

Modified genetic algorithm for optimal design of truss structures

D. Šešok*, R. Belevičius**

*Vilnius Gediminas Technical University, Saulėtekio al. 11, 10223 Vilnius, Lithuania, E-mail: dms@fm.vgtu.lt

**Vilnius Gediminas Technical University, Saulėtekio al. 11, 10223 Vilnius, Lithuania, E-mail: rb@fm.vgtu.lt

1. Introduction

Truss systems are widely used in engineering practice, mainly due to the economy, simple manufacturing, transportation and storage. These systems form the framework of such constructions as bridges, towers, roof supporting structures, etc. Usually a truss system includes a large number of elements (trusses), which can have different parameters (length, cross-sectional area, shape of cross-section, material). Thus, the optimization of truss systems is relevant problem in engineering. Truss systems can be optimized according several criteria: sizing, shape, and topology [1]. Large number of design parameters makes this problem complex. As the sizing optimization usually does not cause serious computational problems, further we will deal with the topology and shape optimization.

In topology optimization the optimal truss placement scheme in the framework of nodes with fixed positions is sought. This problem is discrete; the number of possible variants depends on the number of nodes and is huge even in the case of small-scale structures. Therefore full-search algorithms can not be used for the solution. Instead, different methods and algorithms avoiding the examination of all possible connection combinations are exploited. Thus, in the so-called "ground structure method" [2] the solution begins from an over-connected truss system, and the superfluous trusses are eliminated. Also the simulated annealing method [3], which is the generalization of Monte Carlo method, is used. However, the most natural strategy for topology optimization of truss systems seems to be the use of genetic algorithms (GA), where the solution is adapted to the constraints and objective function [4]. In this paper we solve the topology optimization problems using original modified genetic algorithm, which for this particular class of problems yields better results than the classical GA [5].

In shape optimization the number of nodes and trusses is constant, and only positions of the nodes may vary. Design parameters in this case are the coordinates of certain set of nodes [6]. Thus, the truss system topology must be known prior to the shape optimization. Naturally, in the joint topology/shape optimization the multilevel design strategy can be employed, where in one phase the topology optimization is dealt with, and in the other – the shape optimization [7]. In this paper we also employ this approach; shape optimization is performed using classical GA, thus improving the solution obtained in the topology optimization phase.

An additional research was conducted in order to ascertain what strategy can yield better solutions: to carry out the shape optimization following the topology optimization, or simply perform only the topology optimization with increased number of nodes.

2. Problem formulation

As the objective function for topology and shape optimization the total mass M of the structure is taken

$$\min M(x), x \in D \quad (1)$$

where x denotes the design parameters, and D is the feasible configuration of the truss system. Total mass of structure can be simply obtained summing up the masses of all truss system's elements

$$M = \sum_{e=1}^n L_e \rho_e A_e \quad (2)$$

where L_e is the length of e th element, ρ_e is density, and A_e is cross-sectional area of the same e th element.

The constrains for the problem involve equilibrium (3) and local stability (4) checks. Also stresses in the trusses should not exceed the given threshold value (5).

$$\sum_{j=1}^k \bar{F}_{ij} = 0 \quad (3)$$

where \bar{F}_{ij} is the j th force at the i th node; the forces are obtained solving the main statics equation $[K]\{u\} = \{F\}$, where $[K]$ and $\{u\}$ are stiffness matrix and displacements of the truss system.

$$|F_e| \leq \frac{\pi^2 E_e I_e}{L_e^2} \quad (4)$$

where F_e is maximum allowed compression force at the e th element, E_e is Young's modulus, I_e is moment of inertia, and L_e is the length of the same element.

$$|\sigma_e| \leq \sigma_{max} \quad (5)$$

where σ_e is stress in the e th truss, and σ_{max} is maximum allowable stress.

The objective value is obtained by the finite element method [8, 9]. For this purpose it is possible to use commercially available packages ANSYS, ALGOR, ABAQUS, COSMOS, etc [10]. Original connection technology between the optimization algorithm in C++ and package ANSYS, which examines the requirements of constraints and obtains all the data necessary for objective value, is suggested by the authors [11].

In this paper the original software in C++ was used for the finite element method instead of general-

purpose commercial packages. In the case of commercial packages application, a lot of computer time is wasted for package loading into service application and unloading processes. As the genetic algorithm is also implemented in C++, the unified computational environment dramatically reduces the required computational resources.

