COMPARATIVE ANALYSIS OF METHODS FOR MULTI-OBJECTIVE STRUCTURAL OPTIMIZATION

Aleksandar Milajić [1]
Dejan Beljaković[2]

UDK: 624.072.22 : 519.2
DOI:10.14415/konferencijaGFS 2016.007

Summary: *Optimum design of truss structures is considered as a benchmark problem in the field of the structural optimization. In order to solve this hard combinatorial problem, it is necessary to implement adequate optimization tool that would provide sufficiently wide range of possible solutions within a reasonable time as well as to obtain good exploration and exploitation of search space. The aim of presented study was to compare efficiency of different multi-objective algorithms in solving this task.*

Keywords: *Truss structure, multi-objective optimization, genetic algorithms, tabu search, Big Bag – Big Crunch algorithm*

1. INTRODUCTION

Truss structures optimization usually involves determining optimum values for member cross-sectional areas that would minimize the weight of a given truss structure while satisfying a number of inequality constraints that limit design variables sizes. However, minimum weight is often not the only goal, because limiting deformations is often more important demand in order to achieve usability demands in exploitation during the structure's life-cycle. These cases belong to the field of multi-objective optimization.

The main task of every optimization tool is to perform effective and efficient exploration of a given search space. This search for an optimal solution should be versatile enough to both intensively explore promising areas of the search space around high quality solutions, and to reach its unexplored areas. Two basic concepts for reaching these goals are usually called intensification and diversification. These terms originate from the tabu search field, but same or related concepts can also be found in other methods, such as evolutionary algorithms, denoted as exploitation (related to intensification) and exploration (related to diversification). The main difference between intensification and diversification is that intensification feature of a given search tool focuses on examining neighbors of elite solutions, while the diversification encourages the search process to examine unvisited regions and to generate solutions that differ in various significant

[1] Aleksandar Milajić, Faculty of Construction Management, Union Nikola Tesla University, Cara Dušana 62–64, Belgrade, Serbia, e-mail: aleksandar.milajic@gmail.com
[2] Dejan Beljaković, Faculty of Construction Management, Union Nikola Tesla University, Cara Dušana 62–64, Belgrade, Serbia, e-mail: beljak@mail.com

ways from those seen before [1]. A metaheuristic will be successful on a given optimization problem if it can provide a good balance between the exploitation of the accumulated search experience and the exploration of the search space to identify regions with high quality solutions in a problem specific, near optimal way [2].

The aim of presented research was to explore applicability of three different meta-heuristics, namely genetic algorithm, tabu search and Big Bang – Big Crunch algorithm in finding optimum design features of a given truss structure. Obtained results indicate that the Big Bang – Big Crunch algorithm outperforms two other methods in exploration and exploitation of a search space by producing much better Pareto fronts of possible solutions.

2. PROBLEM FORMULATION

Mathematically, the optimal design of a truss can be formulated as finding set of variables [$A_1, A_2, \ldots, A_n$], $A_i \in D$, where A_i is the cross-sectional area of a member *i*, *n* is the number of members in a givent russ structure, and *D* denotes the allowable set of values for the design variable A_i, in order to minimize both the nodal displacements (1) and the total weight of structure (2):

$$\delta_j = \delta_{j,\min}, \; j = 1,2,\ldots,n_n \quad (1)$$

$$\mathrm{W(A)} = \sum_{i=1}^{n} \gamma_i A_i L_i \quad (2)$$

Where n_n is nuber of nodes; *W(A)* is weight of the structure; *n* is the number of members of the structure; γ_i represents the material density of member *i* and L_i is the length of member *i*, subject to constraints:

$$g_j(A) \leq 0, \; j = 1,2,\ldots,n \quad (3)$$

Since the presented problem has two objective functions, an appropriate solving method is the multi-objective tool that would be able to locate multiple Pareto optimal solutions in a single run. A solution is said to be Pareto optimal if and only if it is not dominated by any other solution in the performance space. If one solution dominates another, it implies that the first one is non-inferior to the second one for all the considered performance criteria but it is better than it for at least one criterion. All Pareto solutions form a Pareto front in the performance space.

Analysis of the whole Pareto front provides useful information on trade-off relationship between the fitness functions and enables a decision maker to consider different alternatives and make a choice that would represent acceptable compromise for conflicting objectives. In hard combinatorial problems such as this one it is impossible to conduct thorough search for Pareto solutions within the whole search space without appropriate optimization tool.

