World Academy of Science, Engineering and Technology
International Journal of Structural and Construction Engineering
Vol:9, No:7, 2015

Optimum Design of Steel Space Frames by Hybrid Teaching-Learning Based Optimization and Harmony Search Algorithms

Alper Akın, İbrahim Aydoğdu

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf

Abstract**—**This study presents a hybrid metaheuristic algorithm to obtain optimum designs for steel space buildings. The optimum design problem of three-dimensional steel frames is mathematically formulated according to provisions of LRFD-AISC (Load and Resistance factor design of American Institute of Steel Construction). Design constraints such as the strength requirements of structural members, the displacement limitations, the inter-story drift and the other structural constraints are derived from LRFD-AISC specification. In this study, a hybrid algorithm by using teaching-learning based optimization (TLBO) and harmony search (HS) algorithms is employed to solve the stated optimum design problem. These algorithms are two of the recent additions to metaheuristic techniques of numerical optimization and have been an efficient tool for solving discrete programming problems. Using these two algorithms in collaboration creates a more powerful tool and mitigates each other's weaknesses. To demonstrate the powerful performance of presented hybrid algorithm, the optimum design of a large scale steel building is presented and the results are compared to the previously obtained results available in the literature.

Keywords**—**Optimum structural design, hybrid techniques, teaching-learning based optimization, harmony search algorithm, minimum weight, steel space frame.

I. Introduction

STEEL buildings are one of the most common structural types as these structures have high strength and ductility as well as fast constructability. In practice, design engineers try to meet required criteria and structural performance, but recent economic conditions in the world push them to consider how to obtain more economical designs as well. However, obtaining optimum structural designs is not an easy mission for designers due to the complexity and nonlinearity in the analysis and the design of steel structures. Generally, that complexity includes selection of discrete design variables, complex design limitations on ultimate strength capacities of structural members, displacement constraints, and stability and geometric compatibilities. At this stage, metaheuristic techniques, important tools for optimum structural design problems, take care of this challenge.

Metaheuristic techniques have become efficient tools for structural optimization problems since their emergence and in recent years large number of optimum design algorithms have been developed that are based on these techniques [1]-[9].

A. Akın is with Trinity Meyer Utility Structures, Memphis, TN 38125 USA (phone: +1 901-486-7570; e-mail: alperakin@ yahoo.com).

I. Aydoğdu is with Akdeniz University, Antalya, 07058 Turkey (e-mail: aydogdu@akdeniz.edu.tr).

In this study, a hybrid version of the teaching-learning based optimization (TLBO) and harmony search (HS) algorithms is developed and employed for the optimum design of steel space frames. The sequence numbers of the W steel section, listed in steel profile table, are treated as design variables. Design constraints are implemented from the specification of LRFD-AISC (Load and Resistance Factor Design [LRFD]) [10] which include the ultimate strength requirements, the displacement limitations, inter-story drift restrictions, and geometric constraints. The solution to the discrete design optimization problem is obtained by the proposed algorithm and the design example is presented to demonstrate the performance of the design optimization algorithm.

II. Proposed Hybrid TLBO-HS Algorithm

A. Teaching-learning Based Algorithm

The TLBO algorithm, population-based global optimization method originally developed by [11]-[13], is inspired by the interaction and outcome of the teacher and learners and mimics the teaching–learning process by simulating the interactivity between the teacher and learners (students) in a class. This optimization algorithm requires only common controlling parameters like the population size and number of generations. In this simulation, the teacher, who is considered the most knowledgeable person in the class (population), desires to improve the average performance and information level on a specific subject of students in the class (individuals in a population). Since the teacher shares his or her knowledge and experience with the learners, the quality of the teacher affects the outcome of the learners. It is obvious that a more qualified teacher generates better student outcomes, in terms of their marks or grades, which are measured by a higher mean value (performance criterion).

The analogy of teaching–learning process, which TLBO algorithm is based on, is summarized in [11]-[13]. Based on the teaching–learning process, the mathematical process of TLBO is divided into two phases. The first is the "teacher phase", which means learning from the teacher; and the second is the "learner phase", which means learning through the interactions between learners through efforts to share information while interacting with each other. The implementation of TLBO algorithm is explained with stepwise manner and briefly discussed below. The flowchart of the TLBO algorithm is presented in Fig. 1.

The TLBO algorithm requires assigning the number of

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf

students (population size) and stopping criteria (maximum number of generations). In this step, the class, which represents the population, is filled with randomly generated students (solutions) according to the population size (n_p) and number of design variables (n_{dv})

$$\text{Class} = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_{n_{dv}-1}^1 & x_{n_{dv}}^1 \\ x_1^2 & x_2^2 & \dots & x_{n_{dv}-1}^2 & x_{n_{dv}}^2 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ x_1^{n_p-1} & x_2^{n_p-1} & \dots & x_{n_{dv}-1}^{n_p-1} & x_{n_{dv}}^{n_p-1} \\ x_1^{n_p} & x_2^{n_p} & \dots & x_{n_{dv}-1}^{n_p} & x_{n_{dv}}^{n_p} \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \end{matrix} \begin{matrix} \rightarrow & f(x^1) \\ \rightarrow & f(x^2) \\ \vdots & \vdots \\ \rightarrow & f(x^{n_p-1}) \\ \rightarrow & f(x^{n_p}) \end{matrix}$$

where, each row represents a student (solution) in the population (class) and $f(x_1, x_2, \dots, n_p)$ represents the corresponding objective function value.

