Advanced Steel Construction – Vol. 15 No. 3 (2019) 274–287
DOI:10.18057/IJASC.2019.15.3.8

DESIGN OPTIMIZATION OF TUBULAR LATTICE GIRDERS

Tugrul Talaslioglu

Department of Civil Engineering, Osmaniye Korkut Ata University, 80000, Osmaniye, Turkey

* *(Corresponding author: E-mail: ttalaslioglu@osmaniye.edu.tr, talaslioglu@cu.edu.tr)*

A B S T R A C T

The lattice girder, members of which are constructed by use of ready profiles with tubular cross-sections, has a simple but an effective structural framing form. In this regard, this study proposes to optimize the design of tubular lattice girders in a way of minimizing its entire weight and joint displacement and maximizing its load-carrying capacity considering the design codes of API RP2A-LRFD. As an optimization tool, a multi-objective optimization methodology named pareto archived genetic algorithm (PAGA) was utilized. The search capability of PAGA was improved by involving a designer module for automatically creation of a lattice girder form. The improved PAGA has a big responsibility of increasing the convergence degree of optimal designs against the stability problem. Furthermore, the content of this study is enriched by evaluating the computing efficiency of PAGA with respect to several multi-objective optimization algorithms. Consequently, the improved PAGA achieves to explore the optimal lattice girder designs with the higher convergence, diversity and capacity degrees. Therefore, the proposed optimum lattice girder design tool is recommended for the designers due to its capability of obtaining a wide range of promising designs.

A R T I C L E   H I S T O R Y

Received: 21 November 2017
Revised: 9 January 2019
Accepted: 31 March 2019

K E Y W O R D S

Multi-objective optimization;
Tubular lattice girder;
Pareto dominance;
Genetic algorithm;

1. Introduction

A lattice girder is constructed through connecting a number of diagonal, longitudinal and horizontal structural members. The lattice girder have a big advantages over the other structural framing forms (configurations) due to: *i)* its ability of spanning the long distances without requiring either any intermediate support or an extra height as in the dome structure *ii)* its flexibility of having an option of using a steel manufacturing factory, *iii)* its capacity of easily disassembling and reassembling, *iv)* its capability of altering their current appearance with the different framing configurations. These significant features lead to a decrease in the manufacturing, building, shipping, and maintaining costs of lattice girder. Therefore, the lattice girders are preferably utilized in a number of structural systems (roofs, bridge, cranes etc.) (Nageim & Macginley [1], Talaslioglu [2]).

Particularly, the development in the steel-related technology accelerates to elevate the variety in the available steel profiles. In fact, this development also leads to a big differentiation in the current framing configuration of lattice girder. Thus, the possibility of making a further decrease in the constructional cost of lattice girder correspondingly increases. Moreover, in order to gain an extra economic profit in the construction of the tubular lattice girder, it is suggested that an optimization tool to be involved into the design stage is the best way. As an optimization tool, the evolutionary algorithms (EAs) have already been utilized for the weight minimization of various structures (Hasancebi and *et al.* [3]; Saka [4], Seyedpoor and *et al.* [5]).

A generational optimal design approach for steel structures has still been utilized by a number of designers. According to the generational optimal design approach for the tubular lattice girder, a designer firstly determines the shape of lattice girder for a certain spanning length and topology. Then, the size of lattice girder members is optimized using only a single objective function, for example the constructional cost of tubular structure. Thus, the entire weight of tubular structure is minimized to obtain a further economic design. The design constraints are assigned to check the strengths of the lattice girder members according to an allowable member stress and/or joint displacement value (Cagnina and *et al.* [6]; Chen & Huang [7]; Lu and *et al.* [8]; Torii and *et al.* [9]). However, it has to be taken into account of being governed the strength of members by a number of coupled strength-related criteria (Beer and *et al.* [10]). Therefore, the best way is to include the provisions of a national or international design specification into the design stage in order to increase the design reliability of lattice girder (Talaslioglu [11]).

Furthermore, the proposed optimal design approach must take the responsibility of not only decreasing the entire weight of lattice girder for an economic constructional cost, but also achieving to provide an increased load-carrying capacity along with a higher serviceability for the design of lattice girders (Talaslioglu [11]).

Thus, the generational optimal design approach with a single-objective function becomes unfortunately to be insufficient in case of optimizing the design of lattice girder considering its load-carrying capacity, serviceability and weight at the same time. In order to deal with this design problem, the most appropriate remedy is to integrate a multi-objective optimization procedure with the proposed design approach. Thus, the designer has also an opportunity to make a trade-off analysis among the multiple objective function values. Fortunately, EAs are also proved to be the perfect optimization tools with multiple objectives (Talaslioglu [11], Salajegheh and et. al. [12]).

A general multi-objective optimization problem is formulated as:

$$\min \; and/or \; \max \; F(X) = \{ f_1(X), f_2(X), ..., f_K(X) \} \tag{1}$$

K objective functions are defined depending on a *design (decision) variable set X*, each which is a member of *design variable space DS*. When a multi-objective optimization algorithm (MOA) is executed, a set of random solutions is generated. Several solutions are *dominated* with respect to the other remaining ones ($f_i(X) \leq f_1(X)$ and $f_i(X) \neq f_1(X)$, $\forall i \in (1,…,K)$)) (Srinivas & Deb [13]). The dominated optimal solutions are defined as *pareto solution*. The state of *pareto dominance* is formulated in DS as:

$$P^* = \left\{ f_i(X) \leq f_i(X^*) \; : \; \begin{cases} f_i(X) \neq f_i(X^*), \forall i \in [1,...,K) \; and \; \forall X \in DS \\ g_n(X^*) \leq 0 \quad , n = (1,...,N) \\ h_m(X^*) = 0 \quad , m = (1,...,M) \end{cases} \right\} \tag{2}$$

The combination of pareto solutions makes a form named *Pareto front*:

$$PF^* = \left\{ F(X), \; X \in P^* \right\} \tag{3}$$

The pareto front is updated throughout the search of MOA. If a new dominant pareto solution is never obtained for the *current pareto front*, then the current pareto front is defined as *true pareto front*. In fact, the current and true pareto front has a big importance for not only the trade-off analysis but also a performance comparison of MOAs.

Although evaluating the computing performance of any optimization algorithm with a single objective is carried out by only assessing the quality of optimal solution, it is difficult for the multi-objective optimization algorithm due to the number of objective functions. In order to evaluate the computing efficiencies of employed MOAs, the several performance metrics named *capacity*, *density*, *convergence-diversity*, *coverage* are utilized. These performance metrics utilized in this study are also called as “unary metrics”, the working mechanisms of which are governed by the predetermined approximation sets (Zitzler and *et al*. [14]). The predetermined approximate sets are generated using the current and true pareto fronts.

The measuring metric named “capacity” informs us about the number of current pareto solutions. Particularly, this quantity metric has a big importance for the optimal engineering design problems since the designer mostly desires to make a trade-off analysis considering pareto solutions. Therefore, the larger size of pareto solutions becomes a big advantage for the designer. The density metric "average distance" is utilized to measure the density degree of current pareto solutions in a way of computing the average value of deviation between Euclidian distance and average Euclidian distance values among current pareto solutions. The lowest average distance value indicates about a higher density degree. The other two diversity metrics "epsilon" and "spread" also measure the diversity in a way of computing the maximum distance and covering degrees of current pareto solutions with respect to true pareto fronts, respectively (Zitzler and *et al*. [14], Wu and Azarm [15], Deb and *et al.* [16], Veldhuizen and Lamont [17]). In fact, the quantity metric "epsilon" is also utilized to indicate about the pure convergence degree since its minimum value shows about being covered the entire region of true pareto front by the current pareto front. This quantity metric is computed using both Euclidian and average Euclidian distances among the current pareto solutions along with true pareto fronts. In order to measure the volume of current pareto solutions with respect to the true pareto fronts, the quantity metric "R2" is utilized (Hansen and Jaszkiewicz, [18]). In fact, R2 indicator has a responsibility for determining both convergence and diversity degree of current pareto solutions in a way of utilizing two extreme data points named "nadir" (maximum point) and "ideal" (minimum point) along with a utility function named "weights" for obtaining the equally scattered solutions on a certain range. It is noted that R2 indicator, which is computed depending on the use of utility function and data of current pareto fronts, is a member of R family. Thus, the value of R2 indicator is obtained as an average value of differences between the maximum values of current data and utility functions. Whereas Matlab scripts of quality indicators R2 and epsilon are available in Reference (Wagner & Kretzschmar [19-20]), the average distance and spread indicators are embedded in the Matlab scripts named "distanceAndSpread.m" (see the further details in Matlab [21].

This study proposes to optimize the design of tubular lattice girder considering its load-carrying capacity, serviceability and entire weight. The design optimization of lattice girder is carried out using a multi-objective optimization tool, named "pareto archived genetic algorithm (PAGA)". PAGA developed by Author has been also utilized to optimize the design of dome structures considering the nonlinear structural behavior (Talaslioglu [22]). In this study, the exploration capacity of PAGA for the optimization of lattice girder design is enhanced through a designer module for automatically creation of a lattice girder configuration. While the shape and topology of lattice girder is automatically arranged, its size is accordingly chosen among the available steel profiles. The computing efficiency of PAGA is assessed through not only the tubular lattice girder designs, but also three MOAs named NSGAII, EVMOGA and SMSEMOA considering the performance metrics named capacity, density, convergence-diversity and coverage.

