



A Linear Approach for Sizing Optimization of Isostatic Trussed Structures Subjected to External and Self-Weight Loads

Flavio Avila Correia Martins¹ · Juan Pablo Julca Avila¹ · Marcelo Araujo da Silva¹

Received: 4 August 2018 / Accepted: 4 December 2018 / Published online: 10 December 2018
© Korean Society of Steel Construction 2018

Abstract

The implementation of a new linear method to optimum weight design of trussed structures subjected to external and self-weight loads is proposed. Design variables are the cross-section areas of the members. Inequality constraints are written based on the force-method for isostatic structures considering maximum and minimum axial stress criteria. The novelty of the proposed approach is the benefit created from the combination of a linear inequality-constrained formulation with interior-point methods to tunnel the solution rapidly and monotonically towards the minimum value through feasible space, also eliminating the need to directly explore the finite-element model. To evaluate the performance of the algorithm, trusses are subject to optimization processes based on different techniques: (i) the proposed method, called by “indirect-method”; (ii) a design problem with constraint evaluated directly from the finite-element model; (iii) optimization based on Genetic Algorithms. The three methods are compared using trusses with 10, 37 and 1240 bar-elements. The results showed that the indirect-method was able to provide great performance for complex topologies, returning weight designs up to 70 times lighter in 1% of the time required by a Genetic Algorithm.

Keywords Sizing optimization · Truss optimization · Linear programming · Optimization with MATLAB

1 Introduction

Optimization of trussed structures is considered a state-of-art area of study and many authors have made great efforts to overcome the barriers of this field. Mostly, design problems are characterized by large numbers of variables and constraints, which usually insert as many local minima points and non-linearities in the solution (Asl et al. 2013) and, in most cases, the optimum design must be assumed in order to avoid confronting these. The optimal design methodology of structures is generally categorised by: sizing, shape and topology optimization. The former consists on finding the optimal set of structural parameter such as length, area or thickness. Shape optimization is intended to find an optimal shape by changing the geometrical configuration. Topology optimisation searches for a shape that satisfies the design criteria, but without a predefined configuration (Gan et al. 2017). This study implements sizing optimization for

optimum weight design of trussed structures. Design variables are the cross-section areas of the elements. Space of solution is restricted by maximum and minimum axial stress criteria, while the slenderness of bar-elements is controlled by inequality constraints.

When it comes to optimizing structures, most of produced research have the goal of minimizing weight (Mohr et al. 2011). Also, most algorithms are initially developed for trusses before their use are extended to higher order structures. Well-known examples of such progression include the technique described by Rabadi and Hanna Al Rabadi (2014), a harmony-search method, which has been progressively implemented in higher order systems (see de Almeida 2016). Other pioneering studies, such as those of Rajan (1995), Wang et al. (2004), Stolpe (2016), Doan and Lee (2018) and Banh and Lee (2018), have also created new paradigms for future studies. The overall premise of these methods is to find a structure that saves as much material as possible while preserving its reliability. Many methodologies have been created to achieve this goal, although the state-of-art of recent works is directed towards improving the efficiency of algorithms, rather than exploring powerful yet high-costly techniques. Nevertheless, the objective function in structural

✉ Flavio Avila Correia Martins
flavio.martins@outlook.com

¹ CECS, UFABC, PO Box 09210-580, Santo Andre, SP, Brazil

optimization problems is usually defined in the design space, while the responses of the constraints are designed in the behaviour response space, using the structural analysis for relating both. This aspect generally leads structural optimization problems to non-linear relations.

So far, it is believed that Evolutionary Algorithms (EA) are particularly suited to handle very complex structures, since they search for the solution mimicking the biological evolution, adapting and selecting discrete variables that produced the best solutions in previous steps (Dominguez et al. 2006). These models became further improved when Holland (1973) introduced what he called by Genetic Algorithm (GA), inserting mutation and recombination (crossover) between the “specimens” along the solution. However, his model had some down-points, such as the fact that its computational cost was very high, while it was still vulnerable to local minima (Sivanandam and Deepa 2008). In addition, the main reason for GA to be widely implemented is due to the lack of necessity for gradient evaluation and sensitive analysis (Bölte and Thonemann 1996), making it a zero-order optimization type. Interesting aspects of GA programming are discussed by Assimi et al. (2017). Other similar approaches are investigated by Kripka (2004) and Farshi and Alinia-Ziazi (2010).

In this work, we present a linear-formulated method, which benefits by being more efficient than non-linear formulations (Ferrier and Lovell 1990). However, linear systems require the removal of classic non-linear constraints and objective functions, such as those on the first natural frequency (Mortazavi and Toğan 2017; Gomes 2011), maximum displacement (Camp and Farshchin 2014) and compliance (Bendsøe et al. 1994), leaving only the admissible stress criterion to constrict the problem. Nevertheless, structures must also be isostatic, since the linearity of the problem is lost in hyperstatic models (Koski 1981). Although this may be true, the decrease in computational time and cost in linear formulations are outstanding considering the average performance of non-linear approaches, outweighing these in most design problems where both formulations are feasible.

Some recent research on optimization of isostatic structures have shown good results in terms of computational efficiency, such as those of Wang et al. (2002) and Hultman (2010), in which elements and nodes that have minor contribution to the stiffness of the structure are removed until it reaches a given objective or become statically indeterminate. In addition, Lamberti and Pappalettere (2003) have described an efficient move limit definitions incorporating a trust region method. Recently, Farshi and Alinia-Ziazi (2010) proposed a method that implements a constraint function based on the force-method to tunnel the solution. They have also introduced stress and displacement constraints into the inscribed hyper-spheres method, performing a fair design-problem. On the other hand, one of the main down-points of such methods

is fact that the self-weight is not considered in the optimization process. In addition, the efficiency of these approaches is relatively low, requiring a few minutes to process design problems with hundreds of variables. In the present method, contrarily, self-weight loads are considered and the algorithm is able to optimize complex structures in seconds using ordinary desktop PCs.

In this study, the use of the force-method to optimize the cross-section areas of trussed structures is investigated. It is proposed a linear inequality-constraint formulation that represents the response of the structure in a similar fashion to that of a finite-element model, removing the need to embed it into the algorithm. This strategy leads to a variety of advantages: (i) the Hessian matrix requires less processing power to be generated (Lyamin and Sloan 2002); (ii) combined with interior-point methods, this approach allows the solution to be tunnelled through feasible solution space in a monotonic, fast drop trend; (iii) the present procedure relies only on linear-programming, making it more efficient in terms of computational cost; (iv) the analysis step is embedded within the optimization stage, avoiding tedious separate analyses; (v) it allows the use of multi-criteria optimization techniques based on inequality-constraint functions, such as slenderness control (see Sect. 2.2); (vi) the solution is non-dependent on the initial value, providing an advantage in design problems of complex structures. Our method, on the other hand, is only feasible in linear and statically determinate structures, subjected to small displacements.

The optimization of three isostatic structures subject to their own weight and external loads is carried out. Each system has 10, 37 and 1240 bar-elements with three fixed degrees-of-freedom. The first two models have been widely implemented in algorithms and benchmark tests (see Asl et al. 2013; Kripka 2004; Assimi et al. 2017; Camp and Farshchin 2014; Gomes 2011; Mortazavi and Toğan 2017; Kaveh and Khayatazad 2013; Miguel and Miguel 2012; Tejani et al. 2017; Frans and Arfiadi 2014; Farshchin et al. 2016; Cazacu and Grama 2014; Ringertz 1985), while the latter is a complex model developed for this work. The efficiency test will be performed by verifying the results of three approaches for the same initial conditions and boundaries. The three implemented methods are: (i) a GA code written by Farshchin et al. (2016); Camp and Farshchin (2014) (referred to as B&A (2014) in this paper). (ii) optimization based on the direct exploitation of the FE model, called by “direct-method” (DM) and, the approach developed for this work, (iii) the “indirect-method” (IM), which is a first-order optimization type, requiring gradient evaluations.