During the teaching phase of the algorithm where learners learn from the teacher, the teacher's objective is to improve their mean outputs (grades) [11]. The best solution which has the minimum objective function value in the population is found and it resembles the teacher. Since the teacher will make an effort to move the mean of the population (class) for the students towards the teacher, an update formula for the solution (student) is applied as;

$$\text{Difference Mean}_i = r_i + (M_{new} - T_f M_i)$$

where, T_f is the teaching factor that decides the value of mean to be changed, and r_i is the random number in the range [0, 1]. The teaching factor may have a value between 1 and 2 and it is not utilized as an input to the algorithm, and its value is randomly assigned by using;

$$T_f = \text{round}[1 + \text{Random}(0,1)\{2,1\}]$$

However, the algorithm performs much better if the value of T_f is either 1 or 2 and hence to simplify the algorithm, the teaching factor is suggested to take a value of either 1 or 2 depending on the rounding up criteria. This difference modifies the existing solution according to;

$$x_{new,i} = x_{old,i} + \text{Difference Mean}$$

In case the new solution has better output than the current solution, the new solution is replaced with the current solution. Otherwise, existing solution is preserved. In other words, if $f(x_{new,i}) < f(x_i)$ $x_i = x_{new,i}$; if $f(x_{new,i}) \geq f(x_i)$ $x_i = x_i$.

In the learner phase of the algorithm, according to the teaching–learning process, learners can also increase their knowledge by means of interaction among themselves. So, a student interacts randomly with other students in the class to learn something new and to enhance his or her knowledge. Obviously, if the other students have more knowledge than him or her, a student can learn new things. At any iteration, sharing new information between the learner i and j in the class and learner modification can be expressed mathematically;

For $i = 1: P_n$

X_i and randomly selected learner X_j, where $i \neq j$

$$X_{new,i} = X_{old,i} + r_i(X_i - X_j) \text{ if } f(X_i) < f(X_j)$$

$$X_{new,i} = X_{old,i} + r_i(X_j - X_i) \text{ if } f(X_j) < f(X_i)$$

End For.

Similar to the teaching process, if the new solution produces a better objective function value $f(x_{new,i})$ than the current solution $f(x_i)$ changes x_i to $x_{new,i}$. Otherwise, it preserves the existing solution.

At the end of the learning phase, a cycle (iteration) is completed for the TLBO and the steps in teaching and learning phase are continued until reaching a termination criterion. Generally, a termination criterion is a predetermined maximum generation cycle number.

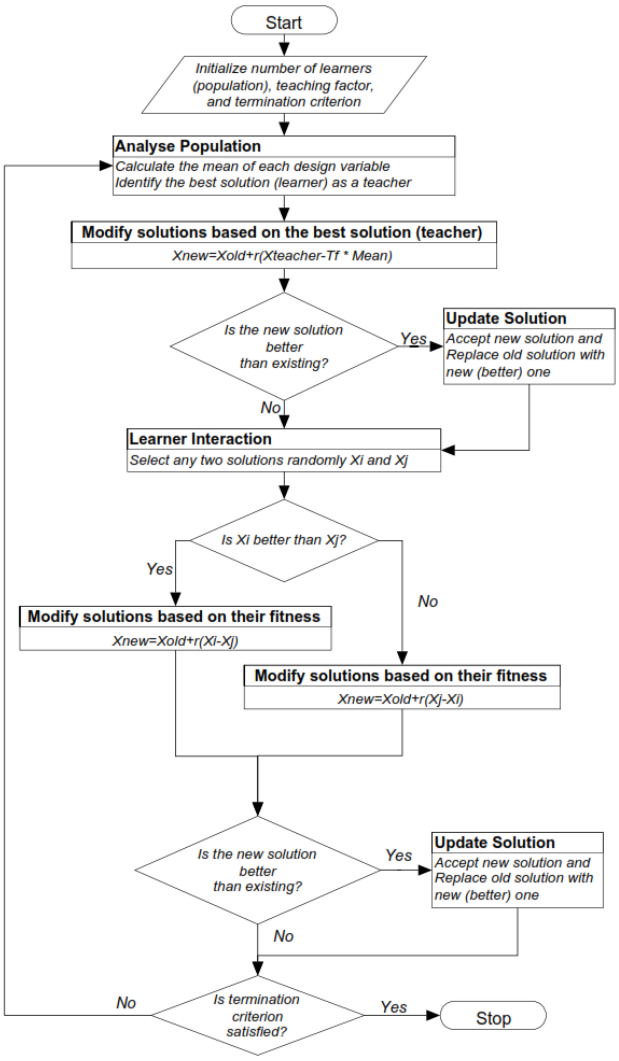


Fig. 1 Flow chart for Teaching-Learning Based Optimization (TLBO) algorithm

B. Harmony Search Algorithm

The HS algorithm, an emerging metaheuristic optimization algorithm, is inspired by the working principles of harmony

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf

improvisation. Since the initial development of HS algorithm by [14], [15], it now has substantial literature. HS algorithms have been applied to a diverse range of structural engineering problems [6], [16]-[20]. As concluded in these studies, HS algorithm is an effective method to find the optimum solutions of such design problems.

