

Reformulations of Big Bang-Big Crunch Algorithm for Discrete Structural Design Optimization

O. Hasançebi and S. Kazemzadeh Azad

Abstract—In the present study the efficiency of Big Bang-Big Crunch (BB-BC) algorithm is investigated in discrete structural design optimization. It is shown that a standard version of the BB-BC algorithm is sometimes unable to produce reasonable solutions to problems from discrete structural design optimization. Two reformulations of the algorithm, which are referred to as modified BB-BC (MBB-BC) and exponential BB-BC (EBB-BC), are introduced to enhance the capability of the standard algorithm in locating good solutions for steel truss and frame type structures, respectively. The performances of the proposed algorithms are experimented and compared to its standard version as well as some other algorithms over several practical design examples. In these examples, steel structures are sized for minimum weight subject to stress, stability and displacement limitations according to the provisions of AISC-ASD.

Keywords—Structural optimization, discrete optimization, metaheuristics, big bang-big crunch (BB-BC) algorithm, design optimization of steel trusses and frames.

I. INTRODUCTION

IN general, the optimum design of a steel structure is an attempt to find the best combination of design variables that results in a minimum weight or cost design of the structure. Meanwhile, for practical applications the optimum design should satisfy a set of predefined constraints imposed according to a given code of practice. In general, optimum design of skeletal structures (either frame or truss structures) can be divided into three main categories as sizing, shape, and topology optimization. In sizing optimization the cross sectional areas of structural members are considered as design variables. This can further be divided into two subcategories in terms of the nature of the design variables employed: continuous and discrete. In continuous sizing optimization any positive value can be assigned to cross sectional areas of elements. However, this is usually not the case in practical applications, where structural members should be adopted from a set of available sections. The latter is addressed to discrete sizing optimization.

The two well-known categories of traditional structural optimization methods are mathematical programming [1] and

optimality criteria [2, 3] approaches. The main shortcomings of traditional design optimization techniques are that these techniques are gradient-based approaches and thus typically work on the basis of continuous design variables. Additionally, computing the gradients of highly nonlinear objective functions of practical instances becomes another obstacle while working with the traditional techniques. The most recent category of structural optimization techniques is referred to as non-traditional stochastic search methods or metaheuristics. These algorithms, such as genetic algorithms (GAs), particle swarm optimization (PSO), ant colony optimization (ACO), etc., are typically nature inspired techniques, which borrow their working principles from natural phenomena [4]. Unlike traditional optimization techniques, metaheuristic algorithms do not perform a gradient based search and are able to handle both discrete and continuous design variables. Additionally, the stochastic nature of metaheuristics makes it more probable to find a near optimum solution (if not the global optimum) even for complicated design optimization problems. Since the optimization approaches based on metaheuristics are robust and quite successful in finding the optimal solutions, these algorithms can efficiently be employed for solving practical structural optimization problems. Nowadays, there are a large number of such metaheuristic algorithms available in the literature. The state-of-the-art reviews of these algorithms as well as their applications in structural design optimization problems can be found in [5-7].

Big bang-big crunch (BB-BC) algorithm is a novel metaheuristic optimization method based on the BB-BC theory of the universe evolution [8]. Due to its competitive performance and ease of use, the BB-BC technique quickly became one of the most popular algorithms of the recent years [9-18]. Afshar and Motaei [9] used the BB-BC to find the optimal solution of large scale reservoir operations problem. Tang et al. [10] used the technique for parameter estimation in structural systems. The first application of the BB-BC for optimum design of skeletal structures was carried out by Camp [11]. Therein, the optimum design of planar and spatial truss structures was performed using a modified variant of the technique. To enhance the efficiency of the BB-BC, he proposed a weighting parameter to control the influence of both the center of mass and the current global best solution on generating new candidate solutions. Additionally, a multiphase search strategy was utilized to increase the quality

O. Hasançebi, Associate Professor, Middle East Technical University, Department of Civil Engineering, Ankara, Turkey (phone: +90 312 210 2456; fax: +90 312 210 1262; e-mail: oguzhan@metu.edu.tr).

S. Kazemzadeh Azad, Ph.D. student, Middle East Technical University, Department of Civil Engineering, Ankara, Turkey (phone: +90 537 910 8622; e-mail: saeid.azad@metu.edu.tr).

of solutions. The study evinced efficiency of the BB-BC in comparison to other metaheuristics, such as GAs, PSO, and ACO. Kaveh and Abbasgholih [12] used the Camp's strategy of generating new candidate solutions for optimum design of planar steel sway frames. Recently, Lamberti and Pappalettere [13] introduced an improved BB-BC algorithm for weight minimization of truss structures and reported promising results using four benchmark truss optimization instances. Kaveh and Talatahari [14-16] developed hybrid variants of the BB-BC for design optimization of different types of skeletal structures. In addition, Kaveh et al. [17] employed a hybrid BB-BC algorithm for optimum seismic design of gravity retaining walls. A recent performance evaluation of the BB-BC algorithm was carried out by Kazemzadeh Azad et al. [18], where efficiency of the method in benchmark engineering optimization problems is investigated.

In the present study BB-BC algorithm is employed for discrete size optimum design of steel truss and frame structures. It is shown that a standard formulation of the BB-BC algorithm may sometimes fail to provide acceptable solutions to discrete sizing problems in structural optimization. The observed deficiencies of the algorithm are attributed to ineffective manipulation of the two search parameters; namely, search dimensionality and step size. Reformulations of the BB-BC algorithm are then proposed, where the formula used by the standard algorithm for generating new candidate solutions around the center of mass is simplistically yet efficaciously reformulated, resulting in the so-called modified BB-BC (MBB-BC) and exponential BB-BC (EBB-BC) variants for truss and frame type structures, respectively. The performances of the proposed algorithms are experimented and compared to its standard version as well as some other metaheuristic techniques using several practical design examples. In these examples the steel trusses and frames are sized for minimum weight subject to stress, stability and displacement limitations according to the provisions of AISC-ASD [19]. The numerical results confirm the efficiency of the proposed approaches in practical design optimization of steel structures. The outline of the paper is as follows. The second section includes a detailed statement of the optimization problem for skeletal steel structures based on AISC-ASD [19] specification. In the third section, the steps involved in implementation of a standard BB-BC algorithm are outlined. In the fourth section the observed deficiencies of the standard BB-BC algorithm in discrete design optimization of steel structures are discussed. In the fifth section the proposed reformulations of BB-BC algorithm (i.e., MBB-BC and EBB-BC) are discussed in detail. The performance evaluations of the proposed algorithms thorough numerical examples are carried out in the sixth section. The last section of the paper covers the concluding remarks.

II. PRACTICAL DESIGN OPTIMIZATION OF SKELETAL STEEL STRUCTURES

For a steel structure consisting of N_m members that are

collected in N_d member groups, the design optimization problem according to AISC-ASD [19] can be formulated as follows.

The objective is to find a vector of integer values \mathbf{I} (Eq. 1) representing the sequence numbers of steel sections assigned to N_d member groups

$$\mathbf{I}^T = [I_1, I_2, \dots, I_{N_d}] \quad (1)$$

to minimize the weight (W) of the structure

$$W = \sum_{i=1}^{N_d} \rho_i A_i \sum_{j=1}^{N_i} L_j \quad (2)$$

where A_i and ρ_i are the length and unit weight of the steel section adopted for member group i , respectively, N_i is the total number of members in group i , and L_j is the length of the member j which belongs to group i .

