A GENETIC ALGORITHM FOR STRUCTURAL OPTIMIZATION OF STEEL TRUSS ROOFS

Eduardo S. Croce[1 2]
Elaine G. Ferreira[2]
Afonso C. C. Lemonge[2]
{*eduardo,elaine,lemonge*}*@numec.ufjf.br*
[1]Bolsista de Iniciação Científica PIBIC CNPq/LNCC
[2]NUMEC - Núcleo de Pesquisas em Métodos Computacionais em Engenharia
Universidade Federal de Juiz de Fora, Faculdade de Engenharia, Departamento de Estruturas
Campus Universitário, Bairro Martelos, CEP 36036-330, Juiz de Fora, MG, Brasil
Leonardo G. Fonseca
Helio J.C. Barbosa
{*goliatt,hcbm*}*@lncc.br*
Laboratório Nacional de Computação Científica – LNCC/MCT
Av. Getúlio Vargas, 333, Petrópolis, RJ, Brasil, CEP 25651-075

Abstract. *A common structural design problem is the weight minimization of structures which is formulated selecting a set of design variables that represent the structural and architectural configuration of the system. The structures are usually subject to stress and displacement constraints and the design variables can be continuous or discrete. In practice, it is often desirable to choose design variables (such as cross-sectional areas) from commercially available sizes. The use of a continuous optimization procedure – although usually more straightforward – will lead to non-available sizes and any attempt to substitute those values by the closest available commercial sizes can make the design unfeasible or unnecessarily heavier.*

In this paper a genetic algorithm is proposed to evolve the structural configuration for weight minimization of industrial buildings, with rectangular geometry projection, made up of uniform planar structures (steel truss roofs) along the longitudinal direction which are interconnected by purlins. The planar structures are trusses and their number, shape, and topology are allowed to change during the optimization process.

Keywords: *Genetic Algorithms, Optimization, Structural Mechanics*

1. INTRODUCTION

A common structural design problem is the weight minimization of structures subject to stress and displacement constraints. The design variables can be continuous and/or discrete and the inclusion of the later usually makes the problem harder. In practice, it is often desirable (or even mandatory) to choose design variables (such as cross-sectional areas) from commercially available sizes. The use of a continuous optimization procedure – although usually more straightforward – will lead to non-available sizes and any attempt to “round” or substitute those values by the “closest” available commercial sizes can potentially make the design unfeasible (constraints are violated) or uneconomical (the weight is unnecessarily increased). A good survey of structural shape optimization is presented in Haftka (1986). Using mathematical programming techniques problems of optimization involving design of trusses with discrete sizing and shape variables can be found in Hansen and Vanderplaats (1990) and Salajegheh and Vanderplaats (1993). Combining fully stressed design optimization and conjugate gradient techniques, shape and cross-section optimization of trusses, is discussed in Gil and Andreu (2001). Recently, in Smith et al. (2002), a method for designing models of truss structures with simultaneous optimization of mass and geometry is presented.

From the early paper by Goldberg and Samtani. (1986), where the classic ten-bar truss is minimized for weight, to the present day, several applications of genetic algorithms (GAs) in structural optimization have appeared in the literature.

In previous works, Lemonge and Barbosa (2000) and Lemonge et al. (2003), GAs are applied to evolve the structural configuration considering weight minimization of a space truss structure (made up of standard modules), presenting both continuous design variables, such as the coordinates of the nodes, as well discrete variables, such as the cross-sectional areas of the bars, which are to be chosen from commercially available sizes. Other works involving GA and shape structural optimization can be found in Duponcheele and Tilley (1995), Lemonge and Barbosa (1998), and Koumousis and Georgiou (1994). Domain knowledge considered within an algorithm for optimization of designs and configurations of steel truss roofs can be found in Hamza et al. (2003).

In this paper a genetic algorithm is proposed to evolve the structural configuration for weight minimization of industrial buildings, with rectangular geometry projection, made up of uniform planar structures (steel truss roofs) along the longitudinal direction which are interconnected by purlins. The planar structures are trusses and they are able to have their shape and topology changed during the search process.