3. Topology optimization

For topology optimization of truss systems the GA concept is employed [5, 12]. The interface between GA and particular truss system, namely, how to convert the truss system into string of bits, is described in details in [11]. The original modified GA was used in this work [13] (Fig. 1).

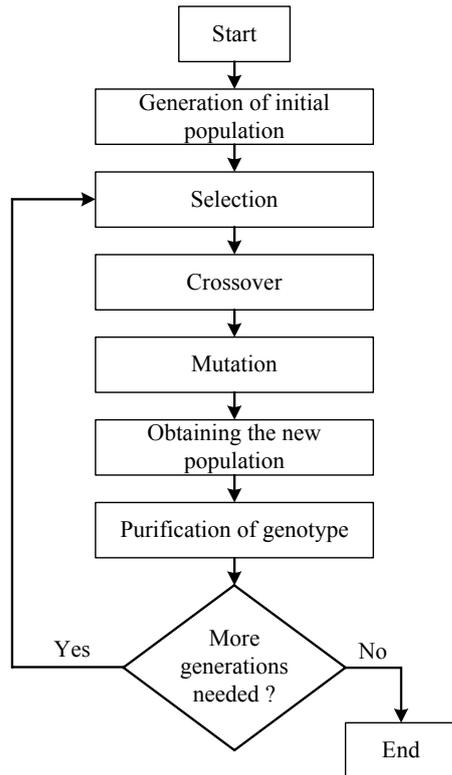


Fig. 1 Scheme of modified genetic algorithm

As it is seen, the modified GA adds to the classical algorithm (Fig. 8) one additional phase – the purification of genotype. Here, beside the selection, cross-over and mutation processes, each individual (which is entire truss system) is additionally analyzed. Provided that there are particular trusses with stresses below some threshold value, they are eliminated from the truss system, and the values of fitness function associated with these individuals are recalculated. However, these individuals are retained in the population. If these individuals prove to be unfit, they will not survive the selection. Evidently, as an alternative for genotype purification the following constraint can be led into mathematical model:

$$|\sigma_e| \geq \sigma_{min} \quad (6)$$

where σ_{min} is minimum allowable stress.

However, the experience of numerous truss systems optimization clearly states, that such a constraint impedes the optimization process, because a number of indi-

viduals must be eliminated from each generated population. It is obvious from engineering point of view: at the beginning of optimization process the truss system contains usually fairly large number of elements, and the probability to obtain the under-stressed trusses is high. The suggested heuristics proved to retain more possibilities for obtaining the global solution.

Let us illustrate the process of optimization on one numerical example: rectangular cell of dimensions 3x2 m with two clamped nodes and nonsymmetrical loading (three concentrated loads; vertical loads of magnitude 25 kN and 10 kN, horizontal – of 15 kN); the positions of boundary conditions and loads are fixed. Nodes are to be connected by trusses with cross-sectional area of 5e-4 m²; material Young's modulus is 200 GPa and density is 7800 kg/m³. At the beginning the cell is divided into 12 nodes (Fig. 2), later on the discrete scheme is refined regularly.

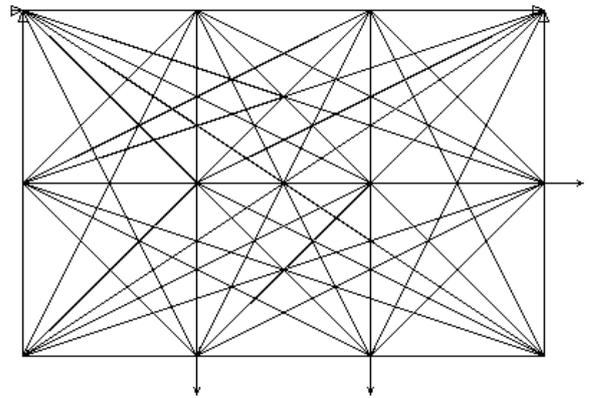


Fig. 2 Connection “all-to-all” in 12-node truss system

In this case there are 66 possible node connections and approximately 7.4E+19 variants of truss system. In order to increase the probability of obtaining the global solution, the problem was solved with different values of genetic parameters. Thus, the population size was varied in the range from 40 individuals to 90 with a step of 2 individuals, and the mutation probability was from 1% to 5% (step 1%) – 130 numerical experiments in total. The best solution was obtained, when the population size is 86 individuals and mutation probability is 1%. The total length of elements of the truss system corresponding to this individual (Fig. 3) is 20371.6 mm, and mass – 794.492 kg.