The members subjected to a combination of axial compression and flexural stress must be sized to meet the following stress constraints:

$$\text{if } \frac{f_a}{F_a} > 0.15 ; \quad \left[\frac{f_a}{F_a} + \frac{C_{mx} f_{bx}}{\left(1 - \frac{f_a}{F_{ex}}\right) F_{bx}} + \frac{C_{my} f_{by}}{\left(1 - \frac{f_a}{F_{ey}}\right) F_{by}} \right] - 1.0 \leq 0 \quad (3)$$

$$\left[\frac{f_a}{0.60 F_y} + \frac{f_{bx}}{F_{bx}} + \frac{f_{by}}{F_{by}} \right] - 1.0 \leq 0 \quad (4)$$

$$\text{if } \frac{f_a}{F_a} \leq 0.15 ; \quad \left[\frac{f_a}{F_a} + \frac{f_{bx}}{F_{bx}} + \frac{f_{by}}{F_{by}} \right] - 1.0 \leq 0 \quad (5)$$

If the flexural member is under tension, then the following formula is used instead:

$$\left[\frac{f_a}{0.60 F_y} + \frac{f_{bx}}{F_{bx}} + \frac{f_{by}}{F_{by}} \right] - 1.0 \leq 0 \quad (6)$$

in Eqs. (3-6), F_y is the material yield stress, and $f_a = (P/A)$ represents the computed axial stress, where A is the cross-sectional area of the member. The computed flexural stresses due to bending of the member about its major (x) and minor (y) principal axes are denoted by f_{bx} and f_{by} , respectively. F'_{ex} and F'_{ey} denote the Euler stresses about principal axes of the member that are divided by a factory of

safety of 23/12. F_a stands for the allowable axial stress under axial compression force alone, and is calculated depending on elastic or inelastic buckling failure mode of the member using Formulas 1.5-1 and 1.5-2 given in AISC-ASD [19]. The allowable bending compressive stresses about major and minor axes are designated by F_{bx} and F_{by} , which are computed using the Formulas 1.5-6a or 1.5-6b and 1.5-7 given in AISC-ASD [19]. It is also required that the computed shear stresses (f_v) in members are smaller than the allowable shear stresses (F_v) as formulated in Eq. (7) where C_v is referred to as web shear coefficient.

$$f_v \leq F_v = 0.40C_vF_y \quad (7)$$

Slenderness limitations are imposed on all members such that maximum slenderness ratio ($\lambda = KL/r$) is limited to 300 and 200 for tension and compression members, respectively.

As for the displacement constraints, the maximum lateral displacements are restricted to be less than $H/400$, and upper limit of story drift is set to be $h/400$ for frame type structures, where H is the total height of the frame building and h is the height of a story.

Finally, we consider geometric constraints between beams and columns framing into each other at a common joint for practicality of an optimum solution generated. For the two beams B1 and B2 and the column shown in Fig. 1, one can write the following geometric constraints:

$$\frac{b_{fb}}{b_{fc}} - 1.0 \leq 0 \quad (8)$$

$$\frac{b'_{fb}}{(d_c - 2t_f)} - 1.0 \leq 0 \quad (9)$$

where b_{fb} , b'_{fb} and b_{fc} are the flange width of the beam B1, the beam B2 and the column, respectively, d_c is the depth of the column, and t_f is the flange width of the column. (8) simply ensures that the flange width of the beam B1 remains smaller than that of the column. On the other hand, (9) enables that flange width of the beam B2 remains smaller than clear distance between the flanges of the column ($d_c - 2t_f$). Further details of the problem formulation can be found in Hasançebi et al. [20].

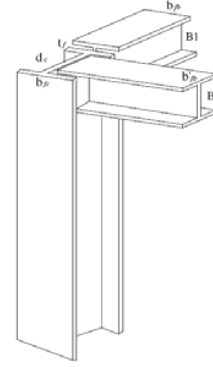


Fig. 1 Beam-column geometric constraints

III. STANDARD BB-BC ALGORITHM

Big bang-big crunch optimization method, developed by Erol and Eksin's study [8], is inspired from the big bang and big crunch theories of the universe evolution. The method is based on continuous application of two successive stages, namely big bang and big crunch phases. During big bang phase, new solution candidates are randomly generated around a point called center of mass. This point is recalculated and updated every time in the big crunch phase with respect to the solution candidates generated.

The steps in the implementation of a standard BB-BC algorithm can be outlined as follows.

Step 1. Initial population: Form an initial population by randomly spreading individuals (solutions) over all the search space (first big bang) in a uniform manner. This step is applied once.

Step 2. Evaluation: The initial population is evaluated, where structural analyses of all individuals are performed with the set of steel sections selected for design variables, and force and deformation responses are obtained under the loads. The objective function values of the feasible individuals that satisfy all problem constraints are directly calculated from (2). However, infeasible individuals that violate some of the problem constraints are penalized using an external penalty function approach, and their objective function values are calculated according to (10).

$$\phi = W \left[1 + p \left(\sum_i c_i \right) \right] \quad (10)$$

In Eq. (10), ϕ is the constrained objective function value, c_i is i -th problem constraint and p is the penalty coefficient used to tune the intensity of penalization as a whole. This parameter is generally set to an appropriate static value of $p = 1$. The fitness scores of the individuals are then calculated by taking the inverse of their objective function values (i.e. fitness = $1/W$ or $1/\phi$ for feasible and infeasible solutions, respectively). The fitness scores are assigned as the mass values for the individuals.

Step 3. Big crunch phase: Calculate the center of mass by taking the weighted average using the coordinates (design variables) and the mass values of every single individual or choose the fittest individual amongst all as their center of mass (the latter approach is adopted in the present study).

Step 4. Big bang phase: Generate new individuals by using normal distribution (big bang phase). For a continuous variable optimization problem, Eq. (11) is used at each iteration to generate new solutions around the center of mass.

$$x_i^{new} = x_i^c + \alpha \cdot N(0,1)_i \frac{(x_i^{\max} - x_i^{\min})}{k} \quad (11)$$

where x_i^c is the value of i -th continuous design variable in the fittest individual, x_i^{\min} and x_i^{\max} are the lower and upper bounds on the value of i -th design variable, respectively, $N(0,1)_i$ is a random number generated according to a standard normal distribution with mean (μ) zero and standard deviation (σ) equal to one, k is the iteration number, and α is a constant.

In the present study, however, a discrete list of ready sections is used for sizing members of a steel structure. Hence, Eq. (12) is employed instead to round off the real values to nearest integers representing the sequence number of ready sections in a given section list

$$I_i^{new} = I_i^c + \text{round} \left[\alpha \cdot N(0,1)_i \frac{(I_i^{\max} - I_i^{\min})}{k} \right] \quad (12)$$

where I_i^c is the value of i -th discrete design variable in the fittest individual, and I_i^{\min} and I_i^{\max} are its lower and upper bounds, respectively.

Step 5. Elitism: Keep the fittest individual found so far in a separate place or as a member of the population.

Step 6. Termination: Go to Step 2 until a stopping criterion is satisfied, which can be imposed as a maximum number of iterations or no improvement of the best design over a certain number of iterations.

IV. SEARCH DIMENSIONALITY AND STEP SIZE

Metaheuristic search techniques offer a general solution methodology for solving a wide range of optimization problems from different disciplines. On the other hand, each optimization problem has unique features of its own, and in most cases a problem-wise reformulation is necessary to achieve the best performance of the algorithm for a particular class of problems. In the following the observed deficiencies of the standard BB-BC algorithm in discrete design optimization of steel structures are discussed in detail. The poor performance of the standard algorithm is attributed to ineffective manipulations of the two search parameters; namely search dimensionality and step size.