These structures can usually be analyzed considering isolated planar structures composed by one truss, columns and foundations. In a previous work this problem was studied considering the truss with a fixed topology. Now it is possible to have trusses with different topologies among the candidate solutions to the optimization problem.

The optimization problem considers continuous as well as discrete design variables: the number of planar structures along the longitudinal direction (discrete), the number of bays in each truss roof (discrete), node coordinates defining the shape of the truss (continuous), and the cross-sectional areas of the bars (discrete).

Additional domain knowledge is introduced in the development (or morphogenesis Angeline (1995)) process that generates the phenotype (the structural design) from the genotype (a string of bits) and also in the fitness evaluation process of each candidate design.

This capability of evolution of the structural configuration in addition to its shape and sizing is an attractive feature because the designer must often deal with the lack of specific computational codes, and the difficulty in evaluating a large number of possible solutions.

2. THE GENETIC ALGORITHM

One reason for GAs popularity is that unlike many traditional optimization methods, GAs demand less from the underlying problem Goldberg (1989):

1. GAs do not require the objective function to be continuous and/or differentiable,
2. GAs do not require extensive problem formulation,
3. GAs are not sensitive to the starting point,
4. GAs are less prone to entrapment in local optima, and
5. GAs are naturally parallel.

Because of these advantages GAs have been applied to a wide variety of problems in science, engineering, finance, etc.

When a GA is used as a minimization (or maximization) algorithm, it differs from the more familiar mathematical programming techniques by

1. employing a *population* of candidate solutions,
2. operating upon the *coding* of a solution and not on the solution itself,
3. using *probabilistic* transition rules, and
4. not requiring additional information (like derivatives) about the function to be optimized.

As a result, the search can be performed over non-convex (and even disjunct) sets for non-convex, non-differentiable functions of variables that can be of different types (e.g. continuous, discrete, boolean).

A generic GA can be stated as:

```
begin
Initialize the population P
Evaluate each string in the population
repeat
   repeat
      Select 2 or more individuals in P
      Apply recombination operators with probability p_c
      Apply mutation operator with rate p_m
      Insert new individuals in P'
   until (population P' complete)
   Evaluate individuals in population P'
   P ← P'
until (termination criterion)
stop
end
```

In the following, the coding, selection, recombination, mutation and evaluation procedures of the GA used in this paper are summarized.

2.1 The coding procedure

The first step is to encode all the variables corresponding to a candidate solution in a chromosome. In this paper we adopted the standard binary coding: each variable is encoded into a string of binary digits of a chosen length and these strings are then concatenated to form a single string which is an individual in the population of candidate solutions.

2.2 The Selection Scheme

In this paper, the *rank-based* selection scheme is adopted. Given the current population, this selection scheme starts by sorting the population according to the values of the fitness function constructing a *ranking*, i.e. better solutions have higher rank. Individuals in the population are then selected in such a way that higher ranking individuals have a higher probability of being chosen for reproduction. This leads to an intermediate population whose elements will then be operated upon by the recombination and mutation operators.

2.3 The Recombination Operators

The recombination of the genetic material of the selected "parent" chromosomes in order to generate the offspring chromosomes will be accomplished here using three *crossover* operators – one-point, two-point and uniform Syswerda (1995). The recombination operation is usually performed with a user-defined probability p_c and, consequently, with probability $1 - p_c$, the operation is not performed and both parents are just copied and sent to the mutation operation step.

2.4 The Mutation Operator

After the recombination step and again inspired in Nature, a mutation operator is introduced to simulate the errors that may arise during the copy process. With a (low) given mutation rate p_m the mutation operator is applied to each bit in the offspring chromosomes. The effect of this operator in the case of a binary alphabet is simple: just change a 1 into a 0 and vice-versa.

2.5 The Evaluation Step

After a new population is created each individual/solution must be evaluated in order to have a fitness value assigned to it. The genotype development, sometimes called morphogenesis (a term also borrowed from Biology), is the process that builds from each chromosome its corresponding phenotype and, in this case, the complete individual to be evaluated.