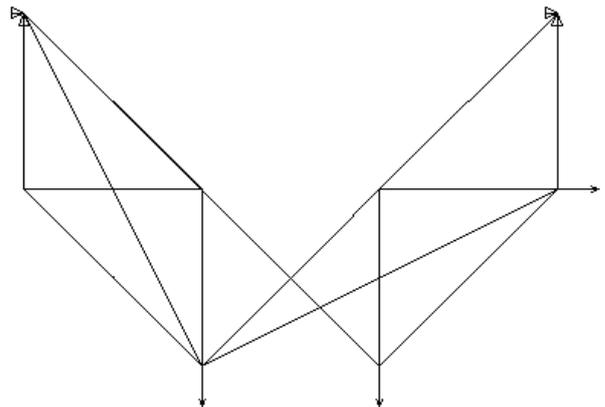


Fig. 3 The best solution of topology optimization problem (12-node truss system)

The problem was solved on the processor AMD Athlon 1.09 GHz, 1 GB of RAM. For each generated population 200 iterations were performed, after what the GA is stopped and the best obtained individual is selected. Processing of 200 iterations for the population, where the best individual was obtained (Fig. 3), took 15 sec.

The evident strategy for better solution is refinement of the discrete scheme of the problem. Thus, let us regularly duplicate the density of the mesh retaining all the initial data of the problem. The obtained scheme with 35 nodes is shown in Fig. 4.

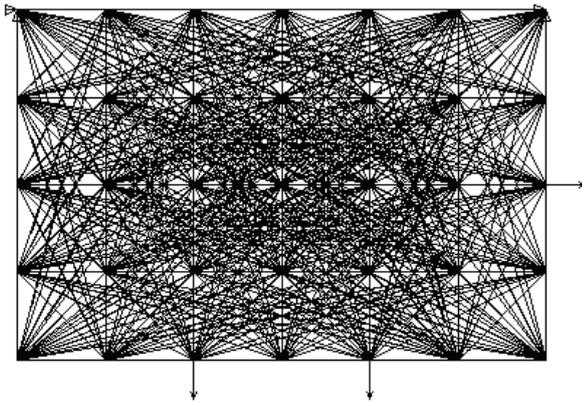


Fig. 4 Connection "all-to-all" in 35-node truss system

Now the size of the problem increases until 595 possible node connections and approximately $1.3E+179$ variants of the truss system. As the length of each individual (i.e., length of the string of bits) now increases until 595 bits comparing to 66 bits in previous case, the population size also must be increased. Practical experience shows, that minimum size of the population should be of the rank of individual's length.

The populations possessing from 150 to 400 individuals with a step of 10 individuals and the mutation probability from 1% to 5% were examined. The best solution (Fig. 5) is obtained in 92 sec. for the population having 220 individuals and with mutation probability 1%.

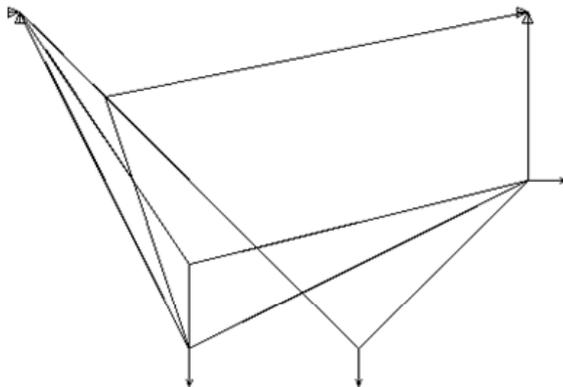


Fig. 5 The best solution of topology optimization problem (35-node truss system)

Total length of the elements of the obtained truss system is 18916.9 mm, mass is 737.759 kg. Thus, the solution is improved from 794.492 kg to 737.759 kg, or by 7.1%. However, the increase of computational time (92 sec. instead of 15 sec.) naturally leads to the exploration of other possible solution strategies. One evident strategy is to

carry out shape optimization of the obtained topologies instead of increasing node mesh density.