2 Methodology

2.1 Statement of the optimization problem

The optimization problem aims to minimize the cross-section areas of the bar-elements. For a truss with n elements, the problem statement is given by:

Find:

$$\min f(\mathbf{x}) = m(\mathbf{x}) = \rho \sum_{j=1}^n L_j A_j = \rho \sum_{j=1}^n L_j x_j$$

Subjected to:

$$|\sigma_i| \leq \sigma_{adm} \text{ for all } i \in [1, 2, \dots, n] \quad (1)$$

For x_i in:

$$\{L_b\} \leq \{x_i\} \leq \{U_b\}$$

where L_i , x_i , σ_i are the length, area and stress associated with the i -th element, respectively. The vector of design variables is given by $\mathbf{x} = \{x_1, x_2, \dots, x_n\}^T$. Further, σ_{adm} is the admissible stress, while the slenderness of an element is limited by lower and upper admissible cross-section areas, L_b and U_p . For the GA, the fitness function is the structural mass and the constraint is the admissible stress. The maximum number of generations has been set to 10 and the number of

admissible stalled generations, to 5. Also, the initial population is selected based on the number of bar-elements, N_{ele} , using:

$$P_{size} = \min(\max(10N_{ele}, 40), 100) \quad (2)$$

as proposed by Chambers (1995). All variables and parameters are listed in Table 1. For the three simulated structures, the mechanical characteristics were based on a generic aluminium alloy, while the minimum and maximum values of areas were defined as $U_b = 1$ and $L_b = 5 \times 10^{-6} m^2$, respectively. The solution is constricted by the maximum axial stress on each member, such as $|\sigma_i| \leq \sigma_{adm}$. All mechanical characteristics and imposed constraints are also listed in Table 1. The stress values that feed the Genetic Algorithm, which has been developed by B&A (2014), are computed using the FE code written in MATLAB by Kaveh and Rahami (2006). The FE code that feeds the IM and DM was written by the authors.

2.2 Formulation of the IM and DM approaches

Equations were developed based on the force-method: the stress on the i -th member is a linear combination of stresses generated by all load sources, including the weight of a j -th element and external forces composition, $F_1 + F_2 + \dots + F_k$, generating a single σ_i value. Hence, for the i -th element:

$$\sigma_i = \sigma_{i,1} + \sigma_{i,2} + \dots + \sigma_{i,n} + \sigma_{F,i} \quad (3)$$

Table 1 Parameters implemented in the simulations

Characteristics of the GA	Values
Maximum number of generations	10
Maximum stalled generations	5
Initial population size	10-Elements: 100 individuals 37-Elements: 100 individuals 1240-Elements: [30, 60, ..., 180] individuals
Characteristics of the IM and DM	
Function variation tolerance	$\Delta f(x)_{min} = 1 \times 10^{-8}$
Maximum number of function evaluations	$N_{f,max} = 1 \times 10^6$
Mechanical characteristics	
ρ	2800 kg/m ³
E	80×10^9 Pa
σ_{adm}	50×10^6 Pa
Boundaries	
$x_{i,0}$	1 m ² for all $i \in [1, \dots, n]$
L_b	5×10^{-6} m ²
U_b	1 × 10 ⁰ m ²

Mechanical characteristics and Boundaries are common for the GA, IM and DM

in which, σ_{ij} is the stress over the i -th element due to the weight of the j -th, and $\sigma_{F,i}$, due to the composition of external forces. Considering a system with all cross-section areas equal to $A_i = 1m^2$, these values will act as coefficients of the influence of the j -th load source on the stress on the i -th element, given by $C_{i,j} \equiv \sigma_{ij}|_{A_j=1}$ and $C_{F,i} \equiv \sigma_{F,i}|_{t=0}$, leading to:

$$\sigma_i = \sum_{j=1}^n C_{i,j} \frac{\rho L_j x_j}{\rho L_{j,0} x_{j,0}} \frac{x_{i,0}}{x_i} + C_{F,i} \frac{x_{i,0}}{x_i} \tag{4}$$

In Eq. 4, $x_{i,0}$ and x_i are the cross-section areas of the i -th element before the optimization process and in current state, respectively. $L_{j,0}$ and L_j represent the original and current length of the j -th member. Defining $x_{i,0} \equiv 1$ and considering small displacements, $L_j \approx L_{j,0}$, Eq. 4 may be re-written such as:

$$\sigma_i = \sum_{j=1}^n C_{i,j} \frac{x_j}{x_i} + \frac{C_{F,i}}{x_i} \tag{5}$$

From the proposed restriction, $|\sigma_i| \leq \sigma_{adm}$, there are two possible sets of criteria: one for $\sigma_i, \sigma_{adm} > 0$ and another for $\sigma_i, \sigma_{adm} < 0$. Therefore, Eq. 1 becomes:

Find:

$$\min f(\mathbf{x}) = m(\mathbf{x}) = \rho \sum_{j=1}^n L_j A_j = \rho \sum_{j=1}^n L_j x_j$$

Subjected to:

$$\begin{aligned} \text{(if } \sigma_i \geq 0) \quad & \sum_{j=1}^n C_{i,j} x_j - \sigma_{adm} x_i \leq -C_{F,i} \\ \text{(if } \sigma_i < 0) \quad & \sum_{j=1}^n -C_{i,j} x_j + \sigma_{adm} x_i \leq C_{F,i} \end{aligned} \tag{6}$$

For x_i in:

$$\{L_b\} \leq \{x_i\} \leq \{U_b\} \equiv \{x_{i,0}\}$$

The constraints as defined in Eq. 6 can be re-written such that $[A]\{x\} \leq \{b\}$, with $[A]$ and $\{b\}$ given by:

$$\begin{bmatrix} (C_{1,1} - \sigma_{adm}) & \dots & C_{1,n} \\ C_{2,1} & \dots & C_{2,n} \\ \vdots & \vdots & \vdots \\ C_{n,1} & \dots & (C_{n,n} - \sigma_{adm}) \\ (-C_{1,1} + \sigma_{adm}) & \dots & -C_{1,n} \\ \vdots & \vdots & \vdots \\ -C_{n,1} & \dots & (-C_{n,n} + \sigma_{adm}) \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} \leq \begin{Bmatrix} -C_{F,1} \\ -C_{F,2} \\ \vdots \\ -C_{F,n} \\ C_{F,1} \\ C_{F,2} \\ \vdots \\ C_{F,n} \end{Bmatrix} \tag{7}$$

For the IM, the coefficients $C_{i,j}$ are obtained individually from the FE model before starting the optimization process, simulating the structure $N_{ele} + 1$ times, each containing

only the j -th load source, including one time to compute the stresses due to the external forces, $C_{F,i}$. After this step is completed, the algorithm builds Eq. 7 automatically and the FE code will not be necessary to perform any further evaluations during the optimization process. The algorithms for the DM and IM are similar, although the former does not make use of Eq. 7, evaluating the constraint function based on the direct exploitation of the FE model, in all loops of the optimization process.