3. OPTIMIZATION OF AN INDUSTRIAL BUILDING CONFIGURATION

In a previous work Lemonge et al. (2003) a GA is applied to evolve the structural configuration considering weight minimization of a space truss structure (consisted of standard modules that includes a steel truss roof), presenting both continuous design variables, such as the coordinates of the nodes, as well discrete variables, such as the cross-sectional areas of the bars, which are to be chosen from commercially available sizes.

The Figure 2 shows two possible solutions for the configuration of an industrial building built by standard modules as shown in the Figure 1. Each module is made up of two columns and a planar truss. As observed, it is possible to have the planar truss with different number of bars as candidate solutions during the evolution process. In the Figure 2, distinct structural solutions are shown for the same area to be covered: the first one, at the left side, presents four regular modules whereas the second one, at the right side, three modules. The first solution uses a greater number of modules, probably with "thinner" cross-sectional areas, whereas the second solution uses a smaller number of modules, probably with "thicker" cross-sectional areas. It can also be observed the different number of bays between the trusses in the Figure 1. The role of the GA is to find the better configuration of the steel truss roof and consequently the complete configuration of the industrial building.

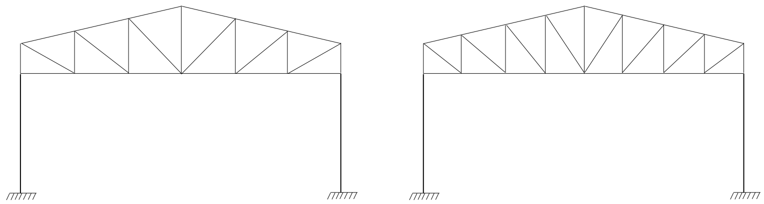

Figure 1: Options of structural confi gurations for a frame presenting different numbers of bays.

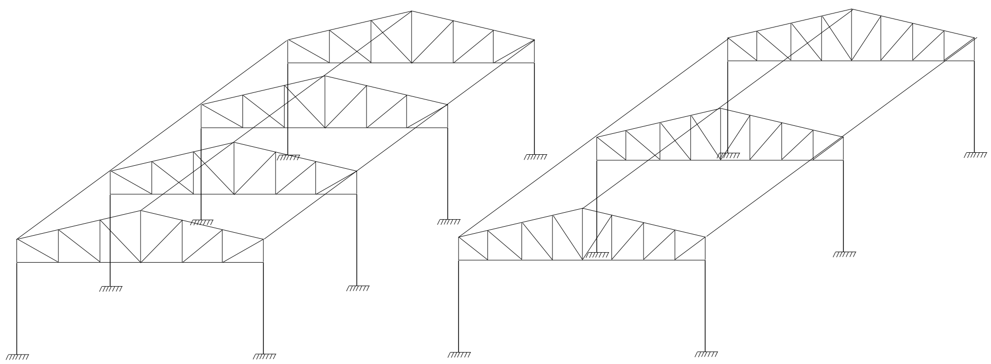

Figure 2: Distinct options for structural confi gurations for an industrial building.

3.1 The optimization problem

The optimization problem proposed here involves as design variables the number of frames, the height of the extreme nodes of the truss and the height of the node in the middle of the truss (continuous). Also, the number of bays, and the cross-section areas of the bars (discrete), linked in three groups, are design variables. One has a mixed discrete-continuous optimization problem and the total number of design variables is equal to 7.

The objective is to minimize the weight of the structure which can be written as:

$$W = \sum_{i=1}^{n} \rho A_i L_i \tag{1}$$

where L_i is the length of the i-th bar of the spatial grid and ρ is the specific weight of the material. The problem is subject to the stress constraints

$$\frac{|\sigma_i|}{\sigma_{max}} - 1 \leq 0 \qquad \text{and} \qquad \frac{|\sigma_i|}{\sigma_E} - 1 \leq 0 \quad \text{if} \quad \sigma_i < 0 \qquad \forall i = 1, 2, ..., n \tag{2}$$

and an upper limit for the nodal displacements in any direction

$$\frac{|u_j|}{u_{max}} - 1 \leq 0 \tag{3}$$

The material properties and constraints are displayed in the Table 1.