4. Shape optimization

During the shape optimization optimal positions of the nodes (or of a particular set of nodes) are sought; the design parameters are coordinates of these nodes. Now the interface program between GA and the truss system must be able to render all coordinates of the nodes and their allowed alteration ranges in the strings of bits, and vice versa.

The coordinates of each node are coded in the following way. Let the x and y are known initial Cartesian coordinates, and the string of 4 bits is given for one coordinate value. Then we have $2^4 = 16$ different positions in x direction and 16 – in y direction. Thus, the coding of one node's position requires 8 bits; the node will be able to take one of $2^8 = 256$ different discrete positions. Allotment of longer string for one coordinate increases the number of possible positions of the node; however, longer computational time is also to be expected. Fig. 6 shows the range of possible node's positions, when each coordinate is coded into 3 bits (i.e., 8 positions); 6 bits for one node. In this case the node can take 64 different positions.

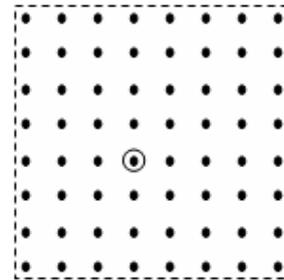


Fig. 6 Feasible range of nodes' positions (coding in 6 bits string)

Possible positions of the node are shown as the black dots; initial position is rimmed. The initial position is not in the centre of the range, because always we have even number of node's positions: 4, 8, 16, 32, etc. Therefore for x coordinate the nearest position from the left side is chosen, and for y coordinate – the nearest one from the bottom side. Now the interface scheme is as shown in Fig. 7.

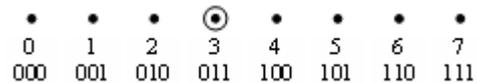


Fig. 7 Transformation of coordinate value

In Fig. 7 the coding of x coordinate is shown using the string of 3 bits. 8 positions are possible (from 0 to 7); the initial (third) position is rendered in the string of bits "011". Thus, if GA yields solution "011", the increment of x coordinate is 0. Correspondingly, the value "100" means transition to the 4th position, etc. Another important parameter for the interface is the distance between adjacent positions. If this value is chosen (the smaller this value, the more precise solutions is expected), decoding of the string into coordinates of the node may be carried out in the following way

$$x_{new} = x_{initial} + (V - V_{transf})S \quad (7)$$

where x_{new} is x coordinate of the node, $x_{initial}$ is initial coordinate, V is the decimal number corresponding to the string of bits, V_{transf} is a value to be subtracted in order to obtain the nearest to the central position from the left side, and S is the distance between adjacent positions. V_{transf} depends on the number of bits N allotted for the coordinate rendering:

$$V_{transf} = 2^{N-1} - 1 \quad (8)$$

Coordinate y is calculated in the same way.

Coding coordinates of the whole truss system, the string of bits is obtained, where x and y coordinates of all nodes in sequence are recorded

$$\underbrace{aaaaaa}_{\substack{x \\ Node_1}} \underbrace{aaaaaa}_{\substack{y \\ Node_2}} \dots \underbrace{aaaaaa}_{\substack{x \\ Node_K}} \underbrace{aaaaaa}_{\substack{y \\ Node_K}}, a \in \begin{Bmatrix} 0 \\ 1 \end{Bmatrix} \quad (9)$$

where a is the value of a bit, x and y are Cartesian coordinates of i th node, and K is the number of nodes to be coded (less than the overall number of system's nodes, as clamped and loaded nodes are not allowed to change their positions).