The flow chart for the DM is shown in Fig. 1. The convergence criterion for the objective function is $\Delta f(x)_{min} = 1 \times 10^{-8}$, with a maximum number of function evaluations of $N_{f,max} = 1 \times 10^6$. The flow chart for the IM is shown in Fig. 2. Simulation characteristics are listed in Table 1. For the IM, the work-flow is given by: an intermediate matrix $[C] = [C_{i \times j}]$, which definition is shown in Fig. 2, must be fed $n = N_{ele}$ times, allocating the stress on the i -th element in the i -th row and j -th column. The process follows to the creation of the matrix $[A_{2n \times n}]$ and the vector $\{b_{2n \times 1}\}$, defined in Eq. 7, which feeds the optimization process with the necessary linear constraints. Since the objective function is the structural mass, with cross-section areas as variables, the design space includes only size quantities.

The linear approach presented in this work takes great advantage of interior-points (IP) methods, since only inequality constraints are applied with a linear formulation, tunnelling the solution through feasible space. Given that IP algorithms benefit from fast gradient evaluations, full linear methods such as the IM provide great advantages to the solver, since the necessary quantities to evaluate the gradients are given in the form of constants, $C_{i,j}$, directly from the objective function, $m(\mathbf{x})$, without the need to perform any further calculations, resulting in a fast drop tendency of the solution (Lustig et al. 1994).

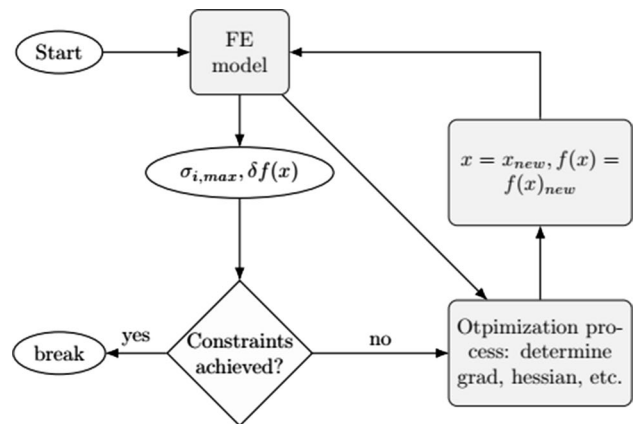


Fig. 1 Flow chart for the DM

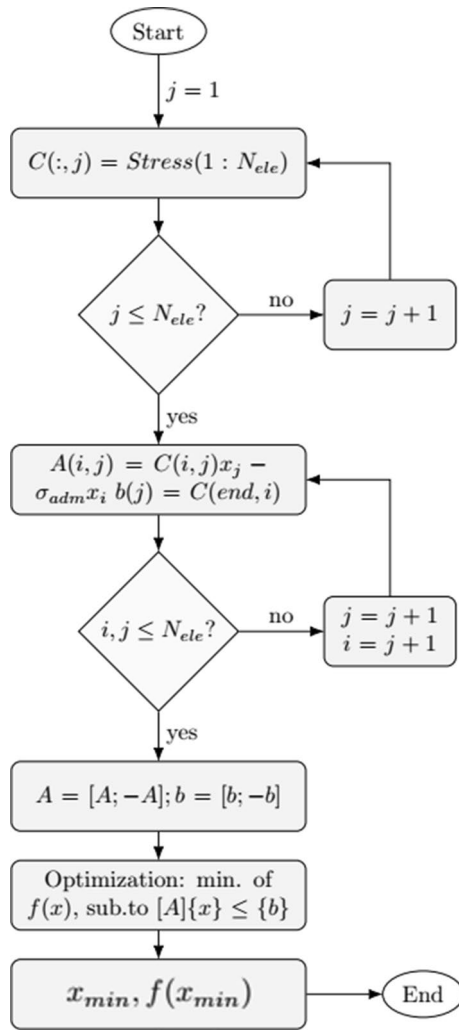


Fig. 2 Flow chart for the IM

3 Results

3.1 10-Elements truss

For the first optimization problem, a 10-elements truss with six nodes is implemented. The system is isostatic and it is illustrated in Fig. 3. All nodal forces have magnitude of $F = 1000\text{ N}$. The length of the vertical and horizontal elements are given by $L = 1\text{ m}$. Results are summarized in Table 2.

Results for the current benchmark are shown in Fig. 4. For the 10-elements truss, the best result has been obtained with the DM, having a final mass 16.6% lower than that of the second best, with IM, and 84.3%, compared with the GA. The performance of the proposed model, IM, is better than those of the two others, obtaining its final results within 27 and 5% the time required by DM and GA, respectively. By comparing Figs. 4a–c, it is clear that the DM was able to

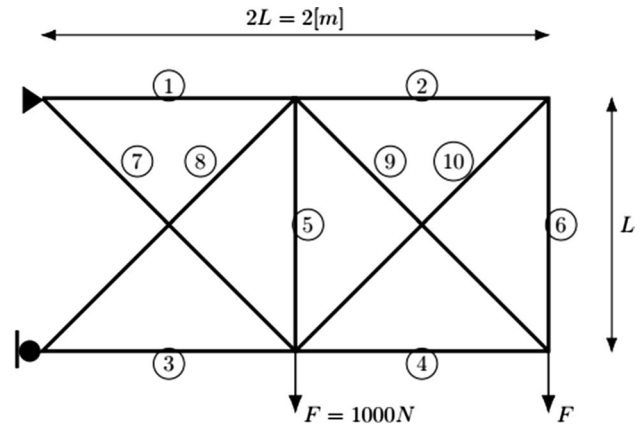


Fig. 3 10-elements truss subjected to the optimization process

perform a near perfect optimization process, since most variables are relatively close to reach the lower boundary, while the constraints are also approximately reached. Despite this, this method has required far more function evaluations than has the IM. The GA algorithm, as predicted (see Farshchin et al. 2016; Rajan 1995), proved to be less interesting when it comes to computational performance. The number of function evaluations of the GA has exceeded the hundreds.

In Fig. 5 the results of the three approaches are shown. The elements in red have $|\sigma| \geq 0.9\sigma_{adm}$, in black, $0.5\sigma_{adm} < |\sigma| < 0.9\sigma_{adm}$ and in gray, $|\sigma| \leq 0.5\sigma_{adm}$. The best results have been obtained with the DM, followed by IM and GA, respectively. The illustrations make it clear that the Genetic Algorithm has not been able to impose an appreciable approximation to the boundaries under the given conditions.

3.2 37-Elements truss

The second benchmark has been performed with a 37-elements truss with 20 nodes. As for the previous model, it is an isostatic structure, as it is shown in Fig. 6. The model is subjected to 9 vertical external forces applied on the upper nodes. Results are summarized in Table 3.

Outputs from the IM and DM are quite similar in terms of final weight, being the final structure obtained with the latter only 0.9% lighter than with the former, with $m(x_{min})_{IM} = 13.35\text{ kg}$. Regardless of this, the IM was able to perform it in only 20% the time the “brute” method required.