In this regard, the outline of this study begins by introducing the working mechanism of proposed evolutionary-based multi-objective optimization methodology "PAGA" and its integration with design of tubular lattice girder in section 2 and 3, respectively. Then, the results outcome from the application of employed MOAs for a tubular lattice girder with various spans and loading situations along with several benchmark examples are discussed in section 4. The summarization of results is given in the conclusion section.

2. The Working Mechanism of Proposed Evolutionary-Based Multi-Objective Optimization Methodology: Pareto Archived Genetic Algorithm (PAGA)

It is mentioned that EAs are also perfect optimization tools for the multi-objective optimization problems. EAs belongs to the evolutionary-based field in the area of artificial intelligent (Yang [23]). In fact, the fundamentals of EAS are completely constituted on the Darwian's evolutionary theory. The evolutionary theory contains the basic biological issues: breeding suitable *individuals* of population, propagating the valuable *genetic material*, which are coded in *chromosomes*, into next *generations*, eliminating the unsuitable one and finally surviving the *individuals* with a higher *fitness*. The concept of *fitness* informs us about the adaptation degree of individuals to both nature and life under an environmental pressure. In this regard, EAs mimics the working mechanism of biological evolution by means of *genetic operators*. The quality of fitness is easily computed by using a *fitness function* and utilized in the management of genetic-based processes. Thus, the evolutionary based computation begins by initiating a *population*. Then, the genetic operators are employed to process the genetic material embedded in the initial population considering the *fitness values* which is computed by fitness function. The genetic material inherited from the initial population is also utilized to generate the new populations. The generation of populations is generally terminated when a pre-defined *maximum generation number* is completed. The evolutionary-based loop is resulted by a number of pareto solutions. In this regard, the closeness, diversity and spread degrees of its current pareto front with respect to a true pareto front are the most important criteria in order to evaluate the computing efficiency of MOAs. In fact, these quality measuring metrics are also utilized to develop or improve the new multi-objective evolutionary algorithms (MEAs) (Reed and *et al.* [24]; Metaxiotis & Liagkouras [25]; Richardson and *et al.* [26]). Some promising approaches are developed as: *i)* aggregation of fitness values (Aggregation Method (Srinivas & Deb [13]; Hwang & Masud [27]), Weighted Metrics (Miettinen [28]), Goal Programming (Charnes & Cooper [29]) etc…) , *ii)* dominance of fitness values (NSGAII (Deb and *et al.* [30]), SPEA2 (Zitzler and *et al.* [31]) etc. (see further details about the other similar approaches in (Talaslioglu [32]), *iii)* quantitative-based assessment of fitness values (IBEA (Zitzler and *et al.* [33]), ESP (Huband and *et al.* [34]) etc.). Especially, non-dominated sorting genetic algorithm II (NSGAII) (Deb and *et al.* [30]) achieve to gain more attention from the scientific research audience (Talaslioglu [11, 35]). The hybridized variants of MOAs have been also developed. Particularly, two recently developed MOAs are e-dominance based MOA (EVMOGA) (Martínez and *et al.* [36]) and hyper-volume dominance-based MOA (SMSEMOA) (Beume and *et al.* [37-38]).

One of MEAs, Genetic Algorithm (GA) achieves to take more attention from different engineering fields due to its simple but effective search mechanism (Yang [23]). However, one of the difficulties in the evolutionary-based multi-objective optimization procedure is generally concerned with determining an optimal solution with higher quality in a way of simultaneously using the multi-objective functions. Although it is known that the valuable genetic material is embedded to the pareto solutions, how to propagate this genetic material to next generations is not known. At this point, the fundamentals of PAGA are constituted on the transmission of pareto solutions to the next populations through pareto-inseminated populations (see the pseudo code of PAGA in Fig. 1). PAGA is managed by four different sub-populations Spop1, Spop2, Spop3 and Spop4, which are obtained thereby dividing the entire population sub-populations by a division number *div_num*. In this regard, an evolutionary search proposed by PAGA begins to randomly generate these four sub-populations. Then, fitness values (f_1, f_2, etc.) and pareto solutions (*pareto_OF_ALL*) are correspondingly computed and determined, respectively. The limit values of objective functions stored in the pareto solutions *max_pareto_OF1_ALL*, *max_pareto_OF2_ALL* and etc. are determined and utilized to compute the size of sub-populations *PS1*, *PS2*, *PS3*, and *PS4* along with *div_num*. Then, the sub-populations sizes are determined. The crossover and mutation operations, the parameter values of which are adjusted depending on sub-generations numbers (*subGN1, subGN2*), are utilized to operate the genetic material in the sub-populations *Spop1, Spop2, Spop3 and Spop4*. After the fitness values of sub-populations *Spop1, Spop2, Spop3 and Spop4* are computed, the pareto solutions are determined. In the last stage, the bounds of individuals located in the each of sub-populations are checked according to the *maxDV* and *minDV*. This main loop is terminated once the maximum generation number is completed. The further details about working principle of PAGA are presented in Reference Talaslioglu [22].

3. The Integration of PAGA with Design of Tubular Lattice Girder

The design optimization of tubular lattice girder is carried out by both minimizing its entire weight, f_1 (*m*, number of lattice girder members; *w*, unit weight per member length) and its joint displacements, f_2 (n, number of nodes; *i*, number of freedom) and maximizing its member forces, f_3 (see Eqn. (1-3)) considering design constraints obtained by use of member-related provisions.

The proposed design constraints are converted into the two ratios for the member-related design constraints and joint displacements. While one of these ratios named $Unity_{Mem}$ is obtained by dividing the current strength of dome member to the allowable nominal strength, the other ratio named $Unity_{Disp}$ is computed by dividing the maximum value of current joint displacement to the predefined value. In order to penalize the unfeasible designs, a penalizing function is accordingly utilized to compute a penalizing value. The penalizing value is added to the fitness value. It is noted that the same penalizing process with an alteration of design constraints is also proposed for the optimization of benchmark design examples.

$$f_1 = \min\left(\sum_{k=1}^{m} (w.l)_k + Pen_1\right) \tag{4}$$

$$f_2 = \min(d_{ij} + Pen_2) \quad (i = 1,...,12 \; and \; j = 1,...,n) \tag{5}$$

$$f_3 = \max(f_{ij} + Pen_3) \quad (i = 1,...,12 \; and \; j = 1,...,n) \tag{6}$$

Where,

$$
\begin{aligned}
Pen_1 &= \left(\sum_{k=1}^{m}(w.l)_k\right).\left(\frac{CGN}{MGN}\right)^{\frac{1}{P_{Mem}+P_{Disp}}} \\
Pen_2 &= \min(d_{ij}).\left(\frac{CGN}{MGN}\right)^{\frac{1}{P_{Mem}+P_{Disp}}} \\
Pen_3 &= \max(f_{ij}).\left(\frac{CGN}{MGN}\right)^{\frac{1}{P_{Mem}+P_{Disp}}}
\end{aligned}
\quad
\begin{cases}
P_{Mem} = \sum_{k=1}^{m} p_k, \left\{ p_k = 1 \; if \; Member\_Strength\_Contsrains \geq 1 \right. \\
P_{Disp} = \sum_{j=1}^{n}\sum_{i=1}^{12} p_d^{ij}, \left\{ p_d^{ij} = 1 \; if \; Unity_{Disp}^{ij} \geq 1 \right. \\
Pen_1 = 0 \; and \; Pen_3 = 0 \quad if \; Member\_Strength\_Contsrains < 1 \\
Pen_2 = 0 \quad if \; Unity_{Disp}^{ij} < 1
\end{cases}
\tag{7}
$$

$$
Member\_Strength\_Contsrains = \begin{cases} Unity_{Axial}^{k}, Unity_{Bending}^{k}, Unity_{CombinedBending}^{k}, Unity_{Shear}^{k}, \\ Unity_{Torsion}^{k}, Unity_{AxialCompr\,\&\,BendingBuck}^{k}, Unity_{Axial\,\&\,BendingYield}^{k} \end{cases} \tag{8}
$$

In Eq. 5, penalizing process is managed by two parameters: *current generation number* CGN and *maximum generation number* MGN. The formulations of design constraints in Eq. 5 are given in References (Talaslioglu [11]).

