The 4th International Conference
″Advanced Composite Materials Engineering ″
COMAT 2012
18- 20 October 2012, Brasov, Romania

WEIGHT MINIMIZATION OF TRUSS STRUCTURES WITH BIG BANG–BIG CRUNCH

C. Casavola, L. Lamberti, C. I. Pruncu

Dipartimento di Meccanica, Management e Matematica
Politecnico di Bari, Viale Japigia 182 - 70126, Bari, Italy
casavola@poliba.it; lamberti@poliba.it; c.pruncu@poliba.it

Abstract: *The Big Bang −Big Crunch (BB−BC) optimization method is a recently developed meta-heuristic algorithm that mimics the process of evolution of the universe. BB−BC has been proven very efficient in design optimization of skeletal structures but yet computationally more expensive than classical meta-heuristic algorithms such as genetic algorithms and simulated annealing. To overcome this limitation, the paper presents a novel hybrid formulation of BB−BC where the meta-heuristic search is hybridized by including gradient/pseudo-gradient information as a criterion to perform new explosions. Each new trial design is formed by combining a set of descent directions and eventually corrected in order to improve it further. The new BB−BC algorithm is successfully tested in two classical weight minimization problems of a spatial 25-bar truss and a planar 200-bar truss.*
Keywords: *Truss structures; Weight optimization; Big Bang–Big Crunch Optimization; Hybrid algorithms.*

1. INTRODUCTION

Weight minimization of truss structures is an important engineering field under continuous development. Random search allows to explore a larger fraction of the design space than in the case of gradient-based optimization. In order to rationalize the search process and approach the region hosting the global optimum by performing only a limited number of structural analyses, a variety of meta-heuristic optimization methods inspired by biology, evolution theory, social sciences, music, physics and astronomy were developed [1-2].
Genetic Algorithms (GA) and Simulated Annealing (SA) were the first meta-heuristic optimization methods to be applied in structural design problems and are still widely utilized nowadays. The second generation of meta-heuristic algorithms released in the last 10 years includes population-based methods such as Particle Swarm Optimization, Ant Colony Optimization, Harmony Search and Big Bang−Big Crunch. In particular, Big Bang−Big Crunch (BB−BC) [3] is one of the most recently developed meta-heuristic algorithms (the first paper on this subject was published in 2006). BB-BC reproduces the evolution of the universe: each explosion generates a state of chaos which is followed by a state of order that will last until the next explosion. In the optimization process, a set of candidate designs is randomly generated over design space (i.e. “explosion phase”). The centre of mass of these designs is determined as a weighted average (i.e. “contraction phase”) where each weighing coefficient depends on the value of cost function evaluated for a trial design. A new population is generated randomly by perturbing optimization variables in the neighborhood of center of mass. The explosion/contraction sequence is repeated until convergence.
The inherent simplicity of BB−BC soon attracted structural optimization experts and several examples of application to design optimization of skeletal structures are documented in literature [4-7]. However, BB−BC implementations have a common feature in the fact that new explosions about the center of mass are always performed after that the position of center of mass has been updated. This entails N_{POP} new structural analyses in each new explosion. Furthermore, there is no guarantee that each newly defined center of mass will always lead to improve the current best record. Starting from these considerations, the present authors developed a novel BB−BC formulation including gradient information as a criterion to generate new trial designs and perform new explosions [8]. The above mentioned formulation is further improved in this study and successfully tested in two classical weight minimization problems of a spatial 25-bar truss and a planar 200-bar truss under multiple loading conditions.

2. WEIGHT MINIMIZATION PROBLEM FOR A TRUSS STRUCTURE

The weight minimization problem for a truss structure comprised of NOD nodes (k=1,…,NOD) and NEL elements (j=1,…,NEL) can be stated as follows:

$$\text{Minimize} \quad W(\mathbf{X}) = \rho g \sum_{j=1}^{NEL} l_j \, x_j$$

$$\text{Subject to} \begin{cases} u^L_{(x,y,z),k} \le u_{(x,y,z),k,ilc} \le u^U_{(x,y,z),k} \\ \sigma^L_j \le \sigma_{j,ilc} \le \sigma^U_j \\ x^L_j \le x_j \le x^U_j \end{cases} \tag{1}$$

where:

- x_j is the cross-sectional area of the j^{th} element of the structure included as sizing variable in the optimization process: each sizing variable can range between the corresponding lower bound x_j^L and upper bound x_j^U;
- l_j is the length of the j^{th} element of the structure;
- g is the gravity acceleration value (9.81 m/s^2); ρ is the material density;
- NLC is the number of independent loading conditions acting on the structure;
- $u_{(x,y,z),k,ilc}$ are the displacements of the k^{th} node in the directions x,y,z, varying between limits $u^L{}_{(x,y,z),k}$ and $u^U{}_{(x,y,z),k}$;
- $\sigma_{j,ilc}$ is the stress in the j^{th} element, varying between the limits σ_j^L (compression stress limit may include critical bucking load) and σ_j^U (tensile stress limit);
- The ilc subscript (ilc=1,…,NLC) refers to the ilc^{th} loading condition. All nonlinear constraints are normalized with respect to displacement and stress limits.

