



Truss layout design and optimization using a generative synthesis approach

Amir Hooshmand ^a, Matthew I. Campbell ^{b,*}

^a Institute for Advanced Study and Product Development Department, Technische Universität München, Lichtenbergstrasse 2a, D-85748 Garching, Germany

^b School of Mechanical, Industrial, and Manufacturing Engineering, Oregon State University, 204 Rogers Hall, Corvallis, OR 97331-6001, United States

ARTICLE INFO

Article history:

Received 23 May 2014

Accepted 23 September 2015

Available online 2 November 2015

Keywords:

Topology optimization

Optimum layout

Design automation

Graph grammar

Truss structures

ABSTRACT

This paper describes a generative methodology for optimum design of truss structures. The novelty of the proposed method is its combination of generative design synthesis methods with conventional gradient-based optimization and simulation models. This combination is able to achieve optimal topologies and shapes for cable trusses under various constraints: stress, displacement, stability. Furthermore, manufacturing issues properly limit the generated solutions. A graph grammar is used to define different topologies for a given load problem. The effectiveness of the method is checked by solving a variety of available test problems found in the literature as well as several new three-dimensional solutions.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

One of the most popular computational design synthesis approaches in engineering design involves topology optimization methods, which is based on using finite element methods (FEM) for the analysis, and various optimization algorithms such as gradient-based optimization techniques [1]. Topology optimization is a mathematical approach that models a given fixed number of decision variables (cells or grids), and optimizes its objective function (material layout) for a given set of boundary conditions and loads. Layout design optimization refers to topology, shape and sizing optimization of structures. These optimization methods are now being used successfully in areas such as electro-magnetics, microelectromechanical systems (MEMSs) and fluids as well [1,2]. For more than two decades, engineering designers have used topology optimization methods for a wide range of structural design problems. The objective of structural optimization is to improve the performance of the structure components in terms of material efficiency in transferring applied loads. Therefore, the performance criterion is usually the weight or cost of the structure subject to geometrical constraints and various performance-based constraints such as stress, displacement, mean compliance, frequency and buckling load.

Truss and space frames are widely used structures, because they are simple and inexpensive to build and can be used in many engineering applications. Literature shows much research based on classical topology optimization methods for the optimal design of truss structures [3–5]. Typical truss topology optimization approaches discretize the design space with a nodal mesh of a large ground structure, in which every node is connected to almost every other node in the domain. The ground structure concept has been first initiated by Dorn et al. [3]. This dense set of potential structural members along with applied loads and boundary conditions are assumed known. The optimization is used to determine the material distribution of cross-sectional areas of the connections. By removing inefficient members with slender areas below a certain threshold the connectivity of the system is changed and the structure is updated [1,6–8]. Truss topology optimization is a combination of three optimization problems; sizing, topology and shape. The objective of sizing optimization is to find the optimal cross-sectional area of structural elements. Topology optimization aims to find the optimum existence and connectivity of the nodes. Shape optimization is concerned with finding the optimum nodal coordinates. For each optimization it is assumed that the variables of the other two optimizations are fixed.

Issues entailed by the formulation of underlying governing mechanics (e.g. local and global instability) are often an important obstacle in topology optimization methods [9]. However, the matter becomes more acute when considering uncertainties associated with the structural stiffness such as geometry and material property imperfections. Consequently most of the researches in this

* Corresponding author. Tel.: +1 (541) 737 6549.

E-mail addresses: hooshmand@pe.mw.tum.de (A. Hooshmand), matt.campbell@oregonstate.edu (M.I. Campbell).



area are focused mainly on deterministic problems with a limited consideration of uncertainty [5,7,10–12]. The main strategy to consider these uncertainties in the formulations has been adding randomness (uncertainty) to the spatial position of the nodes [13–16] or equivalent random forces at nodes [9,17]. Jalalpour et al. [9] aim to be the first who propose a method that is capable of handling both nodal location uncertainties and first order global buckling effects. In their method, random forces at nodal points represent the potential global buckling in imperfect structures. Although considering uncertainty in the spatial location of the nodes creates (theoretically) more stable results, they are less practical structural solutions to be built. Indeed, since 1960, various optimization methods for the layout design of structures have been developed and many papers and books on the mathematical aspects of the structural optimization have been published. These contributions are mainly concerned with theoretical aspects rather than practical applications and engineering aspects [18]. This shows a clear gap between the development of structural layout optimization theory and its practical applications in industry [18,19]. The main reasons behind this gap are the mathematical complexity of structural optimization methods [18], and the fact that structural optimization techniques are developed primarily to save material and not for automating the engineering design process [20]. The work presented in this paper introduces an efficient design tool for mechanical and civil engineers. It is a clear and easy to understand concept and the methodology is an attempt to reduce the gap between theoretical methods and practical design optimization. In many practical applications minimum weight of the structure, maximum nodal displacements and maximum stress in the structure are not the only important criteria. Available materials that can be used to build a structure or financial limitations may play an important role in the design that cannot be neglected. Using the developed approach it is not only possible to consider aforementioned criteria in designing an optimum structure, the designer is able to generate an optimum structure with the available set of components that can be used to build the truss structure.

This paper presents the theory and application of a generative design synthesis method for topology, shape and sizing design of structures. The method incorporates the load flow principal, simulation and optimization methods through generative design synthesis approach into a modern structural layout optimization theory. Furthermore, unlike other conventional methods, the types of bars and cables that are allowed to be used as components of the structure are defined at the beginning of the synthesis. There is no need to post-process the results, because manufacturing issues and limitations can be considered in the synthesis. This method uses a graph grammar interpreter to generate different topological solutions for a structural problem. Through exhaustive search of the design space all valid topologies for a given problem are generated and sorted based on the complexity of the solutions. An optimization algorithm is then used to optimize all (or the top n) topologies. This optimization algorithm changes the spatial positions of the joints, to minimize a desired objective such as stress, displacement, or overall load flow. Finally, the best design candidates are transformed into meaningful 3D shapes. The nodes and arcs of the generated graph represent Constructive Solid Geometry (CSG) shapes. The graph grammars rules work with graph elements to generate a new topological state, as a result the search and generation process is very fast. However, it is vitally important to embed enough information in the graph grammar rules in order to create meaningful structures. To increase the computational effectiveness of the generation process, the design process is carried out in distinct steps. To enter each step, the candidate solution must meet specific requirements.

One of the major limitations, which topology optimization methods in conceptual design are facing, is limited representation

power; the synthesis process and design rules are dependent and integrated into the simulation model, the simulation model is often fixed for a given set of loads and boundary conditions. By utilizing a multiple representation approach for the topology optimization of structures, our algorithm avoids many problems associated with other approaches in setting up the mechanical behavior equations. There is no need for a parameterization scheme because representing the topology is independent of the simulation model. It causes significant computational savings, because the FE analyses and remeshing at each iteration is no longer required. By using multiple representations in our method, dimension (e.g. 2D or 3D) has almost no effect on the computation efforts in finding structure topologies. Furthermore, as the representation and simulation models are fully separated from each other, one can use the same rules for problems with completely different boundary conditions, loads and structural component types.

The proposed methodology not only reproduces the most common designs reported in literature but also increases the freedom of the designer by creating new efficient structural configurations. The effectiveness of the proposed method is checked by solving a variety of available test problems and comparing them with those found in the literature. This paper is organized as follows. Section 2 describes a background about generative design synthesis systems and load flow principal. Section 3 provides details of the proposed approach in this paper. Section 4 presents results and discusses the implications of results; the focus of this section is to present significant benefits of proposed methodology over previously used approaches and check the validity of the proposed approach through several test problems and comparing them with the literature. Section 5 shows the applicability of the proposed approach for large scale problems with multiple numbers of loads. Finally, Section 6 concludes the study and outlines directions of future research.

2. Background

2.1. Graph grammars

A graph grammar is a method to represent elements and their relationships in the design space [21]. Grammars capture large design spaces in a single formalism, and hence can increase the design freedom [22]. Based on a set of pre-defined rules, grammars generate alternative design solutions [23]. A graph grammar may be used as a precise method to model and facilitate design problems due to its formality, extensibility and generality in the modeling and manipulation of structural and non-structural information [24]. A grammar-based approach for truss topology generation and optimization has been first proposed in [25]. Shea further developed the approach for the synthesis of truss structures using finite element simulation and stochastic search methods [26–29] and eifForm is the software tool which has been developed based on this approach [30]. However, the graphs have been used also much earlier in the structural engineering (among others [31,32]). Ref. [33] presents the theorems of graph theory for the structural engineering problems along with many extensive applications.

For graphs, a graph grammar interpreter is required to apply a set of transformative operations on a seed graph. For this study, GraphSynth is used to accomplish graph transformations. GraphSynth is a unique research software for creating, editing, displaying, and manipulating generative grammars. This framework stores graphs, rules and rulesets under XML file format. It allows for the automatic search of creative, optimal or targeted solutions. GraphSynth is an open source framework built on Microsoft Visual Studio .NET. Additionally, it is able to perform various graph trans-

formations such as the double-pushout method and free-arc embedding; these two together cover nearly all types of required graph transformations [34]. One of the most important characteristics of the GraphSynth is its extensibility; through additional compiled on-the-fly functions nearly any capability can be added to the rules and rulesets. It has been used to solve diverse problems such as fluid channels, tensegrities [34–37].

2.2. Load flow path principal

Load flow path is a way in which load ‘flows through a structure’ or mechanism from an input point (point of application) to the output point (support or fixed point). The term load path has been first defined by Kelly and Elsley [38], although various authors [39–41] have used this term – however – in a descriptive sense. A fundamental fact in design process of structures is ensuring an appropriate path for loads and forces to flow in the structure from application input point(s) to the fixed or reaction output point(s). Although this has been always implicitly considered, there is a direct relation between load flow and the deformation behavior of the structure. Hence an insight obtained from load flow path can greatly enhance the design process [38,42]. Load paths are relatively easy to define in simple structures such as trusses which carry only axial loads. The concept of load path has been used in topology optimization methods to reach feasible solutions [43–45]. The main limitation of these techniques is the mathematical formulation of load paths which is based on finite element analysis of previous design step [46]. In this paper the knowledge obtained

from load flow path principal bases a new design methodology for structural topology optimization. This methodology has three main rules;

- forces tend to flow in paths with least resistance; here, resistance is the amount of change in the direction of the load flow path,
- changes in the direction of the load flow requires an extra transmitter member to impose the change,
- structural components become more utilized as their direction tends to the direction of the applied load at a particular joint. For example, if at a joint the applied load is exactly at the same direction as a specific member, the utilization of that member is maximum, because no other force components are created.

Our main aim in this paper is to systematically calculate the load paths based on the rules and generate various topologies that can meet the requirements for the desired force path. This will help us to add only those components which increase the overall performance of the structure. Therefore, stresses and strain energy will be uniformly distributed throughout the topology.

3. Proposed approach to truss topology optimization

The overall approach to the structure synthesis using generative graph grammars is depicted in Fig. 1. The whole process can be divided into two phases: shape and topology synthesis, and

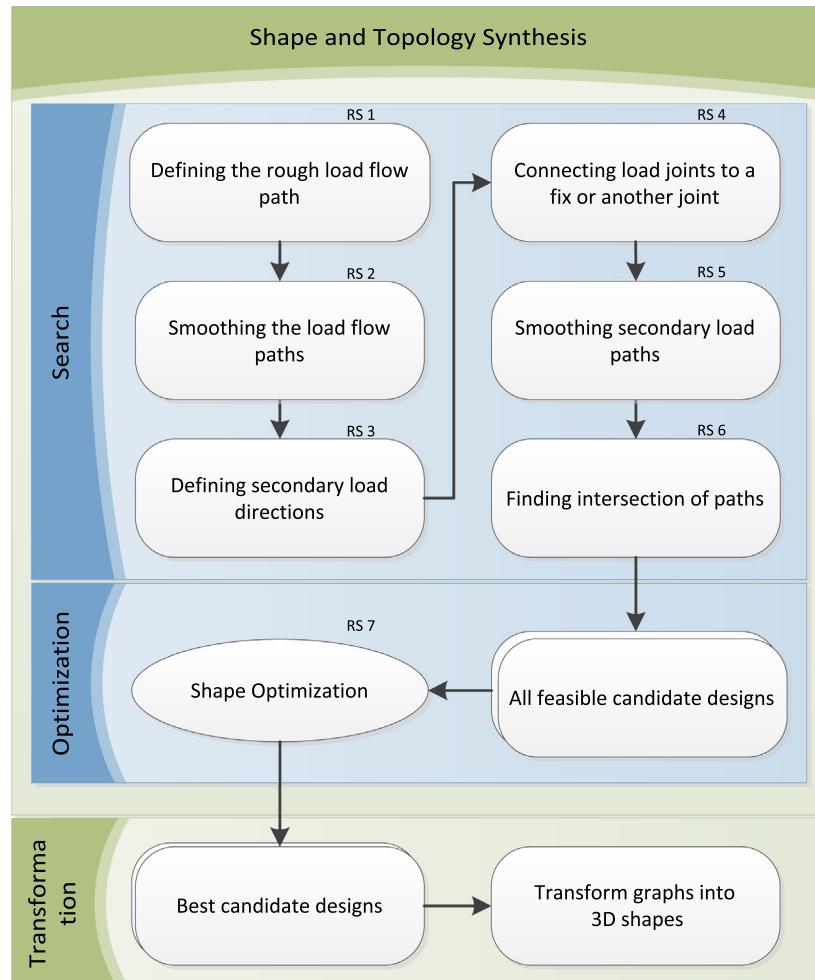


Fig. 1. Flow chart of the proposed methodology for truss structures synthesis.

transformation. The synthesis phase consists of two steps: search and optimization. In the search phase, all valid topologies are generated in six consecutive steps and in the optimization phase the parameters of the topology are optimized. Based on the objective function values, an optimum candidate topology is selected. The synthesis phase uses the graph grammar interpreter to apply graph transformations and generate new topologies. The graph transformation approach, which is based on the load flow principle, is used only for generating structures in the search phase. Rigidity, stress, and stability (in terms of preventing buckling) are addressed by an analysis method in the optimization phase, in which all joints are viewed as contributing zero moment. In the transformation phase the generated topologies, which are represented as graphs are converted to three dimensional (3D) shapes. This step is not necessary for the synthesis of structures and is used just to visualize the final results as 3D shapes. In the following sub-sections all phases of the design are described in detail.

3.1. Analysis of the structures

This section briefly describes how the necessary structural evaluations are carried out. The approach is initially developed for cable truss synthesis, but it can be used without any modification for synthesis of general truss structures. The only difference between cable truss and truss structures is in the individual components that constitute the structures; trusses constitute of tension-compression bars, whereas cable truss structures consist of tension wire ropes and compression bars.

A cable truss structure is a mechanical system composed of tension wire ropes and compression bars, which are connected together through frictionless hinges (nodes), and is loaded only at nodes. Consequently the axial displacement in any individual component (wire rope or bar) is linear and consequently the internal forces, strains and stresses are constant for each component in the structure. In this work the simulation does not consider the weight of the components. But since the simulation and topology generation are separate from each other, the simulation algorithm can be changed.

[Fig. 2](#) shows the shape of the bars and the construction type of the wire ropes that have been used in this study.

Size, shape, material characteristics and construction type of the elements (bars and ropes) can be easily changed at the onset of this approach. In the results section of this paper, the effect of changing elements features is investigated. It is also possible to define a set of options for the structure elements, and leave the decision for choosing adequate element types and sizes for each segment to the optimization. In the optimization Section 3.2.4



Shape = Cylindrical cross section
Diameter = 40 mm
Young's Modulus = 210 N/mm²

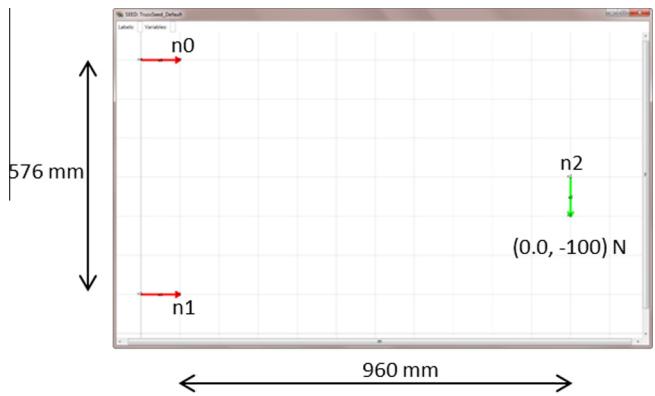


Fig. 3. A seed graph with one load point and two fixed points. (The reaction forces at fixed points throughout this work limit the movement of bars in all directions; x and y for 2D design problems and x , y and z for 3D design problems. However, they have been represented with a single arrow in red.). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

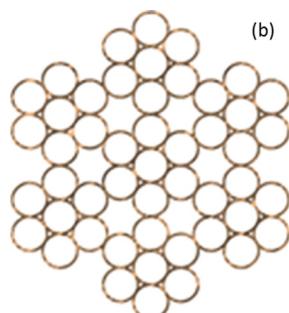
and also the first two sub-sections of the results the steel rope is fixed to a $1 \times 7(1+6)$ construction type and the bar diameter to 24 mm because in these sections our aim is to investigate the capabilities of the shape optimization algorithm. Therefore, we have intentionally created conditions to challenge the shape optimization capability rather than the topology generation. A steel rope with the construction type $1 \times 7(1+6)$ is a wire rope with one strand of wire: the strand consists of seven steel wires. The rope schematized in [Fig. 2](#) has seven strands with the same type.

The procedure for solving the force and displacements of trusses with tension-compression bars can be found in most classical Structural Mechanics books [47]. The analysis approach implemented here is identical. In this process, a stiffness matrix for the structure is defined for all compressed bars and tensile ropes in the truss. This then leads to the determination of displacement and force values for all joints in the true. This scheme has been programmed in C# and can be used for both two and three dimensional problems.

3.2. Shape and topology synthesis of truss layouts

3.2.1. Seed graph

A seed graph defines the scope and boundary conditions of the problem to be solved. In this case, it consists of some arcs and nodes which are labeled as fixed (n_0 and n_1) or loaded (n_2) with different spatial positions. [Fig. 3](#) illustrates a sample seed graph with two fix points and one load in two dimensions (x,y). The ver-



Ultimate Tensile Strength = 1770 N/mm²
Ultimate load = 2540 N / mm²
Diameter = 2 mm
Rope Construction Type = $6 \times 7(1+6) + 1 \times 7(1+6)$

Fig. 2. Cylindrical bar (a), and wire rope (b) used for this study.

tical distance between the fix points is 576 mm and the horizontal distance between fix points and the load is 960 mm. The load is 100 N in -y direction. The goal of grammar rules is to transform this seed graph into a graph that represents a meaningful structure for supporting the load.

3.2.2. Topology generation

The graph grammar interpreter, which is used for structural synthesis, starts with a seed graph, which represents the bounds of a specific problem. The generation (graph transformation) is carried out through 27 rules which are distributed into seven rulesets (RS). A ruleset is a set of rules that transforms the design from one level of maturity to the next level. Rulesets are used as a means to compartmentalize different phases of the generation process. There are two types of rulesets used in this approach: generative rulesets and transformative rulesets. Generative rulesets define the design space of all possible valid candidate designs (RS1, RS2 and RS4). These rulesets define the topology of the candidate designs. Transformative rulesets change the state of generated solutions. They transform a candidate design without changing its topology (RS3, RS5, RS6 and RS7). The branching factor of the transformative rulesets is one whereas the branching factor of the generative rulesets is greater than one.