Search dimensionality (SD) is defined as the number of design variables that are perturbed to generate a new design through Eq. (12). Perhaps a more general term to quantify the degree of search dimensionality irrespective of problem size will be search dimensionality ratio (SDR), which is computed by proportioning the number of perturbed design variables (N_p) to the total number of design variables (N_d) used in a problem, Eq. (13).

$$SDR = \frac{N_p}{N_d} \quad (13)$$

It may be expected that SDR will be different for each individual in the population, and the average search dimensionality ratio for a population, $(SDR)_{ave}$, is obtained by averaging SDR values of all the individuals (Eq. 14), where $(SDR)_j$ is search dimensionality ratio for individual j and N_{pop} is the population size referring to the total number of individuals in the population.

$$(SDR)_{ave} = \frac{\sum (SDR)_j}{N_{pop}} \quad (14)$$

For continuous optimization problems $(SDR)_{ave}$ will always have a value equal or close to 1.0, since all design variables -except those on the value bounds- are subjected to perturbation during generation of a new individual. That is to say an N -dimensional search is performed by the algorithm at any time during the search process. However, for discrete optimization problems some design variables will remain unchanged owing to the fact that the second term on the right hand side of Eq. (12) is rounded off to zero when the random number r_i generated by normal distribution is too small, which implicitly drive $(SDR)_{ave}$ to low values especially when the iteration number k increases.

On the other hand, the step size for a single design variable is equal to $I_i^{new} - I_i^c$. Hence at any iteration one can define an average step size for an individual and the entire population as formulated in Eqs. (15) and (16)

$$(SS)_{j,ave} = \frac{\sum (I_i^{new} - I_i^c)_j}{N_d} \quad (15)$$

$$(SS)_{ave} = \frac{\sum (SS)_{j,ave}}{N_{pop}} \quad (16)$$

where $(SS)_{j,ave}$ and $(SS)_{ave}$ denote the average step size for j -th individual and entire population, respectively.

Typical results obtained from numerical investigations with

the BB-BC algorithm on discrete sizing optimization problem are reflected in Fig. 2, which shows the variations of $(SDR)_{ave}$ and $(SS)_{ave}$ parameters in the course of search process. It is noted that the average search dimensionality ratio of a population is generally in the order of 0.9 in the first iterations, which results in extreme changes in the individuals. Although this helps provide a diverse population, this amount of diversity is more likely to result in convergence difficulties in case of discrete structural design optimization. A useful starting value of $(SDR)_{ave}$ will be in the range of [0.25, 0.50] based on experiments with an evolution strategies integrated search process [21]. On the other hand, towards the later stages, search is implemented mostly in a single direction per design. As the iterations continue, a somewhat decreased search dimensionality might be useful in the sense that it boosts more exploitative search in the design space. Nevertheless, the search capability of the algorithm is significantly restricted when it is limited too much, as observed in the BB-BC algorithm.

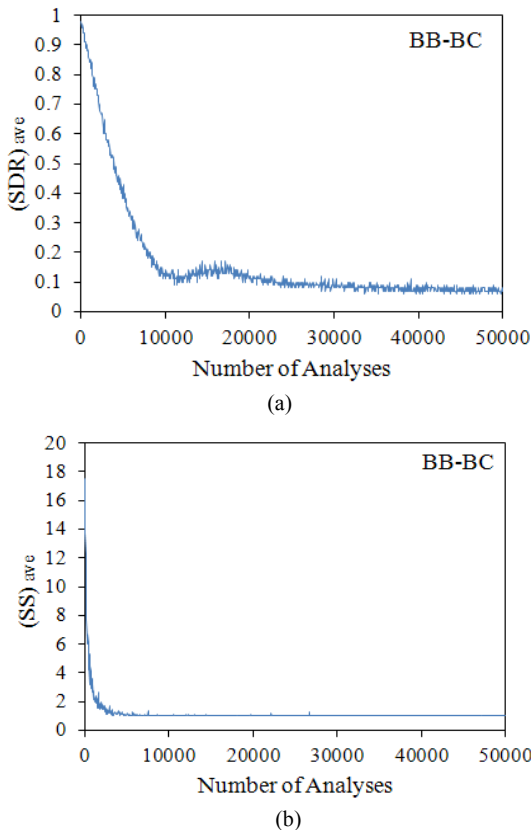


Fig. 2 The variations of (a) $(SDR)_{ave}$ and, (b) $(SS)_{ave}$ in a typical run of BB-BC

V. REFORMULATIONS OF BB-BC ALGORITHM

Noticing the drawbacks of the standard algorithm discussed above, Eq. (17) is proposed in lieu of Eq. (12) to improve the

efficiency of the BB-BC algorithm in discrete structural design optimization. In the new formulation the use of n -th power ($n \geq 2$) of a random number r_i is motivated using any appropriate statistical distribution.

$$I_i^{new} = I_i^c + \text{round} \left[\alpha \cdot r_i^n \frac{(I_i^{\max} - I_i^{\min})}{k} \right] \quad (17)$$

The rationale behind Eq. (17) is to achieve a satisfactory trade-off or compromise between the following two conflicting requirements needed to eliminate the shortcomings of the standard formulation: (i) diminishing search dimensionality in the beginning of the search process and increasing it somewhat towards the latest stage and (ii) enabling large step size from time to time at later optimization stages to facilitate design transitions to new design regions and thereby preventing entrapment of the search in local optima. In fact, at times when the random number is sampled at values below 1, taking n -th power of r_i makes it even much smaller, which helps to fulfill the first requirement. On the other hand, at times when it is sampled at values above 1, it might be amplified to fairly large values by taking its n -th power, helping to satisfy the second requirement.

Two instances of Eq. (17) are generated in Eqs. (18) and (19) by selecting the type of statistical distribution used to sample the random number, where the power of random number n is set to 3 based on extensive numerical experiments.

$$I_i^{new} = I_i^c + \text{round} \left[\alpha \cdot N(0,1)_i^3 \frac{(I_i^{\max} - I_i^{\min})}{k} \right] \quad (18)$$

$$I_i^{new} = I_i^c \pm \text{round} \left[\alpha \cdot E(\lambda=1)_i^3 \frac{(I_i^{\max} - I_i^{\min})}{k} \right] \quad (19)$$

(18) refers to the third power reformulation of big crunch phase according to a normally distributed random number. This reformulation will be referred to as modified BB-BC (MBB-BC) hereafter, and is introduced in relation to discrete design optimization of truss structures [22].

The second reformulation (Eq. 19) is referred to as exponential BB-BC (EBB-BC), where the use of an exponential distribution (E) in conjunction with the third power of random number is favoured when solving problems from discrete design optimization of steel frame structures [23]. It is important to note that unlike normal distribution which samples both positive and negative real numbers, exponential distribution only generates positive numbers. Hence, the rounded term on the right hand side of Eq. (19) should be added to or subtracted from I_i^c under equal probability to allow for both increase and decrease in the value of a design variable.

