using cuckoo search algorithm. *The Structural Design of Tall and Special Buildings, 22*, 1330–1349. https://doi.org/10.1002/tal.1033.

Gandomi, A. H., Talatahari, S., Tadbiri, F., & Alavi, A. H. (2013). Krill herd algorithm for optimum design of truss structures. *International Journal of Bio-Inspired Computation, 5*(5), 281. https://doi.org/10.1504/IJBIC.2013.057191.

Gandomi, A. H., & Goldman, B. W. (2018). Parameter-less population pyramid for large-scale tower optimization. *Expert Systems with Applications, 96*, 175–184. https://doi.org/10.1016/j.eswa.2017.11.047.

Ho-Huu, V., Duong-Gia, D., Vo-Duy, T., Le-Duc, T., & Nguyen-Thoi, T. (2018). An efficient combination of multi-objective evolutionary optimization and reliability analysis for reliability-based design optimization of truss structures. *Expert Systems with Applications, 102*, 262–272 doi:oRg/10.1016/j.eswa.2018.02.040.

Hosseini, S. S., Hamidi, S. A., Mansuri, M., & Ghoddosian, A. (2015). Multi objective particle swarm optimization (MOPSO) for size and shape optimization of 2D truss structures. *Periodica Polytechnica Civil Engineering, 59*(1), 9–14 doi:oRg/10.3311/PPci.7341.

Jain, H., & Deb, K. (2013). An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, Part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation, 18*(4), 602–622.

Kaveh, A., & Khayatazad, M. (2013). Ray optimization for size and shape optimization of truss structures. *Computers and Structures, 117*, 82–94 doi:oRg/10.1016/j.compstruc.2012.12.010.

Kaveh, A., & Laknejadi, K. (2011). A novel hybrid charge system search and particle swarm optimization method for multi-objective optimization. *Expert Systems with Applications, 38*(12), 15475–15488 doi:oRg/10.1016/j.eswa.2011.06.012.

Kaveh, A., & Laknejadi, K. (2013). A hybrid evolutionary graph-based multi-objective algorithm for layout optimization of truss structures. *Acta Mechanica, 224*, 343–364 doi:oRg/10.1007/s00707-012-0754-5.

Kaveh, a., & Mahdavi, V. R. (2014). Colliding Bodies Optimization method for optimum design of truss structures with continuous variables. *Advances in Engineering Software, 70*, 1–12 doi:oRg/10.1016/j.advengsoft.2014.01.002.

Kaveh, A., & Mahdavi, V. R. (2018). Multi-objective colliding bodies optimization algorithm for design of trusses. *Journal of Computational Design and Engineering*, 1–11 doi:oRg/10.1016/j.jcde.2018.04.001.

Kaveh, A., & Zakian, P. (2017). Improved GWO algorithm for optimal design of truss structures. *Engineering with Computers*, 1–23 doi:oRg/10.1007/s00366-017-0567-1.

Kumar, S., Tejani, G. G., & Mirjalili, S. (2018). Modified symbiotic organisms search for structural optimization. *Engineering with Computers.*. https://doi.org/10.1007/s00366-018-0662-y.

Lingyun, W., Mei, Z., Guangming, W., & Guang, M. (2005). Truss optimization on shape and sizing with frequency constraints based on genetic algorithm. *Computational Mechanics, 35*, 361–368 doi:oRg/10.1007/s00466-004-0623-8.

Mirjalili, S., Jangir, P., & Saremi, S. (2017). Multi-objective ant lion optimizer: A multi-objective optimization algorithm for solving engineering problems. *Applied Intelligence, 46*(1), 79–95. https://doi.org/10.1007/s10489-016-0825-8.

Mousa, A. A., El-Shorbagy, M. A., & Abd-El-Wahed, W. F. (2012). Local search based hybrid particle swarm optimization algorithm for multiobjective optimization. *Swarm and Evolutionary Computation, 3*, 1–14 doi:oRg/10.1016/j.swevo.2011.11.005.

Noilublao, N., & Bureerat, S. (2011). Simultaneous topology, shape and sizing optimisation of a three-dimensional slender truss tower using multiobjective evolutionary algorithms. *Computers and Structures, 89*(23–24), 2531–2538 doi:oRg/10.1016/j.compstruc.2011.08.010.

Noilublao, N., & Bureerat, S. (2013). Simultaneous topology, shape, and sizing optimisation of plane trusses with adaptive ground finite elements using MOEAs. *Mathematical Problems in Engineering*, 2013 doi:oRg/10.1155/2013/838102.