The musicians in an orchestra try various possible combinations of the music pitches stored in their memory, while they are improvising a harmony to find a fantastic harmony through musical improvisations. For engineering optimization processes, where the main objective is to find the global solution of a given objective function, this musical process can be imitated to find the global solution of a given objective function in engineering optimization process. In this imitation, decision variables in the optimization problem replace the musicians and the global or near-global solution is treated as the perfect harmony. The flowchart of the basic HS method is given in Fig. 2. HS algorithm can be summarized with following five principal steps.

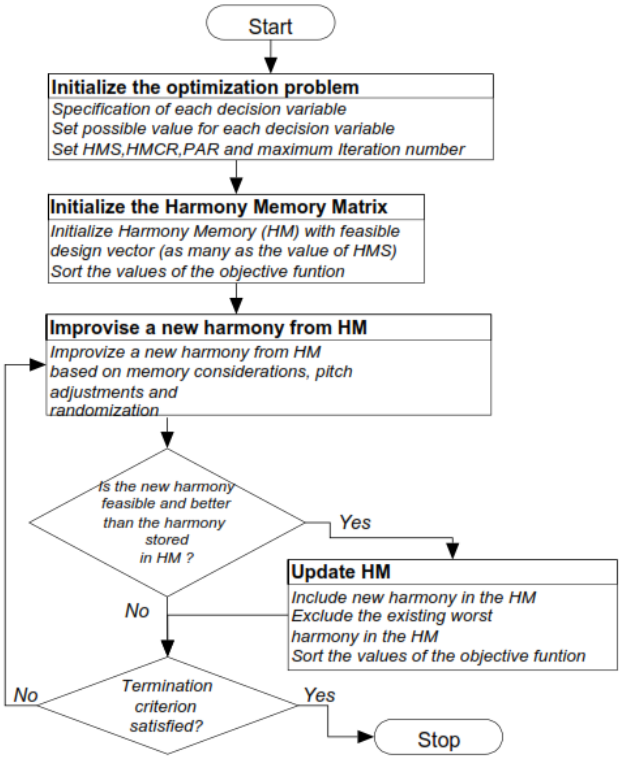


Fig. 2 Flow chart for harmony search (HS) algorithm

Step1. The problem and algorithm parameters are initialized. A possible value range for each design variable of the optimum design problem is specified and a pool is constructed by collecting these values together from which the algorithm selects values for the design variables. Furthermore, HS algorithm parameters; *HMS*; the size of the harmony memory (*HM*) matrix (the number of solution vectors in harmony memory), *HMCR*; harmony memory considering rate, *PAR*; pitch adjusting rate and the maximum number of searches are also determined in this step.

Step2. Initialization of the HM: *HM* matrix is initialized. In *HM* matrix, each row contains the values of design variables which are randomly selected feasible solutions from the design pool for that particular design variable. Hence, *HM* matrix has *n* columns where *n* describes the total number of design variables. The *HM* matrix, the number of rows which is selected in the first step, has the following form:

$$H = \begin{bmatrix} x_{1,1} & x_{2,1} & .. & .. & x_{n-1,1} & x_{n,1} \\ x_{1,2} & x_{2,2} & .. & .. & x_{n-1,2} & x_{n,2} \\ .. & .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. & .. \\ x_{1,hms-1} & x_{2,hms-1} & .. & .. & x_{n-1,hms-1} & x_{n,hms-1} \\ x_{1,hms} & x_{2,hms} & .. & .. & x_{n-1,hms} & x_{n,hms} \end{bmatrix}$$

$x_{i,j}$ describes the value of the i^{th} design variable in the j^{th} randomly selected feasible solution. These candidate designs in *HM* matrix are sorted such that the objective function value corresponding to the first solution vector is the minimum. In other words, the feasible solutions in the *HM* matrix are sorted according to their objective function fitness.

Step3. Improvisation of a new harmony: There are three rules to generate a new harmony vector and these are memory consideration, pitch adjustment, and random selection.