Fig. 3 and 4 are displayed to demonstrate the influence of the proposed reformulation on a BB-BC integrated search

process. They show typical variations of $(SDR)_{ave}$ and $(SS)_{ave}$ parameters during a search captured while performing numerical investigations with the MBB-BC and EBB-BC algorithms, respectively. Figs 3a and 4a indicate that the starting value of average search dimensionality ratio in both MBB-BC and EBB-BC algorithms is around 0.60-0.70. Although this is slightly more than the upper limit of the recommended range, it leads to a more appropriate diversity in the population as compared to the standard algorithm, and provides a more suitable search mechanism in the initial iterations. As the iterations increase, the $(SDR)_{ave}$ parameter is dragged to smaller values, implying that an explorative search is progressively replaced and dominated by an exploitative one. However, unlike the standard algorithm where a rapid and linear reduction is observed in $(SDR)_{ave}$ towards unfavorably too low values, the reduction happens to be slower and more gradual in both the MBB-BC and EBB-BC algorithms. Besides, it is observed that $(SDR)_{ave}$ is always kept at sufficiently high values in both MBB-BC and EBB-BC algorithms, which in turn prevents the search from becoming inefficient or restricted. The EBB-BC algorithm usually provides greater $(SDR)_{ave}$ values in comparison to the MBB-BC. The rate of decrease of $(SDR)_{ave}$ is slower and steadier in EBB-BC algorithm, whereas in the MBB-BC algorithm $(SDR)_{ave}$ is brought down to its minimum value more rapidly and it is practically stabilized around this minimum value thereafter.

Figs 3(b) and 4(b) show that both MBB-BC and EBB-BC algorithms accommodate fairly larger step sizes as compared to the standard algorithm. As the search process goes on, while $(SS)_{ave}$ parameter is strictly reduced to one in the standard algorithm, it takes place in the range of [2, 8] for MBB-BC and of [2, 40] for EBB-BC even in the latest iterations of the optimization. These occasional large step sizes are quite useful for steering the search towards new design regions when the search gets stuck in local optima. This characteristic of the proposed reformulation provide an efficient mechanism to avoid local optima while the standard algorithm is likely to get trapped in local optima.

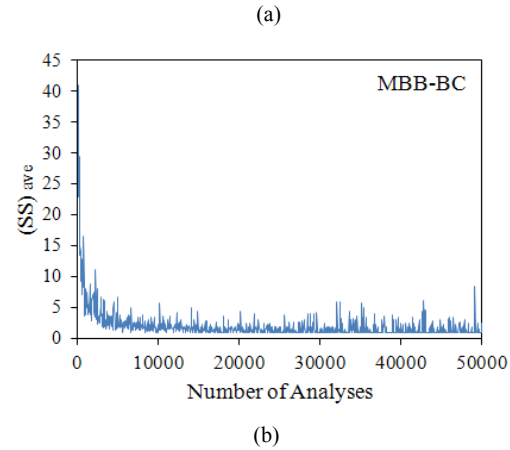
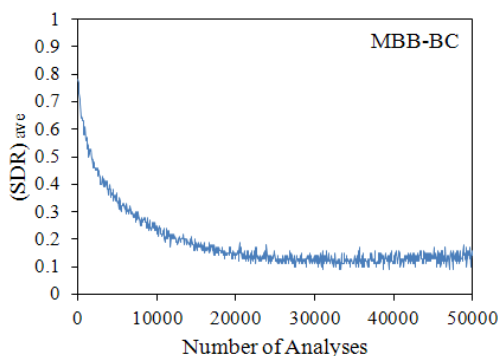


Fig. 3 The variations of (a) $(SDR)_{ave}$ and, (b) $(SS)_{ave}$ in a typical run of MBB-BC

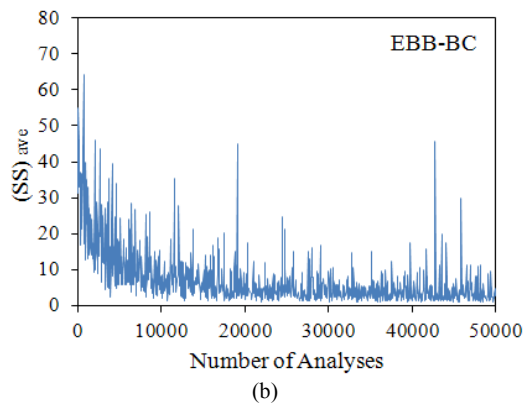
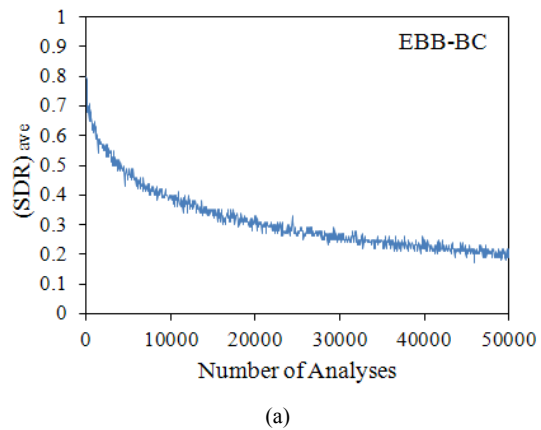


Fig. 4 The variations of (a) $(SDR)_{ave}$ and, (b) $(SS)_{ave}$ in a typical run of EBB-BC

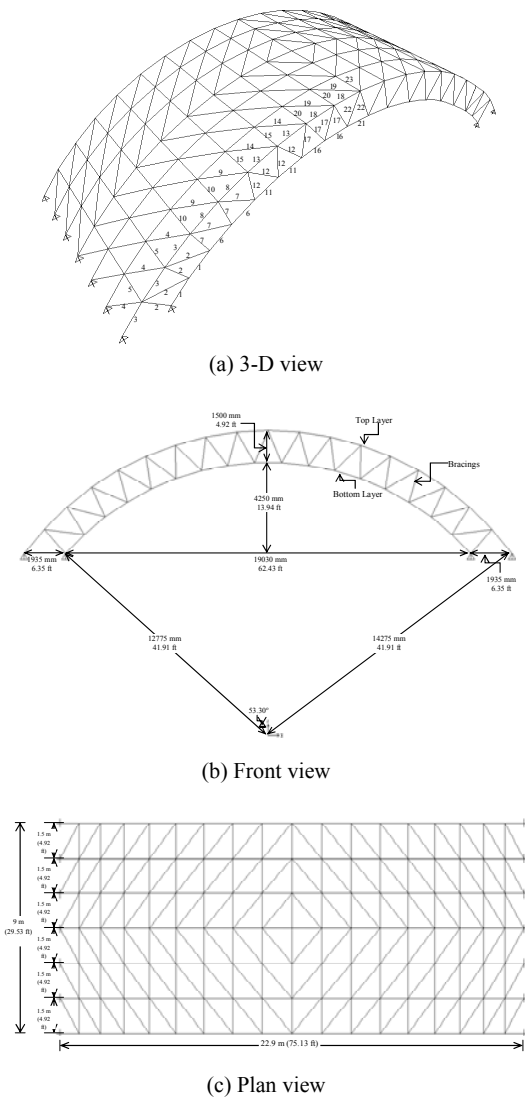


Fig. 5 693-bar braced barrel vault

VI. NUMERICAL EXAMPLES

Several design examples are used to experiment and quantify the performance of MBB-BC in optimum design of steel trusses and MBB-BC algorithms in optimum design of steel frames. The design examples include a 693-bar braced barrel vault (example 1), a 960-bar double layer grid (example 2), a 132-member unbraced steel frame (example 3) and a 209-member industrial factory building (example 4). The optimum solutions produced for these structures with MBB-BC and EBB-BC algorithms are compared to those achieved using the standard version (BB-BC algorithm) as well as other metaheuristic techniques. For a fair comparison of results, the maximum number of structural analyses is limited to the previously reported values in the literature, which is 50,000 for examples 1, 3 and 4, and 100,000 for example 2. For numerical applications, the value of parameter α in Eqs. (12), (18) and (19) is taken as 0.5 for examples 1 and 2, and 0.25

for examples 3 and 4. A population size of 50 is used for all algorithms. The material properties of steel used in all examples are as follows: modulus of elasticity = 29,000 ksi (203,893.6 MPa) and yield stress = 36 ksi (253.1 MPa).