Panagant, N., & Bureerat, S. (2018). Truss topology, shape and sizing optimization by fully stressed design based on hybrid grey wolf optimization and adaptive differential evolution. *Engineering Optimization*, 0273 doi:oRg/10.1080/0305215X.2017.1417400.

Panda, Arnapurna, & Pani, Sabyasachi (2016). A symbiotic organisms search algorithm with adaptive penalty function to solve multi-objective constrained optimization problems. *Applied Soft Computing, 46*, 344–360. doi:10.1016/j.asoc.2016.04.030.

Pholdee, N., & Bureerat, S. (2012). Performance enhancement of multiobjective evolutionary optimisers for truss design using an approximate gradient. *Computers and Structures, 106–107*, 115–124 doi:oRg/10.1016/j.compstruc.2012.04.015.

Pholdee, N., & Bureerat, S. (2013a). Hybrid real-code population-based incremental learning and approximate gradients for multi-objective truss design. *Engineering Optimization, 46*(8), 1032–1051 doi:oRg/10.1080/0305215X.2013.823194.

Pholdee, N., & Bureerat, S. (2013b). Hybridisation of real-code population-based incremental learning and differential evolution for multiobjective design of trusses. *Information Sciences, 223*, 136–152 doi:oRg/10.1016/j.ins.2012.10.008.

Pholdee, N., & Bureerat, S. (2017). A comparative study of eighteen self-adaptive metaheuristic algorithms for truss sizing optimisation. *KSCE Journal of Civil Engineering, 22*(8), 2982–2993 doi:oRg/10.1007/s12205-017-0095-y.

Prayogo, D., & Susanto, Y. T. T. (2018). Optimizing the prediction accuracy of friction capacity of driven piles in cohesive soil using a novel self-tuning least squares support vector machine. *Advances in Civil Engineering*, 2018.

Prayogo, D., Cheng, M.-Y., Wong, F. T., Tjandra, D., & Tran, D.-H. (2018). Optimization model for construction project resource leveling using a novel modified symbiotic organisms search. *Asian Journal of Civil Engineering*.

Ray, T., & Liew, K. M. (2002). A swarm metaphor for multiobjective design optimization. *Engineering Optimization, 34*(2), 141–153. https://doi.org/10.1080/03052150210915.

Richardson, J. N., Bouillard, P., Adriaenssens, S., Bouillard, P., & Filomeno, R. (2012). Multiobjective topology optimization of truss structures with kinematic stability repair. *Structural and Multidisciplinary Optimization, 46*, 513–532 doi:oRg/10.1007/s00158-012-0777-5.

Robič, T., & Filipič, B. (2005). DEMO: Differential evolution for multiobjective optimization, 520–533. doi:oRg/10.1007/978-3-540-31880-4_36

Savsani, V. J., Tejani, G. G., & Patel, V. K. (2016). Truss topology optimization with static and dynamic constraints using modified subpopulation teaching–learning-based optimization. *Engineering Optimization, 48*(11), 1–17 doi:oRg/10.1080/0305215X.2016.1150468.

Schott, J. R. (1995). *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization* Master's Thesis. Cambridge, Massachusetts: Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.

Secui, D. C. (2016). A modified Symbiotic Organisms Search algorithm for large scale economic dispatch problem with valve-point effects. *Energy, 113*, 366–384 doi:oRg/10.1016/j.energy.2016.07.056.

Sierra, M. R., & Coello Coello, C. A. (2005). Improving PSO-based multi-objective optimization using crowding, mutation and ∈-Dominance. *Evolutionary Multi-Criterion Optimization, 1*, 505–519. https://doi.org/10.1007/978-3-540-31880-4_35.

Su, R., Wang, X., Gui, L., & Fan, Z. (2011). Multi-objective topology and sizing optimization of truss structures based on adaptive multi-island search strategy. *Structural and Multidisciplinary Optimization, 43*, 275–286 doi:oRg/10.1007/s00158-010-0544-4.

Techasen, T., Wansasueb, K., Panagant, N., Pholdee, N., & Bureerat, S. (2018). Simultaneous topology, shape, and size optimization of trusses, taking account of uncertainties using multi-objective evolutionary algorithms. *Engineering with Computers* doi:oRg/10.1007/s00366-018-0629-z.

Tejani, G. G., Savsani, V. J., & Patel, V. K. (2016a). Adaptive symbiotic organisms search (SOS) algorithm for structural design optimization. *Journal of Computational Design and Engineering, 3*(3), 226–249. doi:10.1016/j.jcde.2016.02.003.