The new value of the i^{th} design variable in a new harmony can be chosen from any discrete value within the range of i^{th} column of the harmony memory matrix with the probability of *HMCR* which varies between 0 and 1. In other words, one of the discrete values of the vector $\{x_{i,1}, x_{i,2}, x_{i,3},\ldots, x_{i,HMS}\}^T$ can be assigned as the new value of x_i with the probability of *HMCR*. All other design variables are treated the same way. In the random selection, the new value of the i^{th} design variable can also be chosen randomly from the entire pool with the probability of 1-*HMCR*. That is;

$$x_i^{new} = \begin{cases} x_i \in \{x_{i,1}, x_{i,2}, \ldots\ldots, x_{i,hms}\}^T, with\ probability\ HMCR \\ x_i \in \{x_1, x_2, \ldots\ldots, x_{ns}\}^T, with\ probability\ (1 - HMCR) \end{cases}$$

where, *ns* is the total number of values for the design variables in the pool. When the new value of the design variable is selected among those of *HM* matrix, this should be checked whether pitch adjustment requires this new value or not. Pitch adjustment parameter *PAR* is used for that decision as follows:

$$Is\ x_i^{new}\ to\ be\ pitch - adjusted = \begin{cases} Yes & with\ probability\ PAR \\ No & with\ probability\ (1 - PAR) \end{cases}$$

If the value selected for x_i^{new} from the harmony memory is the k^{th} element in the general discrete set and the new pitch-adjustment decision for x_i^{new} came out to be yes from the test, then new x_i^{new} takes the neighboring value k+1 or k-1. This

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf

operation helps to improve the harmony memory for diversity, prevent stagnation and get a greater chance of reaching the global optimum.

Step4. Update the HM: If the newly generated harmony vector gives a better objective function value than the worst harmony in *HM* matrix, the new harmony vector is included in the *HM* while the worst one is taken out and the harmony memory matrix is then sorted in descending order by the objective function value.

Step5. Check the termination criterion: If the stopping criterion (i.e. maximum number of improvisations) is satisfied, computation is terminated. Otherwise, steps 3 to 5 are repeated until the termination criterion which is the pre-selected maximum number of iterations is reached.

C. Hybrid TLBO-HS

Based on the mechanisms of the aforementioned TLBO and HS, these methods are combined together and the hybrid optimization algorithm, hTLBO-HS, is proposed to cope with complex optimization problems. In the TLBO algorithm, although the fitness of the students can be improved using effective and adequate search rules, achieving proper initial solutions and protecting diversity in the population to reach global optima are challenging tasks. The teacher attracting the considerable quantities of students may lead to over-similarity among the class (population). HS method is utilized and combined with TLBO in the new optimization algorithm to eliminate this drawback. That provides more efficient and comprehensive functionality by improvising new solutions (students) into population and by sorting the students according to their grade (fitness). Thus, being trapped into local minima and high similarity among the solution candidates can be avoided by the proposed hybrid optimization method.

In addition to teaching and learning phases, HS plays the role of self-studying and researching by the students in TLBO.

By random selection, memory consideration and pitch adjustment functionalities in improvising a new solution (student), the HS method helps the population (class) to improve the mean of the class, to prevent over-similarity and enhance the overall efficiency of the algorithm. HS is utilized as the third phase of the TLBO algorithm in addition to teaching and learning phases in each generation. In HS, population is sorted and taken as harmony memory (HMS=Class size) and teaching and learning phases use a new population in the following cycle. The flowchart of the hTLBO-HS algorithm is presented in Fig. 3. The computer software is developed to apply the explained steps and numerical examples are analyzed to test the effectiveness of the proposed method for optimum design of steel space frames in Section IV.

III. Mathematical Model for Discrete Optimum Design of Space Steel Frames to LRFD-AISC

The design of space steel frames necessitates the selection of steel sections for its columns and beams from a standard steel section table. The obtained design must satisfy the serviceability and strength requirements specified by the design specifications. The design constraints are implemented from LRFD-AISC and the following discrete programming problem is obtained.

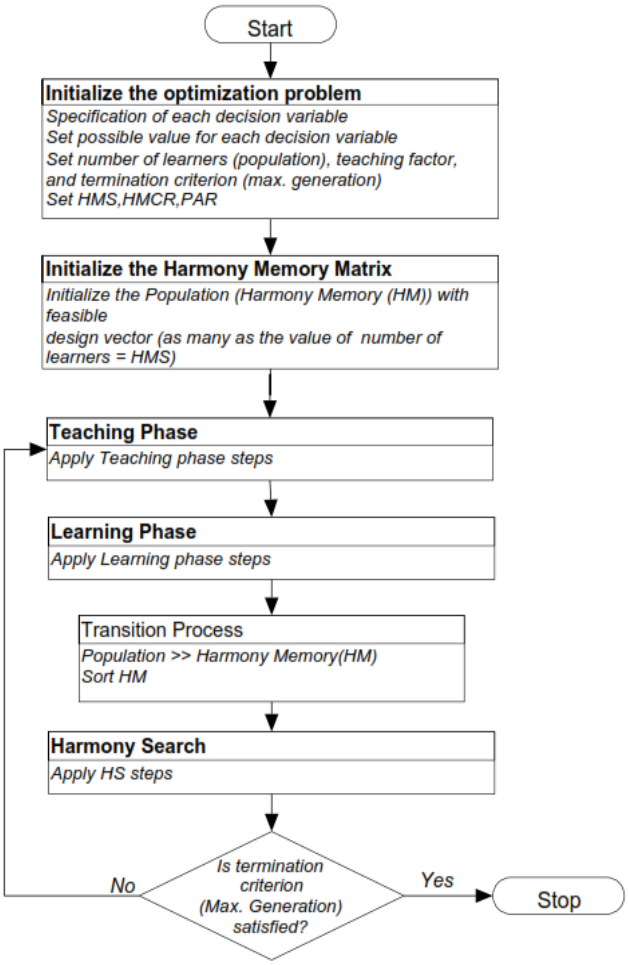


Fig. 3 Flow chart for proposed hTLBO-HS algorithm

The objective function is taken as the minimum weight of the frame to observe the overall economy or the material cost of the frame and it is expressed as;