In Fig. 4 all 27 grammar rules with a short description of each are illustrated. The rules are created in a general way, so that for different types of problems the same rules can be used. The left picture in the Rule column is the left hand side of a rule (LHS) and the right picture is the RHS of the rule. The graph grammar interpreter converts that part of the seed graph which is matched to the LHS into the graph segment depicted in the RHS. Three trigger rules (4, 9 and 14), two optimization rules (3 and 27), and one evaluation rule (26) do not change the graph elements, therefore their LHS and RHS are not depicted. They are referred to as rules to coordinate their execution among the other rules. Fig. 5 shows the tree structure of the synthesis process. Figs. 1, 4 and 5 represent the approach from three different viewpoints with different

levels of detail. As can be seen in Fig. 5, rulesets 1, 2 and 4 increase the number of candidate topologies and other rulesets just transform the existing candidate designs.

The whole approach is developed in such a way that incremental information, which is required in the next step, is added to the design. Aside from the depicted rule conditions in Fig. 4 (like connecting loads to supports), many other additional functions are compiled into the rules to define detailed matching conditions as well as rule actions. For instance, for rule 24, one function aids in the recognition process to find the exact position of the intersection and one function helps in inserting a node at that calculated spatial position. In the following three sub-sections, all seven rulesets including 27 rules are explained in detail.

3.2.2.1. Ruleset 1. The task of the first ruleset (RS1) is to create the main path between load point(s) and the fixed (support) points. In the case of the seed graph in Fig. 3 with only one load and two fix points, just one candidate design is generated. If the suggested design in Fig. 6 with one compression bar and one tension rope meets the objective requirements and does not violate different constraints such as buckling, then the ruleset is done. However in this specific example, due to using a very slender bar (24 mm diameter), the compression bar buckles. Because the types of components and their materials are fixed, one should change the path of the load in a way that causes a reduction in the flow amount. This is carried out through an optimization algorithm, which tries to find the optimum direction for the load carrier vectors (Fig. 6).

It is clear that the optimum load carrier vectors should be in the same direction as the load vectors, because the net load to be carried will be the same as the load itself. Based on this fact, the load carrier vectors in Fig. 7 should be in the same direction as the load, but the optimization result in Fig. 7 shows a different direction. This is due to the fact that changes in the direction of the load flow require an extra transmitter member to impose the change, and the more changes in the direction, the more lateral load flow. Therefore, in the dilemma of minimizing the main load flow and

Ruleset	Type	Rule	Description	Ruleset	Type	Rule	Description
1	Generator	1	Connect loads to fixes	5	Automatic	15	Smooth the path between a load and a fix point
		2	Connect loads to load			16	Smooth the path between a load and a joint point
		3	Optimization I			17	Smooth the path between a joint and a fix point
		4	Trigger rule 1			18	Smooth the path between a joint and a joint point
2	Generator	5	Insert intermediate joint between a load and a fix point	6	Automatic	19	Remove arbitrary arcs
		6	Insert intermediate joint between a load and a joint point			20	Merge Nodes that are very near to each other
		7	Insert intermediate joint between a joint and a fix point			21	Merge nodes that are located on other arcs
		8	Insert intermediate joint between a joint and a joint point			22	Merge adjacent arcs
		9	Trigger rule 2			23	Merge triangle connections
3	Automatic	10	Remove arbitrary arcs	7	Automatic	24	Add intersecting joints
		11	Define direction of load at joints			25	Remove arbitrary nodes
4	Generator	12	Connect joint to fixes			26	Simulation
		13	connect joints to joints			27	Optimization II
		14	Trigger rule 3				Shape Optimization

Fig. 4. Grammar rules.

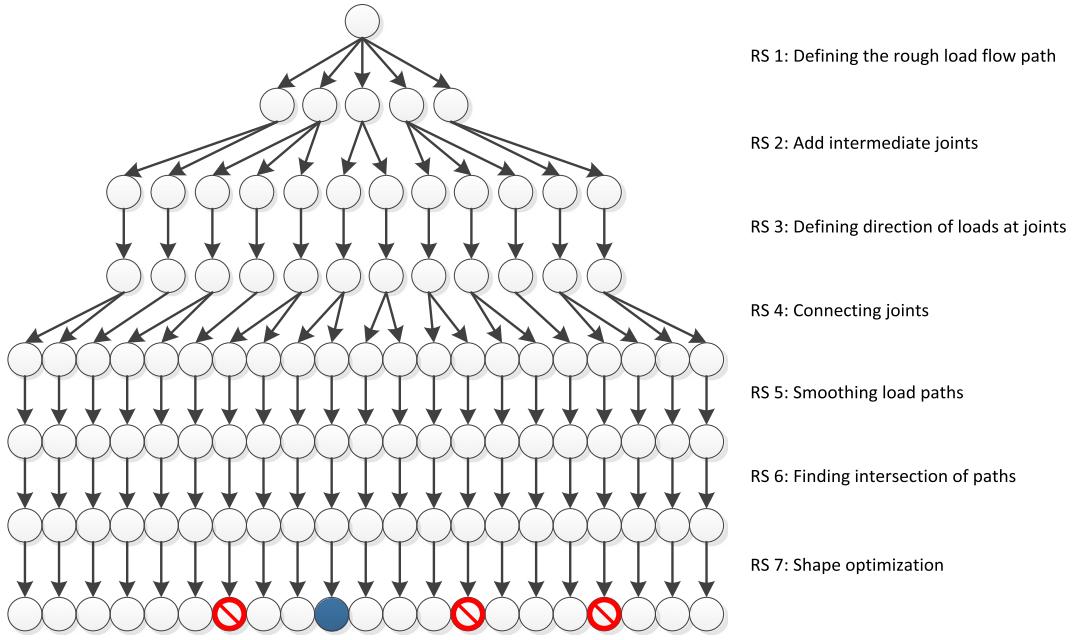


Fig. 5. Tree structure of the synthesis process (the filled circle corresponds to the best candidate design and the three red circles are infeasible candidate designs).

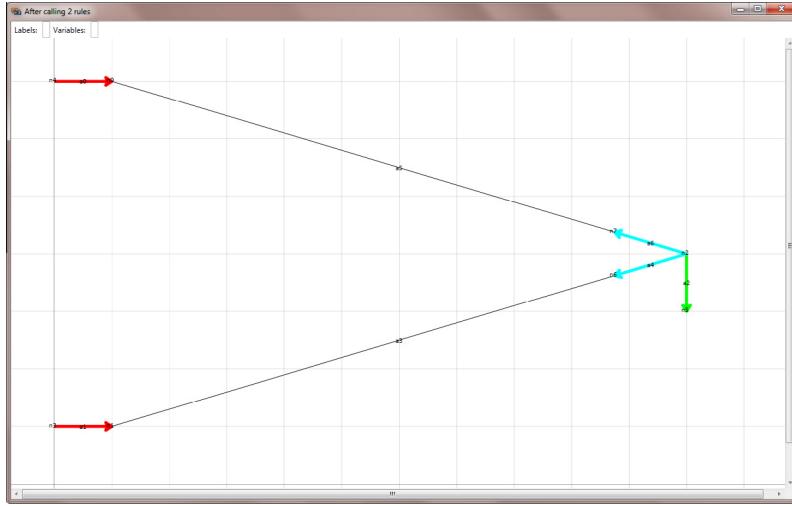


Fig. 6. Connecting the load points to the fix points.

the changes in the direction of load path, a mediating direction is found through the optimization. In the Section 3.2.4, the optimization function is described.

3.2.2.2. Ruleset 2. Based on the load flow direction from the RS1, four rules in this ruleset break the structure elements into smaller ones and one trigger rule is used to exit the rule set. The segmentation of the load paths depends upon the minimum allowed size for structure elements. In this study minimum size of each segment is 225 mm; therefore each path can be divided maximum into four segments. This minimum segmentation size is different from the minimum element size constraint of the shape optimization algorithm and there is no correspondence between segments and elements; segments represent load paths and elements are the components of the structure. However, the load path segmentation size limit indirectly limits the maximum number of the structure elements. It allows the rules of the ruleset 2 to be applied

only on arcs with the minimum length of 450 mm. As illustrated in Fig. 5, this is a generative ruleset for exploring the design space; the single output candidate topology of the ruleset 1 is populated into sixteen candidate topologies with different number of segments for each load path. Fig. 8 shows one of the candidate topologies with one segmentation at each path, which means two new joints are created.

3.2.2.3. Ruleset 3. The newly added joints require extra transmitter members to impose the changes in the direction of the load flow. This relatively small ruleset defines the optimum direction of the force to be imposed at the joint. This is calculated based on the direction of both path segments connected to the joint. The force vector makes the same angle with both path segments that have been connected to that joint. Fig. 9 shows the effect of this ruleset upon the candidate topology from the last ruleset. By applying four rules (each rule two times), the two directing arcs at the load point

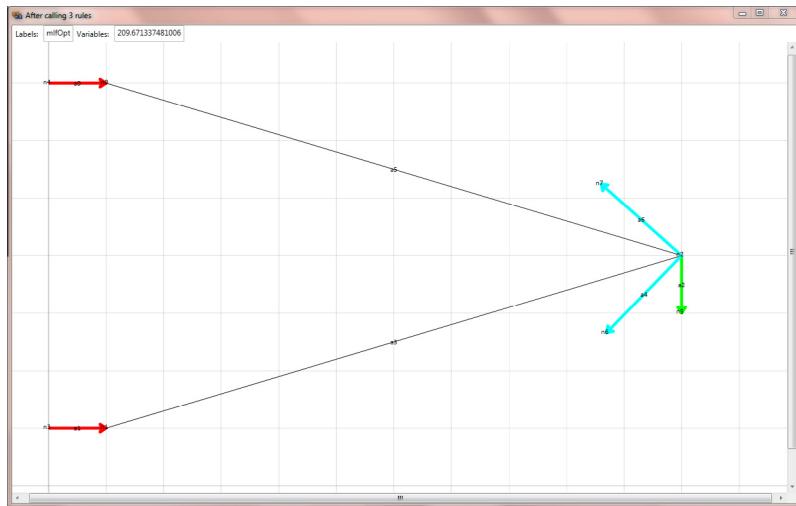


Fig. 7. Defining the direction of main load flows.

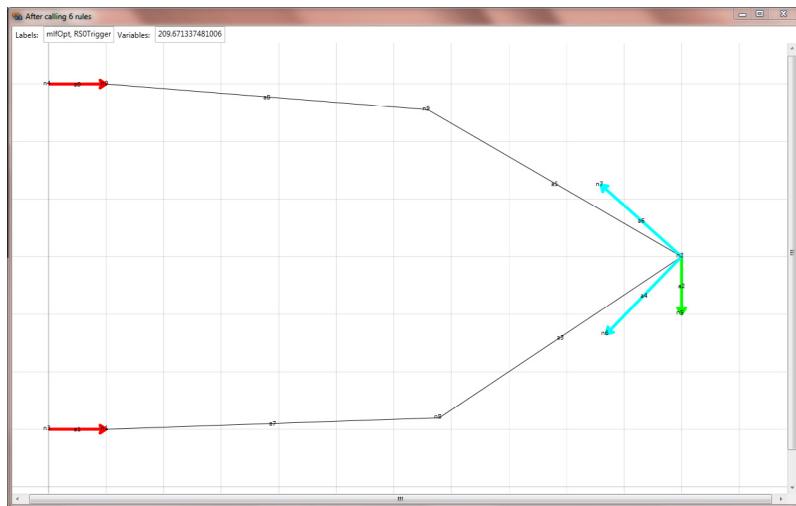


Fig. 8. Dividing load path segments into smaller pieces.

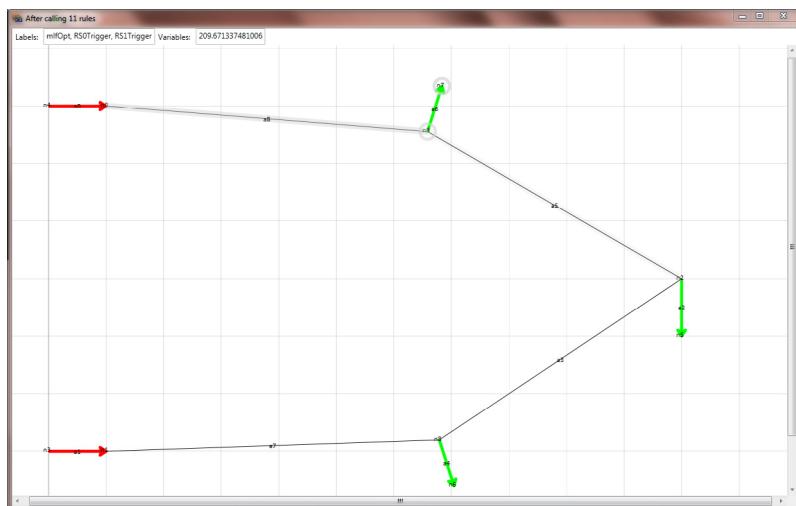


Fig. 9. Defining the direction of load at intermediate joints.

are removed, because they are no longer required and two directing arcs are added to the joints. If the number of segments and consequently the number of joints increases, the number of rule applications also will increase.

3.2.2.4. Ruleset 4. Ruleset 4 is the final generative ruleset. Its rules are designed to connect the joints to other joints or fixed points (Fig. 10). If there is more than one place that we can connect a joint to, the algorithm decides for a connection path that makes the smaller angle with the defined direction of force in the previous step. However the fixed points can be preferred too. This ruleset adds more variety to the design space and increases the number of the candidate designs from 16 to 35. At this stage the topology of the candidate designs is fixed and the remaining exploration is of parametric variation within these 35 topologies.

3.2.2.5. Ruleset 5. The functionality of this ruleset is similar to the ruleset 2, with the segmentation of the paths applied on the secondary load paths and not the primary ones (Fig. 11). This segmentation is necessary to find the intersection place of the load paths. In more complicated candidate topologies the effect of this ruleset is more evident.

3.2.2.6. Ruleset 6. Finally, ruleset 6, which is the final ruleset of the search process, prepares the generated solution candidate topologies for the shape optimization. It finds intersections in the load paths and adds new joints at those places. Rules of this ruleset also remove all unnecessary segmentations of the secondary load paths. Most of the rules in this ruleset such as merging adjacent arcs or uniting near nodes are not used for the test problem illustrated in Fig. 12. However, they are necessary in more complicated design solutions involving sharper changes in the direction of the load flow.

Fig. 13 shows the generation process of a design solution with a finer segmentation. The effect of previous rulesets is illustrated with more clarity in this figure.

3.2.3. Search process

A breadth first search (BFS) algorithm has been used to search the design space for all valid candidate topologies. The BFS algorithm does not leave a ruleset before either generating all possible candidate topologies or transforming all generated candidate topologies in other rulesets. As discussed before, rulesets 1, 2 and 4 explore the design space and rulesets 3, 5, 6 and 7 transform the candidate solutions (Fig. 5). One of the main reasons to use

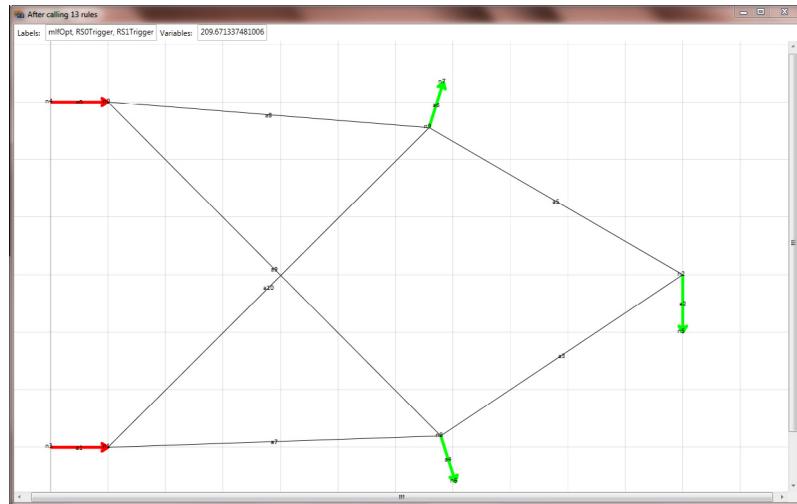


Fig. 10. Connecting intermediate joints to other joints or fix points.

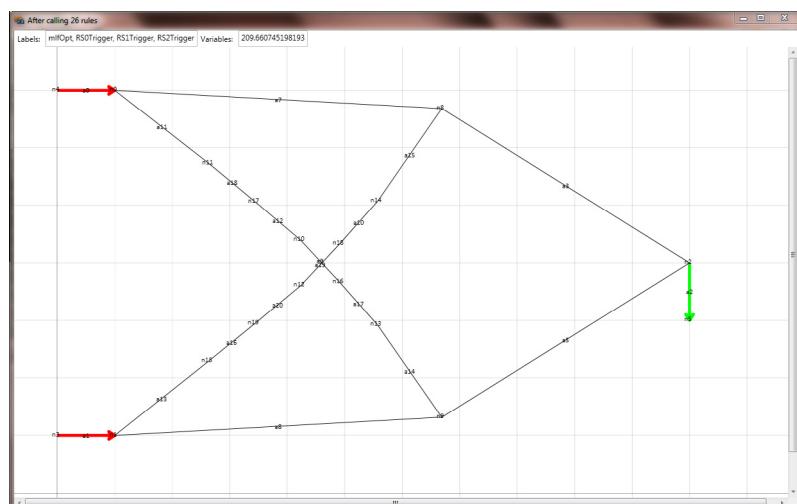


Fig. 11. Smoothing intermediate load paths.

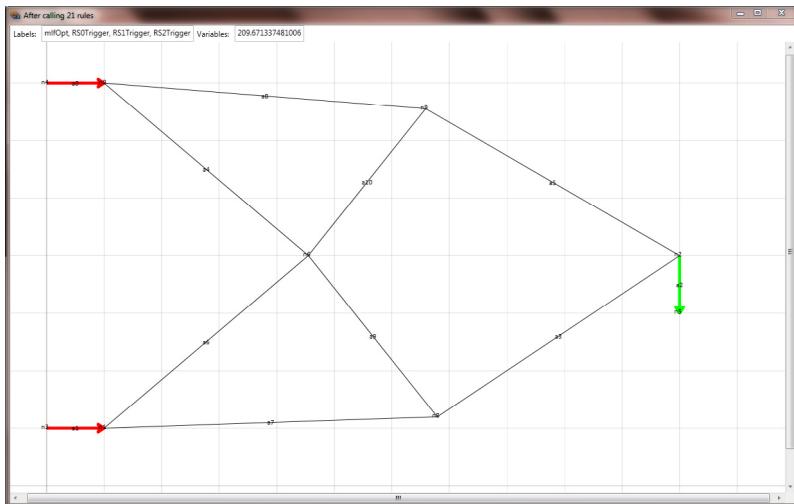


Fig. 12. Post-processing the solutions.