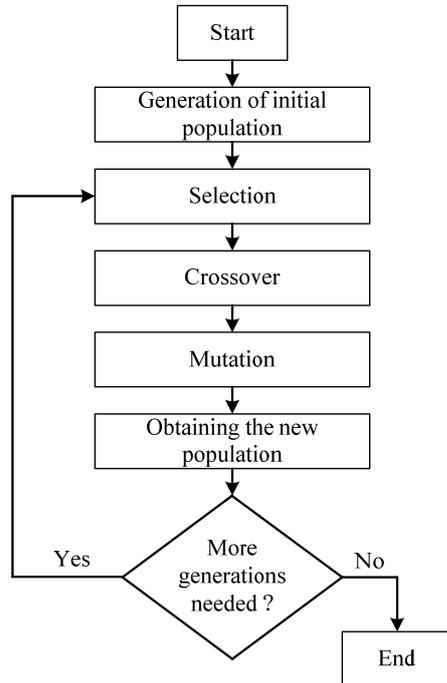


Fig. 8 Scheme of classical genetic algorithm

The described algorithm is implemented in C++. An algorithm was applied for shape optimization of the structure shown in Fig. 3 (the best topology of 12-node system), hoping to improve the obtained topology by varying node positions. In this particular system only positions of 3 nodes can be changed, as remaining nodes are fixed due to boundary and loading conditions. 10 bits were allotted for coding of each node, 30 bits in total for the truss system. Thus, the range of 2^{30} or approximately $1.07E+09$ different

positions were examined. The distance between adjacent positions of the nodes in vertical and horizontal directions was equal; the value was chosen in such a way that two neighboring nodes may nearly meet in the centre of the nodes.

The purification of genotype now is meaningless – the shape optimization starts from the best individual selected by the topology optimization, and the probability to obtain the under-stressed truss is not significant. Thus, classical GA (Fig. 8) was employed for the optimization.

Now the population size may be taken in the range of tens of individuals, because the length of one individual is much shorter. Margins for the values of other genetic parameters are the same as in previous numerical examples.

As in the case of topology optimization, a number of combinations of genetic parameters were examined: the population size varied from 2 to 50 individuals with a step of 2 individuals, and the mutation probability was from 1% to 5%. The best solution (Fig. 9) was obtained with the population size of 42 individuals and the mutation probability of 5%.

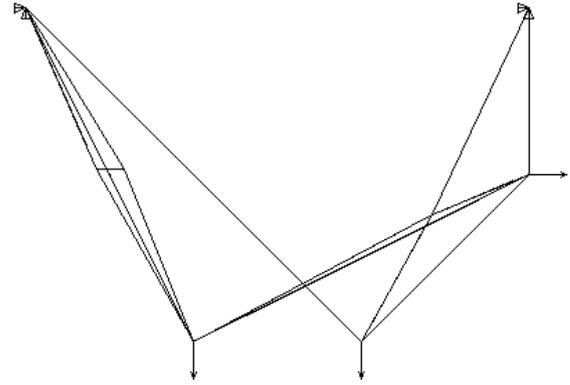


Fig. 9 The best solution of shape optimization (12-node system)

Total length of the elements of the obtained truss system is 18831.9 mm, and mass is 734.444 kg. The solution is obtained per 8 sec. For each population, as in the case of topology optimization, 200 generations were generated.

5. Discussion on numerical experiments

Three numerical experiments for the optimization of the same truss system were performed: topology optimization of initial truss system (serving as the basis for comparison of two different optimization strategies), topology optimization with subsequent shape optimization, and refined topology optimization. The results of all three experiments are presented in Table 1.

Table 1
Numerical results of optimization experiments

Type of optimization	The best mass, kg	Decrease of mass, kg	Increase of time, s
Topology (12 nodes)	794.492	0	0
Topology (35 nodes)	737.759	56.733	77
Topology (12 nodes) + shape	734.444	60.048	8

The results show, that the topology optimization with refined mesh of nodes decreases mass of the truss system by 7.1 %, however, the computational time increases from 15 sec. to 92 sec., i.e., by 513%. Shape optimization for the same case reduces the mass a little bit more – by 7.6%, while the computational time increases by 53.3%. The required computational time is shown in Fig. 10.

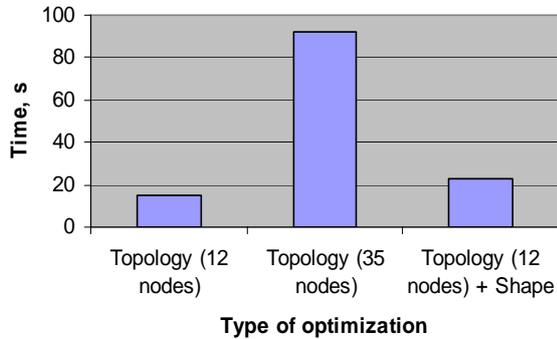


Fig. 10 Dependence of computational time on the type of optimization

It is impossible to judge about the best optimization strategies relying on one numerical example, therefore an additional series of numerical experiments were carried out. This time let us start from the 35-node system (the best topology obtained is shown in Fig. 5) and, as in previous chapter, try to improve the solution using the same strategies: to accomplish topology optimization with duplicated mesh of nodes, and to carry out shape optimization on the obtained best solution of the topology optimization of 35-node system.