Tejani, G. G., Savsani, V. J., & Patel, V. K. (2016b). Modified sub-population teaching-learning-based optimization for design of truss structures with natural frequency constraints. *Mechanics Based Design of Structures and Machines, 44*(4), 495–513 doi:oRg/10.1080/15397734.2015.1124023.

Tejani, G. G., Savsani, V. J., Bureerat, S., & Patel, V. K. (2018a). Topology and size optimization of trusses with static and dynamic bounds by modified symbiotic organisms search. *Journal of Computing in Civil Engineering, 32*(2), 1–11. doi:10.1061/(ASCE)CP.1943-5487.0000741.

Tejani, G. G., Savsani, V. J., Bureerat, S., Patel, V. K., & Savsani, P. (2018b). Topology optimization of truss subjected to static and dynamic constraints by integrating simulated annealing into passing vehicle search algorithms. *Engineering with Computers*. https://doi.org/10.1007/s00366-018-0612-8.

Tejani, G. G., Bureerat, S., Pholdee, N., & Prayogo, D. (2018c). Multiobjective adaptive symbiotic organisms search for truss optimization problems. *Knowledge-based systems*. Elsevier B.V.

Tejani, G. G., Savsani, V. J., Patel, V. K., & Mirjalili, Seyedali (2017). Truss optimization with natural frequency bounds using improved symbiotic organisms search. *Knowledge-Based systems*. Elsevier B.V. doi:10.1016/j.knosys.2017.12.012.

Tran, Duc-Hoc, Luong, Duc-Long, Duong, Minh-Tin, Le, Trong-Nhan, & Pham, Anh-Duc (2017). Opposition multiple objective symbiotic organisms search (OMOSOS) for Time, cost, quality and work continuity tradeoff in repetitive projects. *Journal of Computational Design and Engineering. Society for Computational Design and Engineering*. doi:10.1016/j.jcde.2017.11.008.

Tran, Duc-Hoc, Cheng, Min-Yuan, & Prayogo, Doddy (2016). A novel multiple objective symbiotic organisms search (MOSOS) for time–cost–labor utilization tradeoff problem. *Knowledge-Based Systems, 94*, 132–145. doi:10.1016/j.knosys.2015.11.016.

Veldhuizen, D. A., & Lamont, G. B. (2000). Multiobjective evolutionary algorithms: Analyzing the state-of-the-art. *Evolutionary Computation, 8*, 125–147.

Wang, H., Jiao, L., & Yao, X. (2015). Two Arch2: An improved two-archive algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation, 19*(4), 524–541 doi:oRg/10.1109/TEVC.2014.2350987.

Wei, L., Tang, T., Xie, X., & Shen, W. (2011). Truss optimization on shape and sizing with frequency constraints based on parallel genetic algorithm. *Structural and Multidisciplinary Optimization, 43*(5), 665–682 doi:oRg/10.1007/s00158-010-0600-0.

Yu, V. F., Perwira Redi, A. A. N., Yang, C. L., Ruskartina, E., & Santosa, B. (2017). Symbiotic organisms search and two solution representations for solving the capacitated vehicle routing problem. In *Applied soft computing: 52* (pp. 657–672). Elsevier B.V. doi:10.1016/j.asoc.2016.10.006.

Zhang, B., Sun, L., Yuan, H., Lv, J., & Ma, Z. (2016). An improved regularized extreme learning machine based on symbiotic organisms search. In *Proceedings of the 2016 IEEE 11th conference on industrial electronics and applications, ICIEA 2016* (pp. 1645–1648). doi:oRg/10.1109/ICIEA.2016.7603849.

Zhang, X., Tian, Y., & Jin, Y. (2015). A knee point-driven evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation, 19*(6), 761–776 doi:oRg/10.1109/TEVC.2014.2378512.

Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the strength pareto evolutionary algorithm. *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, 95–100 doi:oRg/10.1.1.28.7571.

Zitzler, E., Laumanns, M., & Thiele, L. (2002). SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. *Evolutionary Methods for Design, Optimization and Control, 1*, 1–6.

Zuo, W., Xu, T., Zhang, H., & Xu, T. (2011). Fast structural optimization with frequency constraints by genetic algorithm using adaptive eigenvalue reanalysis methods. *Structural and Multidisciplinary Optimization, 43*(6), 799–810 doi:oRg/10.1007/s00158-010-0610-y.