Table 1: Material properties and constraints for the industrial building.

Property	Value
Young modulus (E)	2100000.00 kgf/cm^2
Specific weight – steel (ρ)	7850.00 kgf/cm^3
Stress limit (σ_{max})	2500.00 kgf/cm^2
Euler buckling constraint (σ_E)	$4EA/L_i^2$
Displacement limit (u_{max})	5.00 cm

3.2 The design variables

The optimization problem involves a total of seven design variables. The design variable 1 refers to the number of frames, the design variables 2 and 3 refer to the height of the nodes placed at the extreme and in the middle of the truss, respectively given in cm. The fourth variable is the number of bays in a half of the truss roof and, finally, the variables 5, 6 and 7 are the cross sectional areas of the bars, linked in three groups. These design variables are to be chosen from the 32 commercial values listed in the Table 2. The Figure 3 shows details of the design variables 2, 3 and 4.

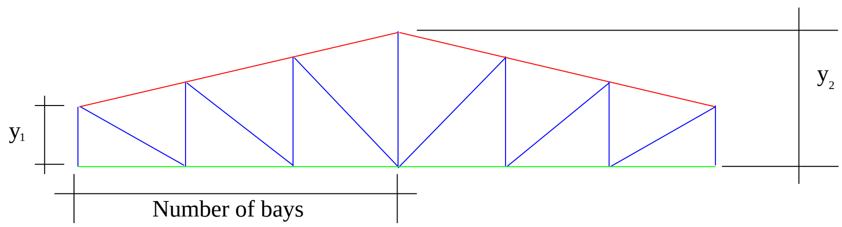

Figure 3: Design variables

In the Figure 3 the red color corresponds to the top chord and the fifth design variable A_1, the green color corresponds to the bottom chord and the sixth design variable A_2 and, finally, the blue color corresponds to the vertical and diagonal bars and the seventh design variable A_3. The variables 5, 6 and 7 are to be chosen from the 32 discrete values of commercially available sizes presented in the Table 2.

Table 2: Commercial cross-sectional areas (cm^2) for the industrial building.

section	area	section	area	section	area	section	area
1	0.71613	9	3.88386	17	10.08385	25	18.58061
2	0.90968	10	4.94193	18	10.45159	26	18.90319
3	1.26451	11	5.06451	19	11.61288	27	19.93544
4	1.61290	12	6.41289	20	12.83868	28	20.19351
5	1.98064	13	6.45160	21	13.74191	29	21.80641
6	2.52258	14	7.92256	22	15.35481	30	22.38705
7	2.85161	15	8.16773	23	16.90319	31	22.90318
8	3.63225	16	9.40000	24	16.96771	32	23.41191

3.3 The fitness function

The fitness function (to be minimized) is given by the expression:

$$f = W + \alpha \sum_i \left(\left(\left[\frac{|\sigma_i|}{\sigma_{max}} - 1 \right]^+ \right)^2 + \left(\left[\frac{[-\sigma_i]^+}{\sigma_E} - 1 \right]^+ \right)^2 \right) + \sum_j \left(\left[\frac{|u_j|}{u_{max}} - 1 \right]^+ \right)^2 \tag{4}$$

where α is a constant penalty coefficient set to 10^6 for all experiments, and $[x]^+ = x$ if $x > 0$, and $[x]^+ = 0$ otherwise. The objective function is the weight of one standard module multiplied by the total number of modules and can be written as:

$$W = N \sum_{i=1}^{n} \rho A_i L_i \tag{5}$$

where N is the number of frames (the first design variable) of a given candidate solution.