In order to improve the search capability of PAGA, a designer module is included to automatically create a lattice girder configuration (see a pseudo code for an automatically generation of lattice girder in Fig. 2). Thus, it is possible to generate various framing configurations using the design variables (size, topology and shape). The cross-sectional properties which represents the design variables (size) *D1*, *D2*, *D3* and *D4* are chosen from 37 available tubular cross-sections (see Fig. 3). The limit values of these design variables Par_{UDV} and Par_{LDV} are taken as 37 and 1. The other design variable (topology) is managed by the division number Par_{DN} in order to form the tubular lattice girder with small nesting parts. The limit values of this design variables Par_{DN} are indicated by Par_{DNU} and Par_{DNL}. Furthermore, the other topology-related design variable Par_{CTop} is also utilized to represent each nesting part of tubular lattice girder named "Cell". The brace members manage the topology of cell. Therefore, the topologic configuration of each cell is represented by four different numbers: 1 for no bracing, 2 for left-sided bracing (\), 3 for right-sided bracing (/) and 4 for double bracing (X). The last design variable (shape) is managed by the first and middle heights of tubular lattice girder H1 and H2 due to the symmetrically adjusted by the designer module and indicated by Par_{H1} and Par_{H2}. The limit values of shape-related design variables are indicated by the parameters Par_{H1U}, Par_{H1L}, Par_{H2U} and Par_{H2L}. Thus, these design parameters are stored in a matrix named "Chrom" ("Chromosome"), which contains the design variables Par_{DN}, Par_{CTop}, Par_{H1}, Par_{H2}, *D1, D2, D3, D4*. However, the main difficulty is related to the determination of topologic configurations for each cell of tubular lattice girder since Par_{CTop}, which varies depending on Par_{DN} is a row matrix. In order to handle with this problem, a unique value for the size of Par_{CTop} is assigned as the half of upper limit value of division number as $Par_{DNU}/2$ due to generating the tubular lattice girder using its symmetrical semi-part. Although Par_{CTop} is generated in a unique size, the required data is easily picked from Par_{CTop}. The tubular lattice girder outcome from the designer module has also a responsibility in the preparation of input data for the ready computer program ANSYS. Thus, it is possible to compute both the structural responses for an external static load and check the strengths of tubular lattice girder members and accordingly objective functions.

```
Step1 Code f1, f2, f3 and assign the values of maxDV and minDV,
Step2 Assign GN=50, PS=50, subGN1=15, subGN2=30 and div_num=2, pareto_OF_ALL==[]
Step3 Generate Spop1, Spop2, Spop3 and Spop4 considering their sizes as PS1=PS2=PS3=PS4=PS/4
Step4 Compute f1, f2, f3
                              for i=1:GN
Step5 if size(pareto_OF_ALL)==0, re-generate Spop1, Spop2, Spop3 and Spop4 considering PS1=PS2=PS3=PS4=PS/4, end
Step6 Compute f1, f2, f3 and determine pareto_OF_ALL
Step7 if size(pareto_OF_ALL)~=0, go to Step5, end
Step8 Determine the bounds of pareto solutions max_pareto_OF1_ALL, max_pareto_OF2_ALL, and etc. using div_num,
Step9 Re-arrange PS1, PS2, PS3 and PS4 (PS= PS1 + PS2 + PS3 + PS4) considering Spop1 and etc. along with
max_pareto_OF1_ALL and etc. and correspondingly re-compute div_num.
Step10 Assign Genetic Parameter values as
         if       i< subGN1
var_crossover_prob=0.20, var_crossover_dist =15.00, var_mutation_prob =0.80, var_mutation_dist =2.00
         elseif  i> subGN1 && i< subGN2
var_crossover_prob=0.80, var_crossover_dist =2.00, var_mutation_prob =0.40, var_mutation_dist =10.00
         else
var_crossover_prob=0.80, var_crossover_dist =2.00, var_mutation_prob =0.01, var_mutation_dist =20.00
         end
Step11 Evaluate Spop1 and etc. considering the crossover and mutation genetic operators and their corresponding genetic
parameter values
Step 12 Correct Spop1 and etc. considering maxDV and minDV
                              end
```

Fig. 1 Pseudo Code for Computing Steps of PAGA

```
Form the cell topology of tubular lattice girder:
%only one of the symmetric parts
%(DivisionNumber must be even number)
  Horizantal_Length=10 (or 20)
  DivisionNumber=Chrom(1,1);
    for i4=1:(DivisionNumber/2)
  CellTopology(i4)= Chrom(1+i4);
    end
Assign the heights of tubular lattice girder:
  Vertical_Length1=Chrom(1, DivisionNumber/2 +2);
  Vertical_Length2=Chrom(1, DivisionNumber/2 +3);
Assign the elements properties of tubular lattice girder for later
use:
  prop_no(1)=1;    % for Chrom(1,DivisionNumber/2+4)
  prop_no(2)=2;    % for Chrom(1,DivisionNumber/2+5)
  prop_no(3)=3;    % for Chrom(1,DivisionNumber/2+6)
  prop_no(4)=4;    %  for Chrom(1,DivisionNumber/2+7)
Determine the nodal coordinates of tubular lattice girder:
  NodeCoor=[];
  CellNodeNo=[];
  IncHorizLeng=Horizantal_Length/DivisionNumber;
  IncVerLeng=(Vertical_Length2 -
  Vertical_Length1)/(DivisionNumber/2);
     %top level generation
  IncHor=0;
  IncVer=0;
     for i1=1:DivisionNumber+1
        if i1==1
  NodeCoor(i1,1)=0;
  NodeCoor(i1,2)=Vertical_Length1;
  IncHor=IncHor+IncHorizLeng;
  IncVer=IncVer+IncVerLeng;
        elseif i1==(DivisionNumber+1)
  NodeCoor(i1,1)=Horizantal_Length;
  NodeCoor(i1,2)=Vertical_Length1;
        elseif (i1 > 1) && (i1 < (DivisionNumber/2)+1)
  NodeCoor(i1,1)=IncHor;
  NodeCoor(i1,2)=Vertical_Length1+IncVer;
  IncHor=IncHor+IncHorizLeng;
  IncVer=IncVer+IncVerLeng;
        elseif i1 == ((DivisionNumber/2)+1)
  NodeCoor(i1,1)=IncHor;
  NodeCoor(i1,2)=Vertical_Length2;
  IncHor=IncHor+IncHorizLeng;
  IncVer=IncVer-IncVerLeng;
        else
  NodeCoor(i1,1)=IncHor;
  NodeCoor(i1,2)=(Vertical_Length1+IncVer);
  IncHor=IncHor+IncHorizLeng;
  IncVer=IncVer-IncVerLeng;
        end
     end
     %below level generation
  IncHor=0;
     for i2=(DivisionNumber+2:DivisionNumber*2+2)
        if i2==DivisionNumber+2
  NodeCoor(i2,1)=0;
  NodeCoor(i2,2)=0;
  IncHor=IncHor+IncHorizLeng;
        else
  NodeCoor(i2,1)=IncHor;
  NodeCoor(i2,2)=0;
  IncHor=IncHor+IncHorizLeng;
        end
     end
Determine the node number for each cell
  CellNodeNo(1,1)=1;
  CellNodeNo(1,2)=2;
  CellNodeNo(1,3)=DivisionNumber + 2;
  CellNodeNo(1,4)=DivisionNumber + 3;
     for i3=2:DivisionNumber
  CellNodeNo(i3,1)=CellNodeNo(i3-1,2);
  CellNodeNo(i3,2)=CellNodeNo(i3-1,2) + 1;
  CellNodeNo(i3,3)=CellNodeNo(i3-1,4);
  CellNodeNo(i3,4)=CellNodeNo(i3-1,4) + 1;
     end
Generate Elements Considering CellNodeNo, CellTopolgy,
NodeCoor and prop_no
      for i4=1:DivisionNumber
      if i4==1
   Generate first cell (symmetrically)
first_one(i4,CellTopology, NodeCoor, ...
CellNodeNo,ElemInc,DivInc,prop_no);
      elseif i4==2
   Generate second cell  (symmetrically)
middle_two_first(i4,CellTopology, NodeCoor, ...
CellNodeNo, ,prop_no);
      elseif i4 <= DivisionNumber
   Generate other cells  (symmetrically)
middle_two_later(i4,CellTopology,NodeCoor, ...
CellNodeNo,DivInc,prop_no);
      end
      end
```

Fig. 2 Pseudo Code for Automatically Generation of Lattice Girder

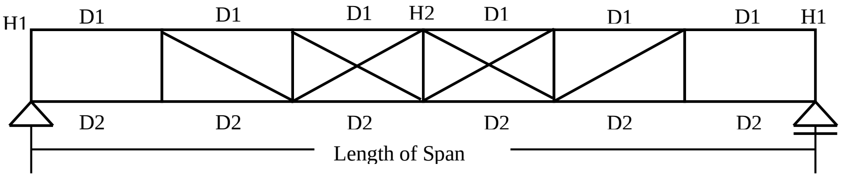


Fig. 3 A Lattice Girder Form Generated by Proposed Designer Module

4. Discussion of Findings

The proposed design approach integrated with the optimization tool, PAGA is utilized for the optimal design of tubular lattice girder. In this regard, this section is divided into two sub-sections in order to summary the results obtained by the application of PAGA into the proposed design examples. In the first part of this section, the efficiency of PAGA is evaluated with respect to three MOAs, NSGAII, EVMOGA and SMSEMOA. For this purpose, three benchmark design examples named I-beam, welded beam and spring designs and a benchmark mathematical function named ZDT3 are utilized for the evaluation of employed MOAs coded in MATLAB. In the second sub-section, the design of tubular lattice girders as the real-world design examples are also optimized by the employed MOAs due to being fallen the benchmark examples into a category of small scaled ones. Thus, it is possible to both re-examine the computing performance of PAGA and determine the governing design factors in the design of lattice girder. The successful algorithms are ranked in two levels and summarized in the separate Tables.

In order to provide an equal competition among the employed MOAs, the most appropriate operator parameter value sets, which manages the search mechanism of employed MOAs, have to be determined. For this purpose, a number of trials with different combinations of operator parameter values are separately performed for each of the employed MOAs. Then, the relatively better values of operator parameters are determined. Also, the values of two important parameters, evolutionary number and population size are taken as 50 and 50, respectively (see Tables 1 and 2).

Moreover, the relationship among them is assessed considering a statistical hypotheses test named “Kruskall Wallis” in a way of measuring the significance degree at a certain level ($P<0.05$). Therefore, the execution number is taken as 100 runs.