$$Minimize\ W = \sum_{r=1}^{n_g} m_r \sum_{s=1}^{t_r} l_s \tag{1}$$

where; W defines the weight of the frame, m_r is the unit weight of the steel section selected from the standard steel sections table that is to be adopted for group r, t_r is the total number of members in group r, n_g is the total number of groups in the frame, and l_s is the length of members which belongs to group r.

The strength capacity of the frame members are required to satisfy the following inequalities specified in Chapter H of LRFD-AISC. These strength constraints for W-sections that are selected for beam-column members are given as;

$$for \frac{P_u}{\emptyset P_n} \geq 0.20$$
$$g_{s,i} = \frac{P_u}{\emptyset P_n} + \frac{8}{9}\left(\frac{M_u}{\emptyset_b M_{nx}} + \frac{M_u}{\emptyset_b M_{ny}}\right) - 1 \quad (2)$$

$$for \frac{P_u}{\emptyset P_n} < 0.20$$
$$g_{s,i} = \frac{P_u}{2\emptyset P_n} + \left(\frac{M_u}{\emptyset_b M_{nx}} + \frac{M_u}{\emptyset_b M_{ny}}\right) - 1 \quad (3)$$

where, M_{nx} is the nominal flexural strength at strong axis (x axis), M_{ny} is the nominal flexural strength at weak axis (y axis), M_{ux} is the required flexural strength at strong axis (x axis), M_{uy} is the required flexural strength at weak axis (y axis), P_n is the nominal axial strength (Tension or compression) and P_u is the required axial strength (Tension or compression) for member *i*. The values of M_{ux} and M_{uy} are to be obtained by carrying out *P-Δ* analysis of the steel frame. This is an iterative process which is quite time consuming. In Chapter C of LRFD-AISC an alternative way is suggested for the computations of M_{ux} and M_{uy} values where two first order elastic analyses are carried out. In the first analysis, the frame is analyzed under the gravity loads only where the sway of the frame is prevented from obtaining M_{nt} values. In the second analysis, the frame is analyzed only under the lateral loads to find M_{lt} values. These moment values are combined as;

$$M_u = B_1 M_{nt} + B_2 M_{lt} \quad (4)$$

where, B_1 is the moment magnifier coefficient and B_2 is the sway moment magnifier coefficient. The details of how these coefficients are calculated are stated in Chapter C of LRFD-AISC [10].

Displacement constraints, the lateral displacements and deflection of beams in steel frames, are limited by the steel design codes due to serviceability requirements. According to the ASCE Ad Hoc Committee report, the accepted range of drift limits by first-order analysis is 1/750 to 1/250 times the building height, H, with a recommended value of H/400.The typical limits on the inter-story drift are 1/500 to 1/200 times the story height. Based on this report the deflection limits recommended are proposed in [9], [10] for general use which is repeated in Table I.

TABLE I
DISPLACEMENT LIMITATIONS FOR STEEL FRAMES

	Item	Deflection Limit
1	Floor girder deflection for service live load	L/360
2	Roof girder deflection	L/240
3	Lateral drift for service wind load	H/400
4	Inter-story drift for service wind load	H/300

It is necessary to limit the mid-span deflections of beams in a steel space frame in an effort to not cause cracks in brittle finishes that they may support due to excessive displacements. Deflection constraints are expressed by the following inequality;

$$g_{dj} = \frac{\delta_{jl}}{\delta_j^u} - 1 \leq 0 \qquad j = 1,..,n_{sm}, \quad l = 1,..,n_{lc} \quad (5)$$

where, δ_{jl} is the maximum deflection of j^{th} member under the l^{th} load case, δ_j^u is the upper bound on this deflection which is defined in the code as span/360 for beams carrying brittle finishers, n_{sm} is the total number of members where deflection limitations are to be imposed, and n_{lc} is the number of load cases.

Drift constraints are of two types. One is the restriction applied to the top story sway and the other is the limitation applied on the inter-story drift. Top story drift limitation is expressed as in;

$$g_{tdj} = \frac{(\Delta_{top})_{jl}}{\frac{H}{Ratio}} - 1 \leq 0$$
$$j = 1,..,n_{jtop}, \qquad l = 1,..,n_k \quad (6)$$

where, H is the height of the frame, n_{jtop} is the number of joints on the top story, n_{lc} is the number of load cases, $(\Delta_{top})_{jl}$ is the top story drift of the j^{th} joint under l^{th} load case and ratio is the drift ratio given the ASCE Ad Hoc Committee report.