3.4 Remark

Since the number of standard modules is a design variable here, the distance between each of them is variable. In this case the loads applied on the nodes of the truss are computed considering influence areas that depend on the distance between the frames leading to different values of equivalent nodal loads for each candidate solution. This scheme is illustrated in the Figure 4. For the experiments reported here only a distributed accidental load p=150.0 Kgf/m^2 is considered. The dead load, wind load, or any other external loads can be included without modifications to the GA code.

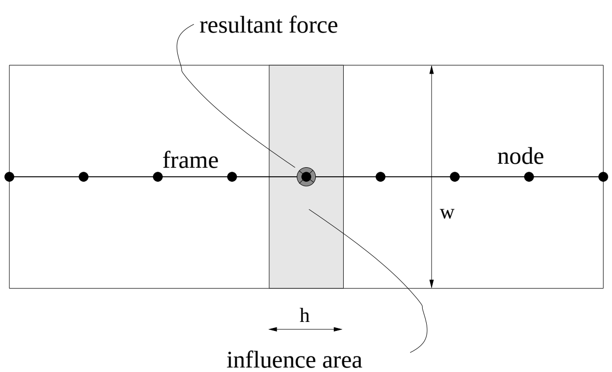

Figure 4: Influence area used to determine equivalent nodal loads to be applied at the standard module.

3.5 Experiments

Four experiments were performed in this optimization problem, varying according to the imposed constraints, as following:

1. Case 1 – constraints in normal stresses;
2. Case 2 – constraints in stresses and displacements;
3. Case 3 – constraints in normal stresses and Euler buckling limits;

4. Case 4 – constraints in normal stresses, Euler buckling limits and displacements.

For each one of these cases, three distinct geometries of the orthogonal in-plan industrial building, are analyzed.The first one is a rectangular area of $l_x = 20$ m $\times$ $l_y = 150$ m, the second one is $l_x = 30$ m $\times$ $l_y = 300$ m and, finally, the third one is $l_x = 40$ m $\times$ $l_y = 300$ m.

3.6 The GA in industrial building optimization

The parameters of the GA, for all experiments are:

1. number of runs: 20,
2. population size: 200,
3. maximum number of generations: 200,
4. crossover probability: p_c=0.8,
5. mutation probability: p_m=0.03.

The binary-coded generational GA implemented uses a Gray code, linear rank-based selection, and elitism (the best element is always copied into the next generation along with 9 copies where one bit has been changed). Standard one-point, two-point, and uniform crossover operators were applied with probabilities $p_c^1 = 0.16$, $p_c^2 = 0.32$, and $p_c^u = 0.32$, respectively.

The Table 3 presents the lower x^L and the upper x^U bounds of the 7 design variables for the first and second geometries. The Table 4 presents the lower and the upper bounds for the third geometry.

Table 3: Design variables, lower and upper bounds, and the number of bits used – first and second geometries.

i	Design variable	x^L	x^U	$nbits$
1	No. of frames	20	40	5
2	y_1	60.00	120.00	9
3	y_2	130.00	250.00	10
4	No. of bays	3	9	3
5	A_1	1	32	5
6	A_2	1	32	5
7	A_3	1	32	5

3.7 Results

The Tables 5, 6 and 7 show the design variables and the total weight of the industrial building found in the three optimization problems analyzed. All of them are feasible solutions.

The GA shown tendentious to chose a smaller number of bays, in several cases, with “thicker” cross-sectional areas of the bars and it might be a logical occlusion. A truss made up of a smaller number of bars with “thick” cross-sectional areas is better than a truss made up of a great number of bars with “thin” cross-sectional areas.

The design variables 2 and 3 correspond to the basic heights of the truss roofs shown tendentious to search values in the direction of the upper limits and it can be justified by the fact of a gain of stiffness using these values.

Table 4: Design variables, their lower and upper bounds, and the number of bits used – third geometry.

i	Design variable	x^L	x^U	$nbits$
1	No. of frames	20	40	5
2	y_1	60.00	300.00	10
3	y_2	130.00	450.00	11
4	No. of bays	5	12	3
5	A_1	1	32	5
6	A_2	1	32	5
7	A_3	1	32	5

The limits of the design variables 2, 3, and 4 for the third geometry are different from the first and second geometries due to the larger area to be covered.