For optimization problems including also nodal coordinates as lay-out variables, Eq. (1) can be rewritten as:

$$\text{Minimize} \quad W(\mathbf{X}) = \rho g \sum_{j=1}^{NEL} x_j \sqrt{(x_{j1} - x_{j2})^2 + (y_{j1} - y_{j2})^2 + (z_{j1} - z_{j2})^2} \tag{2}$$

where $x_{j1,2}$, $y_{j1,2}$, $z_{j1,2}$ are the coordinates of the nodes limiting the j^{th} element of the structure.

Based on the degree of structural symmetry, variable linking can be adopted by grouping the NEL elements in NGR groups: each group includes elements with identical stiffness (i.e. cross-sectional area) properties. This approach allows the number of design variables to be reduced thus simplifying the optimization process.

3. DESCRIPTION OF THE NEW BIG BANG–BIG CRUNCH FORMULATION

There is only one parameter that must be initialized in the Big Bang–Big Crunch algorithm: the number of trial designs N_{POP} included in the population. Let $\mathbf{X}^k(x_1{}^k, x_2{}^k, \dots, x_{NDV}{}^k)$ be the k^{th} design included in the population (k=1,…,N_{POP}). The following generation scheme is utilized to create the initial population for an optimization problem including NDV design variables (j=1,…,NDV):

$$x_j{}^k = x_j{}^L + \rho_{DG,j}{}^k \cdot \left(x_j{}^U - x_j{}^L\right) \tag{3}$$

where $\rho_{DG,j}{}^k$ is the random number in the interval (0,1) for the j^{th} optimization variable considered in the k^{th} generation.

The center of mass $\mathbf{X_{CM}}$ of the initial population, randomly generated over the entire search space with an explosion, is defined as in the classical BB−BC implementations:

$$x_{CM,j} = \left(\sum_{k=1}^{NPOP} \frac{x_j{}^k}{W^k} \right) \Big/ \left(\sum_{k=1}^{NPOP} \frac{1}{W^k} \right) \tag{4}$$

where W^k is the cost function value computed for the k^{th} candidate design.

The definition of the center of mass (4) implies the centre of mass always be at least better than the worst design included in the population. Therefore, two possible cases should be considered to cover any possible scenario: (i) the center of mass is better than the current best record, that is: $W(\mathbf{X_{CM}})<W(\mathbf{X_{OPT}})$; (ii) the center of mass is worse than the current best record, that is: $W(\mathbf{X_{CM}})>W(\mathbf{X_{OPT}})$.

If $W(\mathbf{X_{CM}})<W(\mathbf{X_{OPT}})$ and $\mathbf{X_{CM}}$ is feasible, the new BB−BC algorithm simply replaces the worst design of the population with the center of mass without performing any new explosion. The position of the center of mass is updated until the condition $W(\mathbf{X_{CM}})<W(\mathbf{X_{OPT}})$ remains satisfied and $\mathbf{X_{CM}}$ is feasible. The current best record $\mathbf{X_{OPT}}$ changes with respect to the previous iteration as it coincides with the new center of mass. However, the condition $W(\mathbf{X_{CM}})<W(\mathbf{X_{OPT}})$ is not very likely to occur as the quality of the design corresponding to the center of mass accounts also for the presence

of sub-optimal designs. Two strategies for updating design variables were originally developed in [8] to deal with the case $W(\mathbf{X_{CM}})>W(\mathbf{X_{OPT}})$.

The first design updating strategy (indicated as "Variant 1" in Ref. [8]) is similar to that implemented in the "global annealing" scheme described in [9]. The cost function gradient $\nabla W(\mathbf{X_{CM}})$ is computed at the center of mass $\mathbf{X_{CM}}$. Each optimization variable x_j is randomly perturbed so to have $(\partial W/\partial x_j)\Delta x_j<0$. That is:

$$\begin{aligned}\partial W/\partial x_j>0 &\Rightarrow x_{TR,i}=0.5\cdot(x_{CM,j}+x_{OPT,j})-(x_j^U-x_j^L)\cdot N_{RND,j}\cdot\mu_j\cdot W_{OPT,l-1}/W_{OPT,l}\\ \partial W/\partial x_j<0 &\Rightarrow x_{TR,i}=0.5\cdot(x_{CM,j}+x_{OPT,j})+(x_j^U-x_j^L)\cdot N_{RND,j}\cdot\mu_j\cdot W_{OPT,l-1}/W_{OPT,l}\end{aligned} \quad (5)$$

where: the random numbers $N_{RND,j}$ are chosen in the interval (0,1) for each design variable (j=1,…,NDV); each weighting coefficient μ_j is defined as $|\partial W/\partial x_j|/\|\nabla W(X_{CM})\|$; $W_{OPT,l}$ and $W_{OPT,l-1}$ are the last two current best record values taken by the cost function.