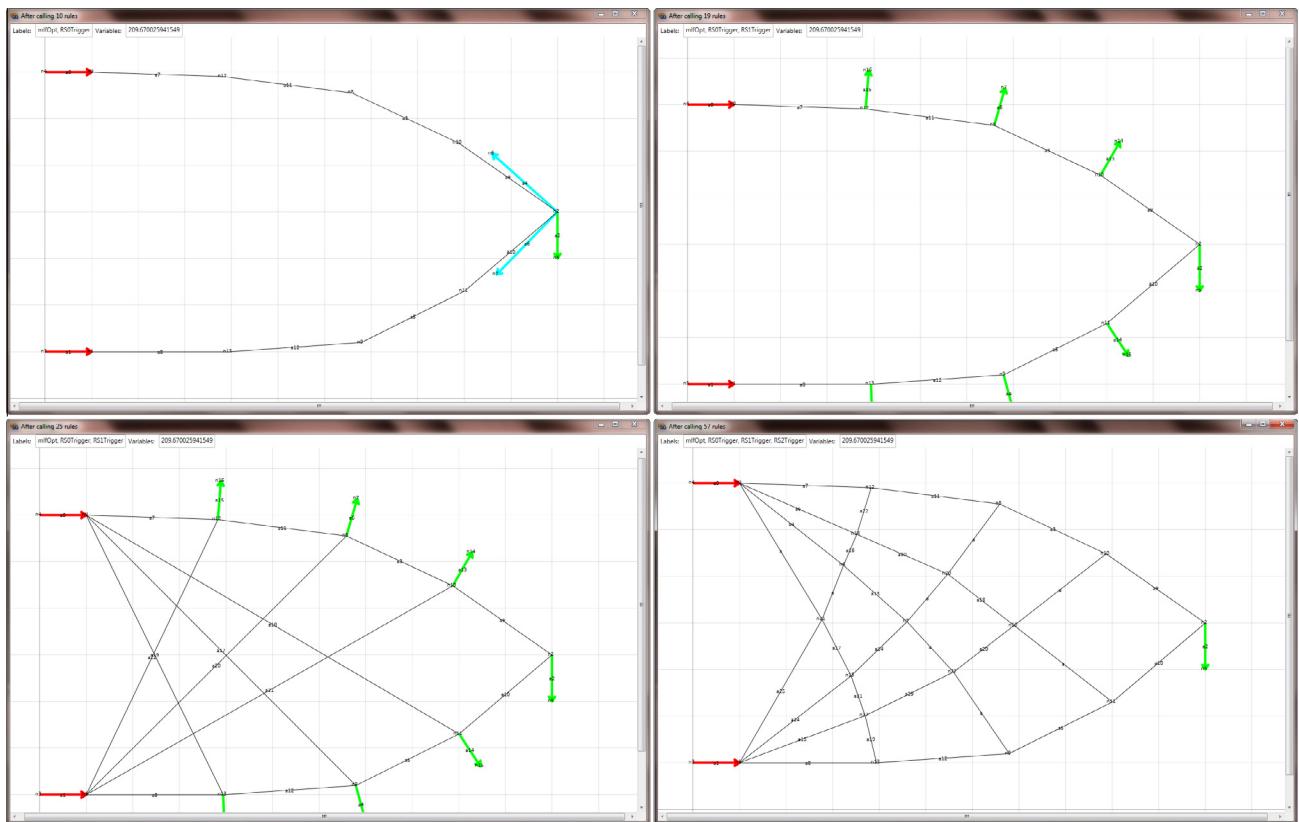


Fig. 13. Generation of the candidate design 32.

BFS algorithm is the possibility to generate the same candidate topology through different sequence of rule applications, therefore after finishing each ruleset the duplicated designs are removed. Two mechanisms have been considered to prevent duplicate designs; the first mechanism is preventing confluent rules from being applied (confluent rules do not invalidate one another, [48]). The second mechanism is a duplicate check. This algorithm – after generating all candidate topologies – compares them with each other and removes those which are repeated.

Fig. 14 illustrates six candidate designs among the 35 generated solutions. The entire approach requires less than 10 s to generate all 35 candidate designs. The top candidate designs in **Fig. 14** are the simplest to generate with less than 20 rule applications. The solutions with more elements in **Fig. 14** require up to 100 rule applications. At this stage, all generated candidate topologies are stored in a sorted list based on their complexity. The complexity criterion in this context is number of structural elements.

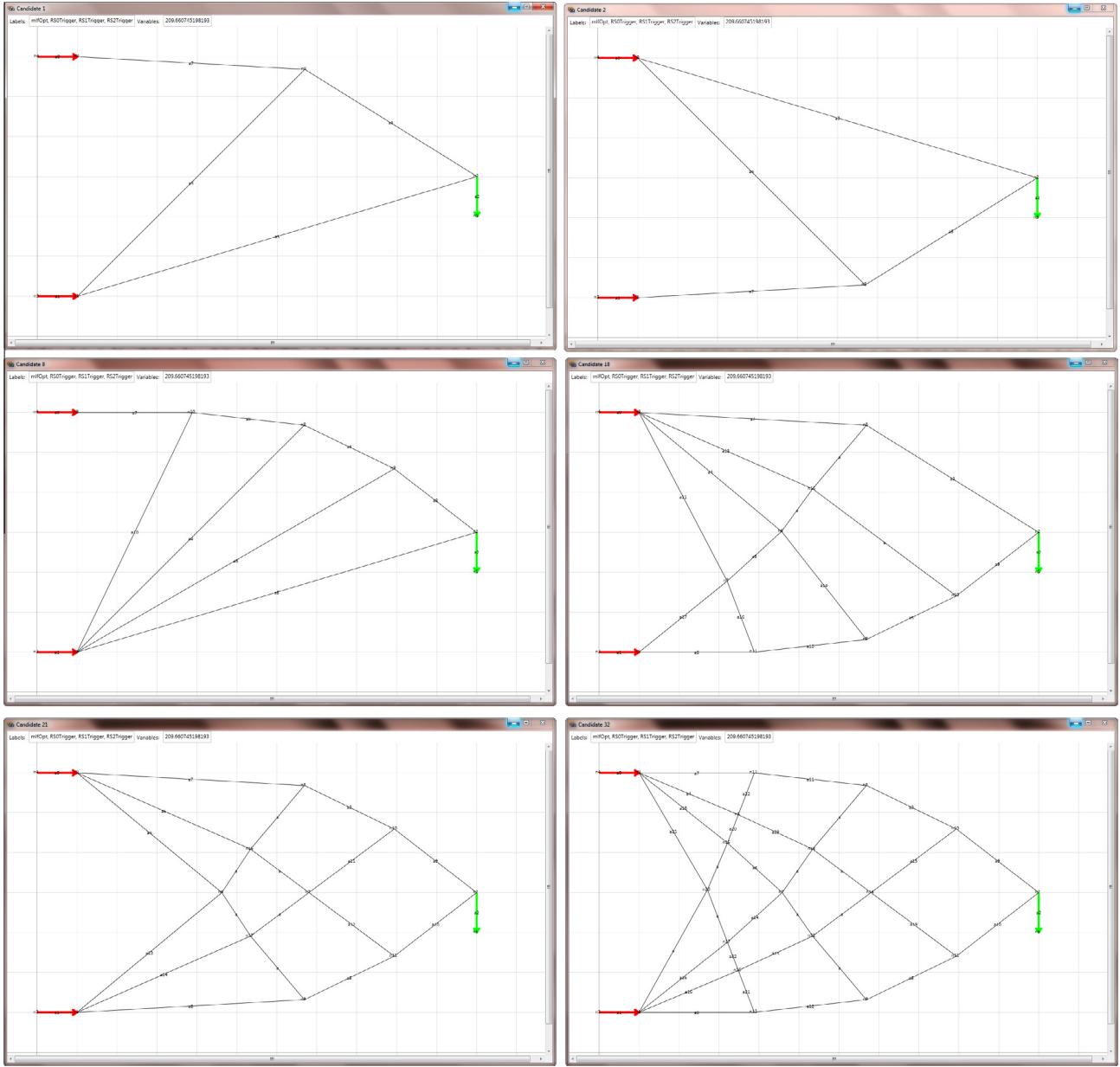


Fig. 14. Six candidate designs randomly selected from a set of 34 feasible designs generated in about 10 s.

It is assumed that construction costs increase with the number of elements. The present approach is to search for the simplest structures that can meet all constraints such as stability, buckling and other spatial constraints, with the best performance (i.e. objective value). Therefore the candidate topologies in the sorted list are fed one by one to the final ruleset for shape optimization. The optimization process is terminated as soon as a candidate design satisfies all requirements. This criterion is indeed a hidden mechanism to meet the important goal of finding optimum design candidates with minimum number of segments that can meet all requirements. Should this objective not be of primary importance for the problem at hand, the above mentioned termination criterion can be deactivated and the shape of all candidate solutions can be optimized.

3.2.4. Optimization process

After storing all results of the exhaustive search in a sorted list, the candidate topologies are further optimized in the second step

of the topology generation phase. Due to the smoothness and unimodality of the design spaces as well as the use of efficient optimization algorithms, most of the candidate topologies can be optimized within a few seconds. Based on the idea of the load path principal, which considers forces like fluid flow, an optimum structure is one that minimizes the amount of force at all elements by best distributing the force between the elements. Although the total load flow in a structure depends on the connectivity of the layout elements such as number of elements, number of nodes, and secondary load paths, we should solve a simple optimization problem to achieve optimum solutions. In fact, defining different layouts with different connectivity is the task of the search algorithm not the optimization.

Although being trapped in a local minimum is one of the main concerns of any optimization effort, this approach uses gradient-based optimization to only change the shape (or positioning of joints) of the candidate solutions rather than the topology. This is because the final topology has already been generated and the

gradient-based optimization is applied to all resulting topologies. Since the topologies are defined using the low-flow principal, the optimized results are not far from the graph generated topologies. This optimization can be tailored to minimize various objective functions such as stress, strain, total load flow, or weight. As an example, the total load flow objective function of the optimization process is for instance defined as (1):

$$f(x) = \sqrt{\sum_{i=0}^N R^2} \quad (1)$$

where x is the spatial position of the joints defining the layout of the structure, N is the number of structure elements in the layout, and R is the amount of load that flows in each element. The objective function of the rule 3 is defined by Eq. (1) plus changes in the direction of load paths, which has been discussed in the sub-section “ruleset 1”. Another objective may be to minimize the total displacement of loaded points (2):

$$f(x) = \sqrt{\sum_{i=0}^L D^2} \quad (2)$$

where x is the spatial position of the joints defining the layout of the structure, L is the number of loads acting on the layout, and D is the displacement at each loaded point. It is also possible to have a combined objective function, but in this study single objective functions with multiple constraints have been mainly used (except for Section 4.2). In order to simplify and speed up the optimization process, conflicting objectives have been considered mainly as constraints. Different objective functions can be subjected to different constraints such as spatial constraints, tension-compression constraints, displacement constraints (if not selected as objective function), and element size constraints.

It should be noted that if multi-objective optimization is performed in order to account for conflicting objectives, weights must be assigned to each objective and candidate designs should hierarchically be sorted with respect to the relative importance of each objective. The same approach should be followed if the structure must carry independent loading conditions each of which may drive the design process by a different extent. However, these issues are beyond the scope of the study and will be addressed in future investigations.

3.2.4.1. Optimization algorithm. For both optimizations (rule 3 and rule 27), the Fletcher-Reeves gradient algorithm is used. Fletcher-Reeves gradient algorithm belongs to a group of methods called conjugate gradient methods. Conjugate gradient method was first generalized by Fletcher and Reeves to solve nonlinear problems in 1964 [49]. The conjugate gradient algorithms converge generally faster than steepest descent directions; because at each iteration the step size is adjusted. In order to determine the step size that minimizes the objective function, a search is performed along conjugate directions. This line search is made to determine the optimal move distance along with the current search direction. The next search direction should be determined in a way that is conjugated to the previous search directions. In general the new search directions are determined by combining the new steepest descent direction with the previous search directions. Various versions of the conjugate gradient methods compute the combination of the new and previous search directions differently. The combination ratio of the Fletcher-Reeves algorithm is the norm squared of the newest gradient to the norm squared of the last gradient.

If the structure includes only one type of compression bar and one type of tension cable rope, the optimization process is continuous. In the present case, no sizing optimization is required. In general, discrete optimization may be performed if there are

different types of elements (i.e. bars and cables) made of different material and including different element cross-sections to be selected from a prescribed set of commercially available values. Layout optimization also can be a mixed discrete and continuous process.

3.2.4.2. Optimization constraints and boundary conditions. Due to the separation of the topology generation from the shape and sizing optimizations, it is possible to easily include additional constraints and boundary conditions in the optimization. However unlike other conventional structural optimization methods [50,51], this separation of the synthesis process does not affect the quality of the results, because the search algorithm is responsible for exploring the whole design space and generating all valid solutions not the optimization. Therefore, the following constraints have been used in this study: (1) stress, (2) displacement, (3) buckling force, (4) outer spatial boundaries to limit the design space, (5) minimum structure element length, and (6) maximum structure element length (this constraint is – aside from the buckling constraint – due to manufacturing restrictions or esthetic aspects).

It is also possible to define other spatial constraints such as specific regions to be avoided like holes or other construction requirements. There is no constraint for possible material or manufacturing imperfections considered in this study. Unlike other conventional methods, which add uncertainty in the position of the nodes or use artificial nodal loads [9,16] to consider imperfections, these issues could be considered in the simulation algorithm, or with a high safety factor for stress, displacement and buckling constraints.

In the next sub section, various examples demonstrate the robust results of the optimization and proposed approach.

3.2.4.3. Optimization results. Analyzing the candidate design in Fig. 6 shows that the amount of load in the compression bar is equal and opposite the amount of load in the tension rope (174 N). With the prescribed components of this study (cylindrical 24 mm diameter bars) and considering the length of the segment, this structure is failing due to buckling; the maximum allowed buckling force for this element was calculated 33.61 N. The objective value for this structure layout (Eq. (1)) is 246.1 N.

Fig. 15 shows the optimized shape of the structure layout which was discussed in Section 3.2. The value of the overall load objective function computed with Eq. (1) is 224.43 N. The maximum load in the structure is 106.96 N in a tension rope and -91.28 N in a compression bar. The reason of this difference lies in the buckling constraint. For this example and the following ones in the approach section, the displacement is not considered as a constraint, but it is determined through the analysis. The displacement at the loaded point is (13.10, -50.58) mm in x and y directions. This seems large, because the selected rope is very thin (just 2 mm diameter) but it has a very high ultimate load (2540 N/mm²). This selection was deliberately done to challenge the optimization process.

Fig. 16 shows an optimized structure with 15 joints and 32 components. The optimization took about 65 s and the best objective function was 265.51 N. This value is higher than the corresponding value obtained for the design shown in Fig. 15. However, the load is much better distributed and the structure layout is more symmetrical. Consequently, the buckling constraint was almost inactive. The maximum tensile force in the cables and maximum compressive force in the bars are 78.05 N and -77.97 N, respectively. Displacement values computed for the loaded node, in particular the displacement in the direction $-y$ of the applied load, improve significantly: (15.90, -31.35) mm in x and y directions, respectively.

The optimized layout shown in Fig. 17 is the same as that in Fig. 16. The only difference is in the fact that no design space

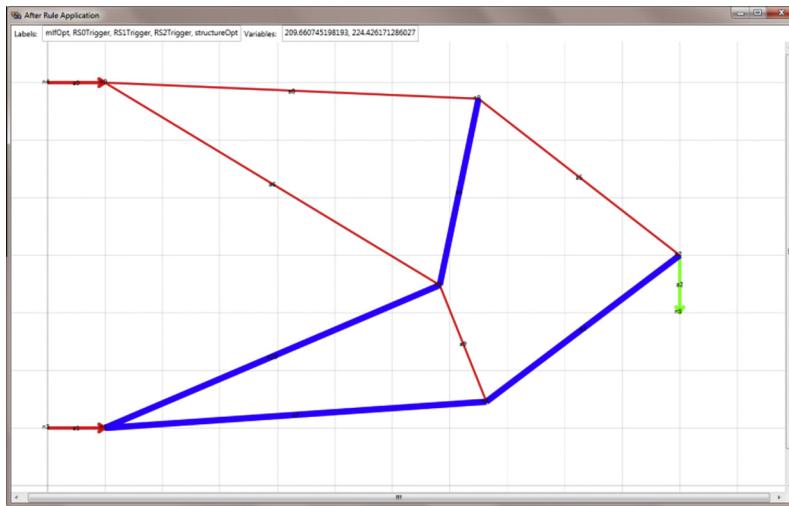


Fig. 15. Optimum design found by minimizing the overall load flow.

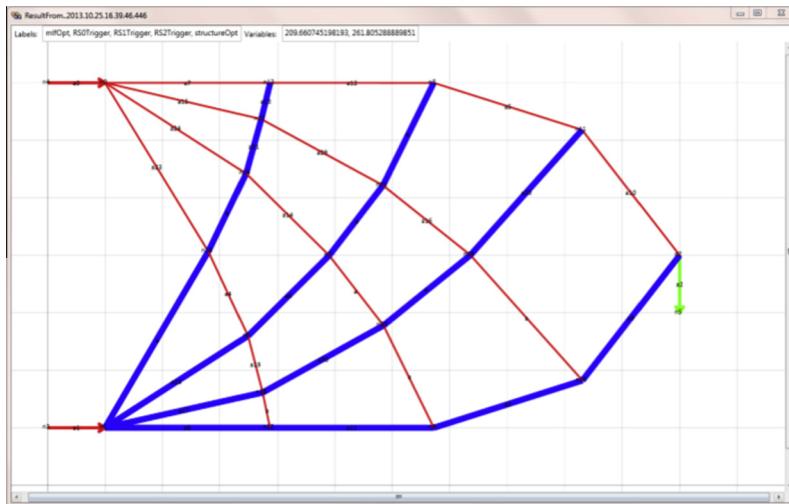


Fig. 16. Optimum design found for a finer segmentation (the overall load flow is minimized).

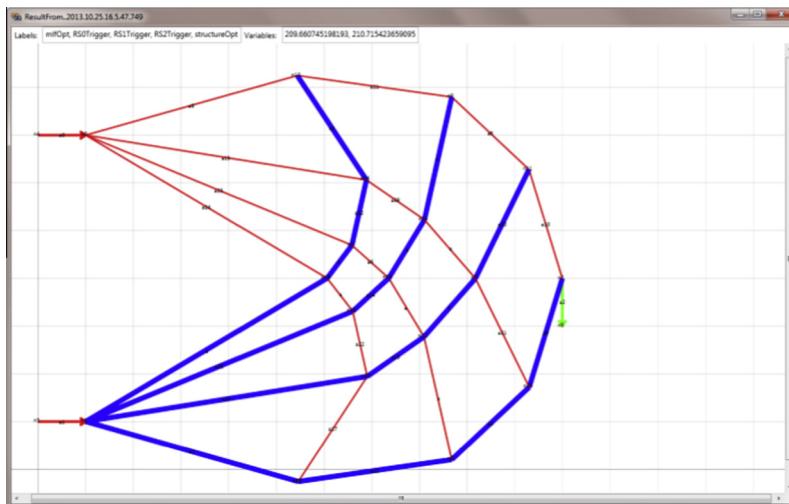


Fig. 17. Optimum design found without including design box constraints (the overall load flow is minimized).

limitation is considered for the structure, therefore the structure extends out of the initially considered design box. This leads to improve significantly the overall load flow and the displacement of the loaded point: the cost function is 210.58 N while nodal displacements are (6.83, -25.99) mm. The maximum tensile force in the cables and maximum compressive force in the bars are 60.11 N and -62.53 N, respectively. Therefore, load is very well distributed in the structure.