Table 2 Results for the 10-elements truss case

Approach	Results
IM	$N_f = 4, T = 1.966\text{ s}, m_{min} = 0.8333\text{ kg}$
DM	$N_f = 76, T = 7.210\text{ s}, m_{min} = 0.7147\text{ kg}$
GA	$N_f = 5319, T = 37.896\text{ s}, m_{min} = 1.317\text{ kg}$

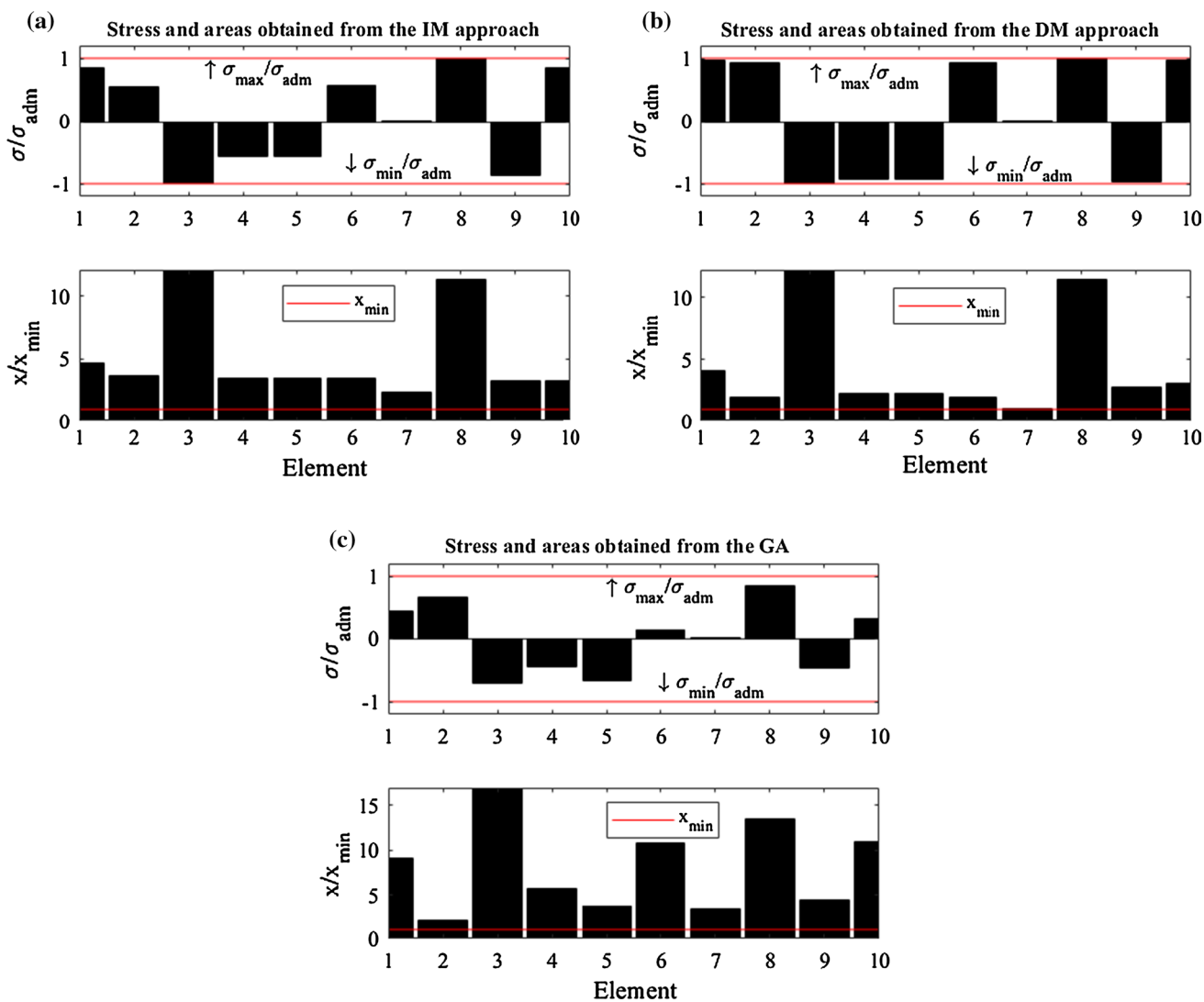


Fig. 4 Results for the **a** IM, **b** DM, **c** GA for the sizing optimization of a 10-elements truss. The lines in red indicate the boundary conditions. The numbering of the elements follows the order shown in Fig. 3. (Color figure online)

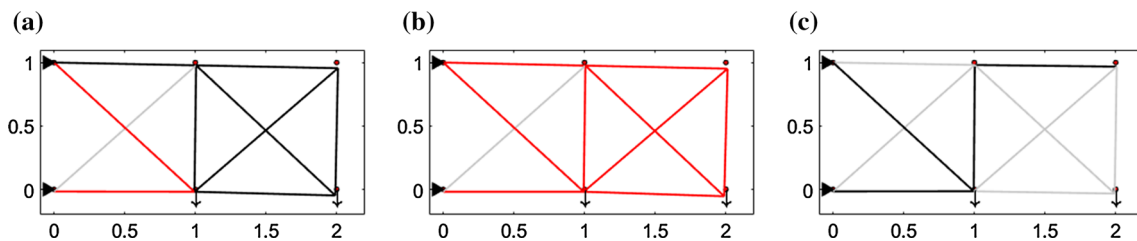


Fig. 5 Results for the **a** IM, **b** DM, **c** GA for the sizing optimization of a 10-elements truss. The elements in red have $|\sigma| \geq 0.9\sigma_{adm}$, in black, $0.5\sigma_{adm} < |\sigma| < 0.9\sigma_{adm}$ and in gray, $|\sigma| \leq 0.5\sigma_{adm}$. (Color figure online)

In comparison to the GA, IM presented better results, optimizing the structure twice as much in less than 2.3% the time required by it, as shown in Table 3. The IM has been able to maximize the stresses on all elements to values close

to the admissible, while the minimum area found is only 3 times higher than the minimum allowed. In addition, it is noticed that the final areas distribution for the DM and IM are quite similar.

Table 3 Results for the 37-elements structure

Approach	Results
IM	$N_f = 4$, $T = 2.210$ s, $m_{\min} = 13.35$ kg
DM	$N_f = 76$, $T = 10.776$ s, $m_{\min} = 13.23$ kg
GA	$N_f = 53211$, $T = 97.058$ s, $m_{\min} = 28.47$ kg

When compared with the optimization of the 10-elements structure, the number of function evaluations performed by IM and DM remained the same, while the GA has performed 10 times more function evaluations than in the previous model. The fact that, for the IM, this number has not increased by the problem size is considered an advantage of the proposed method. From the analysis of the results of the IM and DM, in Fig. 7a, b, it is noticed that the former has left four elements less in the interval of $|\sigma| > 0.9\sigma_{adm}$ when compared to the latter, giving some advantage to the DM. Figure 8 shows the distribution of stress intervals of the present benchmark. Another feature observed in our results is the fact that the GA has not been able to obtain a symmetrical solution from the optimization process, which is shown in Fig. 7c, stalling before the expected final configuration was obtained, even though large margins had been imposed for the solution break.

3.3 1240-Elements truss

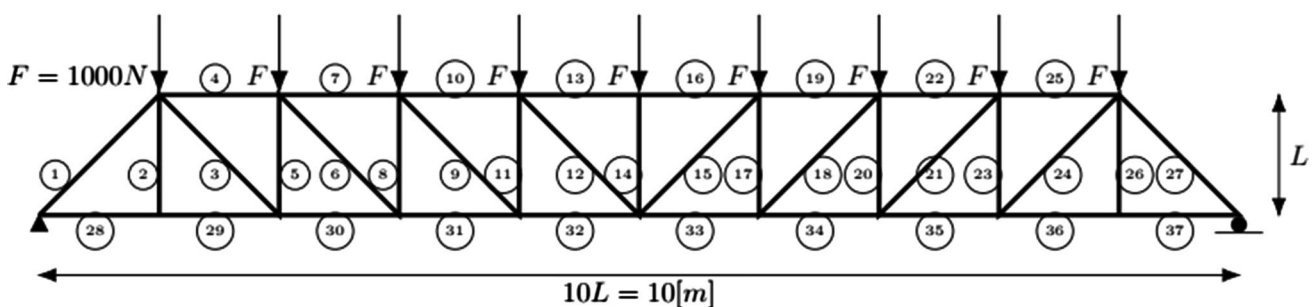
The last benchmark conducted in this study has a 1240-elements truss with 420 nodes, a far greater number than most studies usually implement. The topology of the system is quite complex and may require a few hours to be optimized by less efficient methods. The vector of external forces is shown in Fig. 9.