Equation (5) shows that the new trial design X_{TR} is generated by perturbing optimization variables with respect to the middle point X_{MID} of the segment limited by the current center of mass X_{CM} and the current best record X_{OPT}.

In truss sizing optimization problems, the search direction $\mathbf{S_{TR}}(\Delta x_1,\Delta x_2,\ldots,\Delta x_{NDV})$ defined with Eq. (5) is always a descent direction because cost function gradients are constant over design space as cost function is linear with respect to design variables. The direction $[\mathbf{X_{OPT}}-\mathbf{X_{CM}}]$ also is a descent direction containing the middle point $\mathbf{X_{MID}}$. Therefore, there is a high probability of improving design as the optimization variables are being perturbed in the neighborhood of the current best record by moving along the descent direction $\mathbf{S_{TR}}$. Perturbation of optimization variables with respect to $\mathbf{X_{MID}}$ allows the search process to be maintained close enough to the current best record but, at the same time, far enough from constraint domain boundaries. The latter can reduce the risk of generating infeasible trial points when the search process is converging to the optimum design.

Weighting coefficients μ_j also are constant over design space: this may lead to generate always the same movements if the sequence of random numbers is repeated in the subsequent iterations. The ratio $W_{OPT,l-1}/W_{OPT,l}$ accounts for the current trend taken by the cost function in the optimization process thus forcing the optimizer to take movements large enough to maintain at least the current rate of reduction in cost function.

The second strategy for updating design variables ("Variant 2" in Ref. [8]) deals with the more general case of cost function gradients not available explicitly and resembles the novel harmony search scheme developed in [10].

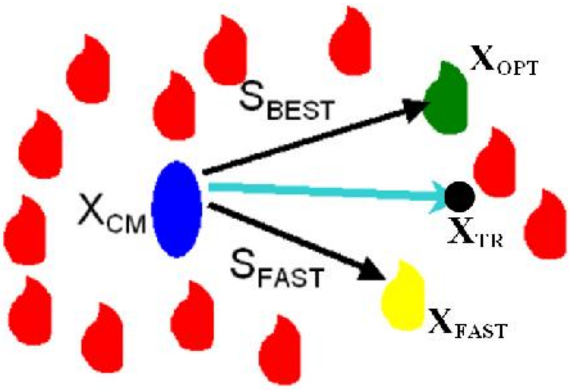

Figure 1. Generation of a new trial design in BB−BC "Variant 2" of Ref. [8] and further improved in this study

Variant 2 is illustrated by Figure 1. Descent directions $\mathbf{S_{BEST}}$ and $\mathbf{S_{FAST}}$ correspond to the largest reduction in weight and to the steepest reduction in weight with respect to the center of mass $\mathbf{X_{CM}}$. Let us assume that there are N_{better} candidate designs that are actually better than the center of mass. The descent direction $\mathbf{S}^k=[\mathbf{X_{better}}^k-\mathbf{X_{CM}}]$ (k=1,...,N_{better}) can be defined: the quantity $\Delta W^k=[W(\mathbf{X_{CM}})-W(\mathbf{X_{better}}^k)]$ represents the reduction in cost that would be achieved by moving away from the centre of mass towards the candidate design $\mathbf{X_{better}}^k$ included in the population. The new trial design $\mathbf{X_{TR}}$ is hence generated as follows:

$$X_{TR}=X_{CM}+\rho_{FAST}S_{FAST}+\rho_{BEST}S_{BEST} \quad (6)$$

where ρ_{FAST} and ρ_{BEST} are two random numbers in the interval (0,1) generated for S_{FAST} and S_{BEST}, respectively. However, the random nature of Eq. (6) may cause some component of the internal product $[X_{TR}-X_{CM}]^T\nabla W(X_{OPT})$ to be positive in sign since S_{FAST} and S_{BEST} are descent directions with respect to the center of mass but not necessarily also with respect to the gradient of cost function $\nabla W(X_{OPT})$. For this reason, the search vector $[X_{TR}-X_{CM}]$ obtained from Eq. (6) was translated to the position of the current best record. Hence, Eq. (6) was rewritten as:

$$\left(X_{TR}\right)_j = X_{OPT,j} + \text{sign}\cdot[\rho_{FAST} S_{FAST} + \rho_{BEST} S_{BEST}]_j \tag{7}$$

if, for the j[th] design variable, the corresponding component of the $[X_{TR}-X_{CM}]$ vector does not lie on a descent direction with respect to $\overline{\nabla}W$ (X_{OPT}). Therefore, the "sign" parameter was set equal to 1 if the corresponding j[th] component of the internal product yields $[X_{TR}-X_{CM}]^T\,\overline{\nabla}W\,(X_{OPT})<0$, or equal to −1 if $[X_{TR}-X_{CM}]^T\,\overline{\nabla}W\,(X_{OPT})>0$.
The quality of the new trial design X_{TR} is evaluated through calculation of structural weight and constraint margins. If the correction strategy (7) is utilized, the j[th] term of the internal product $[X_{TR}-X_{CM}]^T\,\overline{\nabla}W\,(X_{OPT})$ represents the actual change in structural weight for sizing optimization problems of truss structures as their cost function is linear in sizing.