The structural layout shown in Fig. 18 is topologically similar to the ones in Figs. 16 and 17. However, the optimum design was obtained by minimizing the displacement of the loaded point which now is (2.55, -23.04) mm in x and y directions, respectively. This is the lowest value among all 35 candidate solutions; if a displacement less than this is desired, there are two options. The type and size of the used components could be changed, or a finer approximation of the load path could be adopted. This can be achieved by reducing the minimum length of the segmentation which is 225 mm and consequently creating more segments in the main load path. The maximum tensile force in the cables and maximum compressive force in the bars are 47.46 N and -70.17 N, respectively. This indicates that the design is driven by the very thin wire ropes.

Finally, Fig. 19 shows the optimal designs for the same problem as in Fig. 15, with the lower fixed point bearing only forces in the

x-direction. Considering the layout total load flow as the objective function, a displacement of (9.46, -65.70) mm in x and y directions and an objective value of 229.73 N is obtained. By considering the displacement as our objective function the displacement reduces slightly to (9.28, -62.53) mm and the load flow increases to 258.26 N. The displacement is not significantly reduced with respect to the previous case. The buckling constraint leads to a structural layout with higher tensile forces developed in the cables: the maximum tensile force in the ropes is 127.23 N (Fig. 19a) or 116.67 N (Fig. 19b) versus only 70.70 N and 99.72 N maximum compressive forces developed in the bars.

It is possible to consider different parameters of the optimization, but the main point of this methodology is in the way that we approach the topology generation (through load flow principal) and its formulation and representation in grammar rules. Finally, the use of a tree search algorithm to explore the design space appears to be novel. The efficiency of the proposed methodology is verified by solving various test problems (see Sections 4 and 5).

3.3. 2D to 3D transformation for graphical representation

After automated synthesis of the topologies, they can be transformed into three dimensional shapes through a converter which uses the Parasolid geometric kernel [52]. The transformer converts

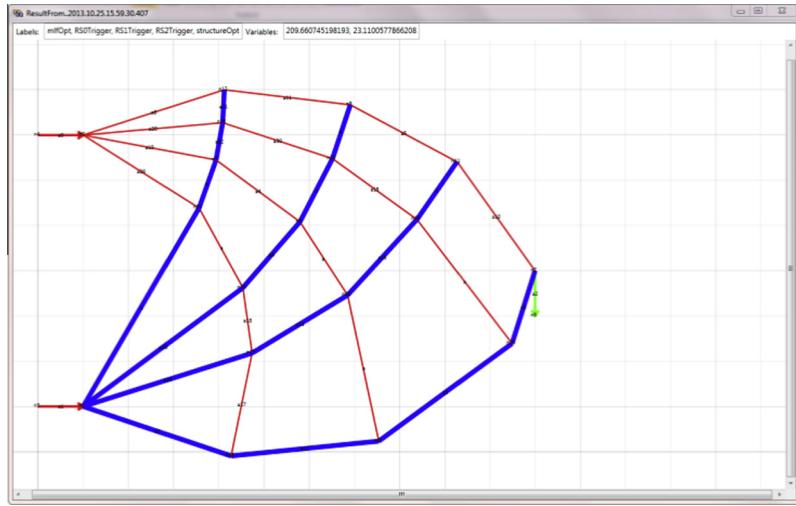


Fig. 18. Optimum design found by minimizing the displacement constraint at the loaded point.

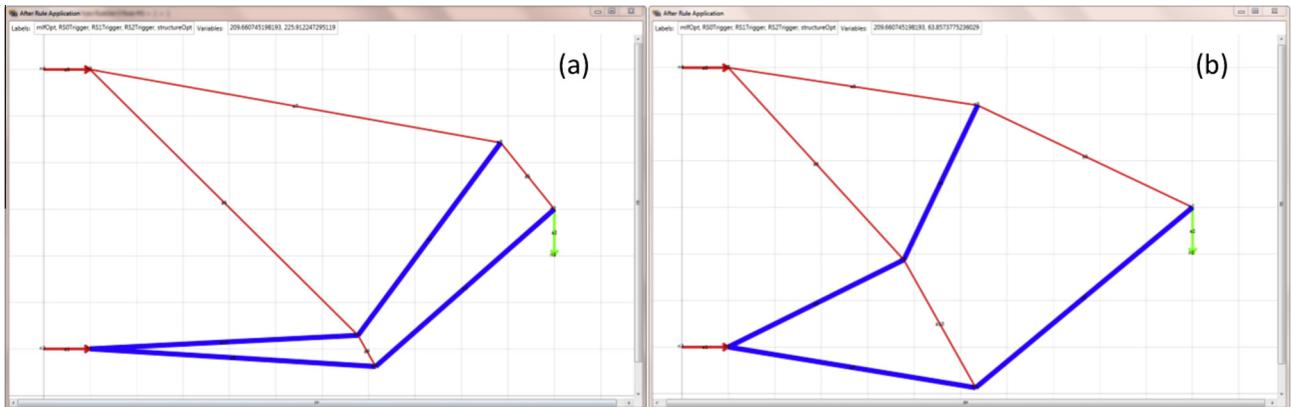


Fig. 19. Optimum design found without constraining the y-displacement of fixed point n1: (a) the overall load flow is minimized; (b) the displacement at the loaded point is minimized.

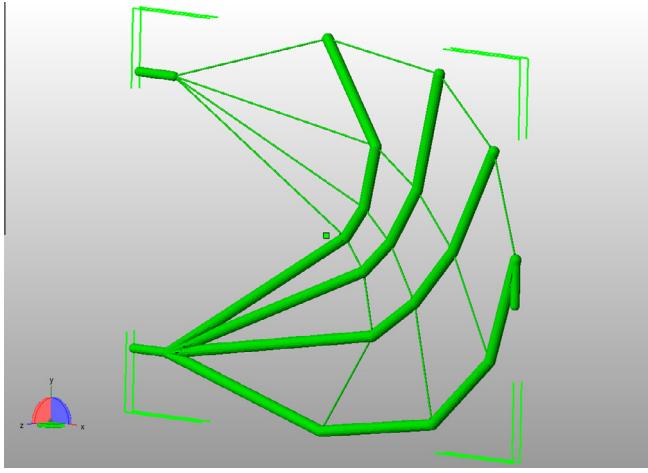


Fig. 20. Optimum topology converted into 3D shape.

nodes in the graph into spheres, and arcs into cylinders. The shapes are saved as STL files. **Fig. 20** shows the candidate topology of **Fig. 17** converted to a 3D shape. For this specific design, the conversion took less than 2 s. As discussed before, this transformation is only for 3D visualization of the results, which is necessary when synthesizing problems in 3D space.

4. Results and discussion

The proposed methodology is tested on various benchmark truss design problems including several variants of problem formulation. The emphasis is on the topology generation phase although some test cases served to evaluate the optimization capability of our approach.

As pointed out in the previous sections, it is possible to include different element types in the structure like rectangular, hollow cylindrical or triangular cross-section bars of different size as well as various construction types and sizes of ropes. Section 4.1 presents a test problem where the bar diameter is increased to 40 mm: this allows sensitivity of structural layout to sizing variables to be evaluated. Section 4.2 is concerned with simultaneous optimization of structural layout and sizing optimization of bar cross-sectional areas. Sections 4.3–4.5 regard structures for which the construction type of the cables is changed to $(6 \times 7(1+6) + 1 \times 7(1+6))$. The objective function considered in the test

problem is always the total load flow computed with Eq. (1), unless explicitly mentioned.

4.1. Effect of sizing variables on optimized layout

Fig. 21 illustrates the effect of sizing variables on the structure layout. The size of bars was increased from 24 mm to 40 mm. **Fig. 21a** and b compares the optimized designs found by minimizing the total load or the loaded point displacement, respectively. The following results were obtained:

- 208.38 N overall load flow, $(8.94, -50.19)$ mm nodal displacement, maximum tensile force 100.32 N, maximum compressive force 98.78 N;
- 231.30 N overall load flow, $(7.99, -39.73)$ mm nodal displacement, maximum tensile force 95.14 N, maximum compressive force 116.55 N.

The design process is marginally driven by buckling. Since the displacements of loaded point depend on the stretching of wire ropes, the compression forces developed in the structure become higher than tension forces as soon as cable stretching must be limited to cope with the chosen objective function.

4.2. Combined shape and sizing optimization

The size of bars and cables were also allowed to vary in the optimization process. The Fletcher–Reeves conjugate gradient algorithm was used to find the optimum structural layout and element sizes. The initial design included cylindrical rod elements of diameter 40 mm. A multi-objective optimization problem searching for the minimum weight of the structure and the minimum nodal displacement was solved. This test case was particularly challenging as the two objectives considered in the optimization problem are conflicting. There are 14 design variables: 6 nodal coordinates (three nodes in 2D) and 8 sizing variables (the radii of all elements). **Fig. 22** shows the optimized size and layout of the structure. **Fig. 22b** refers to the case of the structure optimized with an applied load of 300 N while the structure in **Fig. 22a** refers to the case of 100 N applied load.

Table 1 shows that, when the applied load is 100 N, the radii of the elements were mainly reduced to less than 40% of their initial values. In spite of considering a three times higher load (i.e. 300 N vs. 100 N), the optimizer still reduces the size of most elements to around 45% of the initial design. However, structural layouts are slightly different. Element sizes were optimized with continuous

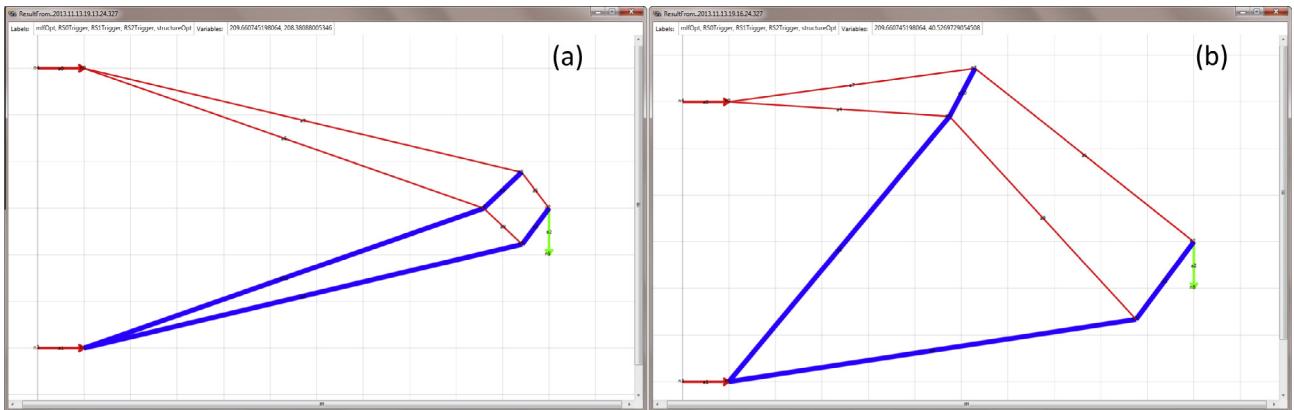


Fig. 21. Optimum designs found by setting bar diameters to 40 mm: (a) the overall load flow is minimized; (b) the displacement at the loaded point is minimized.

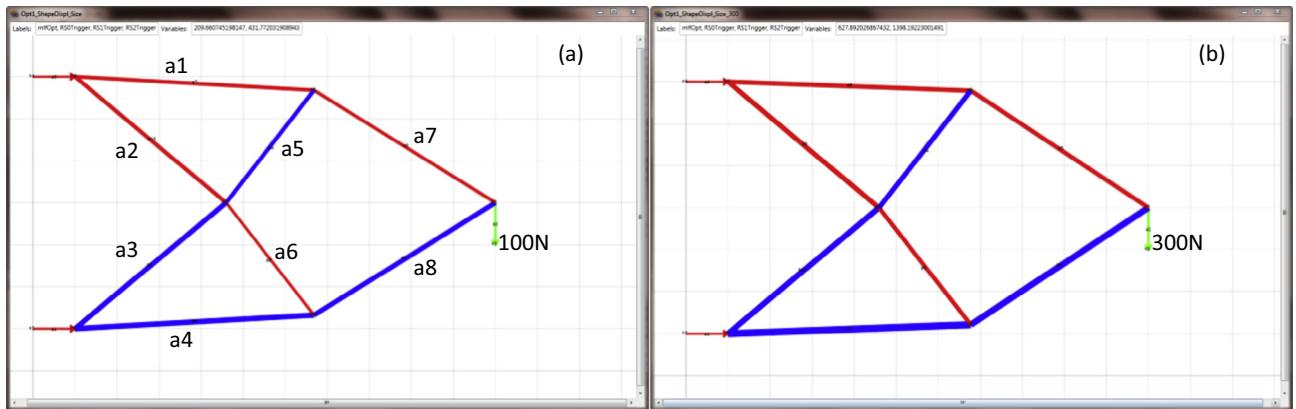


Fig. 22. Results of multi-objective optimization for minimum displacement and minimum weight: (a) applied load of 100 N; (b) applied load of 300 N.

Table 1

Element size and corresponding axial forces for the optimized designs shown in Fig. 22.

Arc no.	100 N.			300 N.		
	Axial load (N)	Radius (mm)	Rounded (mm)	Axial load (N)	Radius (mm)	Rounded (mm)
a1	114.736	8.571	9.00	335.645	10.210	11.00
a2	67.898	7.965	8.00	214.522	11.024	12.00
a3	-67.850	9.594	10.00	-214.592	12.765	13.00
a4	-114.760	12.072	12.50	-335.564	15.880	16.00
a5	-55.205	7.941	8.00	-175.012	11.082	12.00
a6	55.217	6.986	7.00	174.859	9.556	10.00
a7	94.818	7.609	7.50	272.400	9.872	10.00
a8	-94.818	10.834	11.00	-272.353	14.106	15.00

values and then rounded to the nearest standardized value. Discrete optimization of component sizes is currently being investigated to enhance the practical utility of the proposed methodology. To ensure the feasibility of the optimum solutions, they are rounded to the nearest upper sizes.

4.3. Effect of direction and position of the applied load

Fig. 23 shows the optimized layouts for the different directions of the applied load. The layout of the structure does not change but elements are all subjected to tension or compression based on the load direction. This is consistent with the objective of minimizing the total load flow which is equal to 95.15 N. The displacement of the structure under tension is $(13.74, -0.44)$ mm in x and

y -directions. The structure under compression undergoes significantly smaller displacements, $(-0.42, 0.01)$ mm.

Fig. 24 shows the optimization results obtained by changing the direction of the applied load from horizontal to vertical. The load flow is clearly illustrated by the optimized layout. The construction type of ropes is $(6 \times 7(1+6) + 1 \times 7(1+6))$ and the diameter of bars is increased to 40 mm. The latter serves to prevent buckling. The same sizes for bars and cables are chosen in the test cases illustrated later in the paper.

4.4. Evaluation of performance in terms of optimum topology and shape

Design of cantilever beams of different lengths loaded at different positions and Michell type structures are the most common benchmark problems in topology optimization literature [1,2,18,53–58]. The validity of the present approach is evaluated first in terms of ability to create topologies similar to literature and then with respect to layout optimization capability.

Fig. 25 shows two candidate designs generated for the short cantilever beam, selected from a set of 35 candidate designs created for this test case in Section 3. The layout of structure shown in the left part of Fig. 25 is slightly different from its counterpart shown in Fig. 15 because in this case buckling constraints are not active due to using thicker bars (40 mm vs. 24 mm).

Fig. 26 shows other topologies generated for longer cantilevered beams. Different sizes, dimensions, and loads and support numbers were considered always following the load flow principal method. Topologies and layouts are very similar to those reported in

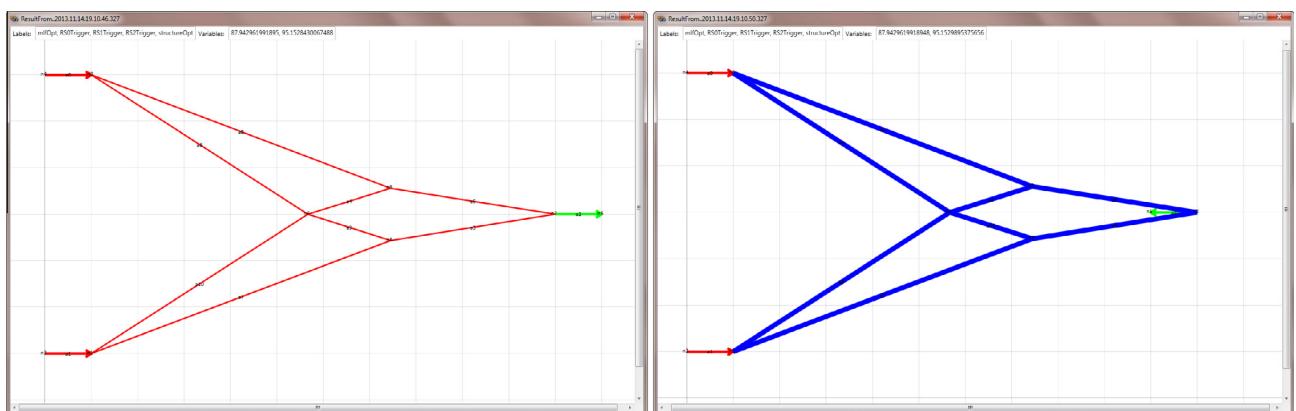


Fig. 23. Sensitivity of optimized layout to load direction.

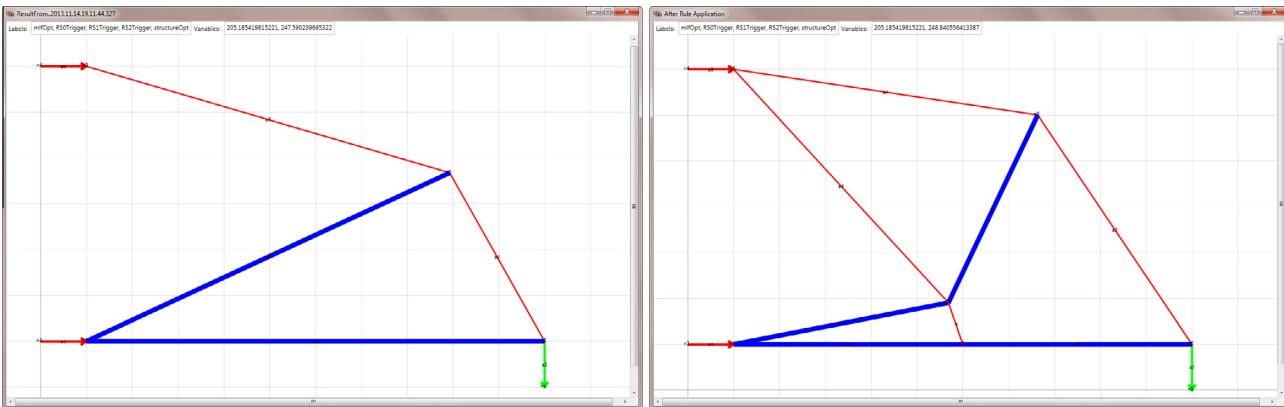


Fig. 24. Sensitivity of optimized layout to load position.