A. 693-Bar Braced Barrel Vault

The first example shown in Fig. 5 is a spatial braced barrel vault [24] consisting of 259 joints and 693 members that are grouped into 23 independent sizing variables considering the symmetry of the structure about the centerline. The member grouping scheme is given in Fig. 5a and the dimensions of the structure are shown in Figs 5b and 5c. It is assumed that the barrel vault is subjected to a uniform dead load (DL) pressure of 35 kg/m², a positive wind load (WL) pressure of 160 kg/m² (32.77 lb/ft²) and a negative wind load (WL) pressure of 240 kg/m² (49.16 lb/ft²). Here, these loads are combined under two separate load cases as follows: (i) $1.5DL+1.5WL = 1.5(35+160) = +292.5$ kg/m² (59.91 lb/ft²) and (ii) $1.5DL-1.5WL = 1.5(35-240) = -307.5$ kg/m² (62.98 lb/ft²), along z-direction. The displacements of all joints in all the x, y, and z directions are limited to a maximum value of 0.254 cm (0.1 in). The strength and stability requirements of steel members are imposed according to AISC-ASD [19]. For design optimization, the structural members are selected from a list of 37 circular hollow sections issued in AISC-ASD [19] code.

In Table I the design optimization results of 693-bar barrel vault obtained using the MBB-BC and BB-BC algorithms are compared to the previously reported results by Hasançebi et al. [25] with different metaheuristic techniques. According to these results, the best solution is attained by the MBB-BC algorithm, which is 4805.96 kg (10595.33 lb). In this example, the BB-BC algorithm shows a more promising performance compared to the other metaheuristic techniques and yields the second best solution, which is 4925.75 kg (10859.42 lb). Amongst the other solutions are 4989.15 kg (10999.20 lb) by ant colony optimization (ACO), 5095.07 kg (11232.71 lb) by harmony search (HS) and 5456.48 kg (12029.49 lb) by simple genetic algorithm (SGA).

B. 960-Bar Double Layer Grid

Fig. 6 shows the second design example which is a double layer grid consisting of 263 joints and 960 members. The symmetry of the structure around x and y axes is used to group the 960 members into 251 independent sizing variables. They are adopted from a database of 28 circular hollow sections in AISC-ASD [19] steel profile list. The lower and upper bounds on sizing variables are set to 1.07 in.² (6.90 cm²) and 21.3 in.² (137.42 cm²), respectively. The structure is subjected to a single load case of snow load under design snow pressure of 0.754 kN/m² (15.75 lb/ft²). The stress and stability limitations of the members are computed in accordance with the provisions of AISC-ASD [19]. Further, the displacements of all nodes are limited to a maximum value of 4.16 in. (10.57 cm) in any direction.

The 960-bar double layer grid is a challenging design

example due to high number of design variables considered. It is noted that only the stress and slenderness ratio constraints are active for this example. The lightest design for the double layer grid system is attained by MBB-BC algorithm, which is

TABLE I
COMPARISON OF RESULTS FOR 693-BAR BRACED BARREL VAULT

Sizing variables	ACO	HS	GA	BB-BC	MBB-BC
1	PXX2.5	PX3.5	PX3	PXX2.5	PX3.5
2	P1	PX.75	P1.25	P1	P1
3	P1	P1	PX1	P.75	P.75
4	P1	P1	P1	P1.25	P1
5	P1	PX.75	P.75	P.75	P.75
6	P.75	P4	PX4	PX3.5	PX3.5
7	PX1	P1.25	PX1	P1	P1
8	P.75	P.75	P.75	P1	P1
9	P3.5	P3.5	PXX2	P1	P1
10	PXX2.5	P1	PX1	P.75	P.75
11	P1	P1.25	P1.25	PX2.5	P3
12	PX1	PX1.25	P1.5	P1.5	P1.5
13	PX1.25	PX1.5	PX1.5	PX1.5	PX1.5
14	PX1	PX1.25	P1.5	P1	P1
15	P.75	P.75	P1	P.75	P.75
16	PX1	PX1.25	P1.25	PX1.5	PX1.25
17	PX1.25	PX1.25	PX1.5	P1.25	P1.25
18	P1	P1.25	P1.5	P2.5	P3
19	P1.25	PX1	P1.25	P1.25	P1
20	P.75	P.75	P.75	P.75	P.75
21	P2.5	P2.5	P3	P1	P1
22	P1.25	P1	P1.25	P1	P.75
23	P1	PX1	PX.75	P.75	P.75
Weight, lb (kg)	10999.20 (4989.15)	11232.71 (5095.07)	12029.49 (5456.48)	10859.42 (4925.75)	10595.33 (4805.96)

24266.7 kg (53498.8 lb). The other designs are 24388.3 kg (53656.7 lb) by simulated annealing (SA), 24780.2 kg (54631.0 lb) by evolution strategies (ESs), 24973.5 kg (55057.1 lb) by particle swarm optimization (PSO), 25320.0 kg (55821.1 lb) by tabu search (TS), 29556.6 kg (65161.2 lb) by ant colony optimization (ACO), 32338.5 kg (71294.2 lb) by simple genetic algorithm (SGA) and 40133.8 kg (88479.9 lb) by harmony search (HS) according to Hasançebi et al. [27]. No feasible solution is obtained with the BB-BC algorithm when the initial population is generated randomly. To facilitate design transitions to feasible regions during the search, the algorithm is started from one feasible design point such that all the member groups are assigned to the strongest section of the discrete set in one individual, while all other individuals in the initial population are created randomly in a

usual manner. The BB-BC algorithm employed under this case produces a final design weight of 31119.8 kg (68607.4 lb).

C.132-Member Unbraced Space Steel Structure

The third design example depicted in Fig. 7 is a three dimensional unbraced (swaying) steel frame composed of 70 joints and 132 members that are grouped into 30 independent sizing variables (Fig. 7b) to satisfy practical fabrication requirements. The columns are adopted from the complete W-shape profile list consisting of 297 ready sections, whereas a discrete set of 171 economical sections selected from W-shape profile list based on area and inertia properties is used to size beam members. Here, both gravity and lateral loads are considered in designing the structure. Gravity loads (G) consisting dead, live and snow loads are calculated according to ASCE 7-05 [26] based on the following design values: a design dead load of 60.13 lb/ft² (2.88 kN/m²), a design live load of 50 lb/ft² (2.39 kN/m²), and a ground snow load of 25 lb/ft² (1.20 kN/m²). This yields the uniformly distributed loads on the outer and inner beams of the roof and floors given in Table 2. As for the lateral forces, earthquake loads (E) are considered. These loads are calculated based on the equivalent lateral force procedure outlined in ASCE 7-05 [26], resulting in the values given in Table 2 that are applied at the center of gravity of each story as joint loads. Gravity (G) and earthquake (E) loads are combined under two loading conditions for the frame: (i) 1.0G + 1.0E (in x-direction), and (ii) 1.0G + 1.0E (in y-direction). The combined stress, stability and geometric constraints are imposed as explained in Section 2. The joint displacements in x and y directions are limited to 1.53 in (3.59 cm) which is obtained as height of frame/400. Additionally, story drift constraints are applied to each story of the frame which is equal to height of each story/400.