Unlike classical BB−BC, no explosions are performed until each new trial design keeps improving the current best record. The center of mass is updated with Eq. (4) as the new best record replaces the worst design of the population. Since new explosions are performed only if all improvement routines failed, the BB−BC formulation [8] allowed the number of structural analyses to be considerably reduced with respect to classical BB−BC: at least by N_{POP} analyses for each optimization iteration. Remarkably, no information on constraint gradients was required in the optimization process if design variables are perturbed with Eqs. (6-7). The optimization algorithm hence becomes a BB−BC scheme with infrequent explosions. Approximate line searches might also be performed if the center of mass $\mathbf{X_{CM}}$ ends up infeasible but the corresponding structural weight/cost is better than for any design included in the population. However, such an eventuality is highly improbable if one considers the definition of center of mass of Eq. (4).

In the present research, Variants 1 and 2 have been merged to generate the new trial design vector $\mathbf{X}_{TR}$ as follows (j=1,…,NDV):

$$x_{TR,j} = \frac{x_{CM,j} + x_{OPT,j}}{2} + \rho_{FAST}(x_{FAST,j} - x_{CM,j}) + \rho_{2ndBEST}(x_{2ndbest,j} - x_{CM,j}) + \rho_{BEST}(x_{OPT,j} - x_{CM,j})\mu_j \tag{8}$$

Therefore, the new BB−BC optimizer is now forced to generate a new trial design taking information from several "good" regions of design space. In fact, the new design is formed by combining four descent directions instead of just two directions such as it happened for the original BB−BC formulation of Ref. [8]. The "mirroring" of the trial design about the current best record is utilized to steer back the search process to a descent direction:

$$X_{add}' = 2\cdot X_{OPT} - \eta_{MIRR}\cdot X_{TR} \tag{9}$$

In the above equation, the random number η_{MIRR} limits the step size of the j[th] variable thus reducing the risk that the corrected design will turn infeasible if it tends to reduce cost function too sharply.

In the present formulation, each new explosion is performed about the middle point X_{MID} between the current best record and the current position of the center of mass while in Ref. [8] each new explosion was performed about the current position of the center of mass, following the classical BB−BC optimization scheme. This elitist strategy allows the search process to better explore the neighborhood of the current best record thus limiting the effect of the presence of less efficient designs intrinsically considered by the definition of center of mass. New explosions are now performed as follows (j=1,…,NDV; k=1,…,N_{POP}):

$$\begin{cases} \rho_{DG,j}{}^k > 0.5 \Rightarrow x_j{}^k = \dfrac{\left(x_{CM,j} + x_{OPT,j}\right)}{2} + (\rho_{DG,j}{}^k - 0.5)\cdot\left[x_j{}^U - \dfrac{\left(x_{CM,j} + x_{OPT,j}\right)}{2}\right] \\ \rho_{DG,j}{}^k < 0.5 \Rightarrow x_j{}^k = \dfrac{\left(x_{CM,j} + x_{OPT,j}\right)}{2} + (\rho_{DG,j}{}^k - 0.5)\cdot\left[\dfrac{\left(x_{CM,j} + x_{OPT,j}\right)}{2} - x_j{}^L\right] \end{cases} \tag{10}$$

An interesting feature should be underlined. Unlike classical BB−BC, perturbation steps given to design variables in each new explosion by using Eq. (10) are not necessarily shrunk as the optimization process progresses. This allows the optimization search not to be confined in regions of the design space containing only local minima and hence enhances the meta-heuristic algorithm's capability to explore the entire design space.

4. TEST PROBLEMS

The new BB-BC algorithm described in this paper is tested in two classical weight minimization problems of truss structures. The first test case is the optimization of the spatial 25-bar truss structure shown in Figure 2a; the structure has 10 nodes. The Young modulus of the material is 68.971 GPa while the mass density is 2767.991 kg/m^3. The structure is optimized with 8 sizing variables corresponding to the cross-sectional area of each group of elements in which the structure can be divided in view of structural symmetry (see Ref. [9] for more details). The structure must carry two independent loading conditions: (i) 4.45 kN acting in the positive X-direction at node 1; 44,5 kN acting in the positive Y-direction and 22.75 kN acting in the negative Z-direction at nodes 1 and 2; 2.28 kN acting in the positive X-direction at nodes 3 and 6; (ii) 89 kN acting in the positive Y-direction at node 1 and in the negative Y-direction at node 2; 22.75 kN acting in the negative Z-direction at nodes 1 and 2.