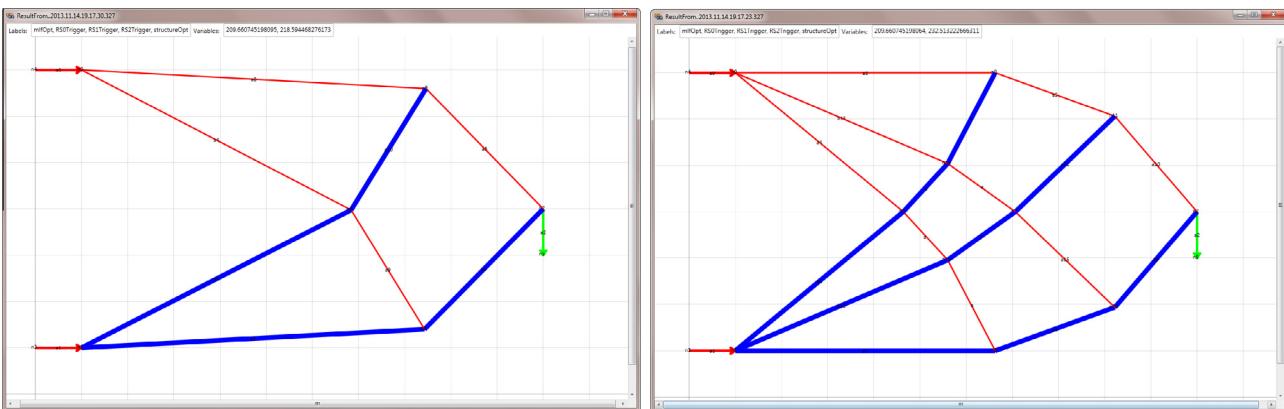


Fig. 25. Examples of optimized designs for short cantilever beams.

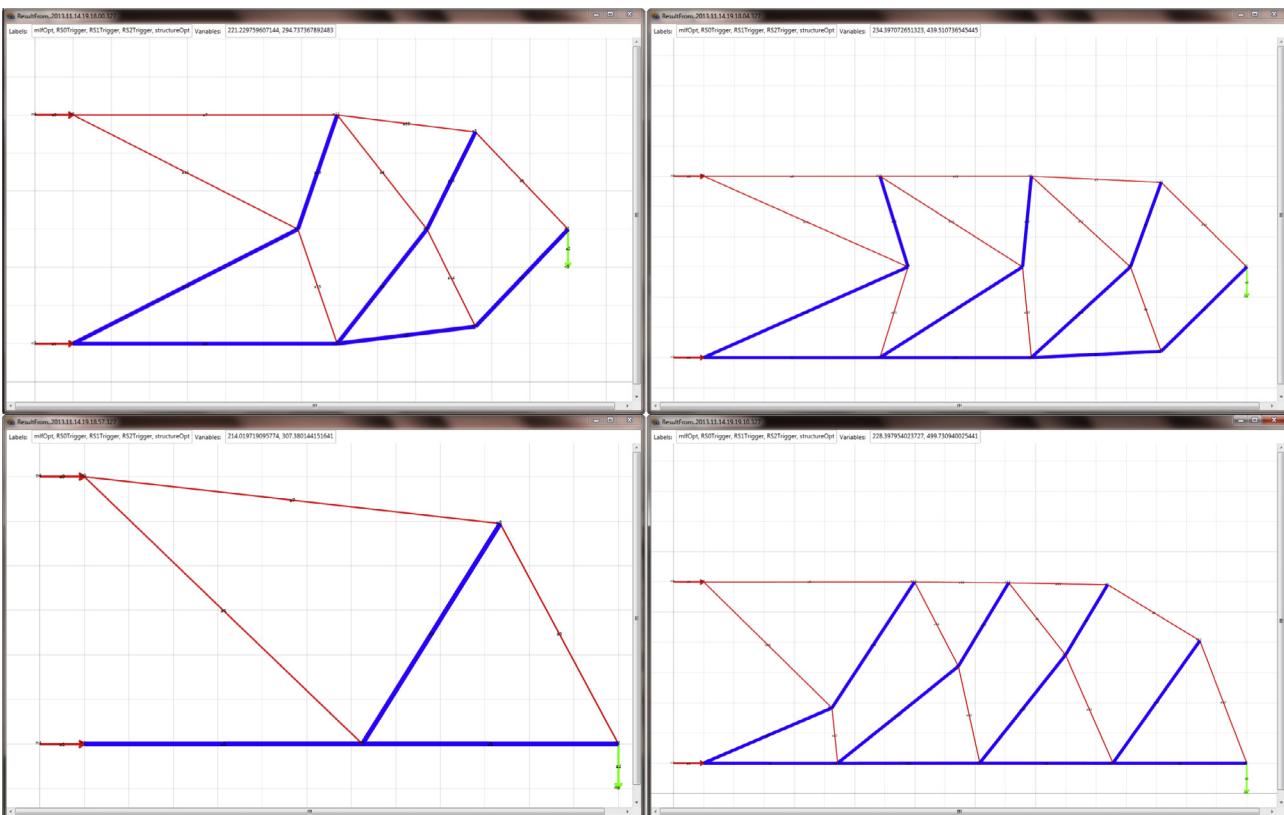


Fig. 26. Examples of optimized designs for long cantilever beams.

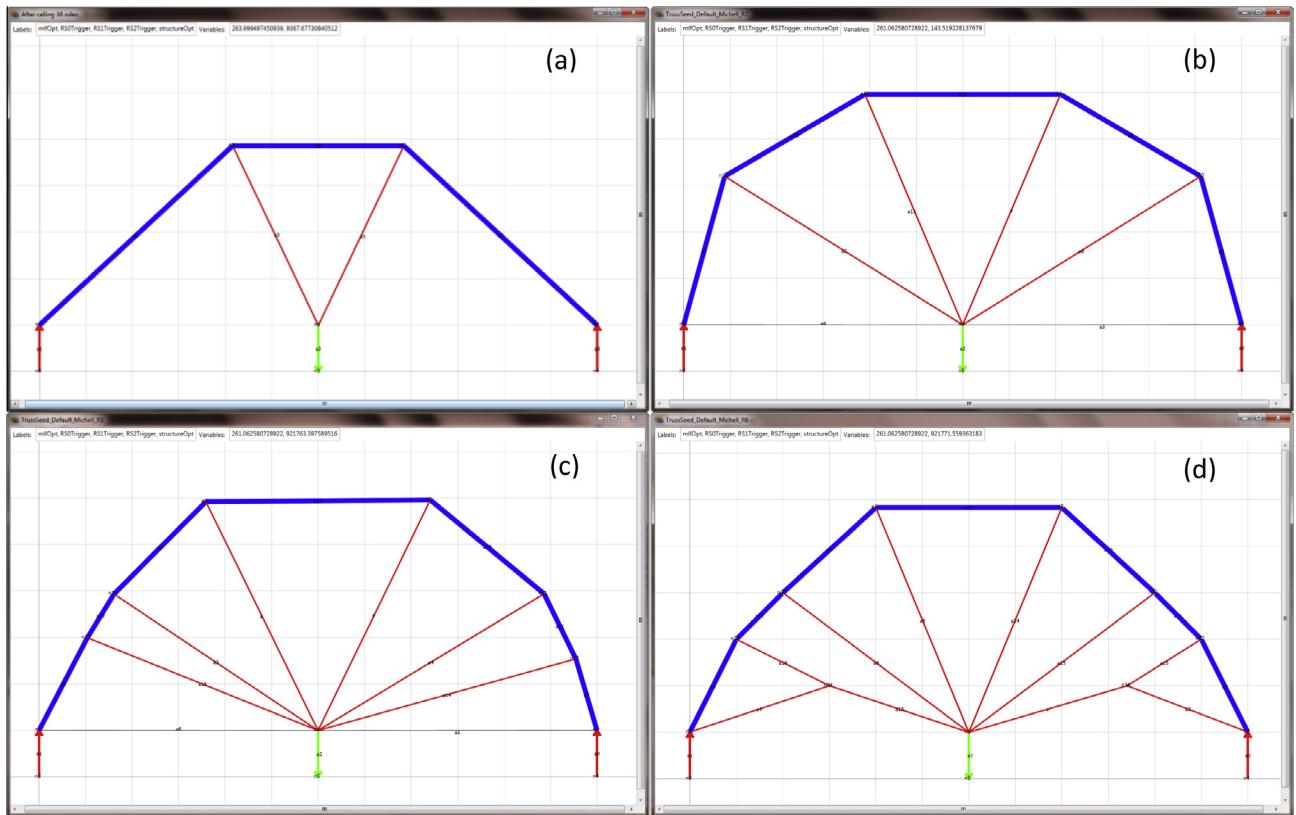


Fig. 27. Examples of Michell beam optimized designs similar to the literature.

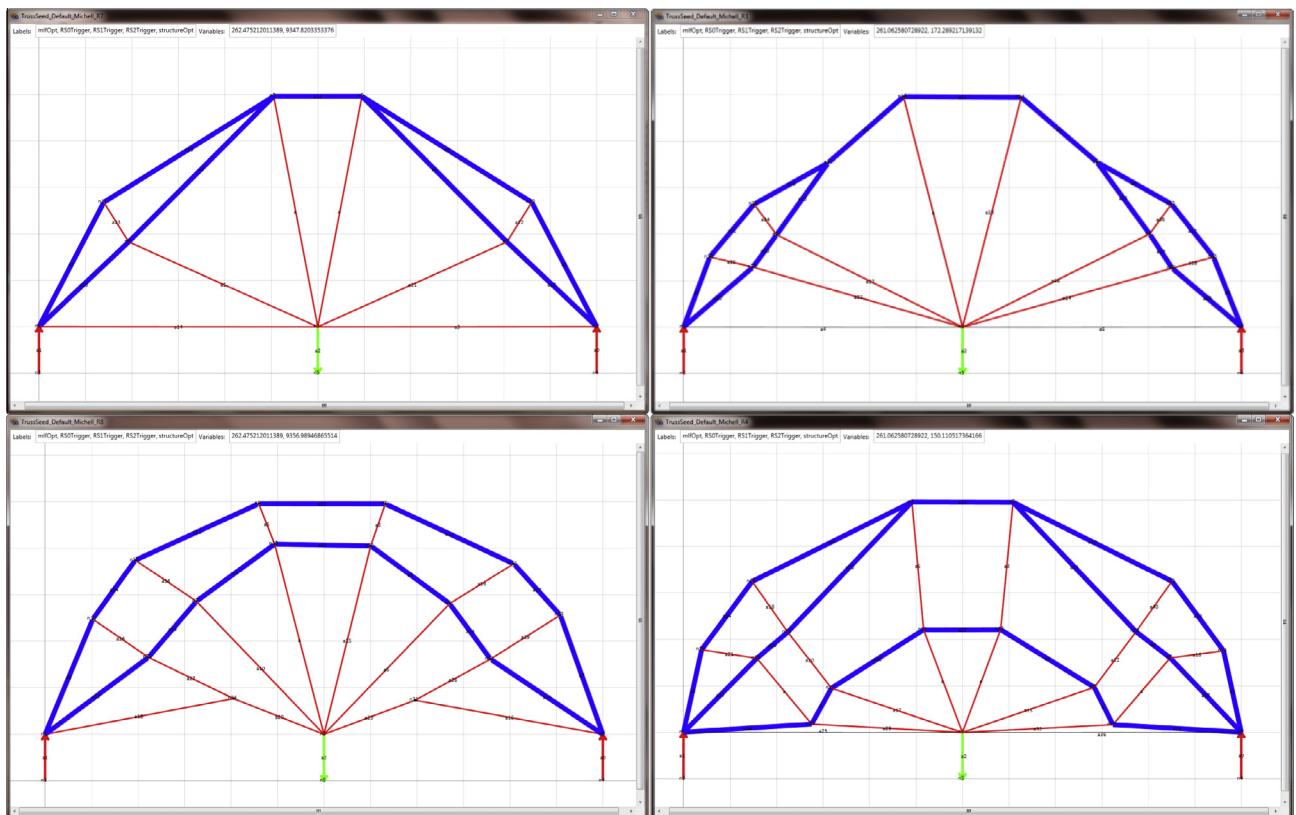


Fig. 28. Examples of Michell beam optimized designs found with a finer segmentation.

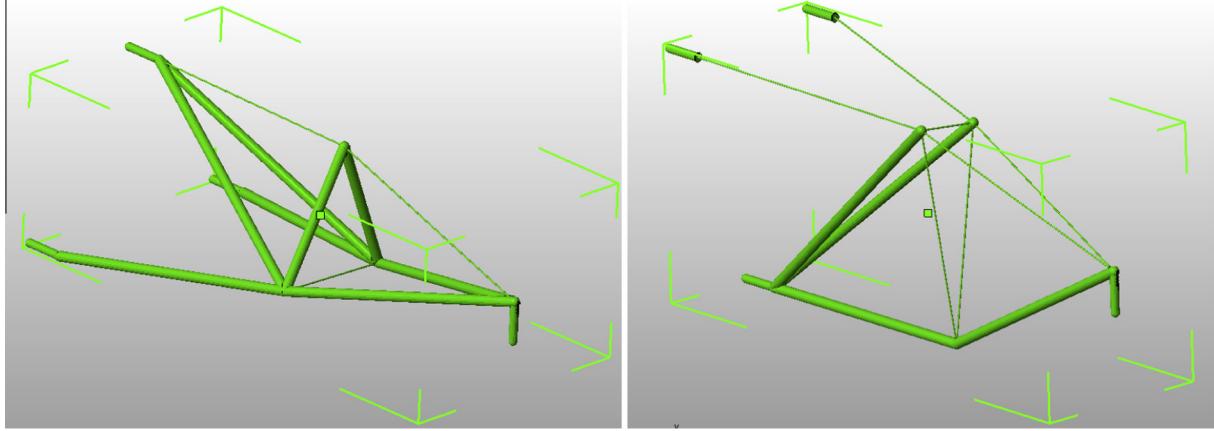


Fig. 29. Optimized design of a 3D short cantilever beam.

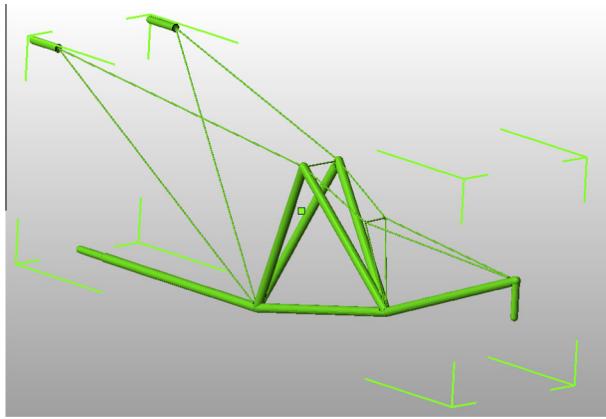


Fig. 30. Optimized design of a 3D long cantilever beam.

literature. However, shape is very sensitive to objective function and active constraints.

Topologies obtained for the Michell type beam problem are shown in Fig. 27. These results are again consistent with literature. This test problem requires a higher computation time due to the

sharper change in the load flow direction: in the previous examples, the load changed by at most 90° while in this example changed by 180°.

By setting the minimum size of the path segments to 225 mm, only two candidate designs are generated in less than 1 s, one of which is shown in Fig. 27a. By reducing the minimum segment size to 125 mm in order to achieve a finer approximation of load paths, it was possible to generate 53 candidate designs in 19 s: three of those designs are shown in Fig. 27b-d. Finally, by reducing the minimum segment size to 100 mm, it was possible to generate more than 500 candidate designs in 10 min: Fig. 28 shows four of those designs. It should be noted that the set of designs generated using finer segmentations always include the set of designs generated with longer segmentations.

Regardless of segmentation size, candidate designs can be sorted with respect to the following criteria: (1) number of elements, (2) initial evaluation of candidate designs, and (3) degree of symmetry of candidate designs (if appropriate). Therefore, candidate designs with lower costs (i.e. including fewer elements), better evaluation results and probably better symmetry are the most suitable for the final optimization. Optimization results show that the total load flow for the structures in Fig. 27 is slightly lower than that evaluated for the designs in Fig. 28. The structural configurations shown in Fig. 27 undergo larger

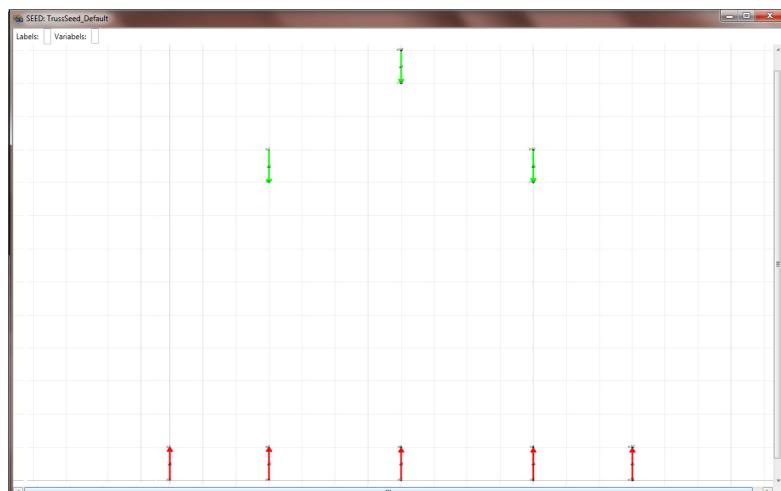


Fig. 31. Seed graph representation of a truss optimization problem with three masses and five fixed points.

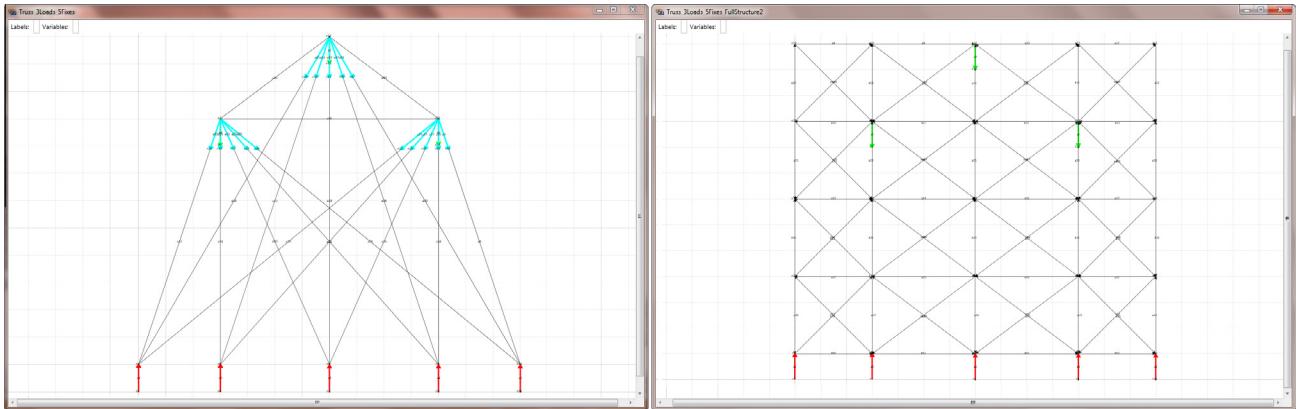


Fig. 32. Ground structures commonly used for the problem defined by Fig. 31.