These results were obtained using a Pentium IV PC (2 GHz) which spent approximately 30 second in each independent run. So as to illustrate the configuration of the steel truss the Figures 5, and 6 7 show the final shapes of the first, second and third geometries, respectively.

Table 5: Design variables and the final weight of the structure considering distinct sets of constraints – $l_x = 20$ m $\times$ $l_y = 150$ m.

i	Design variables	Case 1	Case 2	Case 3	Case 4
1	No. frames	38	30	24	21
2	y_1	116.008	117.065	119.530	119.882
3	y_1	239.091	242.023	250.0	234.633
4	No. of bays	3	3	5	4
5	A_1	6.41289	7.92256	9.40000	11.61288
6	A_2	3.63225	4.94193	6.41289	7.92256
7	A_3	5.06451	6.41289	6.41289	8.16773
Weight (Kgf)		11157.737	11248.878	11568.006	11764.01

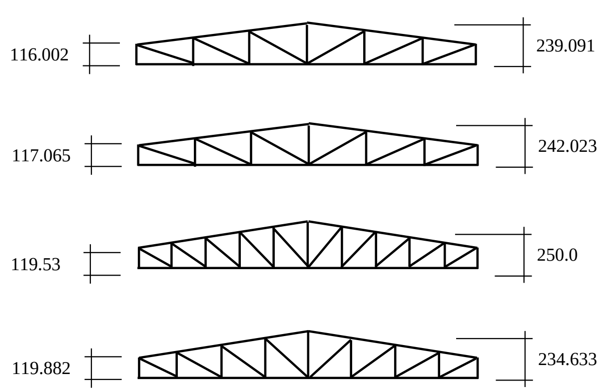

Figure 5: The final shape of the steel roofs of the first geometry.

The Table 8 presents the value of each equivalent nodal load applied at the final optimized steel truss roof respect to each case where P_e and P_i are the loads at the extreme and interior nodes, respectively, of the steel truss roof.

Table 6: Design variables and the final weight of the structure considering distinct sets of constraints – $l_x = 30$ m $\times$ $l_y = 300$ m.

i	Design variables	Case 1	Case 2	Case 3	Case 4
1	No. frames	30	27	23	23
2	y_1	116.947	119.882	118.591	119.295
3	y_1	249.531	248.006	246.012	245.308
4	No. of bays	3	5	5	5
5	A_1	16.90319	18.90319	22.38705	22.38705
6	A_2	10.08385	12.83868	15.35481	15.35481
7	A_3	13.74191	11.61288	13.74191	13.74191
Weight (Kgf)		33271.567	33585.727	33889.26	33890.981

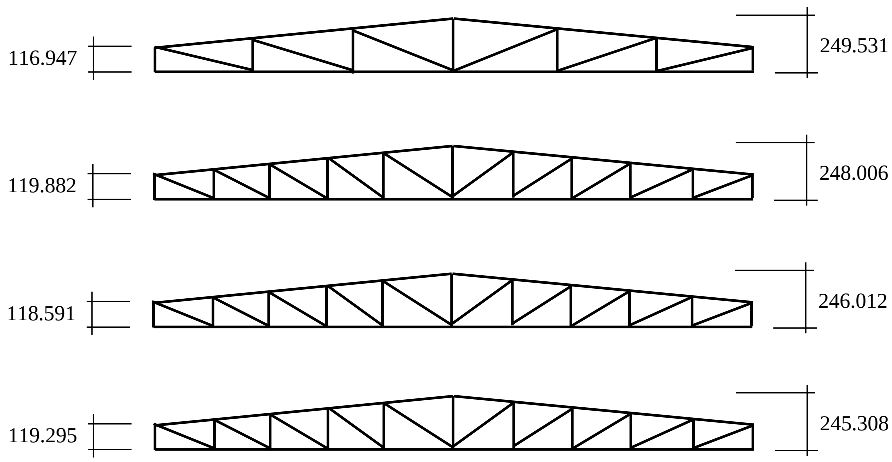

Figure 6: The final shape of the steel roofs of the second geometry.