There are 124 nonlinear optimization constraints on nodal displacements and member stresses. Displacements of the top nodes 1 and 2 in the coordinate directions X, Y and Z must be less than ±0.00889 m (±0.35 in). The stress limit in compression accounts also for buckling strength and hence is different for each group of elements [9]. The stress limit in tension is uniform and equal to 275.89 MPa (40,000 psi). The minimum area gauge is 0.064516 cm^2 (0.01 in^2).

a) b)

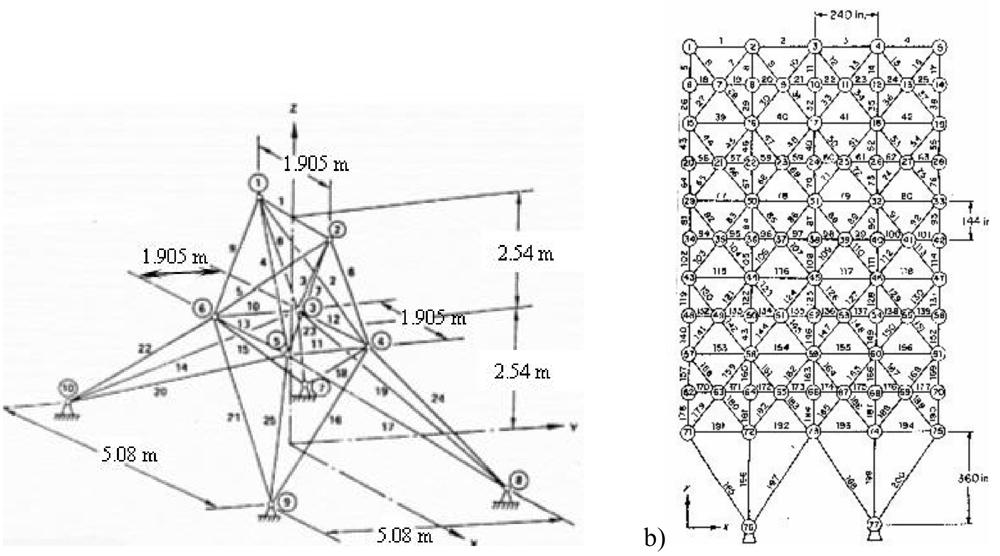

Figure 2. Schematic of the truss structures optimized in this study.

The second test case is the optimization of the planar 200-bar truss structure shown in Figure 2b; the structure has 77 nodes. The Young's modulus of the material is 206.91 GPa while the mass density is 7833.413 kg/m^3. The structure is optimized with 29 sizing variables corresponding to the cross-sectional area of each group of elements in which the structure can be divided from structural symmetry (see Ref. [9] for more details). The structure is subject to three independent loading conditions:

a) 4.45 kN (i.e. 1000 lbf) acting in the positive X-direction at nodes 1, 6, 15, 20, 29, 34, 43, 48, 57, 62 and 71;

b) 44.5 kN (i.e. 10000 lbf) acting in the negative Y-direction at nodes 1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 15, 16, 17, 18, 19, 20, 22, 24, 26, 28, 29, 30, 31, 32, 33, 34, 36, 38, 40, 42, 43, 44, 45, 46, 47, 48, 50, 52, 54, 56, 57, 58, 59, 60,61,62,64, 66, 68, 70, 71, 72, 73, 74, 75;

c) Loading conditions a) and b) acting together.

The optimization must be performed with 1200 non-linear constraints on member stresses. The allowable stress (the same in tension and compression) is 68.97 MPa (i.e. 10000 psi). Cross-sectional areas can vary between 0.64516 cm^2 (i.e. 0.1 in^2) and 645.16 cm^2 (i.e. 100 in^2).

5. RESULTS AND DISCUSSION

The new Big Bang−Big Crunch algorithm developed in this research was implemented in Fortran 90. Optimizations were performed on a standard VAIO laptop equipped with a 2.67 GHz Intel Pentium® I5 processor and 4 GB of RAM memory. The test cases considered in this study are indicative of the ability of the optimizer to find the global optimum as the existence of local minima has been documented in literature (see, for example, Ref. [9]).

The present algorithm was compared with the Big Bang−Big Crunch formulations implemented by Camp [4], Kaveh and Talatahari [5] and two of the present authors [8], and to other state-of-the-art optimization algorithms like basic harmony search developed by Lee and Geem [11], improved harmony search developed by Lamberti and Pappalettere [10], self adaptive harmony search developed by Degertekin [13], heuristic particle swarm optimization (HPSO) developed by Li *et al*. [12], and multi-level and multi-point simulated annealing algorithm [9].