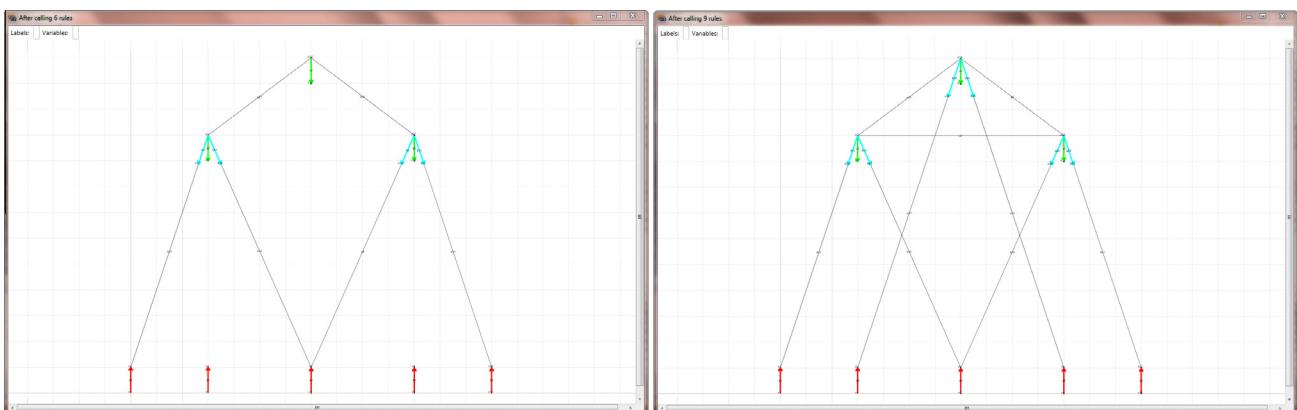


Fig. 33. Topology of candidate designs for the problem defined by Fig. 31.

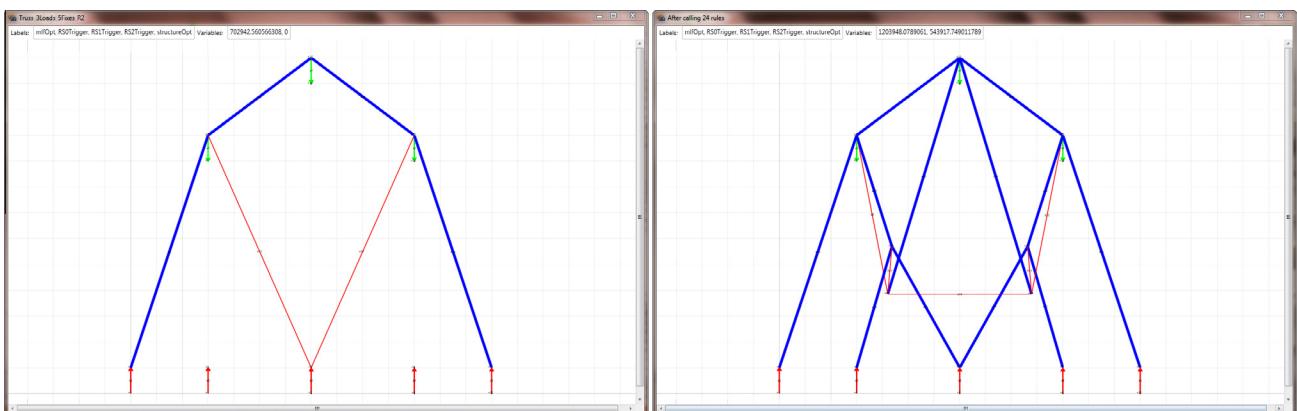


Fig. 34. Optimized layouts for each topology shown in Fig. 33.

displacements than those schematized in Fig. 28. If, aside from stability, robustness and safety considerations, aesthetical aspects are important, the structural layouts shown in Fig. 28 are preferable. The present methodology can cover the whole design space, thus generating many sophisticated designs. This provides the designer with a wide range of possible options. A visual representation like that adopted in this study may help designers to see the tradeoffs in the results.

In this study, segmentation rate of structure components and displacement/total load flow objective functions have been considered separately. However, results show that there is a relationship between segmentation rate and structural performance. While it is not easy to formulate search rules so as to account inherently for this relationship, it appears that the search process of a global optimum having set a specific segmentation rate will vary as the minimum size of the segments changes. The segmentation rate defined

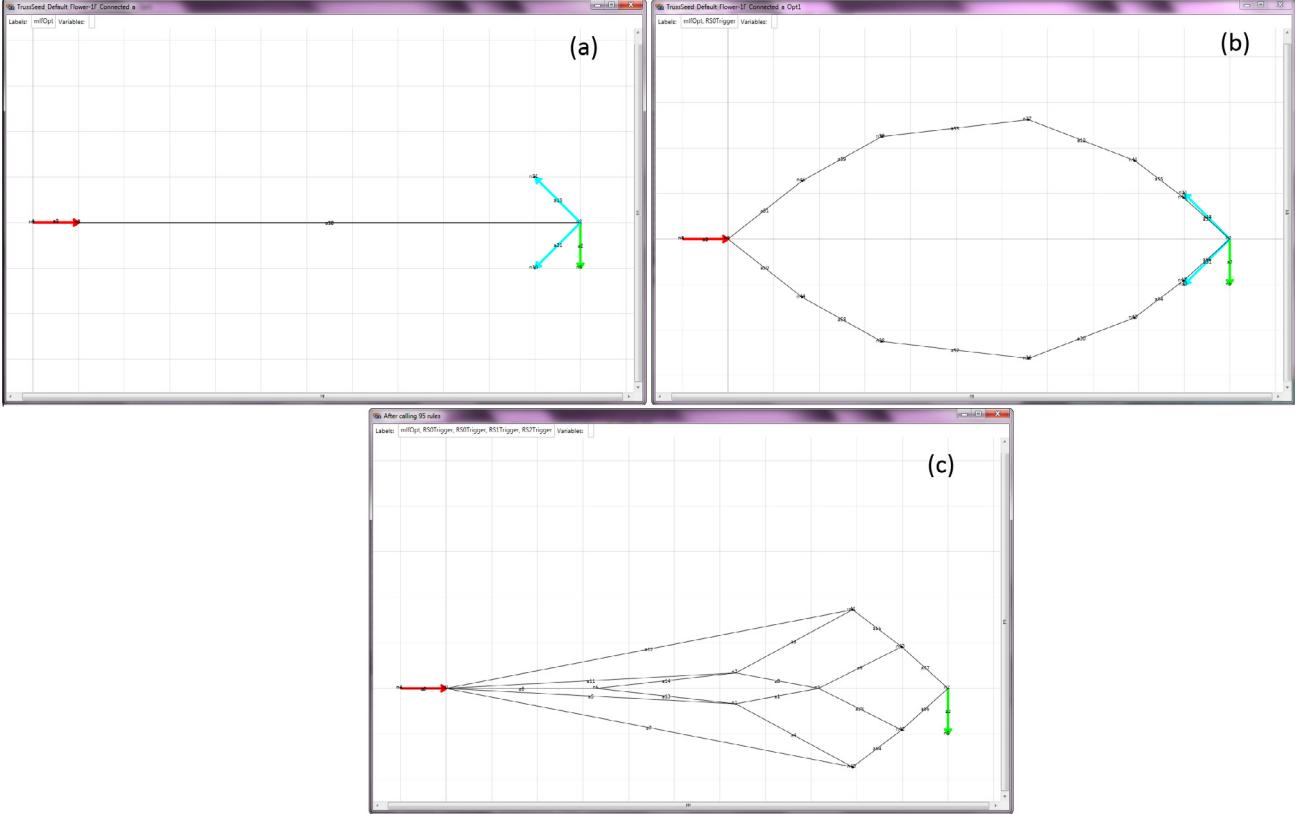


Fig. 35. Test problem with one load and two coinciding fixed points: (a) Main load paths; (b) Optimized load paths; (c) Example of candidate design.

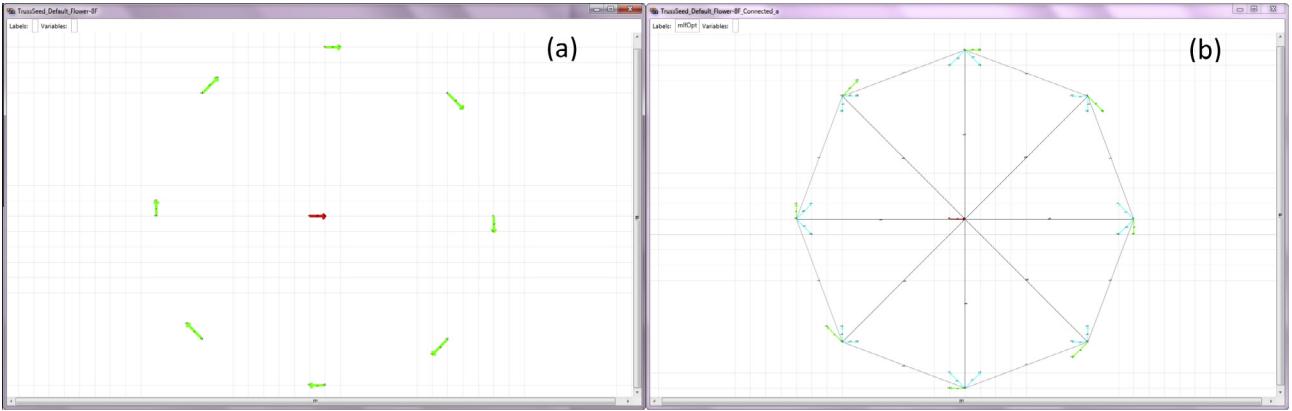


Fig. 36. Design of a structure with circular layout and eight concentrated loads.

in our methodology corresponds to the grid size of grid-based topology optimization methods: the finer is a grid the better will be the final result. However, selecting finer segmentations entails not only higher computation times but may also increase the risk of getting stuck in local minima.

4.5. Three dimensional problems

Since the concept of load flow path applies to both 2D and 3D problems, the methodology developed in this study can design 3D structures with almost no change in the rules. However, the computational cost of the optimization increases by 50% and an extra conversion is required for visualizing graphs. The conversion

may take up to 10 s for large-scale structures. Rule 24 must be modified because finding element intersections in 2D differs from doing the same in 3D.

Figs. 29 and 30 show the results obtained for three test problems characterized by different positions of loads and fixed points, which obviously affects the optimum design. In particular, the structure shown in Fig. 30 is 50% longer than the structures shown in Fig. 29. The number of candidate designs generated for the test problems of Fig. 29 is almost twice as that generated in the similar 2D problem discussed in Section 3 (69 topologies with minimum segmentation 225 mm vs. 35 topologies). Consequently, the computation time increased to 2 min, much higher than the 10 s required for the 2D problem. This is because the tree-search algo-

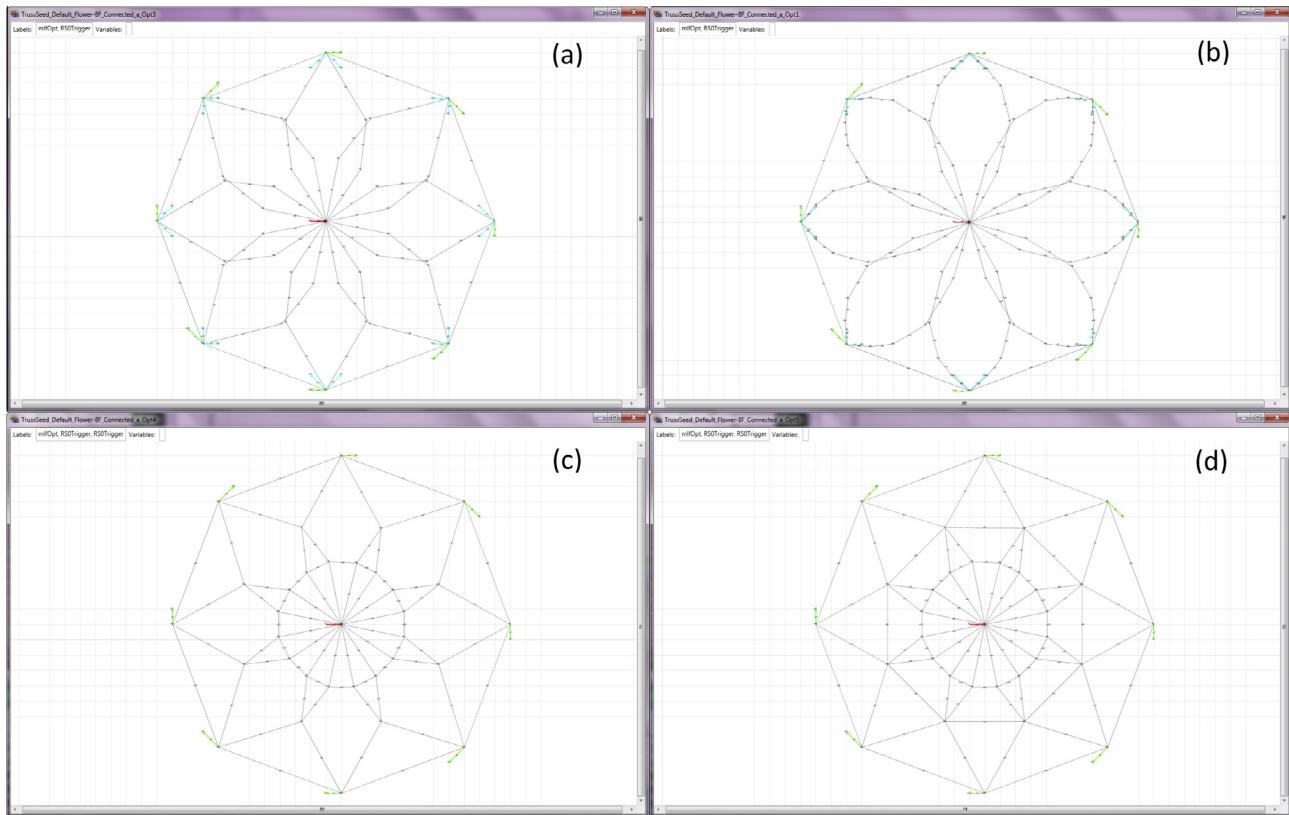


Fig. 37. (a,b) Candidate load paths for the test problem schematized in Fig. 37 obtained for different segmentation rates; (c,d) candidate topologies corresponding to the load paths schematized above.

rithm requires more time to search a larger design space and find feasible designs. An interesting characteristic of the structure shown in Fig. 30 is the concurrence of three wire ropes at two joints. Using cables in truss constructions allows building flexible and foldable lightweight structures that act as rigid structures when fully deployed.

4.6. Exploring an arbitrary ground structure

In this sub-section it is shown how the proposed method explores the design space in a more general fashion than a fixed ground structure. The efficiency of the proposed approach for structures with more than one load and many optional support points is demonstrated through the test problem schematized in Fig. 31. The arrows at the top of Fig. 31 represent three masses (which are 100 N, 200 N and 100 N respectively) and the five arrows at the bottom of Fig. 31 are the optional support points. The objective is to design a truss structure with minimum total load to sustain these masses.

Other approaches to truss topology optimization typically discretize the design space with a nodal mesh as a ground structure, in which every node is connected to almost every other node in the domain and the members have identical cross-sectional area. Fig. 32 shows two typical ground structures for the defined problem in Fig. 31. In the left picture of Fig. 32 all loads are connected to each other and to all optional supports, while in the right picture a grid of nodes is the ground structure. The present methodology in this study has the directness of the first ground structure and at the same time has the flexibility of the second ground structure approach. Similar to ground structure approaches, the maximum number of segments included in the structure is limited. However, the design space is not fixed to a set of nodes (that includes all

possible topologies). The rules utilized in the search phase serve to explore all candidate topologies in the design space while the optimization algorithm finds the optimum layout and size of each candidate design.

As discussed previously, the proposed methodology explores the design space and finds all possible ways to flow the load from the applied load points to the supports. This concept is the same regardless of the number of load or support points. Fig. 33 shows two candidate topologies, which satisfy the minimum requirements for a stable load flow. The left structure is minimally stable as removing any component can cause instability in the structure. Shape and sizing optimizations of the simplest feasible designs (e.g. Fig. 33) are performed. The synthesis process can be terminated as soon as a design is found that meets all requirements and does not violate optimization constraints. However, if the manufacturing cost of the structure is not the only objective of the automated synthesis process, the optimization can be continued until an appropriate solution is obtained. In particular, all candidate topologies must be explored in order to find the global minimum.

Fig. 34 shows the optimized shapes of the candidate topologies in Fig. 33. The right structure shows clearly that the proposed methodology is not limited to the fixed ground structure suggested in Fig. 32. It combines the directness of the first type of the ground structures with the wide covering of the second type of the ground structure to find the optimum size, shape and topology of the structures.

4.7. Comparison to common results

The effectiveness and performance of this method is examined by solving a variety of similar test problems to those found in the

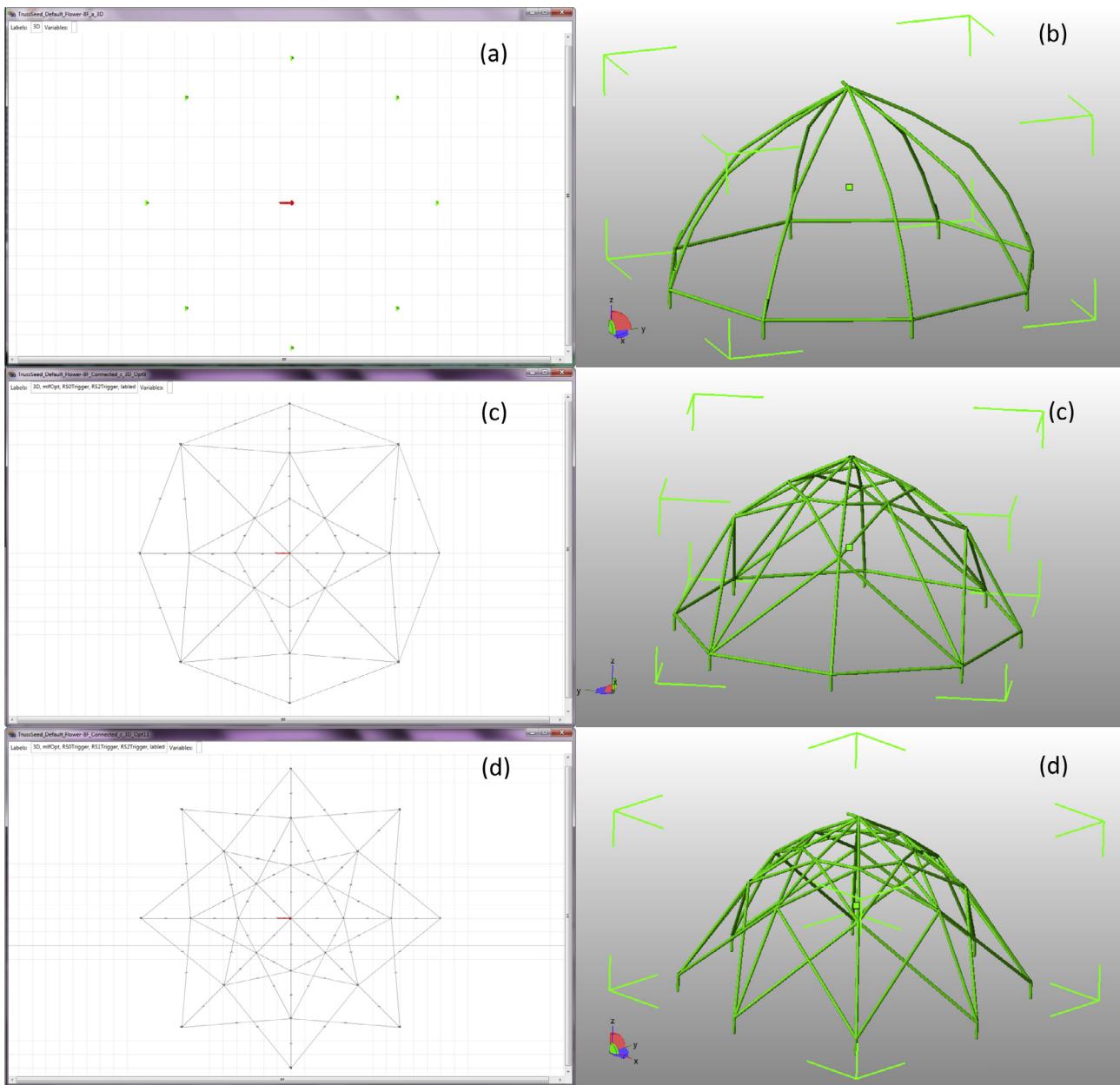


Fig. 38. Spherical dome design problem: (a) Seed graph; (b) segmented load path; (c) candidate design 1 in 2D and 3D; (d) candidate design 2 in 2D and 3D.