Table 1. Parameter Names and Values of Governing Operators Utilized by Employed MOAs

Employed MOAs and Their Genetic Operator Parameter Names		Genetic Operator Parameter Values
NSGAII		
options. Generations		50
options.PopulationSize		50
options.MutationFcn= {@mutationuniform.}		0.2
options.CrossoverFcn= {@crossoverheuristic.}		0.4
options.SelectionFcn={@selectiontournament.}		2
EVMOGA		
dd_ini	(crossover-related)	0.25
dd_fin	(cossover-related)	1
Pm	(mutation-related)	0.1
Sigma_Pm_ini	(mutation-related)	10
Sigma_Pm_fin	(mutation-related)	0.1
Nind_GA	(GA population size)	50
Nind_P	(P population size)	50
Nind_max_A	(A population size)	50
n_iterations	(generation number)	50
SMSEMOA		
defopts.nPop	(size of the population)	50
defopts.maxEval	(maximum number of evaluations)	
defopts.useOCD	(use OCD to detect convergence)	True
defopts.OCD_VarLimit	(variance limit of OCD)	1e-9
defopts.OCD_nPreGen	(number of prec. generations used in OCD)	1
defopts.var_crossover_prob	(variable crossover probability)	0.9
defopts.var_crossover_dist	(distribution index for crossover)	15
defopts.var_mutation_prob	(variable mutation probability)	1/(Number of Design Variables)
defopts.var_mutation_dist	(distribution index for mutation)	20
defopts.var_swap_prob	(variable swap probability)	0.5
defopts.DE_F	(difference weight for DE)	0.2+rand(1)
defopts.DE_CR	(crossover probability for differential evo)	0.9
defopts.DE_CombinedCR	(crossover of blocks)	True
defopts.useDE	(perform differential evo instead of SBX&PM)	True
defopts.refPoint	(refPoint for HV)	0
PAGA		
PS	(population size)	50
GN	(generation number)	50
CombGen1	(gen. num. for first gen. operator comb. application)	20
CombGen2	(gen. num. for second gen. operator comb. application)	30

Table 2. Design Details of The Tubular Lattice Girder For The Design Optimization Purpose

Cases				
	Lattice Girder 1	Lattice Girder 2	Lattice Girder 3	Lattice Girder 4
Material Properties and Geometrical Configuration Details				
Load Values	Loading 1: 70 kipf Loading 2: 20 kipf Loading 3: 30 kipf	Loading 1: 140 kipf Loading 2: 40 kipf Loading 3: 60 kipf	Loading 1: 150 kipf Loading 2: 50 kipf Loading 3: 75 kipf	Loading 1: 300 kipf Loading 2: 150 kipf Loading 3: 75 kipf
Spanning Length	393.70 in. (10 m)	393.70 in. (10 m)	787.40 in. (20 m)	787.40 in. (20 m)
Joint Displ. Limit	3.94 in (100 mm)			
Max. Yielding	36 ksi (248.211 N/mm^2)			
Elast. Mod. Value	29732 ksi (205 kN/ mm^2)			
Design Parameters (Size)				
Par_{ND}	4	4	4	4
Par_{UDV}	37	37	37	37
Par_{LDV}	1	1	1	1
Design Parameters (Topology)				
Par_{UDN}	14	20	40	50
Par_{LDN}	6	6	10	10
Design Parameters (Shape)				
Par_{UH2}	35.433 in (0.9 m)	47.244 in (1.2 m)	59.055 in (1.5 m)	78.740 in (2.0 m)
Par_{LH2}	23.622 in (0.6 m)	23.622 in (0.6 m)	31.496 in (0.8 m)	31.496 in (0.8 m)
Par_{UH1}	23.622 in (0.6 m)	23.622 in (0.6 m)	31.496 in (0.8 m)	31.496 in (0.8 m)
Par_{LH1}	3.937 in (0.1 m)	3.937 in (0.1 m)	7.874 in (0.2 m)	7.874 in (0.2 m)

4.1. Preliminary Results Obtained from Optimization of Benchmark Function and Design Examples with Design Variables of Continuous and Mixed Types

The benchmark function named ZDT3 (Zitzler and *et al.* [33]) is firstly optimized (see the Matlab scripts in Reference Wagner and Kretzschmar [19-20]). This benchmark function ZDT3, which has 30 continuous-type decision variables within a certain interval [0,1] is a mathematical function with a non-contiguous form. Its true pareto front with the optimal solutions is depicted in Fig. 4. Then, three benchmark designs named I beam, welded beam and spring are optimized (see their mathematical expressions in References Yang and *et al.* [39], Deb and Srinivasan [40], He and Wu [41]). While the designs of I and welded beams are represented by use of continues type-decision variables, both discrete and continues type-decision variable is utilized to represent the design of spring (see Fig. 5). Their true pareto fronts along with the optimal solutions are depicted in Figures (6-8). A statistical summarization of quality indicators obtained in the end of 100 runs are tabulated for each of benchmark test examples in Tables (3-6). The computing performances of employed MOAs are summarized in Table 7. Considering Figures (6-8) and Tables (3-7), the preliminary results are listed as:

- There does not exist any relation among the employed MOAs taking into account of the lower probability ratio (Prob>Chi-sq) as 2.42e-77 (see also the note as “3 groups have mean ranks significantly different from PAGA” in Fig. 9).
- The computing performance of PAGA is ranked in a second level according to the measuring metrics obtained for the mathematical function. The capacity, density, convergence-diversity and covering features of PAGA show relatively lower performance than the other employed MOAs (see the rank of MOAs according to the measuring metrics in Table 7 and the true pareto front in Fig. 4)
- PAGA achieves to obtain more accurate approximation with a relatively better distribution for all design examples (see the rank of MOAs according to R2 in Table 7).
- The pure convergence degree of PAGA is ranked in the first place for the design example 1 and 2 but the second place for design example 3 (see the rank of MOAs according to Epsilon in Table 7). It is seen that the true pareto front of design example 3 is obtained as a non-contiguous form in Fig. 8.
- The density degree of pareto solutions obtained by PAGA is relatively higher for design example 1 and 2 but lower for design example 3 (see the rank of MOAs according to Average Distance in Table 7).
- Although PAGA shows a superior performance in obtaining the highest number of pareto solutions, the covering degree of these pareto solutions becomes relatively lower with respect to the other ones obtained by the other employed MOAs (see the rank of MOAs according to the Number of Pareto solutions and Spread in Table 7).

It is easily seen that the decrease in the pure convergence of pareto solutions is arisen from the fast-approximating feature of PAGA due to using the valuable genetic material exploited by the proposed evolutionary search for the exploration of promising candidate solutions. This feature of PAGA leads to an increase in the capacity performance while a decrease in the covering performance.

Table 3. Statistical Values of Pareto Solutions and Elapsed Time (Benchmark Mathematical Function ZDT3)

Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig” .	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	41.872	2.008	3.401		0.363	0.003	0.252		0.213	0.029	0.085		0.407	0.015	0.212		17	5	10
SMSEMOA	9.687	0.080	0.303	2.42E-77	0.040	0.002	0.016	3.79E-77	0.215	0.035	0.059	3.09E-60	0.571	0.106	0.352	1.15E-67	45	18	27
NSGAII	8.323	1.472	1.699		0.187	0.004	0.165		0.197	0.017	0.024		0.918	0.479	0.667		158	80	123
PAGA	13.923	0.288	0.778		0.106	0.004	0.062		0.180	0.033	0.049		0.895	0.314	0.567		80	25	46

Table 4. Statistical Values of Quality Measuring Indicators (Benchmark Design Example 1*:* I-beam Design)

Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	131.492	7.090	59.605		0.110	0.000	0.025		0.193	0.003	0.100		0.771	0.005	0.336		20	4	11
SMSEMOA	5.729	0.004	3.049	1.72E-57	0.079	0.016	0.045	1.40E-62	0.038	0.005	0.015	6.92E-76	1.000	0.175	0.917	2.72E-51	51	50	51
NSGAII	4.692	0.364	2.324		0.083	0.002	0.016		0.060	0.013	0.031		0.998	0.093	0.473		21	21	21
PAGA	5.026	0.000	1.069		0.005	0.000	0.001		0.007	0.002	0.004		1.738	0.589	1.114		994	754	901

Table 5. Statistical Values of Quality Measuring Indicators (Benchmark Design Example 2: Welded Beam Design)

Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	17.447	1.053	6.026		0.032	0.000	0.006		0.416	0.000	0.139		0.797	0.000	0.237		17	3	6
SMSEMOA	0.346	0.001	0.050	1.93E-65	0.001	0.000	0.000	6.32E-62	0.128	0.019	0.036	6.63E-22	1.164	0.262	0.534	1.24E-21	51	50	50
NSGAII	985.499	0.423	70.295		0.043	0.000	0.003		0.202	0.000	0.041		1.302	0.000	0.651		21	1	17
PAGA	0.242	0.000	0.028		0.000	0.000	0.000		0.087	0.005	0.014		1.754	0.283	0.552		233	104	163

Table 6. Statistical Values of Quality Measuring Indicators (Benchmark Design Example 3: Spring Design)

Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	6.43E+10	8.05E+03	2.17E+09		0.510	0.000	0.235		0.600	0.000	0.049		0.998	0.000	0.094		15	1	3
SMSEMOA	5.40E+04	1.11E+01	1.11E+03	2.33E-72	0.080	0.000	0.002	1.08E-54	0.120	0.004	0.033	5.98E-26	1.000	0.232	0.992	1.44E-59	51	48	50
NSGAII	3.41E+06	1.31E+04	1.10E+06		0.002	0.000	0.001		0.232	0.000	0.017		1.000	0.000	0.858		16	1	4
PAGA	3.19E+05	6.93E+03	1.63E+04		0.000	0.000	0.000		0.037	0.000	0.036		1.000	0.895	0.898		243	98	144