3. OPTIMIZATION TOOLS

Genetic algorithms (GA) are a special class of global optimization methods, based on the theory of evolution. GAs do not operate on a single trial solution, but on a group of solutions, called population. A solution (called a string) is a vector of all parameters which are to be optimized. After application of evolution inspired operators such as fitness, crossover and mutation, the best solutions are being transformed and saved, forming the next generation, which means that the whole population moves towards better solutions, and finally to the global optimum [3, 4]. Although GAs have been proven able to locate promising regions for global optima in a search space, they sometimes can have a problem with finding the exact (global) minimum or maximum, especially if the search space is very large. In the field of optimization problems, tabu search (TS) is often used as a 'higher' heuristic procedure for enabling the other methods to avoid the trap of local optimum [5, 6]. TS operates on a single solution at a time and uses problem-specific operators to explore a search space and memory (called the tabu list) while keeping track of parts already visited. By guiding the optimization to the new areas, TS is able to overcome local minima and to reach the global optimum [1].
The Big Bang – Big Crunch algorithm (BB-BC) is relatively new evolution algorithm introduced in 2006 by Erol and Eksin [7], inspired by a theory of the evolution of the universe, namely the Big Bang and Big Crunch theory. Every step of the algorithm consists of two phases: a Big Bang phase, and a Big Crunch phase. In the Big Bang phase, candidate solutions are randomly distributed over the search space. The Big Bang phase is followed by the Big Crunch phase. The Big Crunch is a convergence operator that has many inputs but only one output, named as the center of mass, where the term *mass* refers to the inverse of the fitness function value. After calculating the center of mass, the algorithm generates new candidate solutions for the next Big Bang phase using a normal distribution around the previous center of mass. After the new population is generated, algorithm moves onto the next Big Crunch phase by calculating new center of mass. This sequence of explosion and contraction is repeatedly carried out until a stopping criterion has been met, whether it is reaching maximum number of iterations or obtaining a convergence.

4. COMPARISON CRITERIA

Basically, every multi-objective optimizer aims at three goals: a) to find out the true Pareto front or to converge as close as possible to it; b) to cover as wide as possible span of solutions, and c) to discover solutions as diverse as possible along the obtained Pareto front. In order to provide a quantitative performance assessment for different multi-objective optimizing algorithms, it is necessary to establish exact criteria for measuring and comparing their effectiveness. A variety of such metrics and methods have been proposed in literature [8–10]. Method used in this study is based on a set of three metrics, namely: Coverage, Spacing and Maximum Spread [10, 11].
Coverage (C-metric) provides comparison between two Pareto fronts. If *A* and *B* be are two approximations to the Pareto front, then *C(A, B)* is the percentage of the solutions in *B* that are dominated by at least one solution in *A*:

$$C_{(A,B)} = \frac{\left|\{u \in B \mid \exists v \in A : v \ dom \ u\}\right|}{|B|} \tag{4}$$

Spacing (S-metric) indicates how evenly the solutions are distributed along the discovered Pareto-front:

$$S = \left[\frac{1}{n_{pf}}\sum_{i=1}^{n_{pf}}(d_i - \bar{d})^2\right]^{\frac{1}{2}}, \bar{d} = \frac{1}{n_{pf}}\sum_{i=1}^{n_{pf}} d_i \tag{5}$$

where n_{pf} is the number of members in Pareto front and d_i is the Euclidean distance (in the objective space) between the member i in Pareto front and its nearest member. A smaller value of S implies a more uniform distribution of solutions in Pareto front.

Maximum Spread (MS-metric) measures how “well” obtained Pareto front covers the true Pareto front:

$$MS = \left[\frac{1}{m}\sum_{i=1}^{m}\left[\frac{\min(f_i^{\max}, F_i^{\max}) - \max(f_i^{\min} - F_i^{\min})}{F_i^{\max} - F_i^{\min}}\right]^2\right]^{\frac{1}{2}} \tag{6}$$

where m is the number of objectives, $f_{i\max}$ and $f_{i\min}$ are the maximum and minimum of the ith objective in obtained Pareto front, respectively, and $F_{i\max}$ and $F_{i\min}$ are the maximum and minimum of the ith objective in true Pareto front, respectively. A larger value of MS inicates a better spread of solutions. Since the true Pareto front in this study is not known, $F_{i\max}$ and $F_{i\min}$ are considered as the maximum and minimum of the ith objective in all obtained Pareto fronts by various algorithms.