In multi-story steel frames the relative lateral displacements (Inter-Story Drift) of each floor is required to be limited. This limit is defined as the maximum inter-story drift which is specified as h_{sx} /Ratio where h_{sx} is the story height and ratio is a constant value provided by the ASCE Ad Hoc Committee report;

$$g_{idj} = \frac{(\Delta_{oh})_{jl}}{\frac{h_{sx}}{Ratio}} - 1 \leq 0$$
$$j = 1,..,n_{st}, \qquad l = 1,..,n_{lc} \quad (7)$$

where, n_{st} is the number of story, n_{lc} is the number of load cases and $(\Delta_{oh})_{jl}$ is the story drift of the j^{th} story under l^{th} load case.

In steel frames, we also need the geometric constraints because it is not desired that the column section for upper floor has larger sections than the lower story column for practical reasons as having a larger section for the upper floor requires special joint arrangements which is neither preferred nor is economical. The same applies to the beam-to-column connections. The W-section selected for any beam should have a flange width smaller than or equal to the flange width of the W-section selected for the column to which the beam is to be connected. These are shown in Fig. 1 and named as geometric constraints. These limitations are included in the design optimization model to satisfy practical requirements. Two types of geometric constraints are considered in the mathematical model. These are column-to-column and beam-to-column geometric constraints.

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf

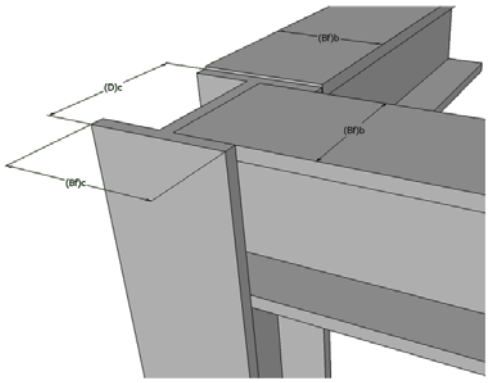


Fig. 4 Beam column geometric constraints

For columns, the depth and the unit weight of W-sections selected for the columns of two consecutive stores should be either equal to each other or the one in the upper story should be smaller than the one in the lower story. These limitations are included in the design problem as;

$$g_{cdi} = \frac{D^a{}_i}{D^b{}_i} - 1 \le 0 \qquad i = 1,..,n_{ccj} \tag{8}$$

$$g_{cmi} = \frac{m^a{}_i}{m^b{}_i} - 1 \le 0 \qquad i = 1,..,n_{ccj} \tag{9}$$

where; n_{ccj} is the number of column-to-column geometric constraints defined in the problem, $m^a{}_i$ is the unit weight of W-section selected for above story, $m^b{}_i$ is the unit weight of W-section selected for below story, $D^a{}_i$ is the depth of W-section selected for above story, and $D^b{}_i$ is the depth of W-section selected for below story.

When a beam is connected to a flange of a column, the flange width of the beam should be less than or equal to the flange width of the column so that the connection can be made without difficulty. In order to achieve this, the flange width of the beam should be less than or equal to (D-2t_b) of the column web dimensions in the connection where D and t_b are the depth and the flange thickness of the W-section respectively as shown in Fig. 4;

$$g_{bci} = \frac{(B_f)_{bi}}{D_{ci} - 2(t_{bc})_i} - 1 \le 0 \qquad i = 1,..,n_{j1} \tag{10}$$

or

$$g_{bbi} = \frac{(B_f)_{bi}}{(B_f)_{ci}} - 1 \le 0 \qquad i = 1,..,n_{j2} \tag{11}$$

where, n_{j1} is the number of joints where beams are connected to the web of a column, n_{j2} is the number of joints where beams connected to the flange of a column, D_{ci} is the depth of W-section selected for the column at joint i, $(t_b)_{cj}$ is the flange thickness of the W-section selected for the column at joint i, $(B_f)_{ci}$ is the flange width of the W-section selected for the column at joint i and is the $(B_f)_{bi}$ flange width of the W-section selected for the beam at joint i.

The discrete optimum design problem of steel space frames necessities the selection of appropriate steel sections for the frame members from W-sections list such that the objective function described in (1) is the minimum while the design constraints given in inequalities from (2) to (11) are satisfied. This is a combinatorial optimization problem.

IV. Design Examples

The plan and 3D views of the five-story, two-bay steel space frame shown in the Fig. 5 and 6 is designed using the hTLBO-HS algorithm and the optimum solutions obtained. The regular steel frame with 54 joints and 105 members are grouped into 11 independent design variables. Gravity loads as well as lateral loads, which the frame is subjected to, are computed per ASCE 7-05. The design dead and live loads are selected as 2.88kN/m2 and 2.39kN/m2 respectively, the ground snow load is considered to be 0.755kN/m2 and the basic wind speed is assumed to be 105mph (65 m/s). The following load combinations are considered in the design of the frame according to the code specification and these load combinations are 1.2D+1.6L+0.5S, 1.2D+0.5L+1.6S, 1.2D+1.6W+0.5L +0.5S where D is the dead load, L is the live load, S is the snow load and W is the wind load. In this design example, the maximum deflection of beam members is restricted at 1.67 cm, the drift ratio limits for inter story drift and top story drift are taken as 1.33 cm and 6.67 cm, respectively.