The numbering of elements has been omitted to make the illustration clearer, although the order follows as: the horizontal members are numbered first, line by line, starting

from the bottom left to the upper right. Sequentially, the vertical members are numbered line by line, from the bottom left to the upper right. Then, the inclined elements are numbered, beginning with the ones at 45° and, finally, the elements inclined in 135° to the horizontal axis. The complete structure is 1×1 m² and the magnitude of the external forces is equal to $F = 1000$ N.

Results of the three algorithms are listed in Table 4. Great differences in performance are observed in the different approaches. The time required by GA and DM are 10 and 100 times greater than the value of 92.14 s from the IM, respectively. Nevertheless, IM presented remarkably better results in terms of weight, given that the objective function converged to $m = 1.032$ kg, which is 5 and 67 times smaller than that of the two others, as shown in Table 4. In addition, there was no appreciable approximation to the boundaries. The population size which achieved the lowest mass for the GA is $P_{size} = 120$ individuals.

Figure 10 shows the performance comparison of the three approaches. Excellent results have been obtained from the IM: although only a small portion of the design variables have approached the stress constraint, the minimum cross-section area value, L_b , has been reached by most of them, indicating that the method could further improve its results. The DM presented poor performance when compared with IM. Starting from the premise that IM is just a conversion of the DM that avoids the use of the FE model throughout the optimization process, similar results were expected from them, indicating that the DM was not able to perform well with so many variables. The stress distributions are shown in Fig. 11. DM and GA presented poor performance, as almost no element could be optimized to $|\sigma_i| \geq 0.5\sigma_{adm}$, even though both methods have performed the optimization process for greater computational times. Another remarkable feature of the IM is the fact that only 12 function evaluations have been performed, while the DM and GA demanded 99 and 9900 function evaluations, respectively.

**Fig. 6** 37-elements truss subjected to the optimization process

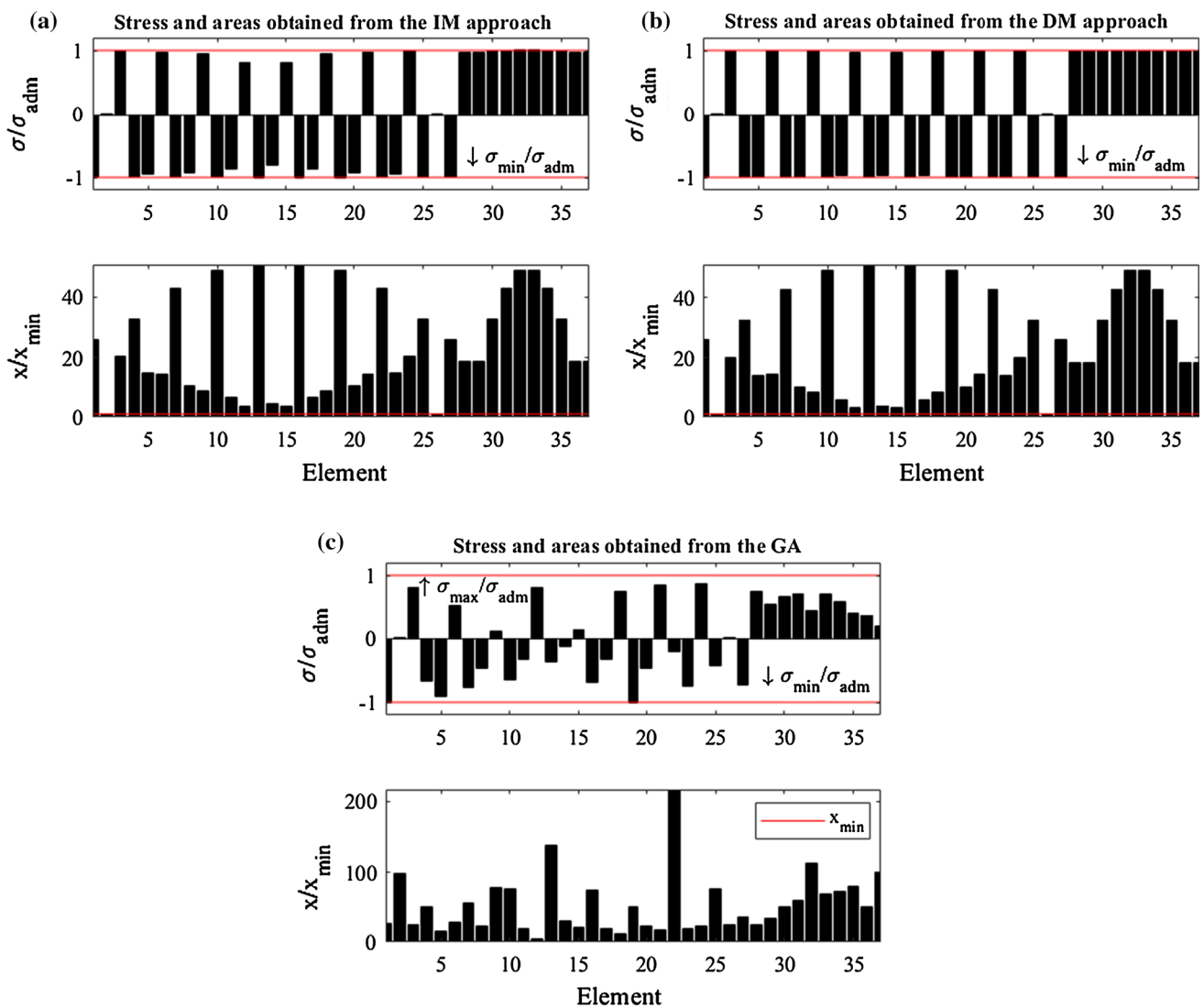


Fig. 7 Results for the **a** IM, **b** DM, **c** GA for the sizing optimization of a 37-elements truss. The lines in red indicate the boundary conditions. The numbering of elements follows the order shown in Fig. 6. (Color figure online)

4 Discussion

As discussed in the previous sections, the formulation of the sizing optimization problem based on linear inequality-constraint equations provided greater performance than those of the two other approaches. By comparing the DM with the IM, it is concluded that by eliminating the need to exploit the FE model throughout the optimization process, great decreases in the computational time are observed, specially for complex topologies. This technique, on the other hand, presents advantages in conditions such that the number of function evaluations is greater than the size of the design variables vector, since the IM will necessarily exploit the FE model $n = N_{ele} + 1$ times. This aspect is noticed from the comparison of the computational times, T , presented by the IM and DM for the 10-elements truss

case, shown in Table 2, in which the gain in performance from the DM to the IM is quite small, but an outstanding gain in performance between the two methods is observed for the 1240-elements case, which is shown in Table 4.