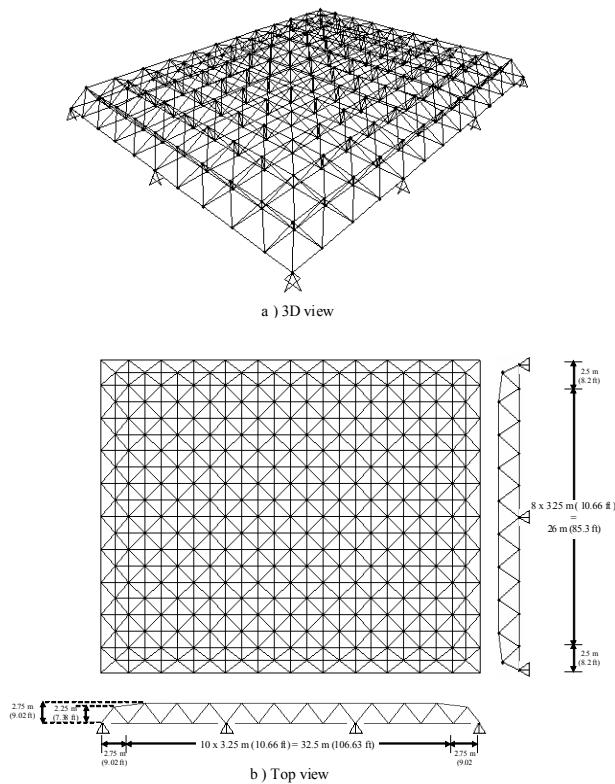


Fig. 6 960-bar double layer grid

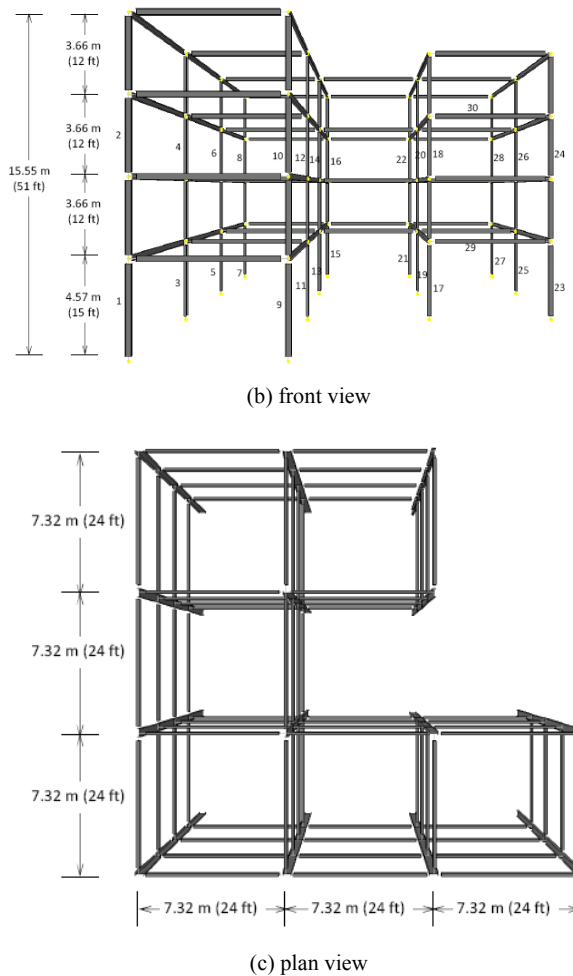
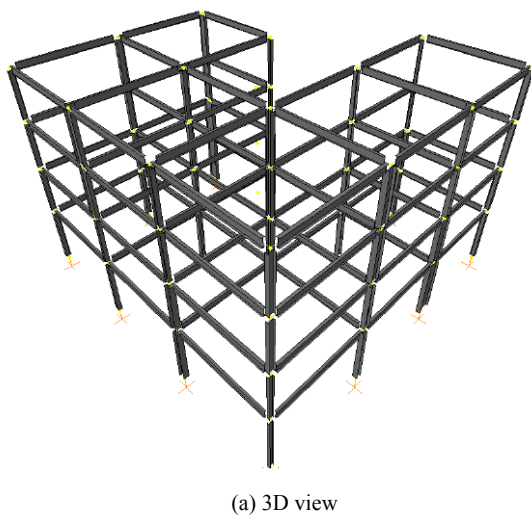


Fig. 7 132-member space steel frame

The BB-BC and EBB-BC algorithms are employed to minimize the weight of the 132-member steel frame. In Table 3 the minimum weight designs of the frame obtained by these algorithms are compared to the previously reported results by Hasançebi et al. [28] using different metaheuristic techniques; namely simulated annealing (SA), tabu search (TS) and harmony search (HS). The EBB-BC algorithm produces a design weight of 60804.31 kg (134050.55 lb) for the frame which is the best solution of this problem reported so far. Relatively higher design weights have been obtained for the frame with other metaheuristic algorithms; namely 62993.55 kg (138874.67 lb) by SA, 64733.69 kg (142710.96) by TS, 64926.17 kg (143135.29) by HS. On the other hand, the BB-BC algorithm exhibits a very poor performance and produces a final design weight of 87468.21 kg (192834.39 lb). Such a significant difference between the results clearly demonstrates the usefulness of the proposed refinement on the performance of the standard algorithm.

TABLE II
THE GRAVITY AND LATERAL LOADING ON 132-MEMBER SPACE STEEL FRAME

Gravity Loads				
Beam Type	Uniformly Distributed Load			
	Outer Span Beams (lb/ft)	(kN/m)	Inner Span Beams (lb/ft)	(kN/m)
Roof beams (Dead + Snow)	1011.74	14.77	1193.84	17.42
Floor beams (Dead + Live)	1468.40	21.49	1732.70	25.29
Lateral Loads				
Floor Number	Earthquake Design Load			
	(kips)	(kN)		
1	6.57	29.23		
2	12.43	55.28		
3	18.52	82.35		
4	24.76	110.15		

D.209-Member Industrial Factory Building

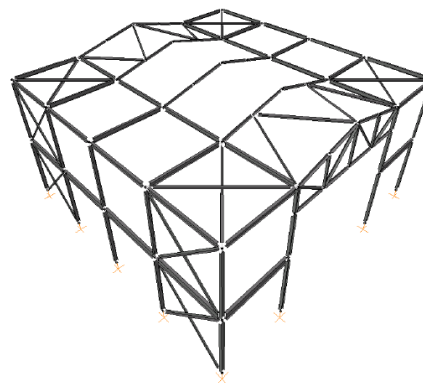
In this example, the design optimization of an industrial factory building (Fig. 8) composed of 100 joints and 209 members is considered. The main system of the structure consists of five identical frameworks lying 6.1 m (20ft) apart from each other in x - z plane. As shown in Fig. 8 (b), each framework includes two side frames and a gable roof truss in between them. The lateral stability of the structure against wind loads in x - z plane is provided through columns fixed at the base as well as the rigid connections of the side frames. Therefore, all the beams and columns in the side frames are designed as moment-resisting axial-flexural members. On the other hand, the gable roof truss is designed to transmit only axial forces through pin-jointed connections. Hence, the web and chord members in the gable roof are all designed as axial members. For longitudinal stability (along y -axis) of the structure, bracing is used in the end bays in the walls and the roof. Considering the symmetry of the structure as well as the fabrication requirements of structural members, the totals of 209 members are collected in 14 member groups (sizing design variables). Fig. 8 (a) gives the member grouping details.