Table 7: Design variables and the final weight of the structure considering distinct sets of constraints – $l_x = 40$ m $\times$ $l_y = 300$ m.

i	Design variables	Case 1	Case 2	Case 3	Case 4
1	No. frames	31	31	22	22
2	y_1	292.727	294.135	261.759	261.525
3	y_1	449.531	449.687	448.280	437.962
4	No. of bays	5	5	6	6
5	A_1	15.35481	15.35481	22.38705	22.90318
6	A_2	10.08385	10.08385	15.35481	15.35481
7	A_3	9.40000	9.40000	12.83868	12.83868
Weight (Kgf)		46231.197	46264.915	48863.260	48985.867

4. CONCLUSIONS

In this paper a GA is proposed to solve the shape, sizing and structural configuration weight minimization problem for an industrial building consisted of steel roof structures. In a previous work similar structures were submitted to an optimization problem where the steel roof had a fixed topology.

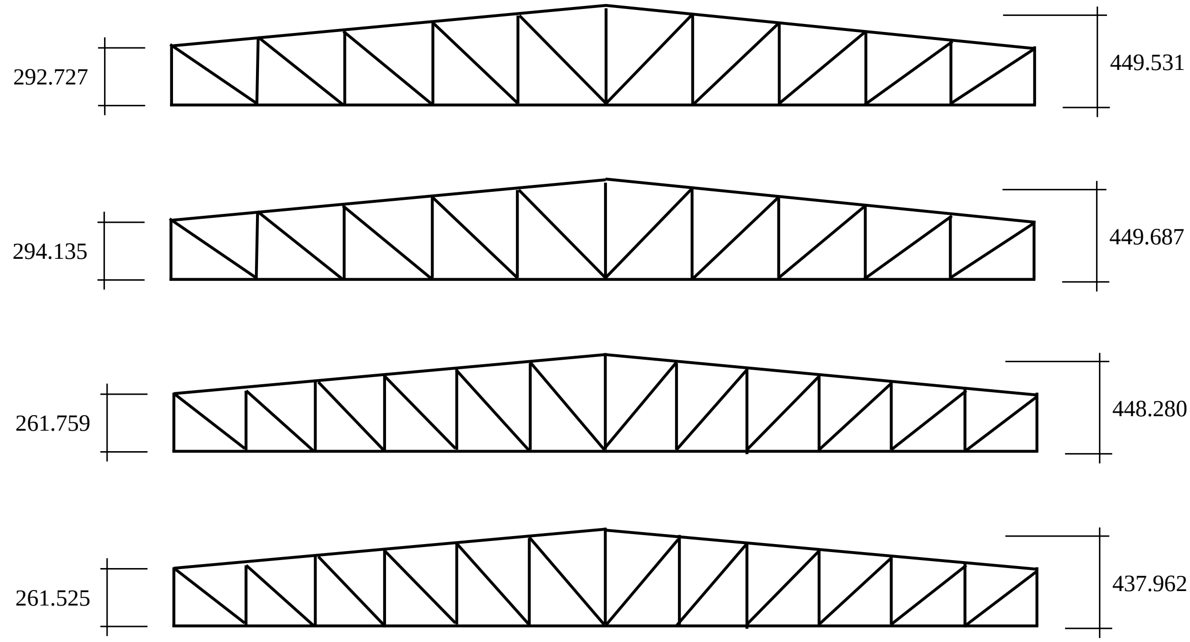

Figure 7: The final shape of the steel roof of the third geometry.

Table 8: Equivalent nodal loads (in Kgf) applied at the final optimized steel truss roof.