Even though the GA is a well-known and vastly implemented technique, it demands way more sub-steps and function evaluations in order to optimize a given system. These characteristics become crucial in design problem of structures based on the finite-element method, given that directly exploiting the numerical model is intensive computationally, increasing even more the computational time, which is avoided in the IM. It should also be mentioned that the theory predicts a polynomial time to solve the design problem using IM, based on the overall performance of linear methods (Potra and Wright 2000).

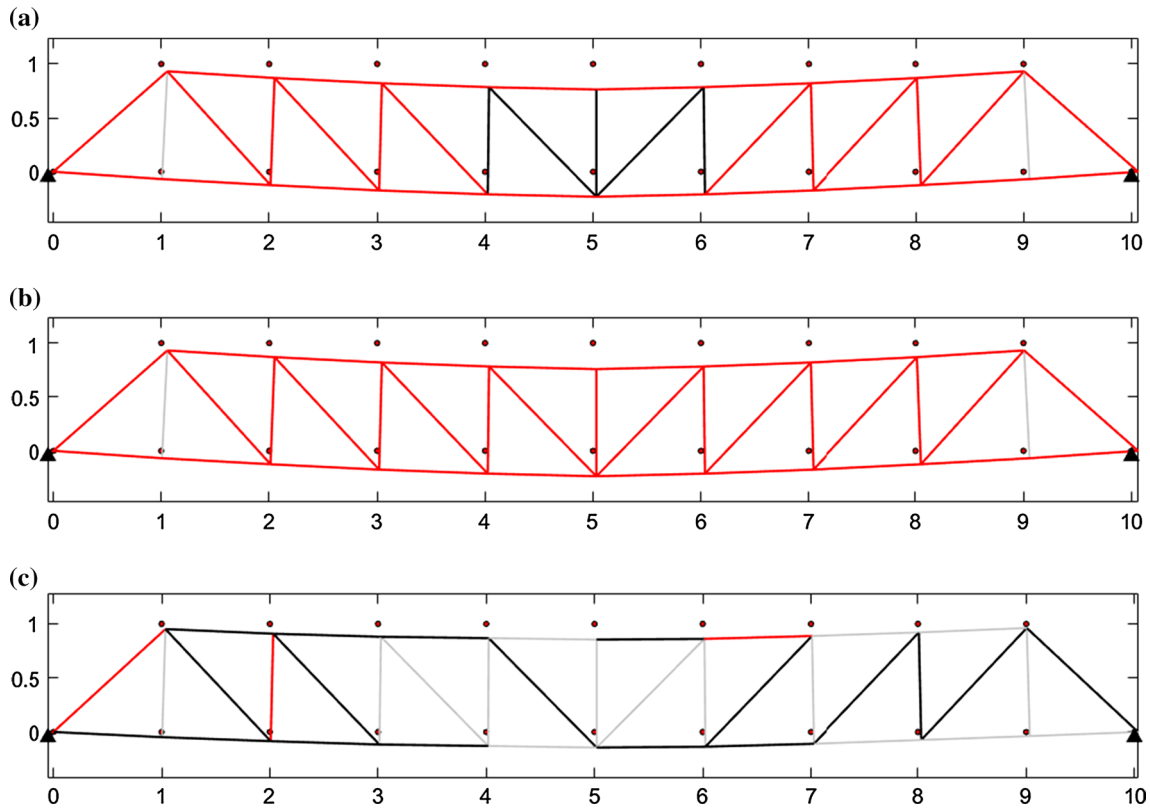


Fig. 8 Results for the **a** IM, **b** DM, **c** GA for the sizing optimization of a 37-elements truss. The elements in red have $|\sigma| \geq 0.9\sigma_{adm}$, in black, $0.5\sigma_{adm} < |\sigma| < 0.9\sigma_{adm}$ and in gray, $|\sigma| \leq 0.5\sigma_{adm}$. (Color figure online)

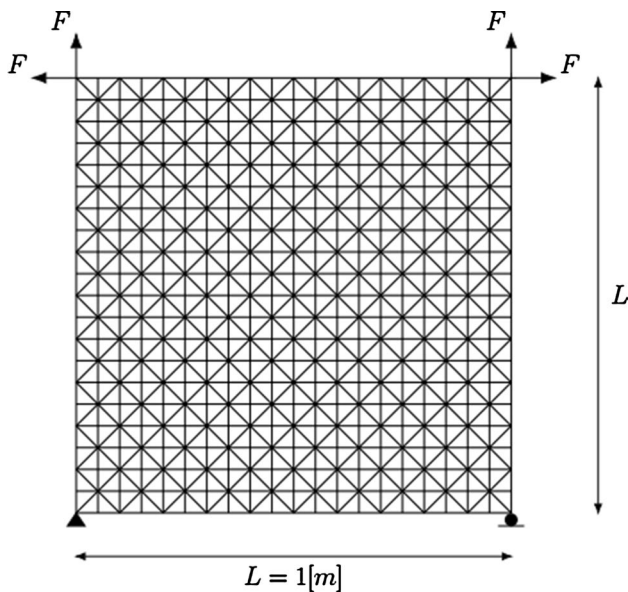


Fig. 9 1240-elements truss subjected to the optimization process. For the proposed case: $F = 1000\text{ N}$ and $L = 1\text{ m}$

Table 4 Results of the 1240-elements structure optimization

Approach	Results
IM	$N_f = 12, T = 92.14\text{ s}, m_{\min} = 1.032\text{ kg}$
DM	$N_f = 99, T = 10709.96\text{ s}, m_{\min} = 5.373\text{ kg}$
GA	$N_f = 9900, T = 1173.51\text{ s}, m_{\min} = 66.608\text{ kg}$

As previously mentioned, this type of linear system has two particular advantages over other classic methods of sizing optimization: the monotonic and fast decrease trend of the objective function and the non-dependence on the initial value. In order to investigate the former, Fig. 12a, showing the convergence trend of the three benchmarks performed with the IM, has been plotted. For the structures with 10 and 37-elements, 8 iterations were necessary to optimize the systems. For the 1240-elements truss, only 12 iterations have been performed. These results show that our method is able to scale the design problem with outstandingly small increases in the number of processing steps.

The second feature of this approach, the non-dependence on the initial conditions, is investigated. For all the results shown in Fig. 12b, with $x_{i,0} = 1.00, 0.50, 0.25$ and 0.01 m^2 , the 1240-elements structure converges to $m_{\min} = 1.036\text{ kg}$,