5. NUMERICAL EXAMPLE

Comparative analysis was performed on standard benchmark problem of 56-bar truss structure [12, 13] with members grouped into three groups as shown in Figure 1. Optimization task is to simultaneously minimize total structural volume $F_{1(X)}$ (and consequently the weight and ammount of material) and total displacement of the node 1, $F_{2(X)}$. Therefore, objective functions are:

$$\mathrm{F}_{1(\mathrm{x})} = \sum_{\mathrm{i}=1}^{56} \mathrm{A}_{\mathrm{i}} l_{\mathrm{i}} \tag{7}$$

$$\mathrm{F}_{2(\mathrm{x})} = \sqrt{\delta_{1x}^2 + \delta_{1y}^2 + \delta_{1z}^2} \tag{8}$$

Load consists of 4 kN force in the Y-direction and 30 kN force in the Z-direction in joint 1, while other free nodes are loaded with 4 kN force in Y-direction and 10 kN force in Z-direction. Vertical displacements of joints 4, 5, 6, 12, 13 and 14 are restricted to 4 mm, and displacement in joint 8 in Y-direction is limited to 2 mm. The modulus of elasticity is 210 kN/mm^2 for all members, while minimum and maximum cross-sectional areas of members are 200 mm^2 and 2000 mm^2, respectively.

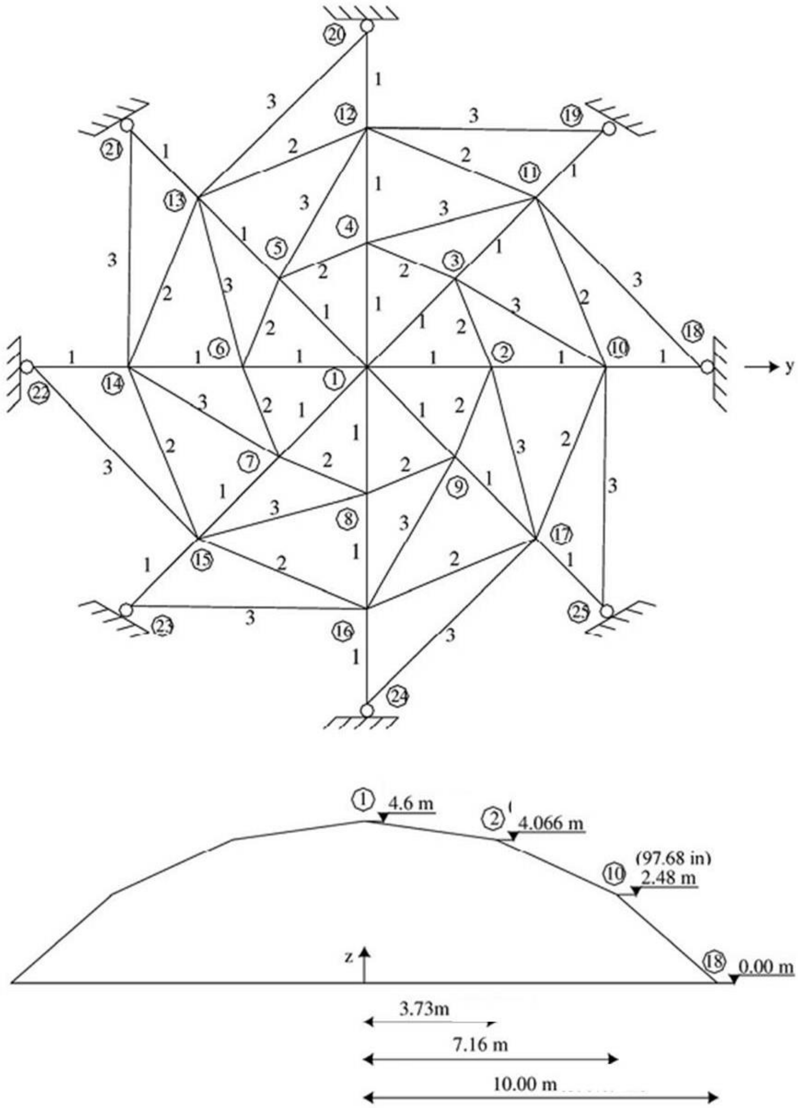

Figure 1. Truss geometry and member grouping

6. RESULTS AND DISCUSSION

In order to evaluate performance of described algorithms, they were run five times. Extreme solutions after five runs of each algorithm are presented in Table 1 and performance metrics values are presented in Table 2. Comparison of results in Table 1 indicates that BB-BC algorithm outperforms GA and TS in solving this task and obtained better extreme results for the both fitness functions.