In the 25-bar truss problem, the size of the population N_{POP} was set as 20. Different population sizes ranging between 10 and 200 candidate designs were instead considered for the 200-bar truss problem to investigate the sensitivity of the new BB−BC algorithm to the number of candidate designs.

Table 1 presents the optimization results obtained in the 25-bar truss problem. The BB−BC algorithm described in this paper found the lightest weight overall but the optimized design violated displacement constraints. However, constraint violation is much smaller than for the BB-BC algorithm developed by Kaveh and Talatahari [5]: 0.05% vs.

0.206%. By including this constraint violation as weight penalty, the optimized weight raises to 247.383 kg which is absolutely competitive with the structural weight obtained by Camp [4].

The present BB–BC algorithm is slightly less efficient than the BB–BC formulation presented in Ref. [8] as it performed more explosions (i.e. 8 vs. 3) and converged to a slightly infeasible design. However, in spite of having required more than two times the number of explosions of [8], the present BB–BC formulation required about the same number of structural analyses to complete the optimization process. Therefore, the present algorithm is absolutely competitive with that of Ref. [8]. Furthermore, convergence curves plotted in Figure 3 indicate that the present BB–BC algorithm reduced the truss weight more quickly than the other meta-heuristic algorithms taken as basis of comparison.

Table 1. Optimization results obtained for the spatial 25-bar truss structure

Design variables	BB-BC (Present)	BB-BC [8] (Average)	BB-BC [4]	BB-BC [5]	SA [9]	Improved HS [10]	HPSO [12]
1	0.0100	0.0100	0.010	0.010	0.0100	0.0100	0.010
2	2.0738	1.9871	2.092	1.993	1.9870	1.9871	1.970
3	2.8665	2.9934	2.964	3.056	2.9935	2.9935	3.016
4	0.0100	0.0100	0.010	0.010	0.0100	0.0100	0.010
5	0.0100	0.0100	0.010	0.010	0.0100	0.0100	0.010
6	0.6761	0.6839	0.689	0.665	0.6840	0.6839	0.694
7	1.6637	1.6769	1.601	1.642	1.6769	1.6769	1.681
8	2.7067	2.6623	2.686	2.679	2.6621	2.6622	2.643
Structural weight (kg)	247.259	247.282	247.380	247.280	247.281	247.282	247.294
Number of structural analyses	593 (8)	*Var. 1*: 582 (3) *Var. 2*: 503 (3)	20566	12500	400	1050	750
Constraint tolerance (%)	0.050	None	None	0.206	None	None	None

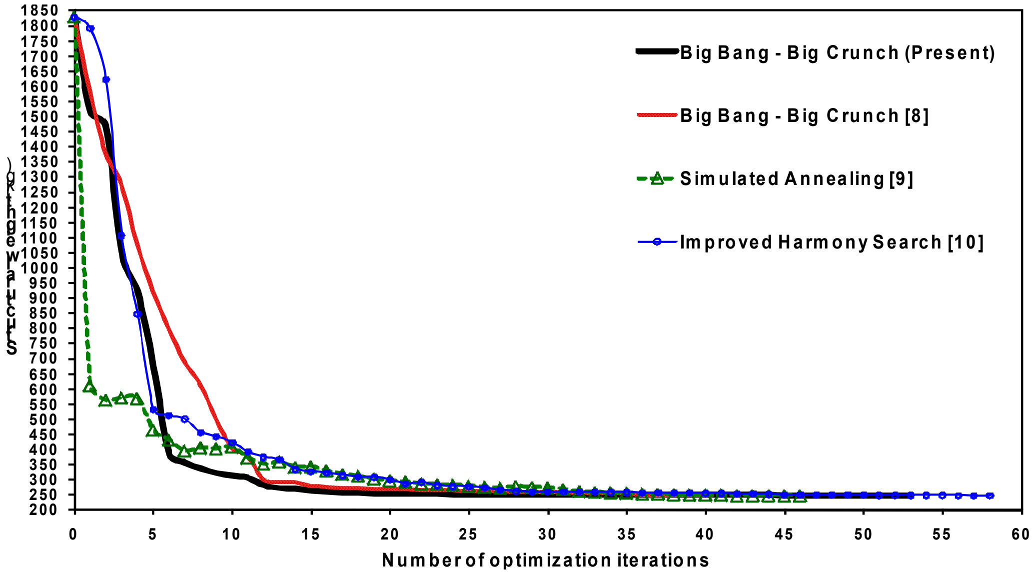

Figure 3. Convergence curves for the spatial 25-bar truss problem

The optimization results obtained for the 200-bar truss structure are listed in Table 2. It can be seen that the average optimized weight found by the present BB–BC algorithm practically coincides with the target optimum weight quoted in literature: 11542.417 kg vs. 11542.409 kg. The present BB–BC formulation outperformed the improved BB–BC algorithm [8] from which it was originated. In fact, the optimized designs obtained in the present study are practically insensitive to the choice of initial population and the optimization algorithm never got trapped in the local minimum of 11544 kg which was instead found in [8] by setting N_{POP}=20 and N_{POP}=50. However, optimization constraints are slightly violated in the present study while they were fully satisfied in [8].