Table 7. A summary for Ranking The Employed MOAs Considering Benchmark Tests

Qual. Ind.	Rank	Mathematical Function	Design Ex. 1	Design Ex. 2	Design Ex. 3
Epsilon	Rank 1	SMSEMOA	PAGA	PAGA	SMSEMOA
	Rank 2	PAGA	NSGAII	SMSEMOA	PAGA
R2	Rank 1	SMSEMOA	PAGA	PAGA	PAGA
	Rank 2	PAGA	NSGAII	NSGAII	NSGAII
Average Distance	Rank 1	NSGAII	PAGA	PAGA	NSGAII
	Rank 2	PAGA	NSGAII	SMSEMOA	SMSEMOA
Spread	Rank 1	EVMOGA	EVMOGA	EVMOGA	EVMOGA
	Rank 2	SMSEMOA	NSGAII	SMSEMOA	NSGAII
Number of Pareto Sol.	Rank 1	NSGAII	PAGA	PAGA	PAGA
	Rank 2	PAGA	SMSEMOA	SMSEMOA	SMSEMOA

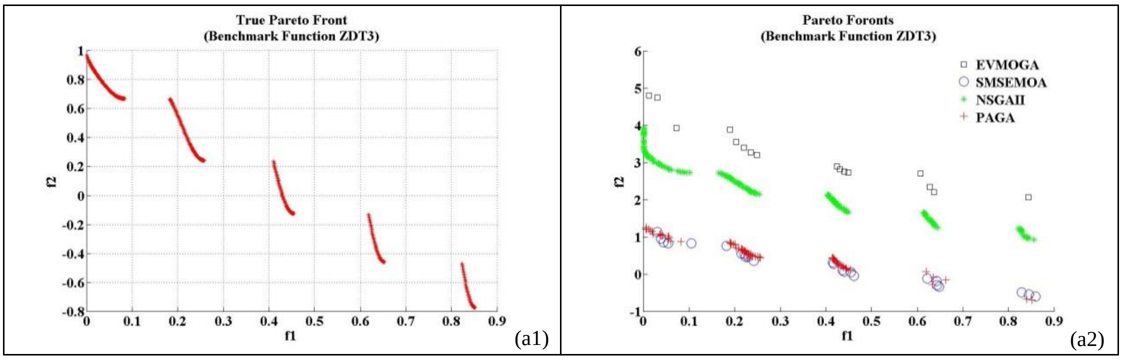


Fig. 4 True Pareto Front (a1) and Current Pareto Fronts Obtained by Use of Employed MOAs (a2) (Benchmark Function ZDT3)

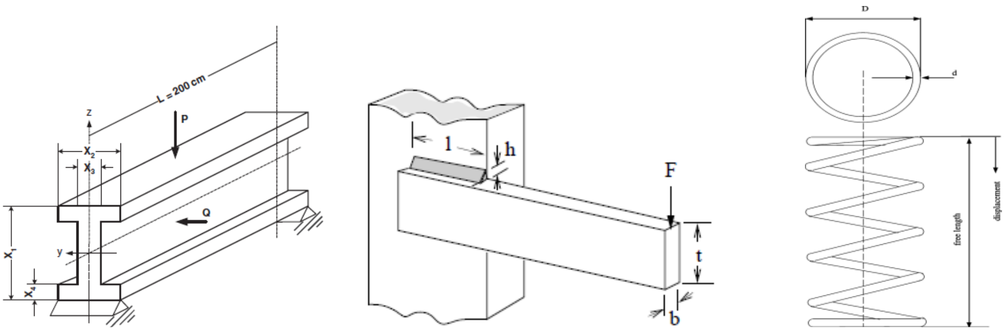


Fig. 5 Benchmark Design Examples I-Beam (Yang *et. al.* [39]), Welded Beam (Deb & Srinivasan [40]) and Spring Design (He and e*t al.* [41])

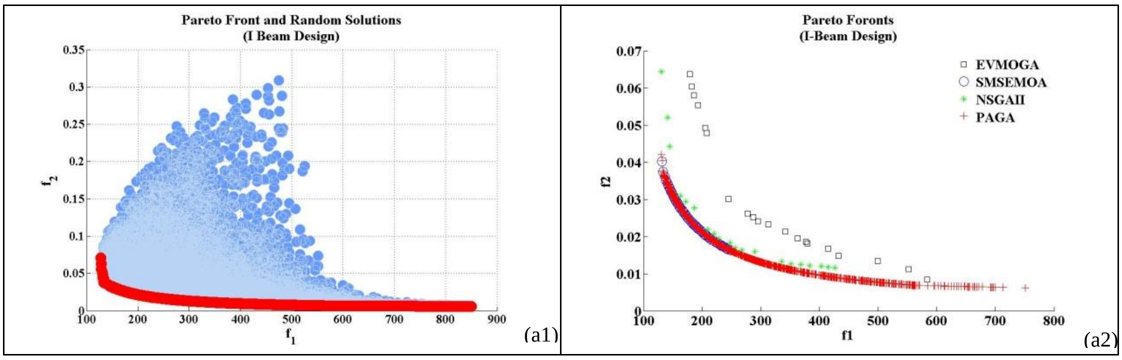


Fig. 6 True Pareto Front along with Random Solutions (a1) and Current Pareto Fronts Obtained by Use of Employed MOAs (a2) (Benchmark I-beam Design)

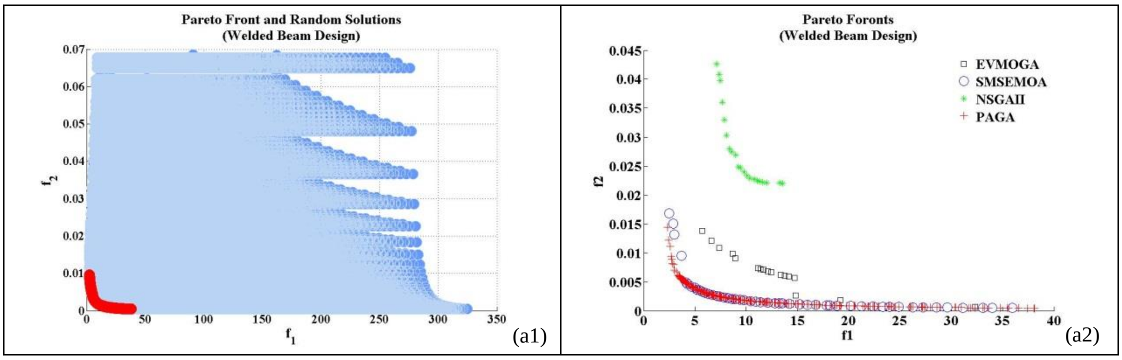


Fig. 7 True Pareto Front along with Random Solutions (a1) and Current Pareto Fronts Obtained by Use of Employed MOAs (a2)

(Benchmark Welded Beam Design)

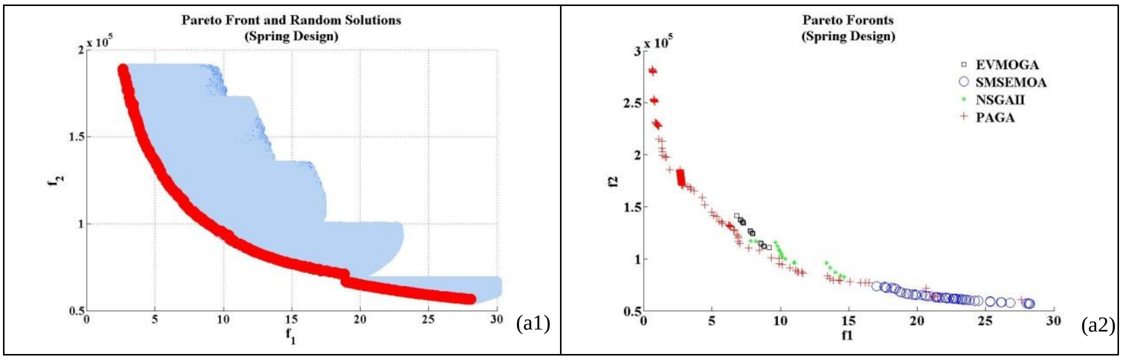


Fig. 8 True Pareto Front along with Random Solutions (a1) and Current Pareto Fronts Obtained by Use of Employed MOAs (a2)