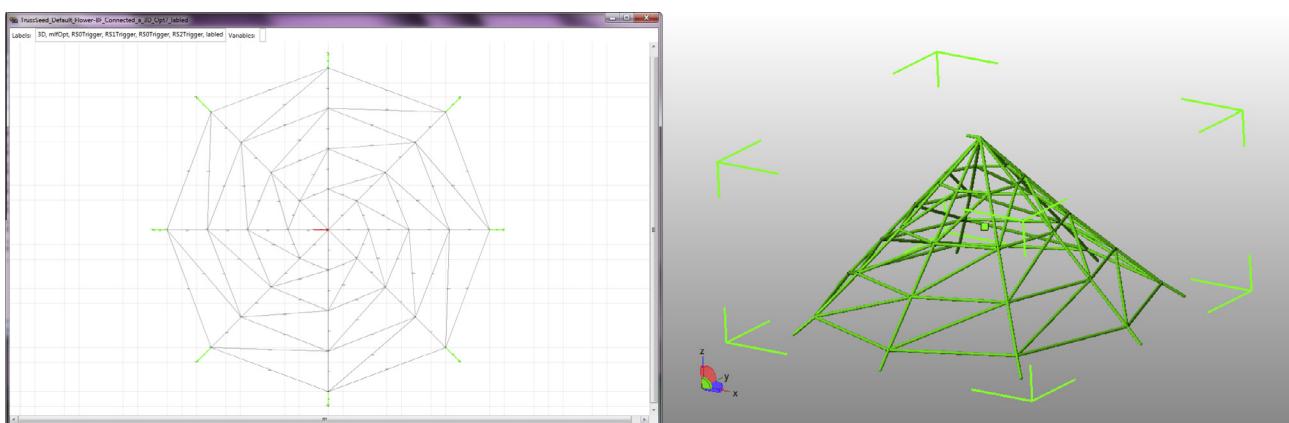


Fig. 39. Candidate design generate for a spherical dome subject to inclined loads.

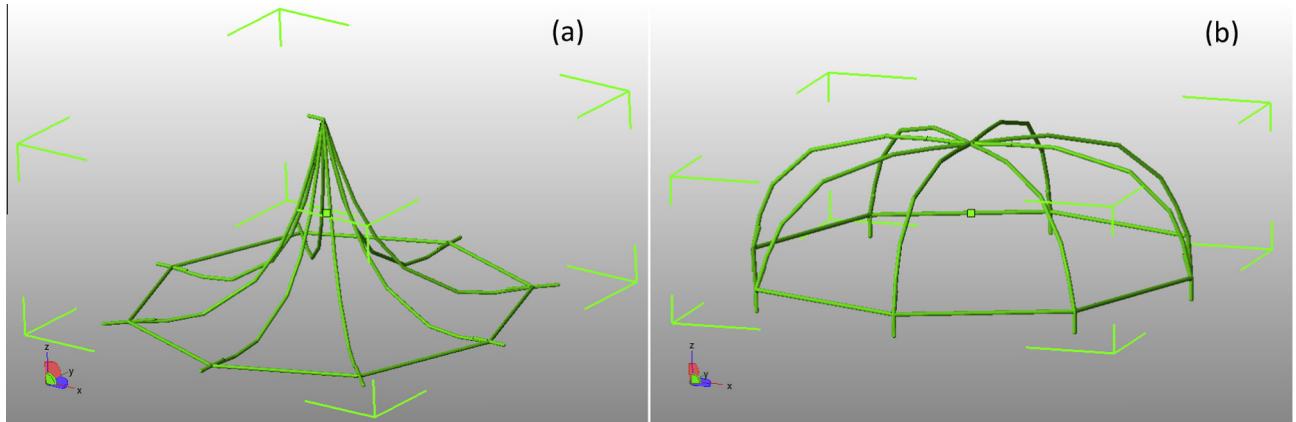


Fig. 40. Effect of structure dimensions and load direction on candidate load paths generated for spherical domes.

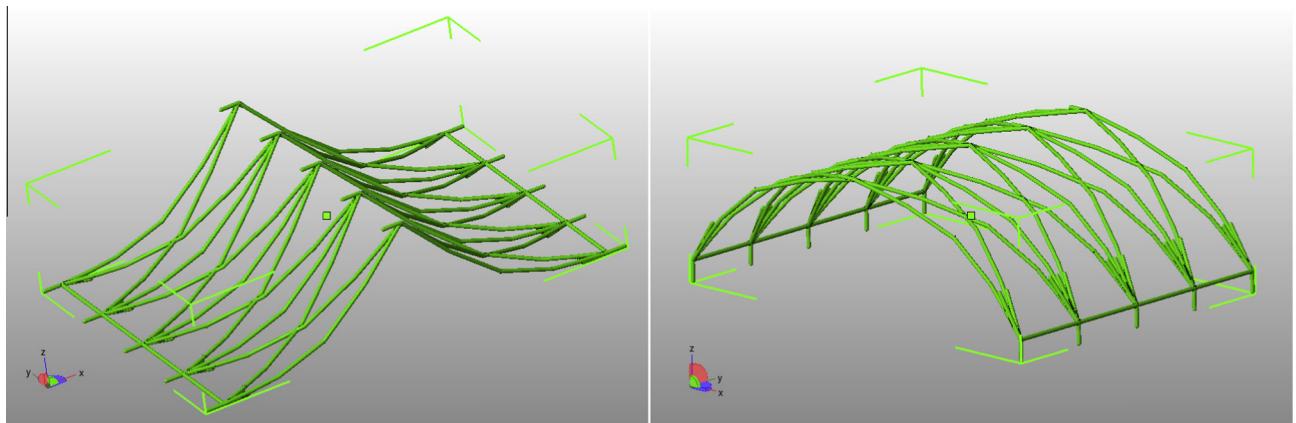


Fig. 41. Effect of load direction on the candidate load path generated for cylindrical domes.

literature. Design of cantilever beams of different lengths loaded at different positions and Michell type structures are the most common benchmark problems in topology optimization literature [1,18,53–55]. It is important to note that the trusses in this paper are defined for fixed compression and tension members. The method is unable to define freeform structures like those produced by relative density methods (e.g. SIMP [1]) or level set methods [53,55,58]. Nonetheless, Figs. 14–26 show a variety of topologies generated for short to long cantilevered trusses that mimic the known optimal freeform shapes. Figs. 27 and 28 show different topologies which are simplifications to those reported in the literature using different approaches such as level set methods [1,2,15,18,53,54,56–61]. Results for the test problem in Figs. 33 and 34 are also consistent with the results of the Hajirasouliha [62].

While these comparisons for benchmark problems are based only on visual inspection, it is difficult to know what the true optimal would be for a prescribed set of compression and tensile cross-sections. The method takes advantage of this initial constraint to limit the search first to the set of valid topologies and then the refined positioning of those topologies. This results in a much lower computational cost than classical ground structure methods.

5. Problems with multiple loads

This section demonstrates the suitability of the present approach for generating efficient topological designs for 2D/3D large structures subject to multiple loads. Remarkably, increasing

problem size or number of loads is not a challenge for the rules as those rely on the load flow principle which is always valid. Since increasing problem complexity and number of loads may lead to miss the global optimum shape and considering that the aim of this study was to formalize the load flow principle into less than 30 general rules usable for any kind of structure and level of structural complexity, the shape optimization phase will not be discussed in this section.

The test problem schematized in Fig. 35a is similar to the pilot problem of Fig. 3 except for the fact that the two fixed points now coincide. Fig. 35a shows the main load flow paths and the corresponding optimized path directions. Fig. 35b shows the optimized load paths while Fig. 35c shows a candidate topology. This very intuitive problem is a key to understand more complex problems such as that schematized in Fig. 36a, where there are eight loads symmetrically circulating around a ground point. Fig. 36b shows the main load flow paths from concentrated forces to fixed points. If only one load is left, this problem becomes similar to that schematized in Fig. 35a.

Fig. 37a and b illustrate two candidate load paths obtained for a different level of segmentation: indeed, Fig. 37b is derived from Fig. 37a by dividing load path segments into smaller segments. Considering individual loads only, the paths in Fig. 35b are exactly similar to paths in Fig. 37b. Fig. 37c and d shows two candidate designs created from the load path solution of Fig. 37a. The design process of these complex configurations is as simple as for the problem of Fig. 35a. The same approach can be used to design 3D dome structures.

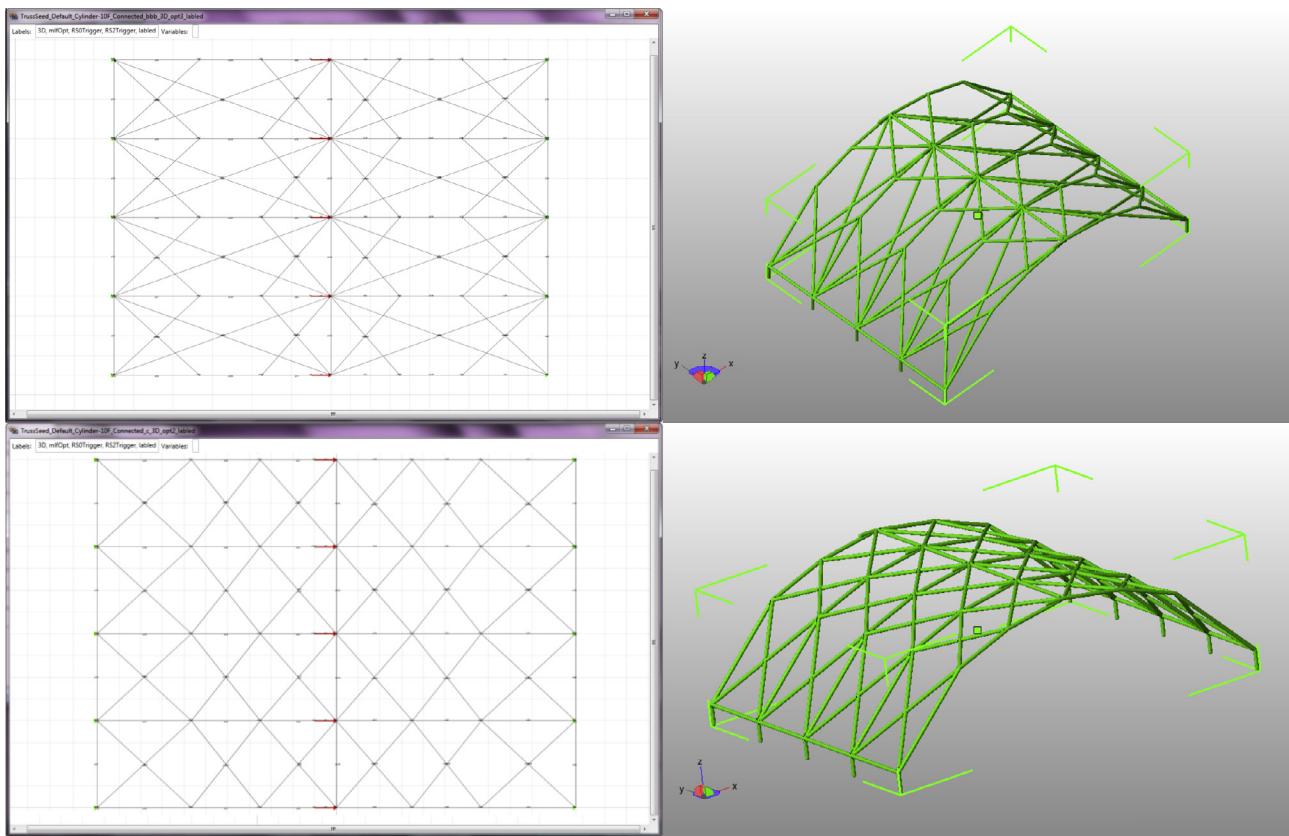


Fig. 42. Candidate designs obtained for cylindrical domes by considering the load paths schematized above.

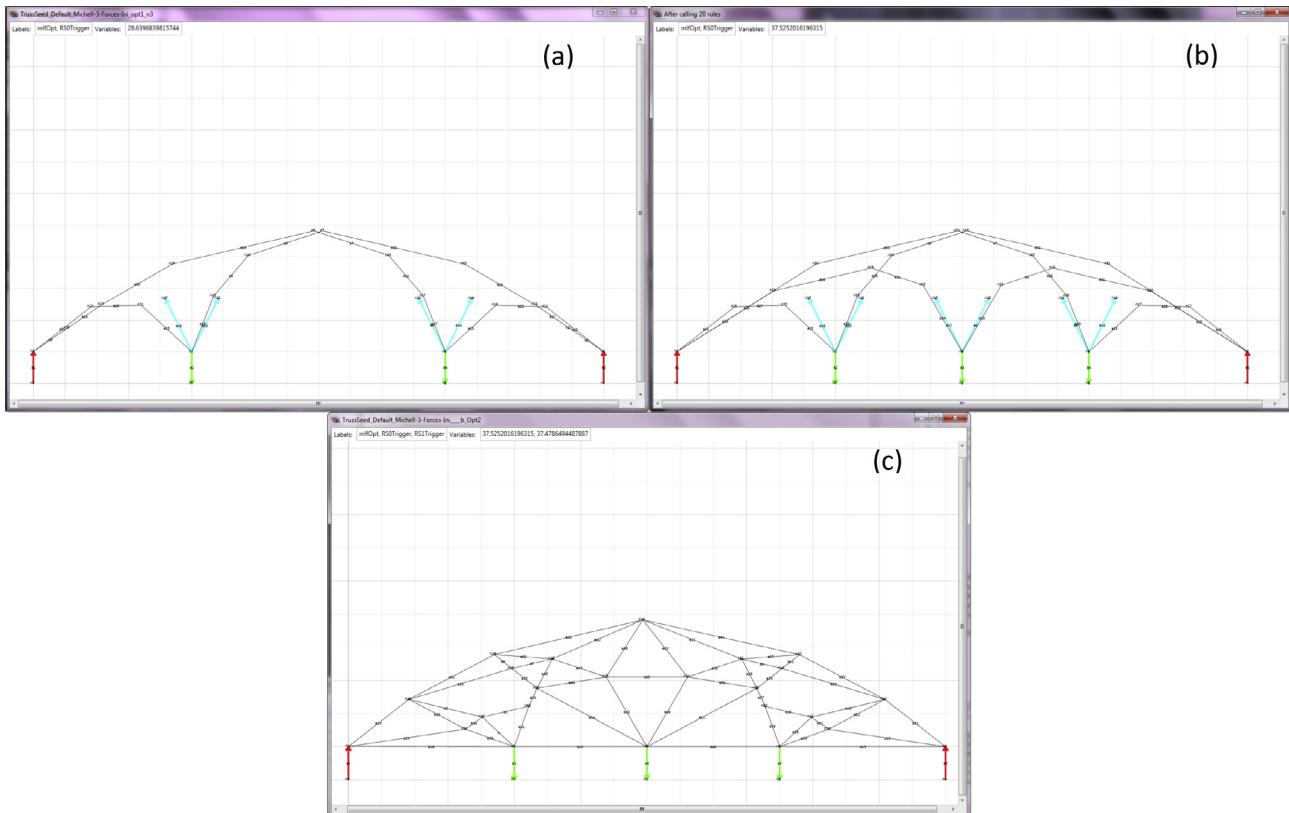


Fig. 43. Candidate load paths generated for bridges subject to (a) two or (b) three concentrated loads; (c) candidate design generated for the three-point loaded bridge.

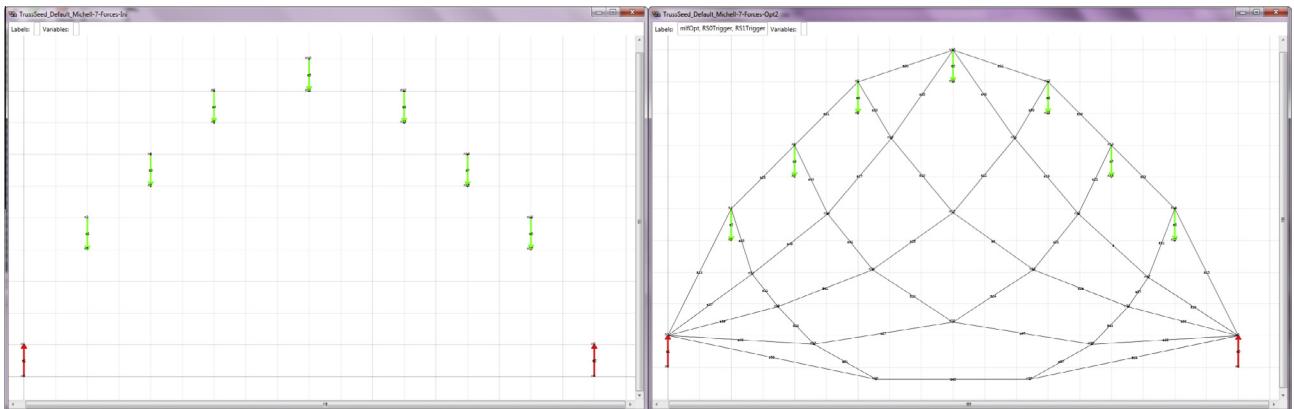


Fig. 44. Seed graph and candidate design for a structure subject to seven concentrated loads.

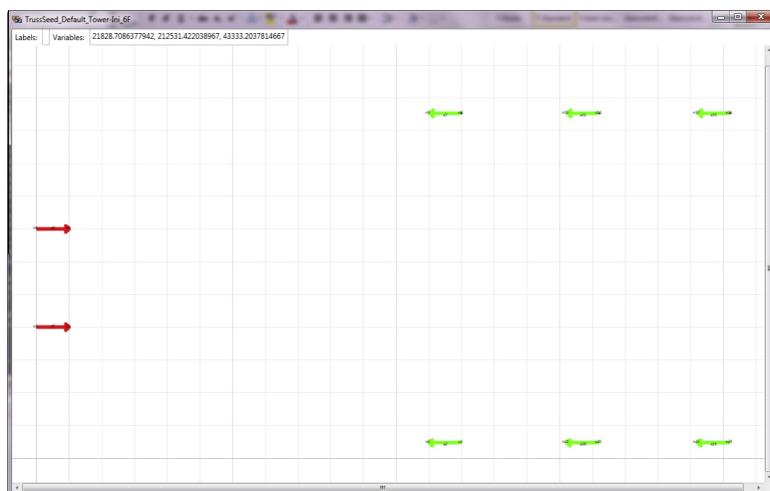


Fig. 45. Seed graph for a power tower subject to six concentrated loads.