The present algorithm required much more optimization iterations (i.e., on average 103 vs. 37) and structural analyses (i.e. on average, 2593 vs. 983) than the original formulation reported in [8]. However, the number of structural analyses per optimization cycle required in the present case (i.e. about 25) is practically the same as in Ref. [8]. This leads to conclude that the present search mechanism has the same computational cost as the search mechanism employed in [8] but is considerably more efficient as it never got trapped in the local minimum of 11544 kg.

The new BB−BC algorithm is superior over classical harmony search [11] because it converged to a slightly better design (i.e. on average, 11542.417 kg vs. 11542.565 kg), required much less structural analyses to complete the optimization process (i.e. on average, 2593 vs. 48000) and the optimized design violated much less stress constraints (i.e. on average, 0.0180% vs. 3.69%). The present algorithm also outperformed the self adaptive HS algorithm developed very recently by Degertekin [13]: in fact, the average weight of 11542.417 kg is lower than the best weight of 11562.931 kg quoted in [13], and the optimization process required on average about 2600 structural analyses vs. about 19700 analyses required in [13].

The SA optimum design reported in Table 2 corresponds to the average of the optimum designs quoted in [8] which were obtained by starting the optimization process from three different initial points. The slightly lighter weight found by SA is however associated to a larger constraint violation. As far as it concerns the computational cost of the optimization, the present BB−BC algorithm required about 25% more design cycles than SA but much less structural analyses to converge to the optimum design.

Table 2. Summary of optimization results obtained for the planar 200-bar truss structure

Design variables	**Present BB-BC Average**	**BB-BC** [8] **Average**	**SA** [9] **Average**	**Standard HS** [11]	**Self adaptive HS** [13]
1	0.1460	0.1437	0.1467	0.1253	0.154
2	0.9401	0.9400	0.9400	1.0157	0.941
3	0.1007	0.1000	0.1000	0.1069	0.100
4	0.1002	0.1000	0.1000	0.1096	0.100
5	1.9425	1.9400	1.9400	1.9369	1.942
6	0.2971	0.2945	0.2962	0.2686	0.301
7	0.1002	0.1000	0.1000	0.1042	0.100
8	3.1110	3.1022	3.1041	2.9731	3.108
9	0.1002	0.1000	0.1000	0.1309	0.100
10	4.1117	4.0707	4.1041	4.1831	4.106
11	0.4016	0.3979	0.4034	0.3967	0.409
12	0.1722	0.1980	0.1922	0.4416	0.191
13	5.4137	5.3894	5.4283	5.1873	5.428
14	0.1002	0.1000	0.1000	0.1912	0.100
15	6.4173	6.3894	6.4283	6.2410	6.427
16	0.5643	0.5748	0.5737	0.6994	0.581
17	0.2672	0.3093	0.1326	0.1158	0.151
18	7.9787	7.8150	7.9723	7.7643	7.973
19	0.1154	0.1000	0.1000	0.1000	0.100
20	8.9745	8.8150	8.9723	8.8279	8.974
21	0.7879	0.8026	0.7048	0.6986	0.719
22	0.3114	0.1413	0.4199	1.5563	0.422
23	10.9761	10.9546	10.8656	10.9806	10.892
24	0.1002	0.1000	0.1000	0.1317	0.100
25	11.9746	11.9546	11.8646	12.1492	11.887
26	0.9996	0.8877	1.0342	1.6373	1.040
27	6.4952	6.7895	6.6831	5.0032	6.646
28	10.7175	10.8818	10.8093	9.3545	10.804
29	13.9036	13.7489	13.8324	15.0919	13.870
Structural weight (kg)	11542.417 (±0.00532)	11543.205 (±0.919)	11542.319 (±0.164)	11542.565	11562.931
Optimization iterations	103 (±8)	37 (±6)	82 (±1)	N/A	N/A
Structural analyses	2593 (±1246)	983 (±232)	9650 (±1050)	48000	19670
Constraint tolerance (%)	0.0180 (±0.00131)	None	0.0662 (±0.00503)	3.69	None

The convergence curves plotted in Figure 4 indicate that the present algorithm is competitive with SA but slower than the original BB-BC formulation of Ref. [8] which however often missed the global optimum design of 11542.4 kg.