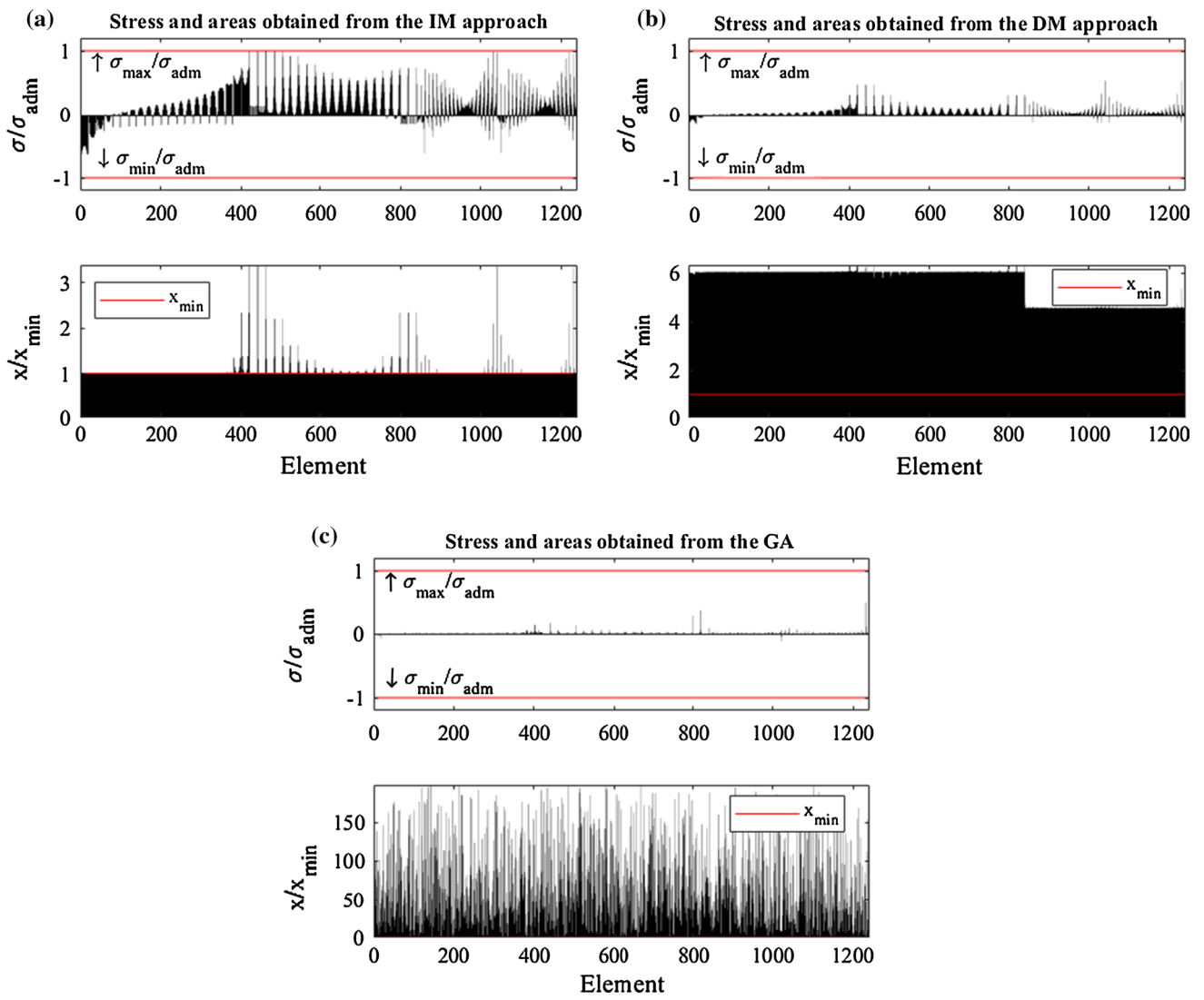


Fig. 10 Results for the **a** IM, **b** DM, **c** GA for the sizing optimization of a 1240-elements truss. The lines in red indicate the boundary conditions. (Color figure online)

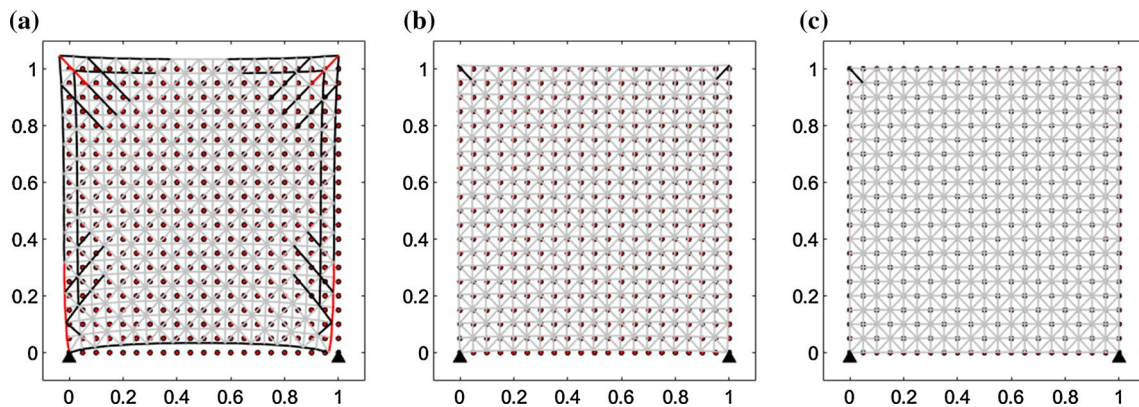


Fig. 11 Results for the **a** IM, **b** DM, **c** GA for the sizing optimization of a 1240-elements truss. The elements in red have $|\sigma| \geq 0.9\sigma_{adm}$, in black, $0.5\sigma_{adm} < |\sigma| < 0.9\sigma_{adm}$ and in gray, $|\sigma| \leq 0.5\sigma_{adm}$. (Color figure online)

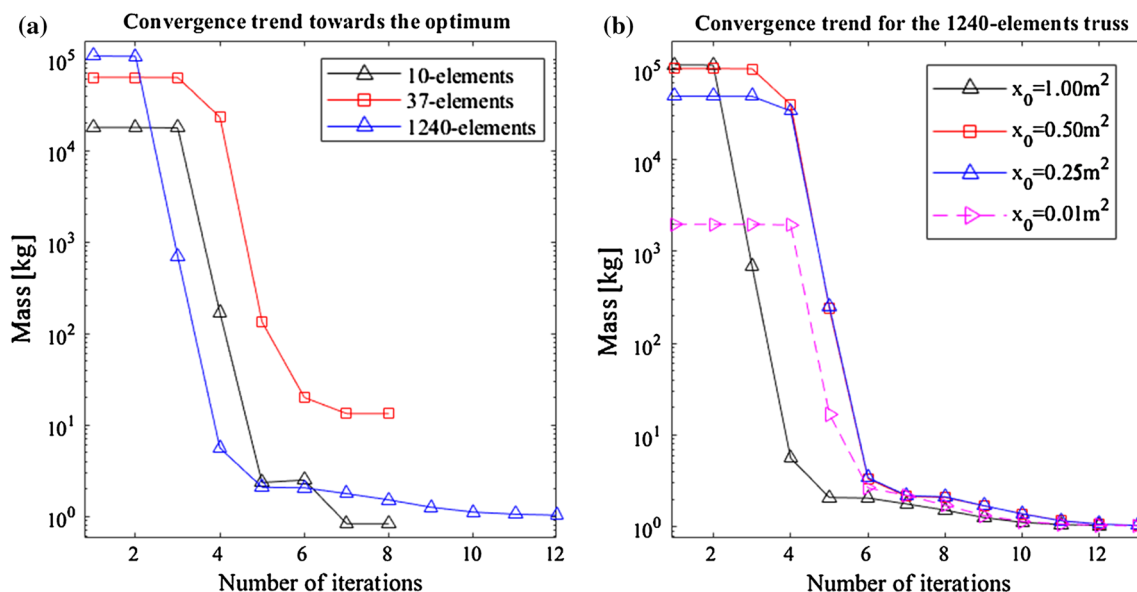


Fig. 12 (Left) convergence trend for the results presented for the three benchmarks in previous sections. (Right) analysis of the dependence of the final result on the initial condition

after 12 or 13 iterations. This behaviour indicates that, for an arbitrary initial solution, the space of solutions tends towards a single point of convergence. Further, combined with the monotonic decrease trend for the solution, which is also evident in Fig. 12b, the polynomial time and low-computational cost, it is believed that the IM is a fair alternative to structural optimization of linear models, being a valid substitute for optimum weight designs of isostatic trussed structures.