For the first case the number of nodes increases until 117 nodes; there are 6786 possible node connections and approximately $6.02E+2042$ variants of truss system (all possible variants of the truss system now can not be shown graphically, see Fig. 11). The problem size increased dramatically, and precise tuning of genetic operators is needed in order to obtain reasonable solutions. Thus,

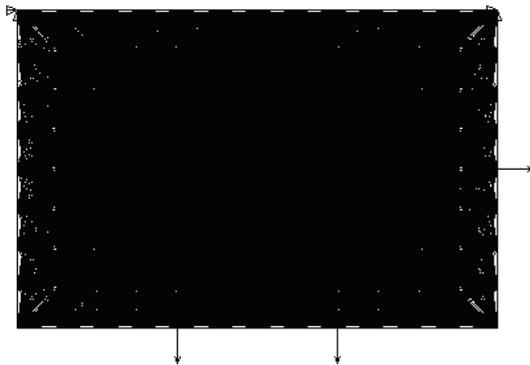


Fig. 11 Connection “all-to-all” in 117-node truss system

numerical experiments with small-size populations (from 50 to 500 individuals) do not render any better solutions. The same is valid for large-scale populations (500 to 1700 individuals) using the mutation probabilities from 1% to 5% (in previous numerical examples these probabilities always allow to obtain good solutions). This can be explained by the following reasons: now the length of each individual is 6786 genes, however, approaching end of the

solution it is expected to have a sparse populated individual (for the problem under consideration only about ten truss elements should be presented in the final scheme). Even the mutation of 1% will exchange approximately 70 genes per time – 70 genes of the value “1” will appear in the individual. Now, even we will arrive to a reasonable individual during the solution due to crossover, the mutation will ruin all the achievements.

The mutation for similar problems should change the values of only several genes. Thus, in this case the mutation probabilities of rank 0.01% proved a success. The best solution was obtained with the mutation probability 0.02% and the population size of 775 individuals (Fig. 12). The overall length of trusses here is 17897.9 mm, and mass is 698.018 kg. 200 iterations of the solution took 1530 sec.

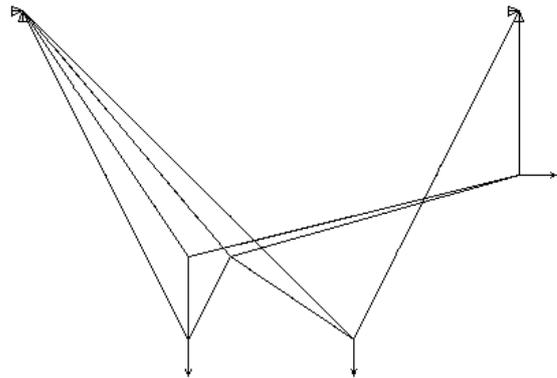


Fig. 12 The best solution of topology optimization problem (117-node truss system)

Thus, mass of the truss system was reduced, if compare with 35-node problem, by 39.741 kg (or 96.474 kg, if compared to 12-node problem).

Alternative strategy supposes to carry out additional shape optimization on the best solution of the topology optimization (Fig. 5) instead of increasing number of the nodes. Here only 2 nodes may change their positions. Coordinates of the nodes are coded by 28 bit string. As in the case of previous shape optimization of 12-node problem, several numerical experiments with different population sizes and mutation probabilities (2 to 50 individuals with step of 2 individuals; 1 to 5% with step of 1%) were carried out. The best solution (Fig. 13) was obtained using the population size of 30 individuals and the mutation probability of 4%; 200 iterations took 4 sec.