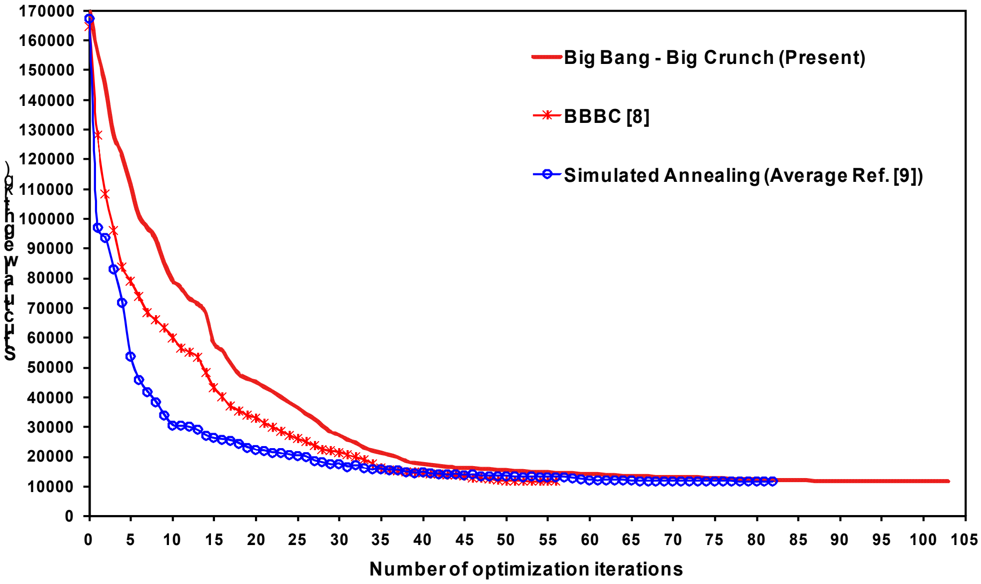

Figure 4. Convergence curves for the planar 200-bar truss problem

6. SUMMARY AND CONCLUSIONS

This paper presented a novel formulation of the Big Bang-Big Crunch meta-heuristic optimization algorithm. Numerical results obtained in two weight minimization problems of truss structures subject to constraints on nodal displacements and member stresses proved that the new algorithm is superior over other advanced meta-heuristic algorithms recently published in literature. Future investigations should be aimed at improving convergence speed as well as to enhance global optimization capability.

REFERENCES

1. M.P. Saka. Optimum design of steel frames using stochastic search techniques based on natural phenomena: a review. In: *Civil Engineering Computations: Tools and Techniques* (B.H.V. Topping, Ed.), Chapter 6, pp. 105–147. Saxe-Coburg Publications, Stirling (UK), 2007.
2. L. Lamberti, C. Pappalettere. Metaheuristic design optimization of skeletal structures: a review. *Computational Technology Reviews* 4 (2011) 1–32.
3. O.K. Erol, I. Eksin. New optimization method: Big Bang–Big Crunch. *Advances in Engineering Software* 37 (2006) 106–111.
4. C.V. Camp. Design of space trusses using Big Bang–Big Crunch optimization. *ASCE Journal of Structural Engineering* 133 (2007) 999–1008.
5. A. Kaveh, S. Talatahari. Size optimization of space trusses using Big Bang-Big Crunch algorithm. *Computers and Structures* 87 (2009) 1129–1140.
6. A. Kaveh, S. Talatahari. Optimal design of Schwedler and ribbed domes via hybrid Big Bang-Big Crunch algorithm. *Journal of Constructional Steel Research* 66 (2010) 412–419.
7. A. Kaveh, A. Zolghadr. Truss optimization with natural frequency constraints using a hybridized CSS-BBBC algorithm with trap recognition capability. *Computers and Structures* 102–103 (2012) 14–27.
8. L. Lamberti, C. Pappalettere. A fast Big Bang–Big Crunch optimization algorithm for weight minimization of truss structures. In: *Proceedings of the Second International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering* (Y. Tsompanakis and B.H.V. Topping, Eds.). Civil Comp Press, Dun Eaglais (UK), 2011.
9. L. Lamberti. An efficient simulated annealing algorithm for design optimization of truss structures. *Computers and Structures* 86 (2008) 1936–1953.
10. L. Lamberti, C. Pappalettere. An improved harmony-search algorithm for truss structure optimization. In: *Proceedings of the Twelfth International Conference on Civil, Structural and Environmental Engineering Computing* (B.V.H. Topping, L.F. Costa Neves, and R.C. Barros, Eds.). Civil Comp Press, Dun Eaglais (UK), 2009.
11. K.S. Lee, Z.W. Geem. A new structural optimization method based on the harmony search algorithm. *Computers and Structures* 82 (2004) 781–798.
12. J.L. Li, Z.B. Huang, F. Liu, Q.H. Wu. A heuristic particle swarm optimizer for optimization of pin connected structures. *Computers and Structures* 85 (2007) 340–349.
13. S.O. Degertekin. Improved harmony search algorithm for sizing optimization of truss structures. *Computers and Structures* 92–93 (2012) 229–241.