TABLE III
COMPARISON OF RESULTS FOR 132-MEMBER SPACE STEEL FRAME

Sizing vars.	SA	TS	HS	BB-BC	EBB-BC
1	W8X35	W8X31	W14X53	W24X176	W10X33
2	W18X86	W12X65	W12X120	W21X132	W12X79
3	W12X79	W27X129	W30X48	W27X336	W40X167
4	W18X65	W8X58	W16X77	W24X279	W12X65
5	W12X65	W12X79	W18X119	W14X193	W14X120
6	W27X161	W12X106	W24X104	W14X109	W14X109
7	W24X117	W18X97	W30X148	W12X87	W14X99
8	W10X54	W8X58	W10X68	W27X94	W14X90
9	W18X86	W12X72	W18X158	W30X292	W10X100
10	W12X96	W14X90	W12X120	W18X283	W12X106
11	W10X60	W36X135	W36X150	W10X49	W33X152
12	W10X49	W10X49	W16X67	W21X62	W12X53
13	W12X87	W12X96	W10X112	W18X311	W14X90

14	W12X50	W10X49	W24X117	W33X141	W36X160
15	W24X55	W24X55	W18X40	W18X40	W18X40
16	W24X55	W10X33	W14X61	W12X210	W12X53
17	W12X58	W18X76	W12X65	W16X67	W21X111
18	W12X67	W21X83	W18X119	W12X65	W12X65
19	W12X40	W8X40	W14X82	W14X211	W14X43
20	W10X49	W14X61	W18X86	W14X211	W10X60
21	W12X72	W18X76	W14X90	W40X277	W12X106
22	W12X79	W12X72	W18X97	W33X141	W10X88
23	W8X48	W12X40	W21X73	W12X65	W8X48
24	W24X68	W24X76	W12X87	W30X326	W27X84
25	W14X61	W10X77	W18X71	W12X72	W14X61
26	W21X50	W16X50	W27X102	W8X28	W10X39
27	W8X40	W10X49	W8X48	W30X124	W12X40
28	W8X67	W14X61	W24X117	W24X94	W18X76
29	W10X39	W18X97	W18X97	W16X89	W24X68
30	W21X44	W16X45	W16X40	W21X44	W18X40
Weight, lb(kg)	138874.67 (62993.55)	142710.96 (64733.69)	143135.29 (64926.17)	192834.39 (87468.21)	134050.55 (60804.31)

For the design of this industrial building three different types of loads namely dead, crane and wind loads are considered in six load combinations. The details of the loadings considered can be found in [28]. All structural members are sized using the AISC standard sections. Accordingly, the beam and column members are adopted from wide-flange sections (W), and side wall and roof bracings are chosen from back to back equal leg double angle sections. The combined stress, stability and geometric constraints are imposed according to AISC-ASD [19] provisions. Further, displacements of all the joints in x and y directions are limited to 3.43 cm (1.25 in), and the maximum allowable value of inter-story drifts is taken as 1.52 cm (0.6 in).



(a) 3D view

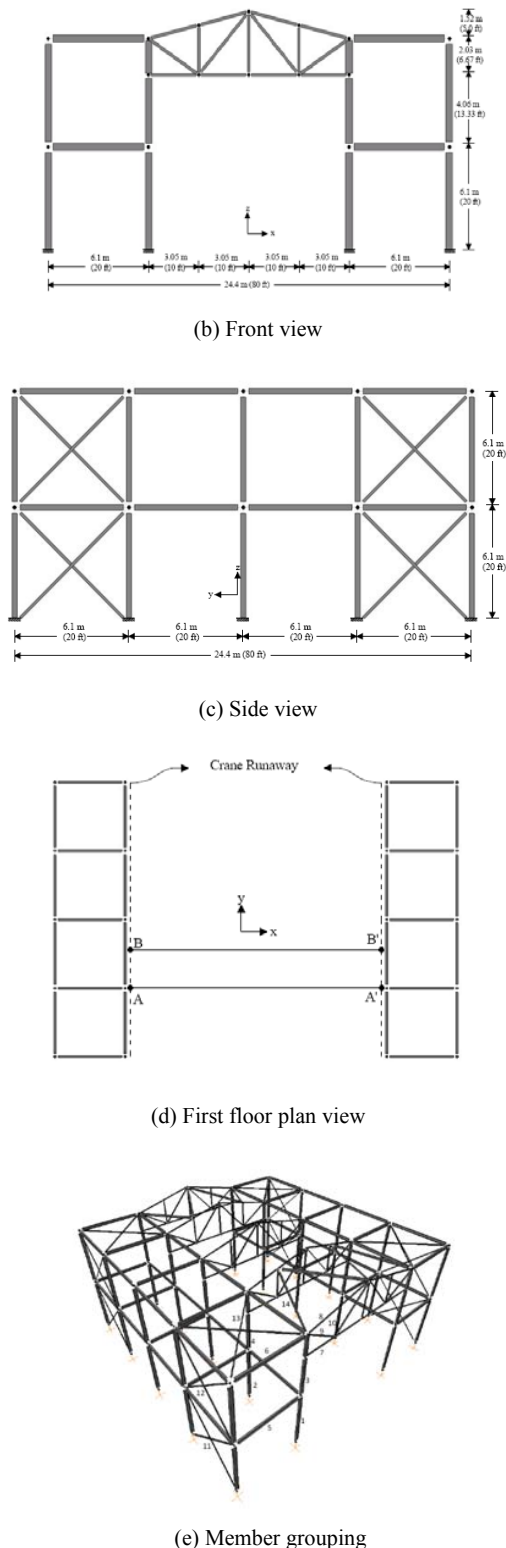


Fig. 8 209-member industrial factory building

TABLE IV
COMPARISON OF RESULTS FOR 209-MEMBER INDUSTRIAL FACTORY BUILDING

Sizing vars.	HS	AHS	BB-BC	EBB-BC
1	W8X31	W8X31	W16X57	W10X33
2	W12X40	W10X39	W16X57	W10X33
3	W8X31	W12X26	W8X28	W8X24
4	W8X40	W8X40	W21X68	W10X33
5	W24X62	W24X62	W24X62	W24X62
6	W12X26	W10X26	W21X44	W12X26
7	2L2.5X2X3/16	2L2X2X1/8	2L5X5X5/8	2L2X2X1/8
8	2L2X2X1/8	2L2X2X1/8	2L2X2X1/8	2L2X2X1/8
9	2L3X3X3/16	2L3X3X3/16	2L4X4X5/8	2L3X3X3/16
10	2L3X2.5X5/16	2L2X2X1/8	2L2.5X2.5X3/16	2L2X2X1/8
11	2L6X6X7/16	2L6X6X5/16	2L6X6X3/4	2L6X6X5/16
12	2L6X6X3/8	2L6X6X5/16	2L8X8X3/4	2L6X6X5/16
13	2L6X6X5/16	2L6X6X5/16	2L6X6X5/8	2L6X6X5/16
14	2L6X6X5/16	2L5X5X5/16	2L5X5X7/16	2L5X5X5/16
Weight, lb (kg)	102924.73 (46685.83)	97121.3 (44053.45)	161764.99 (73375.37)	94631.38 (42924.07)

The BB-BC and EBB-BC algorithms are employed to minimize the weight of the industrial factory building. In Table 4 the minimum weight designs of the structure obtained by these algorithms are compared to the previously reported results by Saka and Hasancebi [28] using harmony search (HS) and its adaptive variant (AHS) techniques. Again the EBB-BC algorithm performs very well and produces the best known solution of the problem, which is 42924.07 kg (94631.38 lb). The final designs attained for this problem with AHS and HS techniques were 44053.45 kg (97121.3 lb) and 46685.83 kg (102924.73 lb), respectively. On the other hand, a substandard performance is exhibited by BB-BC algorithm, in which the structural weight could only be decreased to 73375.37 kg (161764.99 lb).