(Benchmark Spring Design

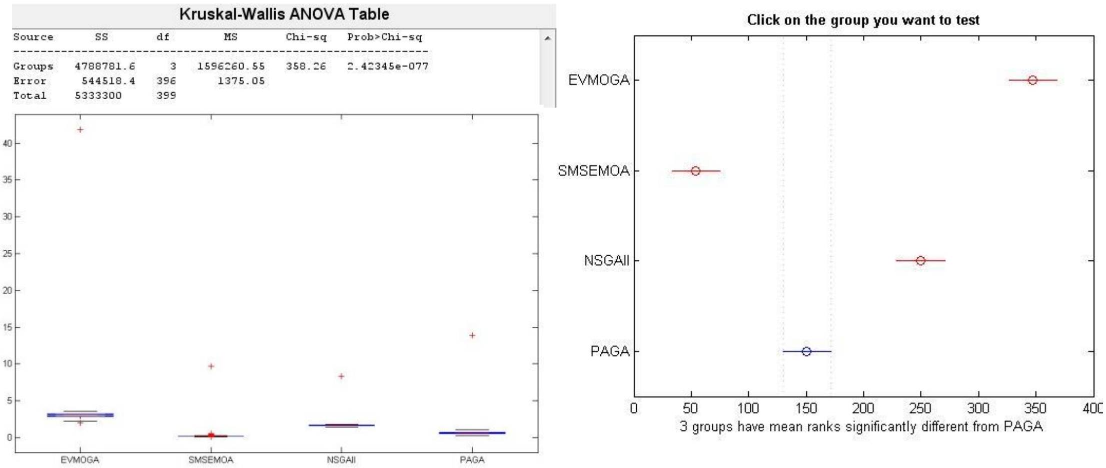


Fig. 9 Sketches Regarded to Statistical Analysing Results Outcome from MATLAB

4.2. Evaluation of Optimal Results Obtained from Design Optimization of Tubular Lattice Girders

This sub-section has two objectivities: *i)* evaluating the computing performance of PAGA considering the optimal design solutions of tubular lattice girders as a real-world design example, *ii)* determining the design factors which have the big responsibilities in the structural behavior of tubular lattice girders. Thus, it is also possible to examine the load-carrying capacity, serviceability and constructional cost of tubular lattice girders depending on its framing configurations and loading conditions. For this purpose, the designs of four tubular lattice girders with different spanning length and loading conditions are optimized (see Table 2). The extended formulations of design constraints in Eq. 5 are given in References (Talaslioglu [11 and 42]). The statistical data regarded to the proposed quality indicator values obtained by the employed MOAs are tabulated in Table (8-11). The true pareto fronts along with random solutions are displayed in Fig. 10. A summary of data tabulated in Table (8-11) is presented in Table 12. Thus, the preliminary results are summarized in the following part as:

- PAGA shows a success in obtaining more accurate approximation with a relatively better distribution for the designs of four tubular lattice girders (see the rank of MOAs according to R2 in Table 12 and see the true pareto fronts in Fig. 10).
- PAGA achieves to obtain the highest pure convergence degree in the designs of four tubular lattice girders (see the rank of MOAs according to Epsilon in Table 12).
- The density degree of pareto solutions obtained by PAGA is relatively lower for the designs of four tubular lattice girders except for the design of tubular lattice girder 3 (see the rank of MOAs according to Average Distance in Table 12).
- While PAGA shows a superior performance in obtaining the highest number of pareto solutions, the covering degree of these pareto solutions becomes relatively lower with respect to the other ones (see the rank of MOAs according to the Number of Pareto solutions and Spread in Table 12).
- The extreme optimal designs are tabulated in Table 13 including their framing configurations. Considering the entire weight values (1345.229 lb, 17565.640 lb) and maximum member forces (169.242 lb, 3829.295 lb) and joint displacements (1.267 in, 2.227 in) corresponding to tubular lattice girder 1 and 4, it is obvious that an increase in the weight of tubular lattice girder correspondingly leads to an elevation in both its load-carrying capacity and joint displacement value (see Table 13). Although this claim seems to be an expected result, its invalidation is easily investigated by taking into account of the increased entire weight values from 3301.855 lb to 4360.518 lb a and the decreased member force values from 1881.772 lb to 514.984 lb obtained for tubular lattice girders 1 and 2. In fact, there is also a similar dilemma between entire weight and joint displacement. In other words, it is expected that an increase in the entire weight of tubular lattice girder causes to an increase in the joint displacement value. But, its invalidation is approved considering the increased entire weight value from 1345.229 lb to 2181.964 lb and the decreased joint displacement value from 1.267 in to 0.895 (see the tubular lattice girders 1 and 2 in Table 13). The reason behind this dilemma is arisen from mainly the stability-related structural behavior. In other words, the inclusion of a slender member into current framing configuration of tubular lattice girder causes to inevitably a decrease in the joint displacements of tubular lattice girder although the entire weight of tubular lattice girder is increased.
- The other important result is concerned with the use of diagonal members in the construction of tubular lattice girders. Considering the sketches regarded to the framing configurations of tubular lattice girder in Table 13, a double bracing of diagonal members leads to an elevation in the load-carrying capacity even if under the severe loading conditions and expanded spanning lengths. Particularly, if the double bracing is increased along with an elevation of the first and last heights of tubular lattice girder, its load-carrying capacity is exponentially increased. This claim is easily confirmed considering the sketches corresponding to the increased member forces as 885.094 lb, 1229.182 lb, 1850.709 lb and 3829.295 lb for the tubular lattice girder 1,3 and 4 (see Table 13). Nevertheless, it is noted that there is a possibility of including a slender member into the current framing configuration of tubular lattice girder. At this point, the designer module gains a big importance in order to prevent the inclusion of slender members through its ability of continually differentiating the framing configurations of lattice girder. Therefore, the designer module increases the flexibility of PAGA. Thus, PAGA utilizes these promising optimal designs in order to generate new framing configurations of tubular lattice girder.

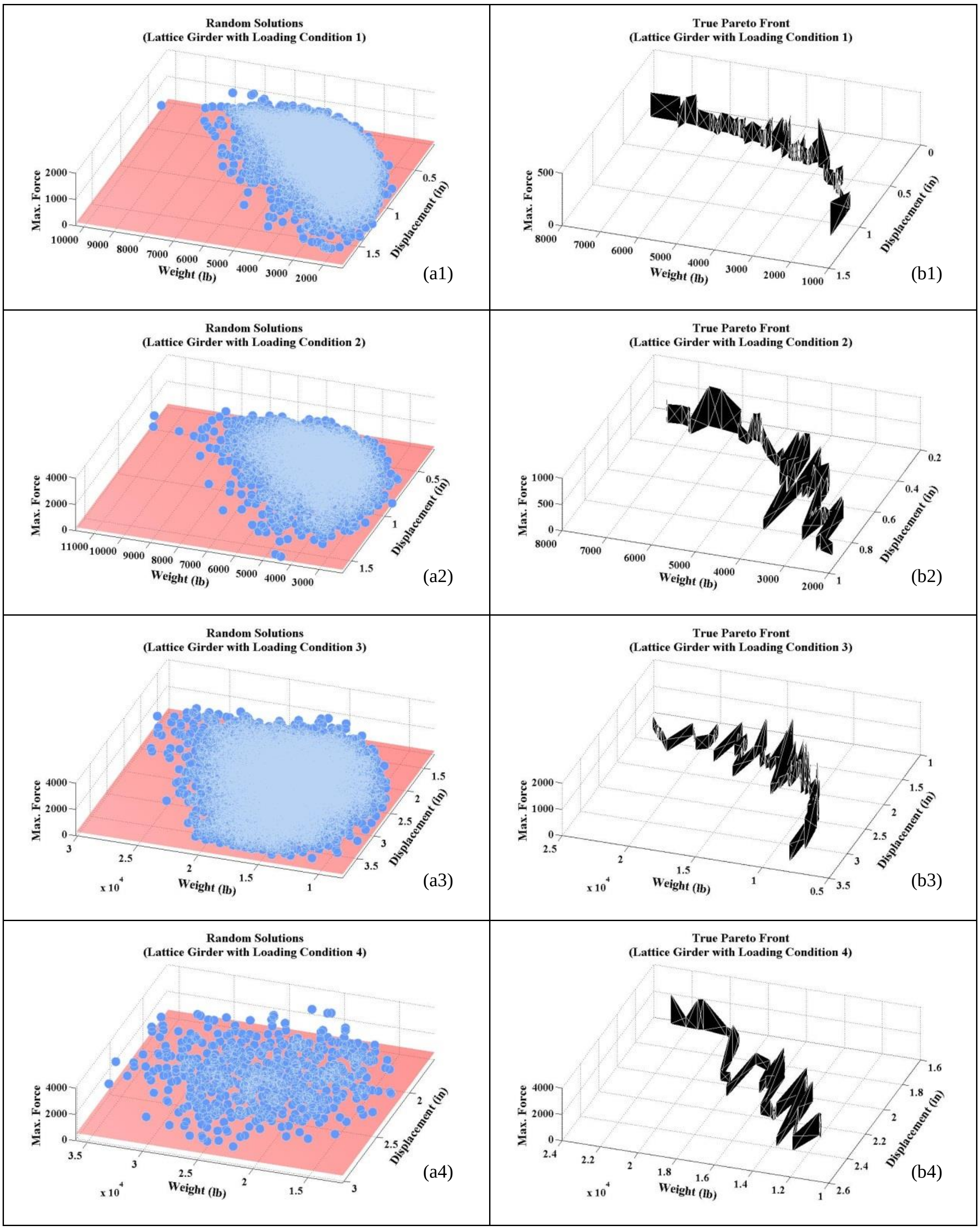


Fig. 10 True Pareto Front and Random Solutions (Tubular Lattice Girder 1 (a1,b1), Tubular Lattice Girder 2 (a2,b2), Tubular Lattice Girder 3 (a3,b3) and Tubular Lattice Girder 4 (a4,b4))

Table 8. Statistical Values of Quality Measuring Indicators (Lattice Girder 1)

Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	1078.469	35.669	699.984		0.010	0.002	0.006		0.187	0.046	0.126		0.982	0.222	0.650		14	5	12
SMSEMOA	1072.896	38.978	636.824	0.00014691	0.010	0.001	0.006	0.00072398	0.185	0.046	0.122	0.00313576	0.985	0.232	0.645	0.01802209	18	11	16
NSGAII	1079.239	62.646	732.861		0.010	0.002	0.006		0.187	0.051	0.126		0.989	0.221	0.675		31	16	24
PAGA	1037.890	100.228	530.509		0.010	0.000	0.005		0.186	0.049	0.124		0.990	0.205	0.579		37	26	32