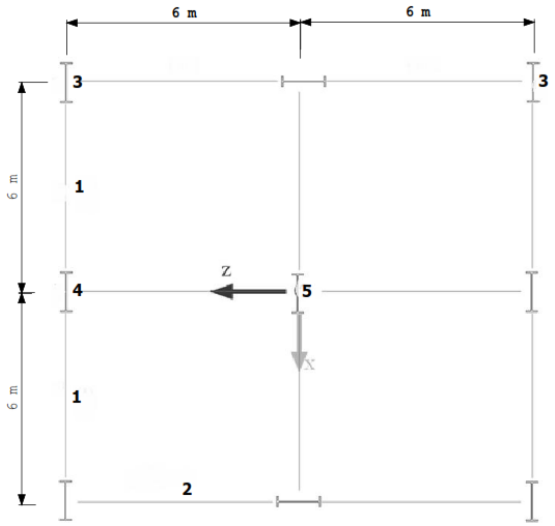


Fig. 5 Plan view of five-story, two bay steel frame

Optimum design problem of the five-story, two-bay steel frame is solved by using proposed hTLBO-HS algorithm described. In these examples, following TLBO and HS parameters are used: teaching factor T_f =1.20, population and harmony memory size (HMS) = 70, pitch adjusting rate (PAR) = 0.3, harmony memory considering rate (HMCR) = 0.9, and maximum generation number = 200. The optimum designs obtained by hTLBO-HS are shown in Table II. It is clearly illustrated in the table that the lightest weight is obtained as 269.184 kN by using hTLBO-HS algorithm The design histories of these algorithms for the best solutions are also

plotted in Fig. 7. It is apparent from the figure that the proposed algorithms show remarkable performance.

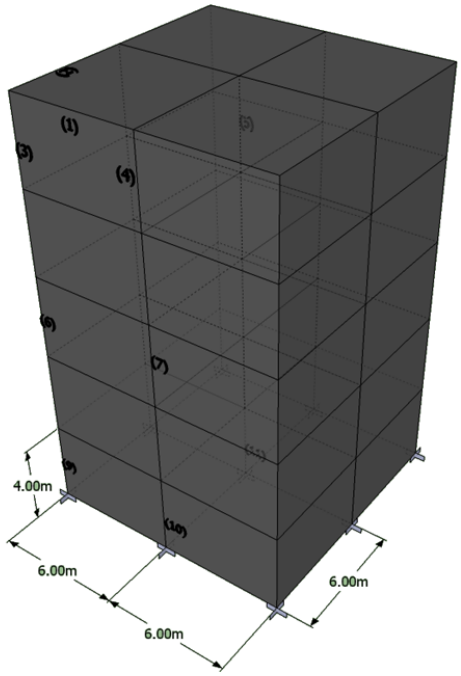


Fig. 6 3D View of the 105-member steel frame

TABLE II
DESIGN RESULTS OF THE FIVE-STORY, TWO BAY STEEL FRAME

Member group	Type	HS	hTLBO-HS
1	Beam	W530X66	W460X52
2	Beam	W310X38.7	W200X35.9
3	Column	W200X35.9	W310X38.7
4	Column	W200X35.9	W200X46.1
5	Column	W360X44	W360X44
6	Column	W310X38.7	W310X74
7	Column	W360X72	W250X73
8	Column	W610X92	W610X101
9	Column	W410X53	W460X74
10	Column	W360X72	W250X73
11	Column	W760X147	W760X173
Max. Strenght Ratio		0.979	0.921
TopDrift (cm)		4.837	4.708
Inter Storey Drift (cm)		1.333	1.325
Maximum Iteration		50000	42000
Weight (kN)		278.196	269.184

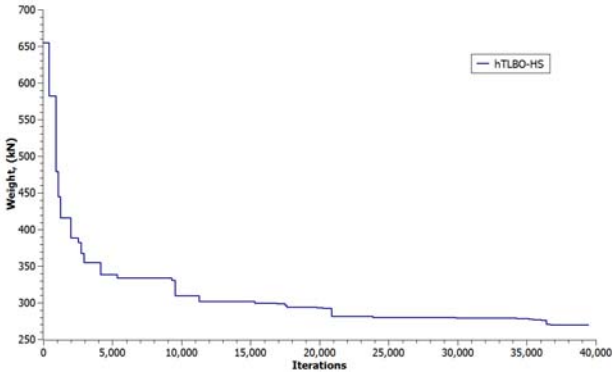


Fig. 7 Design history for 1105-member steel frame

V. CONCLUSION

A hybrid discrete optimization (hTLBO-HS) algorithm is proposed and utilized to calculate a minimum weight for steel space frame structures, where the design constraints are implemented as stated in the LRFD-AISC provisions, by optimizing the beam and column sections. The recently developed hTLBO-HS heuristic algorithm is simple, mathematically less complex and consists of three phases; teaching and learning phases of TLBO and HS phase. In the design, the cross-sectional areas of W-section are considered as design variables. Design example given is performed to test the efficiency of the proposed optimization algorithm in this paper for complex grillage systems.