Table 1. Best results after five runs

Algorithm	Best extreme solution 1 (mm, m^3)	Best extreme solution 2 (mm, m^3)
GA	(2.25; 0.4035)	(7.67, 0.1278)
TS	(2.75; 0.4111)	(7.87, 0.1315)
BB-BC	(2.23; 0.4029)	(7.54, 0.1245)

Table 2. Performance metrics values

C-metric					
Algorithm	Run 1	Run 2	Run 3	Run 4	Run 5
GA vs. TS	0.34	0.38	0.29	0.31	0.38
TS vs. GA	0.11	0.16	0.22	0.24	0.20
GA vs. BB-BC	0.29	0.26	0.22	0.31	0.17
BB-BC vs. GA	0.38	0.36	0.34	0.35	0.40
TS vs. BB-BC	0.11	0.15	0.22	0.17	0.20
BB-BC vs. TS	0.58	0.51	0.47	0.60	0.52
S-metric					
Algorithm	Run 1	Run 2	Run 3	Run 4	Run 5
GA	22.32	19.98	24.66	23.40	25.00
TS	47.54	42.49	38.99	41.71	39.96
BB-BC	11.98	10.44	13.02	15.93	12.12
MS-metric					
Algorithm	Run 1	Run 2	Run 3	Run 4	Run 5
GA	0.95	0.96	0.94	0.97	0.92
TS	0.92	0.91	0.89	0.92	0.90
BB-BC	1.00	1.00	0.97	0.95	0.99

Differences in quality of obtained Pareto fronts can be further discussed according to the results of three considered performance metrics given in Table 4. Values of the preformance metrics indicate that the BB-BC algorithm obtained better Pareto fronts in comparison with other two methods. Values of the S-metric indicate that BB-BC provided very uniform distribution of solutions along the Pareto front, while the solutions obtained by the GA can be considered as satisfying. High values for the TS indicate that this method had problems with being trapped in local optimum points. Values of the MS-metric indicate that solution obtained by all three methods are close enough to the real Pareto front.

7. CONCLUSION

In this paper comparative analysis of three metaheuristics is performed in order to find the most apropriate metodology for solving combinatorial problem of multi-objective truss optimization. Results indicate that the Big Bang – Big Crunch algorithm outperforms tabu search and genetic algorithm both in finding better extreme solutions and considering quality of pbtained Pareto front.

REFERENCES

[1] Glover, F. and Laguna, M. *Tabu Search*. Kulwer Academic Publishers, **1997**.
[2] Stutzle, T. *Local Search Algorithms for Combinatorial Problems - Analysis, Algorithms and New Applications*. DISKI - Dissertationen zur Kunstlik ¨ en Intelligenz. infix, Sankt Augustin, Germany, **1999**.
[3] Holland, J. *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, MA, **1992**.
[4] Goldberg, D. *Genetic Algorithms in Search, Optimisation and Machine Learning*, Addison-Wesley, Reading, MA, **1989**.
[5] Jourdan, C. L., Basseur, M. and Talbi, D. E. Hybridizing Exact Method and Metaheuristics: A Taxonomy. *European Journal of Operational Research*, **2008**.
[6] Talbi, D. E. A Taxonomy of Hybrid Metaheuristics. J. of Heur. 8(5), 541-564, **2002**.
[7] Erol, O. K. and Eksin, I. *New optimization method: Big Bang–Big Crunch*, Adv Eng Software, 37, **2006**, 106–11.
[8] Deb, K. *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, Chichester, UK, **2001**.
[9] Tan, K. C., Lee, T. H., Khor, E. F. *Evolutionary algorithms for multiobjective optimization: performance assessments and comparisons*, Artificial Intelligence Review, 17 (4), **2002**, 253–290.
[10] Zitzler, E. et al.: Performance assessment of multiobjective optimizers: an analysis and review, *IEEE Transactions on Evolutionary Computation*, 7 (2), **2003**, 117–132.
[11] Kaveh, A., Laknejadi, K. A hybrid evolutionary graph-based multi-objective algorithm for layout optimization of truss structures, *Acta Mechanica*, 224, **2013**, 343–364.
[12] Kelesoglu, O.: Fuzzy multi objective optimization of truss-structures using genetic algorithm. *Advances in Engineering Software*, **2007**, Vol. 38, pp. 717–721.
[13] Kaveh, A. And Laknejadi, K.: A Hybrid Multi-objective Optimization and Decision Making Provedure for Optimal Design of Structures, *IJST Transactions of Civil Engineering*, **2011**, Vol. 35 (C2), p.p. 137-154

УПОРЕДНА АНАЛИЗА ВИШЕКРИТЕРИЈУМСКИХ МЕТОДА ОПТИМИЗАЦИЈЕ КОНСТРУКЦИЈА

Резиме: *Оптимално пројектовање решеткастих носача сматра се стандардним проблемом у пољу оптималног димензионисања конструкција. Да би се решио овај тежак комбинаторни проблем, неопходно је применити одговарајућу методу оптимизације која ће за прихатљиво време пружити довољно широк спектар могућих решења али и обезбетиди темељну и свеобухватну претрагу области дефинисаности. Циљ приказаног истраживања био је да се упореди делотворност различитих вишекритеријумских метода оптимизације у решавању овог задатка.*

Кључне речи: *Решеткаста конструкција, вишегритеријумска претрага, генетски алгоритми, табу претрага, Велики прасак – велико сажимање*