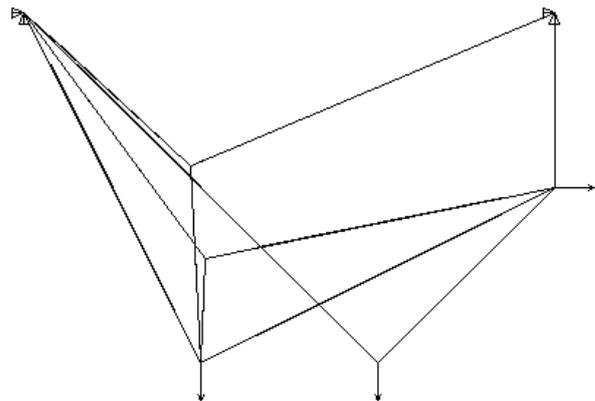


Fig. 13 The best solution of shape optimization (35-node system)

Here the overall length of trusses is 18715.9 mm, and the mass is 729.92 kg, i.e. the value of fitness function was reduced by 7.839 kg compared to the starting value.

Table 2 summarizes results of the both strategies. Thus, the topology optimization on the regularly refined computational scheme yields substantially better solution than the topology optimization/shape optimization strategy, however, at the expense of 15.9 times longer computations.

Table 2
Numerical results of optimization experiments
on 35-node system

Type of optimization	The best mass, kg	Decrease of mass, kg	Increase of time, s
Topology (35 nodes)	737.759	0	0
Topology (117 nodes)	698.018	39.741	1438
Topology (35 nodes) + shape	729.92	7.839	4

6. Conclusions

Genetic algorithms can be naturally adapted to the joint topology and shape optimization of truss systems. For that the interface program must be designed rendering the truss system as the string of bits, and vice versa. The interface for shape optimization renders node positions into strings in a similar way.

The solution of a particular problem may be improved in two ways. First, the topology optimization of the same problem with more dense initial mesh of nodes yields always a better solution. Second, the alternative strategy of solution improvement is to carry out the topology optimization with a more rough mesh of nodes and to employ the shape optimization on the defined solution; this usually further improves the solution.

What is the best strategy of optimization? The pure topology optimization on regularly refined computational scheme, perhaps, yields slightly better solution than the topology optimization of initial scheme and successive shape optimization. However, the required computer resources compared to the second strategy increase dramatically.

Acknowledgement

This work was supported by the Lithuanian State Science and Studies Foundation within the project on B-03/2007 "Global optimization of complex systems using high performance computing and GRID technologies".

References

1. **Smith, J., Hodgins, J., Oppenheim, I.** Creating models of truss structures with optimization. -ACM Transactions on Graphics (SIGGRAPH 2002), 2002, 21(3), p.295-301.
2. **Pederson, P.** Topology optimization of three dimensional trusses.-Topology Designs of Structures.-Kluwer Academic Publishers, 1992, p.19-30.
3. **Reddy, G., Cagan, J.** An improved shape annealing algorithm for truss topology.-J. of Mechanical Design. -ASME, 1995, p.315-321.
4. **Chapman, C., Saitou, K., Jakiela, M.** Genetic algorithms as an approach to configuration and topology design.-Advances in Design Automation.-ASME, 1993, v.65, p.485-498.
5. **Goldberg, D.** Genetic Algorithms in Search, Optimization and Machine Learning. -New-York: Addison-Wesley, 1989.-412p.
6. **Topping, B.** Shape optimization of skeletal structures: A review. -J. of Structural Engineering, 1983, v.109, p.1933-1951.
7. **Spillers, W.** Iterative Structural Design.-North Holland Publishing Co, 1975.-187p.
8. **Spyrakos, C., Raftoyiannis, J.** Linear and nonlinear finite element analysis in engineering practice.-Algor: Inc. Publishing Division Pittsburgh, PA. 1997, p.1-23.
9. **Barauskas, R., Belevičius, R., Kačianauskas, R.** Basics of Finite Element Method.-Technika, 2004.-612p. (in Lithuanian).
10. **Puiša, R.** The adaptive stochastic algorithms for CAD-based structure optimization of mechanical parts. -Doctoral dissertation.-Vilnius, 2005, p.129-144.
11. **Šešok, D., Belevičius, R.** Use of genetic algorithms in topology optimization of truss structures.-Mechanika, 2007, Nr.2(64), p.34-39.
12. **Holland, J.H.** Adaptation in natural and artificial systems.-Ann Arbor: The University of Michigan Press, 1975.-211p.
13. **Šešok, D., Belevičius, R.** Adaptation of genetic algorithms in topology optimization of truss structures.-J. of civil engineering and management. In press.