The performance of the proposed hTLBO-HS clearly shows that the proposed hTLBO-HS algorithm has outperformed the HS algorithm in terms of better result. The results verify that the hTLBO-HS requires less iterations of structural analysis. However, it should be mentioned that its performance is dependent upon the initial values selected for the algorithm parameters (Population (number of students)(=HMS), Tf, HMCR, PAR), which are general characteristics of the stochastic optimization methods.

Consequently, the obtained results by hTLBO-HS are powerful and efficient in finding the optimum solution for discrete structural optimization problems. It can be clearly stated that the proposed algorithm can be easily customized to suit the optimization of any system involving a large number of variables and objectives.

REFERENCES

[1] S. K. Azad, O. Hasancebi and M. P. Saka, "Guided stochastic search technique for discrete sizing optimization of steel trusses: A design-driven heuristic approach," *Computers & Structures*, vol.134, no. pp. 62-74, Apr 1 2014.

[2] O. Hasancebi and S. K. Azad, "Discrete size optimization of steel trusses using a refined big bang-big crunch algorithm," *Engineering Optimization*, vol.46, no. 1, pp. 61-83, Jan 2 2014.

[3] O. Hasancebi, T. Teke and O. Pekcan, "A bat-inspired algorithm for structural optimization," *Computers & Structures*, vol.128, no. pp. 77-90, Nov 2013.

[4] A. Kaveh and S. Talatahari, "Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures," *Computers & Structures*, vol.87, no. 5-6, pp. 267-283, Mar 2009.

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf

[5] A. Kaveh and S. Talatahari, "Charged system search for optimum grillage system design using the LRFD-AISC code," *Journal of Constructional Steel Research*, vol.66, no. 6, pp. 767-771, Jun 2010.
[6] M. P. Saka and Z. W. Geem, "Mathematical and Metaheuristic Applications in Design Optimization of Steel Frame Structures: An Extensive Review," *Mathematical Problems in Engineering*, no. pp. 2013.
[7] C. V. Camp and M. Farshchin, "Design of space trusses using modified teaching-learning based optimization," *Engineering Structures*, vol.62-63, no. pp. 87-97, Mar 15 2014.
[8] C. V. Camp and F. Huq, "CO2 and cost optimization of reinforced concrete frames using a big bang-big crunch algorithm," *Engineering Structures*, vol.48, no. pp. 363-372, Mar 2013.
[9] C. V. Camp and A. Akin, "Design of Retaining Walls Using Big Bang-Big Crunch Optimization," *Journal of Structural Engineering-Asce*, vol.138, no. 3, pp. 438-448, Mar 2012.
[10] "*Load and Resistance Factor Design, Volume 1, Structural Members Specifications Codes*," American Institute of Steel Construction 2001., pp. 2001.
[11] R. V. Rao, V. J. Savsani and D. P. Vakharia, "Teaching-Learning-Based Optimization: An optimization method for continuous non-linear large scale problems," *Information Sciences*, vol.183, no. 1, pp. 1-15, Jan 15 2012.
[12] R. V. Rao, V. J. Savsani and J. Balic, "Teaching-learning-based optimization algorithm for unconstrained and constrained real-parameter optimization problems," *Engineering Optimization*, vol.44, no. 12, pp. 1447-1462, 2012.
[13] R. V. Rao, V. J. Savsani and D. P. Vakharia, "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol.43, no. 3, pp. 303-315, Mar 2011.
[14] Z. W. Geem, "*Optimal design of water distribution networks using harmony search.* ," Korea University vol.PhD, pp. 2000.
[15] Z. W. Geem, J. H. Kim and G. V. Loganathan, "A new heuristic optimization algorithm: Harmony search," *Simulation*, vol.76, no. 2, pp. 60-68, Feb 2001.
[16] A. Akin and M. P. Saka, "Harmony search algorithm based optimum detailed design of reinforced concrete plane frames subject to ACI 318-05 provisions," *Computers & Structures*, vol.147, no. pp. 79-95, Jan 15 2015.
[17] G. Bekdas and S. M. Nigdeli, "Estimating optimum parameters of tuned mass dampers using harmony search," *Engineering Structures*, vol.33, no. 9, pp. 2716-2723, Sep 2011.
[18] S. O. Degertekin, "Optimum design of steel frames using harmony search algorithm," *Structural and Multidisciplinary Optimization*, vol.36, no. 4, pp. 393-401, Oct 2008.
[19] O. Hasancebi, F. Erdal and M. P. Saka, "Adaptive Harmony Search Method for Structural Optimization," *Journal of Structural Engineering-Asce*, vol.136, no. 4, pp. 419-431, Apr 2010.
[20] A. Kaveh and A. S. M. Abadi, "Cost optimization of a composite floor system using an improved harmony search algorithm," *Journal of Constructional Steel Research*, vol.66, no. 5, pp. 664-669, May 2010.

Open Science Index, Structural and Construction Engineering Vol:9, No:7, 2015 publications.waset.org/10002405/pdf