5 Conclusions

In this study, we propose an original linear formulation to perform sizing optimization of isostatic trusses that provides great computational performance. As noticed in the previous analyses, the method is able to optimize structures by up to 1% the time required by a GA. In addition, the increase in computational time by the number of variables is up to 1 and 2 orders smaller than those presented by the DM and the GA, due to the polynomial time of solution, making this approach particularly interesting for complex isostatic topologies. The greatest features of the method are:

- i. the non-dependence of the solution on the initial condition.
- ii. characteristic monotonically rapid drop tendency.
- iii. analysis step is embedded within the optimization stage, avoiding tedious separate analyses.
- iv. the number of optimization iterations in the proposed procedure does not increase appreciably by the problem size.

Motivating the use of the present approach in linear and statically determinate structures, subjected to small displacements.

References

- Al Rabadi, H. F. H. (2014). Truss size and topology optimization using harmony search method. The University of Iowa.
- Asl, R. N., Aslani, M., & Panahi, M. S. (2013). Sizing optimization of truss structures using a hybridized genetic algorithm. arXiv preprint [arXiv:1306.1454](https://arxiv.org/abs/1306.1454)
- Assimi, H., Jamali, A., & Nariman-zadeh, N. (2017). Sizing and topology optimization of truss structures using genetic programming. *Swarm and Evolutionary Computation*, 37, 90.
- Banh, T. T., & Lee, D. (2018). Multi-material topology optimization design for continuum structures with crack patterns. *Composite Structures*, 186, 193.
- Bendsøe, M. P., Ben-Tal, A., & Zowe, J. (1994). Optimization methods for truss geometry and topology design. *Structural Optimization*, 7(3), 141.
- Bölte, A., & Thonemann, U. W. (1996). Optimizing simulated annealing schedules with genetic programming. *European Journal of Operational Research*, 92(2), 402.
- Camp, C., & Farshchin, M. (2014). Design of space trusses using modified teaching–learning based optimization. *Engineering Structures*, 62, 87.
- Cazacu, R., & Grama, L. (2014). Steel truss optimization using genetic algorithms and FEA. *Procedia Technology*, 12, 339.
- Chambers, L. (1995). *The practical handbook of genetic algorithms: New frontiers*. v. 2. CRC Press. <https://books.google.com.br/books?id=9RCE3pgj9K4C>
- de Almeida, F. S. (2016). Stacking sequence optimization for maximum buckling load of composite plates using harmony search algorithm. *Composite Structures*, 143, 287.
- Doan, Q. H., & Lee, D. (2018). Optimal formation assessment of multi-layered ground retrofit with arch-grid units considering buckling

- load factor. *International Journal of Steel Structures*. <https://doi.org/10.1007/s13296-018-0115-x>
- Dominguez, A., Stiharu, I., & Sedaghati, R. (2006). Practical design optimization of truss structures using the genetic algorithms. *Research in Engineering Design*, 17(2), 73.
- Farshchin, M., Camp, C., & Maniat, M. (2016). Multi-class teaching–learning-based optimization for truss design with frequency constraints. *Engineering Structures*, 106, 355.
- Farshi, B., & Alinia-Ziazi, A. (2010). Sizing optimization of truss structures by method of centers and force formulation. *International Journal of Solids and Structures*, 47(18–19), 2508.
- Ferrier, G. D., & Lovell, C. K. (1990). Measuring cost efficiency in banking: Econometric and linear programming evidence. *Journal of Econometrics*, 46(1–2), 229.
- Frans, R., & Arfiadi, Y. (2014). Sizing, shape, and topology optimizations of roof trusses using hybrid genetic algorithms. *Procedia Engineering*, 95, 185.
- Gan, B. S., Hara, T., Han, A., Alisjahbana, S. W., & Asad, S. (2017). Evolutionary ACO algorithms for truss optimization problems. *Procedia Engineering*, 171, 1100.
- Gomes, H. M. (2011). Truss optimization with dynamic constraints using a particle swarm algorithm. *Expert Systems with Applications*, 38(1), 957.
- Holland, J. H. (1973). Genetic algorithms and the optimal allocation of trials. *SIAM Journal on Computing*, 2(2), 88.
- Hultman, M. (2010). Weight optimization of steel trusses by a genetic algorithm—size, shape and topology optimization according to Eurocode. TVBK-5176
- Kaveh, A., & Khayatazad, M. (2013). Ray optimization for size and shape optimization of truss structures. *Computers & Structures*, 117, 82.
- Kaveh, A., & Rahami, H. (2006). Analysis, design and optimization of structures using force method and genetic algorithm. *International Journal for Numerical Methods in Engineering*, 65(10), 1570.
- Koski, J. (1981). *Multicriterion optimization in structural design*. Technical report. Tampere University of Technology, Finland.
- Kripka, M. (2004). Discrete optimization of trusses by simulated annealing. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 26(2), 170.
- Lamberti, L., & Pappalettere, C. (2003). Move limits definition in structural optimization with sequential linear programming. Part II: Numerical examples. *Computers & Structures*, 81(4), 215.
- Lustig, I. J., Marsten, R. E., & Shanno, D. F. (1994). Interior point methods for linear programming: Computational state of the art. *ORSA Journal on Computing*, 6(1), 1.
- Lyamin, A. V., & Sloan, S. (2002). Upper bound limit analysis using linear finite elements and non-linear programming. *International Journal for Numerical and Analytical Methods in Geomechanics*, 26(2), 181.
- Miguel, L. F. F., & Miguel, L. F. F. (2012). Shape and size optimization of truss structures considering dynamic constraints through modern metaheuristic algorithms. *Expert Systems with Applications*, 39(10), 9458.
- Mohr, D. P., Stein, I., Matzies, T., & Knappek, C. A. (2011). Robust topology optimization of Truss with regard to volume. arXiv preprint [arXiv:1109.3782](https://arxiv.org/abs/1109.3782)
- Mortazavi, A., & Toğan, V. (2017). Sizing and layout design of truss structures under dynamic and static constraints with an integrated particle swarm optimization algorithm. *Applied Soft Computing*, 51, 239.
- Potra, F. A., & Wright, S. J. (2000). Interior-point methods. *Journal of Computational and Applied Mathematics*, 124(1–2), 281.
- Rajan, S. (1995). Sizing, shape, and topology design optimization of trusses using genetic algorithm. *Journal of Structural Engineering*, 121(10), 1480.
- Ringertz, U. T. (1985). On topology optimization of trusses. *Engineering optimization*, 9(3), 209.
- Sivanandam, S., & Deepa, S. (2008). *Introduction to genetic algorithms* (pp. 165–209). Berlin: Springer.
- Stolpe, M. (2016). Truss optimization with discrete design variables: A critical review. *Structural and Multidisciplinary Optimization*, 53(2), 349.
- Tejani, G. G., Savsani, V. J., Patel, V. K., & Savsani, P. V. (2017). Size, shape, and topology optimization of planar and space trusses using mutation-based improved metaheuristics. *Journal of Computational Design and Engineering*, 5, 198–214.
- Wang, D., Zhang, W., & Jiang, J. (2002). Truss shape optimization with multiple displacement constraints. *Computer Methods in Applied Mechanics and Engineering*, 191(33), 3597.
- Wang, D., Zhang, W., & Jiang, J. (2004). Truss optimization on shape and sizing with frequency constraints. *AIAA Journal*, 42(3), 622.