Table 9. Statistical Values of Quality Measuring Indicators (Lattice Girder 2)

Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	1305.984	157.895	850.258		0.010	0.001	0.007		0.264	0.039	0.174		0.997	0.704	0.873		12	6	9
SMSEMOA	1297.372	138.568	816.020	2.14E-05	0.010	0.001	0.007	0.0045	0.267	0.046	0.172	0.000506	0.998	0.701	0.858	0.015631	18	11	14
NSGAII	1305.783	138.647	862.117		0.010	0.001	0.006		0.267	0.039	0.175		0.999	0.703	0.863		15	7	12
PAGA	1297.845	5.301	567.545		0.010	0.000	0.005		0.265	0.039	0.188		0.997	0.706	0.864		29	19	21

Table 10. Statistical Values of Quality Measuring Indicators (Lattice Girder 3)

Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	3548.864	237.106	2212.249		0.019	0.002	0.011		0.259	0.040	0.162		0.996	0.708	0.859		9	5	8
SMSEMOA	3546.729	451.603	2209.754	4.08E-03	0.019	0.001	0.013	0.003554	0.258	0.048	0.179	0.001337	0.997	0.702	0.857	0.034427	14	8	10
NSGAII	3534.038	240.117	2266.605		0.019	0.001	0.012		0.257	0.043	0.175		0.999	0.702	0.858		17	9	11
PAGA	3535.471	136.815	1702.061		0.019	0.002	0.011		0.257	0.040	0.144		0.998	0.703	0.856		24	14	18

Table 11. Statistical Values of Quality Measuring Indicators (Lattice Girder 4)

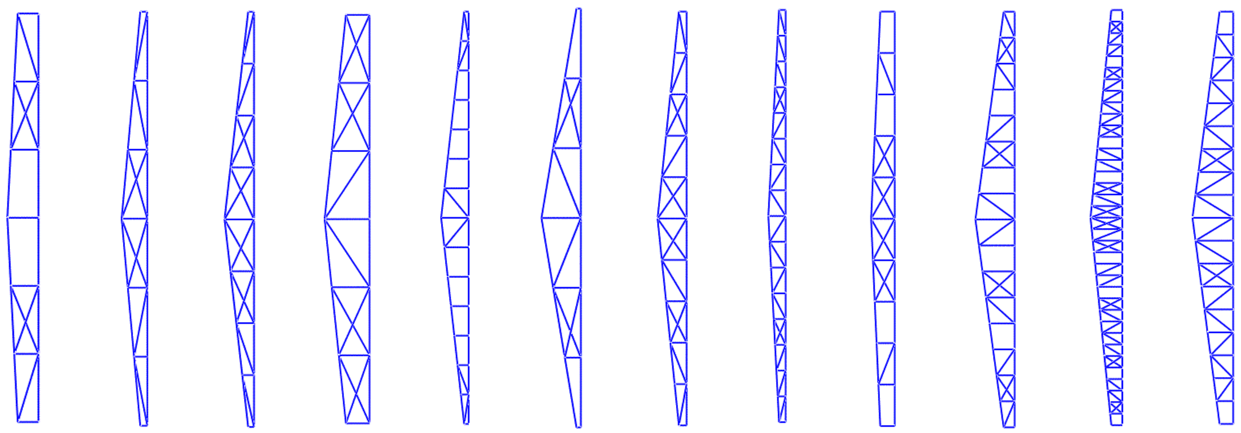
Employed MOAs	Epsilon Indicator				R2 Indicator				Average Distance				Spread				Number of Pareto Sol.		
	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.	Stat. Sig.	Max. Val.	Min. Val.	Aver. Val.
EVMOGA	16404.696	2685.283	10888.279		0.117	0.013	0.079		0.329	0.038	0.210		0.998	0.705	0.875		8	4	6
SMSEMOA	16413.865	982.807	10636.180	4.48E-06	0.117	0.004	0.069	0.00331346	0.332	0.027	0.190	0.00018697	0.999	0.701	0.849	0.00714808	12	6	9
NSGAII	16473.704	1230.183	10395.459		0.117	0.013	0.081		0.330	0.023	0.216		1.000	0.702	0.847		9	4	7
PAGA	16367.764	758.003	7518.177		0.114	0.004	0.064		0.333	0.016	0.192		0.996	0.703	0.854		17	11	14

Table 12. A summary for Ranking The Employed MOAs Considering Tubular Lattice Girder Designs

Qual. Ind.	Rank	Lattice Girder 1	Lattice Girder 2	Lattice Girder 3	Lattice Girder 4
Epsilon	Rank 1	PAGA	PAGA	PAGA	PAGA
	Rank 2	SMSEMOA	SMSEMOA	SMSEMOA	NSGAII
R2	Rank 1	PAGA	PAGA	PAGA	PAGA
	Rank 2	SMSEMOA	SMSEMOA	NSGAII	SMSEMOA
Average Distance	Rank 1	SMSEMOA	SMSEMOA	PAGA	SMSEMOA
	Rank 2	PAGA	PAGA	NSGAII	PAGA
Spread	Rank 1	PAGA	SMSEMOA	PAGA	NSGAII
	Rank 2	SMSEMOA	NSGAII	SMSEMOA	SMSEMOA
Number of Par. Sol.	Rank 1	PAGA	PAGA	PAGA	PAGA
	Rank 2	NSGAII	SMSEMOA	NSGAII	SMSEMOA

Table 13. Design Variables (Size, Shape and Topology) Along with Objective Functions Obtained by Use of Proposed Tubular Lattice Girders

	Lattice Girder 1			Lattice Girder 2			Lattice Girder 3			Lattice Girder 4		
	Des1 (MW)	Des2 (MJD)	Des3 (MEF)	Des1 (MW)	Des2 (MJD)	Des3 (MEF)	Des1 (MW)	Des2 (MJD)	Des3 (MEF)	Des1 (MW)	Des2 (MJD)	Des3 (MEF)
Par_{DN}	6	6	8	6	14	6	10	16	10	16	36	18
Par_{H2}	19.752	6.673	5.331	22.780	4.324	3.992	14.323	12.234	27.457	21.055	20.118	25.845
Par_{H1}	29.565	24.233	26.889	43.298	26.004	36.030	54.123	32.324	44.808	74.913	58.106	77.284
D1	PIPEST3-1/2 (3.44 in^2)	PIPST8 (7.85 in^2)	PIPEEST2 (2.51 in^2)	PIPEST5 (5.72 in^2)	PIPEEST6 (14.7 in^2)	PIPEEST2 (2.51 in^2)	PIPST10 (11.1 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST10 (15.0 in^2)	PIPEEST8 (20.0 in^2)	PIPEEST6 (14.7 in^2)
D2	PIPST5 (4.03 in^2)	PIPEEST2 (2.51 in^2)	PIPST3 (2.08 in^2)	PIPST5 (4.03 in^2)	PIPEEST5 (10.7 in^2)	PIPEST4 (4.14 in^2)	PIPST10 (11.1 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST6 (14.7 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST2 (2.51 in^2)
D3	PIPEEST2-1/2 (3.81 in^2)	PIPST5 (4.03 in^2)	PIPST4 (2.97 in^2)	PIPEEST2-1/2 (3.81 in^2)	PIPST10 (11.1 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST6 (14.7 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST8 (20.0 in^2)	PIPEST8 (11.9 in^2)	PIPEEST2 (2.51 in^2)	PIPST12 (13.6 in^2)
D4	PIPST3 (2.08 in^2)	PIPEST6 (7.88 in^2)	PIPST5 (4.03 in^2)	PIPEST3-1/2 (3.44 in^2)	PIPEEST5 (10.7 in^2)	PIPEEST2 (2.51 in^2)	PIPST4 (2.97 in^2)	PIPEEST2 (2.51 in^2)	PIPEEST10 (15.0 in^2)	PIPEST3-1/2 (3.44 in^2)	PIPEEST6 (14.7 in^2)	PIPEEST10 (15.0 in^2)
Entire Weight (lb)	1345.229	2651.333	3301.855	2181.964	4360.518	5608.938	8009.778	15905.602	14148.830	11895.340	24631.810	17565.640
Joint Disp. (in)	1.267	1.791	1.202	0.895	1.640	0.932	3.075	3.937	2.613	2.289	3.023	2.227
Member Force (lb)	169.242	885.094	1881.772	245.068	514.984	3118.521	1229.182	1531.490	3628.809	1850.709	1861.192	3829.295



5. Conclusions

This study proposes the design optimization of tubular lattice girders. For this purpose, the tubular lattice girders with different loading conditions and spanning lengths are utilized. As an optimization tool, a multi-objective optimization methodology named pareto archived genetic algorithm (PAGA) is employed to execute the optimization-related computing procedures. The exploring capacity of PAGA is improved utilizing a designer module for the automatically generation of tubular lattice girder. Furthermore, the computing performance of PAGA is also evaluated considering several MOAs named NSGAII, EVMOGA and SMSEMOA. The preliminary results are summarized as:

- PAGA achieves to obtain relatively better convergence-diversity, pure convergence, density and capacity degrees of current pareto fronts for the small-scaled benchmark applications.
- PAGA has a capability of simultaneously handling the design variables with continuous, discrete and integer.
- PAGA is employed to optimize the tubular lattice girder design problem as large-scaled real-world design problem. It is shown that its success is increased obtaining better convergence-diversity, pure convergence, capacity and covering degrees of current pareto fronts. These results imply that PAGA is a successful optimization tool with an ability of both exploring an increased pareto optimal solutions and accurately approximating to a pareto front with a relatively better distribution.
- It is shown that the inclusion of designer module into the proposed optimal design approach increases the flexibility of PAGA, thereby preventing the stability problem.
- A diagonal lattice girder with increased double braced members increases the load-carrying capacity. However, it has to be noted that a possibility of including a slender member into the current framing configuration of tubular lattice girder causes to a stability loss. In this regard, the proposed designer module has a big importance for the design of tubular lattice girder due to its ability of exploring the stable frame configurations. Thus, it is possible to use these frame configurations in order to generate a tubular lattice girder with higher stability.
- It is emphasized that the multi-objective optimization procedure instead of the single-objective has to be utilized in order to obtain a tubular lattice girder with a higher load-carrying, serviceability capacities and a lower constructional cost at the same time.