Geometry	load	Case 1	Case 2	Case 3	Case 4
1	P_e	1013.514	1293.104	978.261	1406.250
	P_i	2027.027	2586.207	1956.522	2812.500
2	P_e	1293.104	865.385	1022.727	1022.727
	P_i	2586.207	1730.769	2045.455	2045.455
3	P_e	750.000	750.000	892.857	892.857
	P_i	1500.000	1500.000	1785.714	1785.714

The advantage of considering the different topologies for the steel roofs can be justified by the fact that it is not necessary to chose a pre-defined number of bars for the truss. If this choice has a range of options it can be set as a design variable within the GA. Although the experience of the designer may help in this case, that possibility is an attractive feature because it allows the designer to automate the search of the global configuration of the industrial building simultaneously with the search of topology, shape and sizing design variables for the steel truss roofs.

As a result, a much larger space of possible solutions is searched, potentially leading to counter-intuitive or non-traditional solutions. The results found in the optimization process help the designer providing, if not definitive, valuable preliminary conclusions guiding to good final solutions while avoiding the repetitive evaluation of alternative designs in a time consuming process.

Acknowledgments

The authors acknowledge the support received from CNPq and the LNCC – Laboratório Nacional de Computação Científica.

REFERENCES

Angeline, P., 1995. Morfogenic evolutionary computations: Introduction, issues and examples. In Reynolds, J. M. B. & Fogel, D., eds, *Fifth Annual Conference on Evolutionary Programming*. MIT Press.

Duponcheele, G. & Tilley, D., 1995. Shape optimization by GA. In *Proceedings of the INWGA*, pp. 43–68, Vaasa, Finland.

Gil, L. & Andreu, A., 2001. Shape and cross-section optimisation of a truss structure. *Computers & Structures*, vol. 79, pp. 681–689.

Goldberg, D., 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Pub. Co.

Goldberg, D. & Samtani., M., 1986. Engineering optimization via genetic algorithms. In *9th Conf. Electronic Computation*, pp. 471–482, New York, N.Y.

Haftka, R., 1986. Structural shape optimization – a survey. *Computers Methods in Applied Mechanics and Engineering.*, vol. 57, pp. 91–106.

Hamza, K., Mahmoud, H., & Saitou, K., 2003. Design optimization of N-shaped roof trusses using reactive taboo search. *Applied Soft Computing.*, vol. 3, pp. 221–235.

Hansen, S. & Vanderplaats, G., 1990. Approximation method for configuration optimization of trusses. *AIAA Journal*, vol. 28, n. 1, pp. 161–168.

Koumousis, V. & Georgiou, P., 1994. Genetic algorithms in discrete optimization of steel truss roofs. *Journal of Computing in Civil Engineering*, vol. 8, n. 3, pp. 309–325.

Lemonge, A. & Barbosa, H., 2000. A genetic algorithm for shape, sizing and configuration optimization of framed structures. In *proc. of 21st Iberian Latin American Congress on Computational Methods in Engineering in CD-ROMProc. of the 9th Conference in Electronic Computation*, Rio de Janeiro.

Lemonge, A. & Barbosa, H., November 19-21, 1998. A genetic algorithm for shape and sizing optimization of truss structures. In *III Simpósio Mineiro de Mecânica Computacional*, pp. 269–277, Ouro Preto, Minas Gerais.

Lemonge, A., Barbosa, H., & Fonseca, L., 2003. A genetic algorithm for the design of space framed structures. In *XXIV CILAMCE – Iberian Latin-American Congress on Computational Methods in Engineering*, Ouro Preto, Brasil.

Salajegheh, E. & Vanderplaats, G., 1993. Optimum design of trusses with discrete sizing and shape variables. *Structural Optimization*, vol. 6, pp. 79–85.

Smith, J., Hodgins, J., & Oppenheim, I., 2002. Creating models of truss structures with optimization. Technical report, Carnegie Mellon, http://www-2.cs.cmu.edu/ jeffrey/.

Syswerda, G., 1995. Uniform crossover in genetic algorithms. In Shaffer, J., ed, *Third International Conference on Genetic Algorithms and Their Applications*, pp. 2–9. Morgan Kaufmann, San Mateo, CA.