D. Šešok, R. Belevičius

SANTVARŲ OPTIMIZAVIMAS MODIFIKUOTU
GENETINIŲ ALGORITMU

R e z i u m ė

Straipsnyje aprašoma strypinių sistemų (santvarų arba rėmų) topologijos ir formos optimizacija. Optimizacija pradama nuo į baigtinį mazgų skaičių sudalytos struktūros; mazgai strypais vieni su kitais sujungiami visais galimais variantais. Netinkami strypinių sistemų variantai atmetami. Nagrinėjamos dvi alternatyvios optimizavimo strategijos: didesnio mazgų skaičiaus pradinės struktūros vien tik topologijos optimizavimas ir topologijos optimizavimas, pradedant nuo mažesnių mazgų skaičių turinčios struktūros ir papildomai optimizuojant gautą topologijos formą. Topologijai optimizuoti naudojamas autorių modifikuotas genetinis algoritmas, kuris tos klasės uždaviniams spręsti tinka geriau negu klasikinis genetinis algoritmas. Į algoritmą vietoj apribojimų sistemos plėtimo įtrauktas papildomas žingsnis – genomo išgryninimas, leidžiantis papildomai pagerinti atskirus populiacijos individus bei išlaikyti daugiau galimybių optimizavimo procesui, nei kad leistų ribojimų griežtinimas. Formai optimizuoti taikomas klasikinis genetinis algoritmas. Abi strategijos efektyviai gerina sprendinį, tačiau taikant bendrąją topologijos ir formos optimizavimo strategiją reikia mažesnių kompiuterio išteklių. Visi straipsnyje aprašomi skaitiniai pavyzdžiai išspręsti naudojantis originalia autorių sukurta programine įranga.

D. Šešok, R. Belevičius

MODIFIED GENETIC ALGORITHM FOR OPTIMAL DESIGN OF TRUSS STRUCTURES

S u m m a r y

In this paper, topology and shape optimization of truss or frame structures is discussed. The optimization starts from a structure, into which a finite number of nodes are set; all the nodes are connected together by trusses in all possible variants. Unfit variants of the truss system are rejected. Two alternative ways of the optimization are compared: topology optimization starting from initial structure with a larger number of nodes, and topology optimization starting from initial structure with smaller number of nodes but with additional shape optimization of the obtained topology. The topology optimization is solved with original modified genetic algorithm, giving better results in comparison with classical genetic algorithm. Instead of further development of constraint system, the additional step is introduced into algorithm – purification of genotype, which allows complementary improvement of particular population individuals, and together for the optimization process retains more possibilities than stiffening of constraints. The shape optimization is solved by classical genetic algorithm. Both strategies effectively improve the solution, however the common topology/shape optimization requires less computer resources. All numerical examples are obtained with original software developed by the authors.

Д. Шешок, Р. Белявичюс

МОДИФИЦИРОВАННЫЙ ГЕНЕТИЧЕСКИЙ АЛГОРИТМ ДЛЯ ОПТИМИЗАЦИИ ФЕРМЕННЫХ КОНСТРУКЦИЙ

Р е з ю м е

В статье описывается оптимизация топологии и формы ферменных конструкций. Оптимизация начинается со структуры, поделенной на конечное число узлов. Узлы соединяются стержнями друг с другом всеми возможными вариантами. Некорректные варианты стержневых систем отбрасываются. Сравняются две альтернативные стратегии оптимизации: оптимизация топологии с большим количеством узлов и оптимизация топологии с меньшим количеством узлов, но с последующей дополнительной оптимизацией формы. Для оптимизации топологии используется модифицированный авторами генетический алгоритм, который для данного класса задач дает лучшие результаты, чем классический генетический алгоритм. В алгоритм вместо расширения системы ограничений добавлен дополнительный шаг – очищение генотипа, который позволяет дополнительно улучшить некоторые индивиды популяции и сохранить большие возможности для процесса оптимизации, чем это бы позволило ужесточение ограничений. Для оптимизации формы используется классический генетический алгоритм. Обе стратегии эффективно улучшают решение, однако совместная стратегия оптимизации топологии-формы требует меньших компьютерных ресурсов. Все приведенные в статье численные примеры были решены с использованием оригинального программного обеспечения, разработанного авторами.

Received October 11, 2007