In further research, the new design constraints regarded to the welding strengths will be employed along with the current ones. Also, PAGA will be improved thereby enriching its exploiting feature. For this purpose, a self-adaptive mechanism is developed in conjunction with new implementations in its current recombination procedure.

Acknowledgement

Author thanks the reviewers for improving the content of this manuscript. Furthermore, Author also thanks Dr. Tobias Wagner for both his support for highlighting the details about the use of quality indicators named epsilon and R, and his effort for coding SMSEMOA and R2-EMOA in Matlab.

References

[1] Nageim H.K. and Macginley T.J., Steel structures, practical design studies, 3th Edition, EF Spon Press, 2005.

[2] Talaslioglu T., "Optimization of geometrically nonlinear lattice girders. Part I: Considering member strengths", Journal of Civil Engineering and Management, 21(4), 423-443, 2015

[3] Hasancebi O., Çarbaş S., Doğan E., Erdal F. and Saka M.P., "Performance evaluation of meta-heuristic search techniques in the optimum design of real size pin jointed structures", Computers and Structures, 87(5-6), 284-302, 2009.

[4] Saka M.P. "Optimum design of steel frames using stochastic search techniques based on natural phenomena: a review", In Proceedings of Civil engineering computations: tools and techniques, Saxe-Coburgh Publications, U.K., 2007.

[5] Seyedpoor S.M., Gholizadeh S. and Talebian S.R., "An efficient structural optimization algorithm using a hybrid version of particle swarm optimization with simultaneous perturbation stochastic approximation", Civil Engineering and Environmental Systems, 27(4),.295-313, 2010.

[6] Cagnina L.C., Esquivel S.C. and Coello C.A.C., "Solving constrained optimization problems with a hybrid particle swarm optimization algorithm", Engineering Optimization. 43(8), 843-866, 2011.

[7] Chen T.Y. and Huang J.H., "An efficient and practical approach to obtain a better optimum solution for structural optimization", Engineering Optimization. 45(8), 1005-1026, 2013.

[8] Lu Y.C., Jan J.C., Hung S.L. and Hung G.H., "Enhancing particle swarm optimization algorithm using two new strategies for optimizing design of truss structures", Engineering Optimization, 45(10), 1251-1271, 2013.

[9] Torii A.J., Lopez R.H. and Biondini F., "An approach to reliability-based shape and topology optimization of truss structures", Engineering Optimization, 44(1), 37-53, 2012.

[10] Beer F, Johnston ER, DeWolf J and Mazurek D "Mechanics of material", 6th Edition, McGraw-Hill 2011.

[11] Talaslioglu T., "Multiobjective size and topolgy optimization of dome structures", Structural Engineering and Mechanics", 43(6), 795-822, 2012a.

[12] Salajegheh E., Mashayekhi M., Khatibinia M. and Kaykha M., "Optimum Shape Design of Space Structures by Genetic Algorithm", International Journal of Space Structures, 24, 45-57, 2009.

[13] Srinivas N. and Deb K., "Multi-objective optimization using non-dominated sorting in genetic algorithms", Evolutionary Computation, 2(3), 221-248, 1995.

[14] Zitzler E., Thiele L., Laumanns M., Fonseca C. and Grunert da Fonseca V., "Performance Assessment of Multiobjective Optimizers:An Analysis and Review", IEEE Transactions on Evolutionary Computation, 7(2), 117-132, 2003.

[15] Wu J. and Azarm S., "Metrics for quality assessment of a multiobjective design optimization solution set," J. Mech. Design, 123, 1, 18–25, 2001.

[16] Deb K., Pratap A., Agarwal S. and Meyarivan T., "A fast and elitist multiobjective genetic algorithm: NSGA-II," IEEE Trans. Evol. Comput., 6(2), 182–197, 2002.

[17] Veldhuizen D.A. and Lamont G.B., "Multiobjective evolutionary algorithm test suites," in Proc. ACM Symp. Appl. Comput., 351–357, 1999.

[18] Hansen M.P. and Jaszkiewicz A., "Evaluating the quality of approximations of the non-dominated set," Technical University of Denmark, Tech. Rep. IMM-REP-1998-7, 1998.

[19] http://inriadortmund.gforge.inria.fr/r2emoa/

[20] Wagner T. and Kretzschmar F.S., "Metric Selection Evolutionary Multi-objective Algorithm (SMSEMOA)" [online]. Available from: http://ls11-www.cs.uni-dortmund.de/rudolph/hypervolume/start

[21] Matlab, [online], Available from: http://www.mathworks.com/ help/gads/gamultiobj.html.

[22] Talaslioglu T., "Optimal dome design considering the member-related design constraints", accepted for the publication in the journal Frontiers of Structural and Civil Engineering, Springer, 2018.

[23] Yang X.S., "Engineering Optimization: An Introduction with Meta-heuristic Applications", John Wiley and Sons, 2010.

[24] Reed P.M., Hadka D., Herman J.D., Kasprzyk J.R. and Kollat J.B., "Evolutionary multi-objective optimization in water resources: The past, present, and future", Advances in Water Resources, 51, 438–456,2013.

[25] Metaxiotis K. and Liagkouras K., "Multiobjective Evolutionary Algorithms for Portfolio Management: A comprehensive literature review", Expert Systems with Applications, 39, 11685–11698, 2012.

[26] Richardson J.N., Nordenson G., Laberenne R., Coelho R.F. and Adriaenssens S., "Flexible optimum design of a bracing system for façade design using multi-objective genetic algorithms", Automation in Construction, 32, 80-87, 2013.

[27] Hwang C.L. and Masud "ASM Multiple objective decision making-methods and applications: A state-of-the-art survey", Lectures Notes in Economics and Mathematical Systems, 164, 1979.

[28] Miettinen K., "Nonlinear Multiobjective Optimization", Academic Kluwer Press, 1999.

[29] Charnes A. and Cooper W.W., "Goal programming and multiple objective optimization", European Journal of Operational Research, 1(1), 39-45, 1977.

[30] Deb K., Agrawal S., Pratap A. and Meyarivan T., "A fast and elitist multi-objective genetic algorithm: NSGA-II", IEEE Transactions on Evolutionary Computation, 6(2), 156-198, 2002.

[31] Zitzler E., Laumanns M. and Thiele L., "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization", In Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems, International Center for Numerical Methods in Engineering (CIMNE), 2002.

[32] Talaslioglu T., "Multi-objective Design Optimization of Grillage Systems according to LRFD-AISC", Advances in Civil Engineering, doi:10.1155/2011/932871, 2011.

[33] Zitzler E., Deb K. and Thiele L., "Comparison of Multi-objective Evolutionary Algorithms: Empirical Results", Evolutionary Computation, 8(2), 173-195, 2000.

[34] Huband S., Hingston P., While L. and Barone L., "An evolution Strategy with Probabilistic Mutation for Multi-objective Optimization", In Congress on Evolutionary Computation (CEC 2003), IEEE Press, 2003.

[35] Talaslioglu T., "Multi-objective Design Optimization of Geometrically Nonlinear Truss Structures", Kuwait Journal of Science and Engineering (Issue 1B)", 39(2), 47-77, 2012b.

[36] Martínez M., Herrero J.M., Sanchis J., Blasco X. and García N.S., "Applied Pareto Multi-objective Optimization by Stochastic Solvers", Engineering Applications of Artificial Intelligence, 22, 455-465, 2009.

[37] Beume N., Naujoks B. and Emerich M., "SMS-EMOA: Multiobjective selection based on dominated hyper-volume", European Journal of Operational Research, 181, 1653-1669, 2007.

[38] Beume N., Naujoks B. and Rudolph G., "SMS-EMOA-Effektive evolutionäre Mehrzieloptimierung (SMS-EMOA-Effective Evolutionary Multiobjective Optimization). Automatisierungstechnik, 56(7), 357-364, 2008.

[39] Yang B.S., Yeun Y.S. and Ruy W.S., "Managing approximation models in multi-objective optimization", Structural and Multidisciplinary Optimization, 24, 141-156, 2002.

[40] Deb K. and Srinivasan A., "Innovization: Innovative Design Principles Through Optimization", Kanpur Genetic Algorithms, [online]. Available from: http://www.iitk.ac.in/kangal/pub.htm KanGAL Report Number 2005007, 2005.

[41] He S., Prempain E. and Wu Q.H., "An Improved Particle Swarm Optimizer for Mechanical Design Optimization Problems", Engineering Optimization, 36(5), 585-605, 2004.

[42] Talaslioglu T., "Weight Minimization of Tubular Dome Structures by a Particle Swarm Methodology", Kuwait Journal of Science and Engineering (Issue 1B), 1, 145-180, 2013.