VII. CONCLUSION

In the present study an easy-to-implement and efficient design optimization algorithm based on a big bang-big crunch algorithm is implemented for discrete sizing optimization of steel structures. Through simple modifications of the standard algorithm (BB-BC) two enhanced variants of the algorithm; namely MBB-BC and EBB-BC are introduced and applied for practical design optimization of steel trusses and frames, respectively. The numerical efficiencies of the proposed algorithms are quantified using four different practical design examples; namely a 693-bar braced barrel vault (example 1), a 960-bar double layer grid (example 2), a 132-member unbraced steel frame (example 3) and a 209-member industrial factory building (example 4). In all the examples, the steel

structures are designed for minimum weight subject to strength and serviceability limitations according to AISC-ASD specification. Based on the results obtained in these examples it is shown that the performance of the BB-BC algorithm can be improved to a great extent with the proposed reformulations. Further, in comparison to the previously reported results with various meta-heuristic techniques, the MBB-BC and EBB-BC algorithms consistently obtained lighter designs, demonstrating the robustness of the optimization process with the latter. The advantages of the proposed algorithms are basically due to their abilities to provide more advantageous mechanism for adjusting search dimensionality ratio when dealing with discrete design optimization problems as well as making it possible to have occasional increments in the step size values throughout the optimization process.

REFERENCES

- [1] F. Erbatur, M.M. Al-Hussainy, "Optimum design of frames", *Comput. Struct.*, 45(5-6), 887-891, 1992.
- [2] E.I. Tabak, P.M. Wright, "Optimality criteria method for building frames", *J. Struct. Div.*, ASCE, 107(7), 1327-1342, 1981.
- [3] M.P. Saka, "Optimum design of steel frames with stability constraints" *Comput. Struct.*, 41(6), 1365-1377, 1991.
- [4] X-S. Yang, "Nature-inspired metaheuristic algorithms" Luniver Press, 2008.
- [5] L. Lamberti, C. Pappalettere, "Metaheuristic design optimization of skeletal structures: a review", *Computational Technology Reviews*, 4, 1-32, 2011. doi:10.4203/ctr.4.1
- [6] M.P. Saka, "Optimum design of steel frames using stochastic search techniques based in natural phenomena: a review", in B.H.V. Topping, (Editor), "Civil Engineering Computations: Tools and Techniques", Saxe-Coburg Publications, Stirlingshire, UK, Chapter 6, pp 105-147, 2007.
- [7] M.P. Saka, E. Doğan, "Recent developments in metaheuristic algorithms: a review" in B.H.V. Topping, (Editor), "Computational Technology Reviews", Saxe-Coburg Publications, Stirlingshire, UK, 31-78, 2012.
- [8] O.K. Erol, I. Eksin, "New optimization method: big bang-big crunch", *Adv. Eng. Software*, 37, 106-111, 2006.
- [9] M.H. Afshar, I. Motaei, "Constrained big bang-big crunch algorithm for optimal solution of large scale reservoir operation problem", *Int. J. Optim. Civil Eng.*, 1(2), 357-375, 2011.
- [10] H. Tang, J. Zhou, S. Xue, L. Xie, "Big bang-big crunch optimization for parameter estimation in structural systems", *Mech. Syst. Signal Proc.*; 24(8), 2888-2897, 2010.
- [11] C.V. Camp, "Design of space trusses using big bang-big crunch optimization", *J. Struct. Eng.*, ASCE, 133, 999-1008, 2007.
- [12] A. Kaveh, H. Abbasgholizadeh, "Optimum design of steel sway frames using big bang-big crunch algorithm", *Asian J. Civ. Eng.*, 12(3), 293-317, 2011.
- [13] L. Lamberti, C. Pappalettere, "A fast big bang-big crunch optimization algorithm for weight minimization of truss structures", in Y. Tsompanakis, B.H.V. Topping, (Editors), *Proceedings of the Second International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering*, Civil-Comp Press, Stirlingshire, Paper 11, 2011.
- [14] A. Kaveh, S. Talatahari, "Size optimization of space trusses using big bang-big crunch algorithm", *Comput. Struct.*, 87(17-18), 1129-1140, 2009.
- [15] A. Kaveh, S. Talatahari, "Optimal design of Schwedler and ribbed domes via hybrid Big Bang-Big Crunch algorithm", *J. Construct. Steel Res.*, 66 (3), 412-419, 2009.
- [16] A. Kaveh, S. Talatahari, "A discrete big bang-big crunch algorithm for optimal design of skeletal structures", *Asian J. Civ. Eng.*, 11(1), 103-123, 2010.
- [17] A. Kaveh, S. Talatahari, R. Sheikholeslami, "Optimum seismic design of gravity retaining walls using the heuristic big bang-big crunch algorithm", in Y. Tsompanakis, B.H.V. Topping, (Editors), *Proceedings of the Second International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering*, Civil-Comp Press, Stirlingshire, UK, Paper 4, 2011.
- [18] S. Kazemzadeh Azad, O. Hasançebi, O.K. Erol, "Evaluating efficiency of big-bang big crunch algorithm in benchmark engineering optimization problems", *Int. J. Optim. Civil. Eng.*, 1(3), 495-505, 2011.
- [19] AISC-ASD, "Manual of steel construction-allowable stress design", 9th ed., Chicago, Illinois, USA, 1989.
- [20] O. Hasançebi, S. Çarbaş, E. Doğan, F. Erdal, M.P. Saka, "Optimum design of real size steel frames using non-deterministic search techniques", *Computers and Structures*, 88 (17-18), 1033-1048, 2010.
- [21] O. Hasançebi, "Adaptive evolution strategies in structural optimization: enhancing their computational performance with applications to large-scale structures", *Comput. Struct.*, 86 (1-2), 119-132, 2008.
- [22] O. Hasançebi, S. Kazemzadeh Azad, "Discrete Size Optimization of Steel Trusses using a Refined Big Bang-Big Crunch Algorithm", *Engineering Optimization*, accepted, 2012.
- [23] O. Hasançebi, S. Kazemzadeh Azad, "An Exponential Big Bang-Big Crunch Algorithm for Discrete Design Optimization of Steel Frames", *Computers and Structures*, accepted, 2012.
- [24] G.S. Ramaswamy, M. Eekhout, G.R. Suresh, "Analysis, design and construction of steel space frames", Thomas Telford Publishing, 2002.
- [25] O. Hasançebi, S. Çarbaş, M.P. Saka, "A reformulation of the ant colony optimization algorithm for large scale structural optimization" In *Proceedings of the Second International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering*, Tsompanakis Y, Topping BHV, (Editors), Civil-Comp Press, Stirlingshire, UK, Paper 12.
- [26] ASCE 7-05, "Minimum design loads for building and other structures", American Society of Civil Engineering, 2005.
- [27] O. Hasançebi, S. Çarbaş, E. Doğan, F. Erdal, M.P. Saka, "Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures", *Comput Struct*, 87(5-6):284-302, 2009.
- [28] O. Hasançebi, S. Çarbaş, M.P. Saka, Improving the performance of simulated annealing in structural optimization, *Struct. Multidisc. Optim.*, 41, 189-203, 2010.