5.1. Case example 1: Dome-like structures

Domes are spherical or semi-spherical architectural designs where the load is distributed over the whole surface. The whole weight can be either at the center or at the outer verge of the dome. The latter assumption is made in this study. Fig. 38a shows a simplified seed graph representation of a spherical dome subject to eight loads at its outer verge and including a central ground point; loads are perpendicular to the page. Fig. 38b shows a 3D view of the segmented load paths created in ruleset 1. Fig. 38c and d shows two candidate topologies generated using only the first six rulesets but not optimizing shape. Rules of ruleset 4 were manually selected. In order to obtain finer structural configurations, it is necessary to reduce the minimum segment size and increase the number of loads.

Fig. 39 shows a candidate design generated for a dome subject to inclined loads at the outer border. The sensitivity of load paths to load directions and the dome height clearly appears from Fig. 40a and b.

Fig. 41 shows two candidate load paths generated for the same cylindrical dome (with eight concentrated loads and four ground points) by changing load directions. Fig. 42 presents two candidate designs generated for cylindrical domes. Again, finer structural schematizations can be obtained by increasing the number of loads and reducing the minimum segment size.

5.2. Case example 2: Bridge-like structures

Bridges are similar to a Michell structure (see Section 4.4) but are usually subject to multiple loads. For example, Fig. 43 shows the load paths generated for two bridges of similar size but subject to a different number of loads. Distributing the total load over a different number of points leads to change in the structural configuration. Fig. 43c presents a candidate solution generated for the three-point loaded bridge. Rules in ruleset 4 again were manually selected to preserve structural symmetry of candidate designs. However, automatic rule selection can be implemented so as to detect and enforce symmetry.

Fig. 44 shows the seed graph and the corresponding candidate design generated for a structure subject to seven concentrated loads. Since angles made between the direction of applied loads and the directions defined by the loading points and the fixed points are much smaller than in the case of bridges, this problem was much simpler to solve.

5.3. Case example 3: Tower-like structures

Fig. 45 shows the seed graph of a power tower subject to six concentrated loads. In order to improve the visualization of results, the distance between loaded points and ground is considered much smaller than in reality. For clarification purposes, let us con-

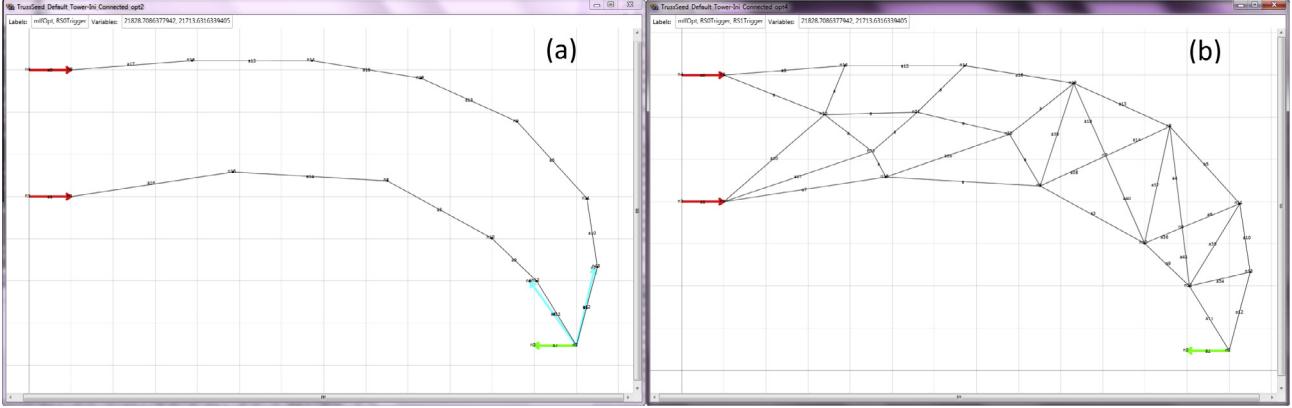


Fig. 46. Candidate load path (a) and design solution (b) generated for a single-load truss tower.

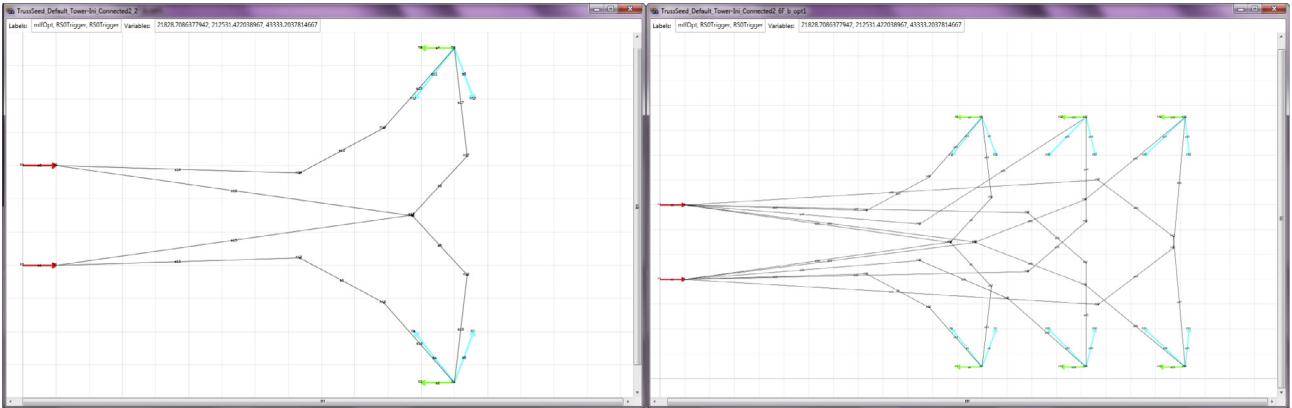


Fig. 47. Candidate load paths generated for power truss towers subject to two or six concentrated loads.

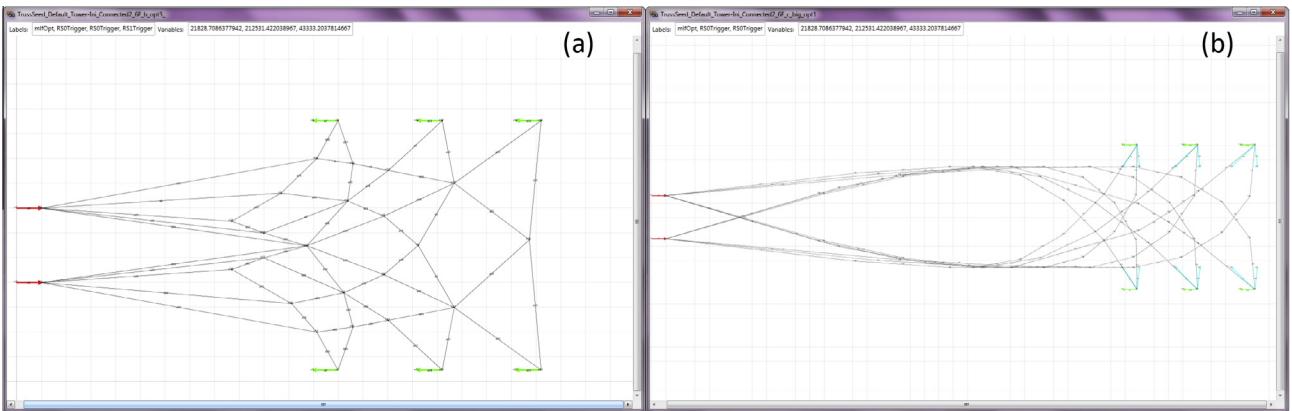


Fig. 48. Candidate designs generated for six-point loaded power truss towers: (a) Structural configuration with "short legs"; (b) structural configuration with "long legs".

sider just one of the six loads first. Fig. 46a shows the load paths while Fig. 46b shows a candidate topology generated for the single-load tower.

Fig. 47 presents the load paths obtained for two power towers loaded by two and six concentrated forces, respectively. The candidate design shown in Fig. 48a for the six-point loaded tower was obtained without using ruleset 4, because the intersection of principal load paths often created meaningful design patterns. However, ruleset 4 is necessary for designing tower

legs. Fig. 48b presents the load paths obtained for a power tower with longer legs. It can be seen that load paths merge in their way over long distances. Therefore, a new rule should be implemented to merge multiple adjacent load paths into a single main load path. This will be very similar to rule 22 that merges adjacent arcs. A new rule should also be implemented to merge long skinny triangles such as those found in the leg section of Fig. 48a. This will turn useful in designing very long or very wide structures.

6. Conclusions

A new approach to shape, sizing and topology optimization of cable truss structures using a generative design method has been presented in the article. The proposed methodology utilizes graphs to represent both topology and shape of structure. This allows a very fast generation of topological solutions for a given design problem. After exhaustive search of the design space, the solutions are stored in a list in the database for further detailed shape design and optimization. In order to visualize the solutions, the graphs are converted to 3D shapes via Parasolid. These shapes can be used to extract the construction drawings. Since the simulation model is fully separated, it should be fairly easy to include seismic loads, uncertainty in materials and construction with any number of loads and supports. Dividing the synthesis process in the shape and topology phases does not affect the quality of the results, because the tree-search algorithm serves to explore all candidate topologies in the design space while the optimization algorithm finds the optimum shape and size of each candidate design. However, the current limitation of the approach is in optimizing the shape of large structures with hundreds of nodes. New optimization algorithms are required to solve this issue and prevent trapping in local minima.

It should be noted that comparisons with optimum designs reported in literature for benchmark problems were only qualitative. However, this is not a major limitation because the present study focused on generating valid topologies without need of new structural analyses; hence at much lower computational cost than classical ground structure methods. Furthermore, the practicality of the generative design synthesis approach was demonstrated by the fact that real cables and bars available in the market are included in the design process.

Future investigations will try to account for the effect of the components weight; this will improve the shape and sizing optimization phases. Since rules are flexible enough and independent of the simulation model, the proposed methodology may be used, for example, to design compliant mechanisms as their performance is strictly related with the ability to transfer loads from one or more points to other points. A better understanding of the load flow principle will facilitate the creation of highly general and generic rules that can be used for any problem that has a load component. This investigation may finally lead to the development of a more formal definition for the principle.

Another important field of research that can increase the generality of the proposed approach is to handle obstacles in the problem specifications. The reason for this investigation is that obstacles are an undeniable part of the real world design problems. It may be interesting to use the output results of the proposed methodology as input for conventional continuous topology optimization methods. Due to good initial designs, convergence can be faster and many problems might be solved that are hitherto not solvable.

Acknowledgements

With the support of the Technische Universität München – Institute for Advanced Study, funded by the German Excellence Initiative.

References

- [1] Bendsoe MP, Sigmund O. Topology optimization. Berlin, Heidelberg: Springer Berlin Heidelberg; 2003.
- [2] Eschenauer HA, Olhoff N. Topology optimization of continuum structures: a review. *Appl Mech Rev* 2001;54:331–90.
- [3] Dorn WS, Gomory RE, Greenberg HJ. Automatic design of optimal structures. *J Mec* 1964;3:25–52.
- [4] Kirsch U. Optimal topologies of truss structures. *Comput Methods Appl Mech Eng* 1989;72:15–28.
- [5] Diaz AR, Bendsøe MP. Shape optimization of structures for multiple loading conditions using a homogenization method. *Struct Optim* 1992;4:17–22.
- [6] Achztiger W, Bendsøe MP, Ben-Tal A, Zowe J. Equivalent displacement based formulations for maximum strength truss topology design. *IMPACT Comput Sci Eng* 1992;4:315–45.
- [7] Bendsøe MP, Ben-Tal A, Zowe J. Optimization methods for truss geometry and topology design. *Struct Optim* 1994;7:141–59.
- [8] Kirsch U. On the relationship between optimum structural topologies and geometries. *Struct Optim* 1990;2:39–45.
- [9] Jalalpour M, Igusa T, Guest JK. Optimal design of trusses with geometric imperfections: accounting for global instability. *Int J Solids Struct* 2011;48:3011–9.
- [10] Lógo J. New Type of Optimality Criteria Method in Case of Probabilistic Loading Conditions #. *Mech Based Des Struct Mach* 2007;35:147–62.
- [11] Lógo J, Ghaemi M, Rad MM. Optimal topologies in case of probabilistic loading: the influence of load correlation. *Mech Based Des Struct Mach* 2009;37:327–48.
- [12] Yonekura K, Kanno Y. Global optimization of robust truss topology via mixed integer semidefinite programming. *Optim Eng* 2010;11:355–79.
- [13] Sandgren E, Cameron TM. Robust design optimization of structures through consideration of variation. *Comput Struct* 2002;80:1605–13.
- [14] Calafiori GC, Dabbene F. Optimization under uncertainty with applications to design of truss structures. *Struct Multidisc Optim* 2008;35:189–200.
- [15] Guest JK, Igusa T. Structural optimization under uncertain loads and nodal locations. *Comput Methods Appl Mech Eng* 2008;198:116–24.
- [16] Asadpoor A, Toottaboni M, Guest JK. Robust topology optimization of structures with uncertainties in stiffness – application to truss structures. *Comput Struct* 2011;89:1131–41.
- [17] Tyas A, Gilbert M, Pritchard T. Practical plastic layout optimization of trusses incorporating stability considerations. *Comput Struct* 2006;84:115–26.
- [18] Liang Q. Performance-based optimization of structures: theory and applications. London and New York: Taylor & Francis Ltd; 2005.
- [19] Cohn MZ, Dinovitzer AS. Application of structural optimization. *J Struct Eng* 1994;120:617–50.
- [20] Liang Q. Performance-based optimization method for structural topology and shape design. Australia: Victoria University of Technology; 2001.
- [21] Cagan J. Engineering shape grammars: where we have been and where we are going. In: Antonsson EK, Cagan J, editors. *Form Eng Des Synth*. New York: Cambridge University Press; 2001. p. 65–92.
- [22] Alber R, Rudolph S. On a grammar-based design language that supports automated design generation and creativity. In: Borg J, Farrugia P, Camilleri K, editors. *Knowl Intensive Des Technol*. Springer; 2004. p. 19–35.
- [23] Chase SC. A model for user interaction in grammar-based design systems. *Automat Constr* 2002;11:161–72.
- [24] Mullins S, Rinderle JR. Grammatical approaches to engineering design, part I: An introduction and commentary. *Res Eng Des* 1991;2:121–35.
- [25] Reddy G, Cagan J. An improved shape annealing algorithm for truss topology generation. *J Mech Des* 1995;117:315–21.
- [26] Shea K, Cagan J. Topology design of truss structures by shape annealing. In: Proc DETC98 1998 ASME des eng tech conf 1998; DETC98/DAC. p. 1–11.
- [27] Shea K, Cagan J. The design of novel roof trusses with shape annealing: assessing the ability of a computational method in aiding structural designers with varying design intent. *Des Stud* 1999;20:3–23.
- [28] Shea K. Essays of discrete structures: purposeful design of grammatical structures by directed stochastic search. Carnegie Mellon University; 1997.
- [29] Shea K, Cagan J, Fenves SJ. A shape annealing approach to optimal truss design with dynamic grouping of members. *ASME J Mech Des* 1997;119:388–94.
- [30] Shea K, Aish R, Gourtovaia M. Towards integrated performance-driven generative design tools. *Automat Constr* 2005;14:253–64.
- [31] Kaveh A. Topological transformations applied to structural mechanics. *Comput Struct* 1997;63:709–18.
- [32] Kaveh A. Graphs and structures. *Comput Struct* 1991;40:893–901.
- [33] Kaveh A. Structural mechanics: graph and matrix methods. 3rd ed. Exeter, U.K.: Research Studies Press (John Wiley); 2004.
- [34] Kurtoglu T, Swantner A, Campbell MI. Automating the conceptual design process: "From black box to component selection". *Artif Intell Eng Des Anal Manuf* 2010;24:49–62.
- [35] Hooshmand A, Campbell MI. Layout synthesis of fluid channels using generative graph grammars. *AI EDAM* 2014;28.
- [36] Hooshmand A, Campbell MI. Topology optimization of fluid channels using generative graph grammars. In: 39th Des automat conf. ASME; 2013.
- [37] Hooshmand A, Campbell MI, Shea K. Steps in transforming shapes generated with generative design into simulation models. In: 38th Des autom conf parts A B, vol 3. ASME; 2012. p. 883–92.
- [38] Kelly D, Elsley M. A procedure for determining load paths in elastic continua. *Eng Comput* 1995;12:415–24.
- [39] Kermode AC. Aeroplane structure. 2nd ed. Pitman Publishing; 1964.
- [40] Osgood CC. Fatigue design. Wiley-Interscience; 1970.
- [41] Hart-Smith LJ. An engineer's viewpoint on design and analysis of aircraft structural joints. *Arch Proc Inst Mech Eng Part G J Aerosp Eng* 1995;209:105–29. 1989–1996 (vols 203–210).
- [42] Skakoon JG. The elements of mechanical design. Three Park Avenue New York, NY 10016-5990: ASME; 2008.

- [43] Harasaki H, Arora JS. Topology design based on transferred and potential transferred forces. *Struct Multidiscip Optim* 2002;23:372–81.
- [44] Harasaki H, Arora JS. New concepts of transferred and potential transferred forces in structures. *Comput Methods Appl Mech Eng* 2001;191:385–406.
- [45] Hoshino H, Sakurab T, Takahashic K. Vibration reduction in the cabins of heavy-duty trucks using the theory of load transfer paths. *JSAE Rev* 2003;24:165–71.
- [46] Marhadi K, Venkataraman S. Comparison of quantitative and qualitative information provided by different structural load path definitions. *Int J Simul Multidiscip Des Optim* 2009;3:384–400.
- [47] Armenàkas A. Classical structural analysis: a modern approach. New York, USA: McGraw-Hill Press; 1988.
- [48] Heckel R, Küster J, Taentzer G. Confluence of typed attributed graph transformation systems. *Graph Transform* 2002;2505:161–76.
- [49] Fletcher R. Function minimization by conjugate gradients. *Comput J* 1964;7:149–54.
- [50] Luh G-C, Lin C-Y. Optimal design of truss-structures using particle swarm optimization. *Comput Struct* 2011;89:2221–32.
- [51] Noilublao N, Bureerat S. Simultaneous topology, shape and sizing optimisation of a three-dimensional slender truss tower using multiobjective evolutionary algorithms. *Comput Struct* 2011;89:2531–8.
- [52] Siemens PLM Software Inc., Parasolid; 2013.
- [53] Yulin M, Xiaoming W. A level set method for structural topology optimization and its applications. *Adv Eng Softw* 2004;35:415–41.
- [54] Bulman S, Sienz J, Hinton E. Comparisons between algorithms for structural topology optimization using a series of benchmark studies. *Comput Struct* 2001;79:1203–18.
- [55] Luo Z, Tong L, Kang Z. A level set method for structural shape and topology optimization using radial basis functions. *Comput Struct* 2009;87:425–34.
- [56] Rong JH, Liang QQ. A level set method for topology optimization of continuum structures with bounded design domains. *Comput Methods Appl Mech Eng* 2008;197:1447–65.
- [57] Wang MY, Wang X. “Color” level sets: a multi-phase method for structural topology optimization with multiple materials. *Comput Methods Appl Mech Eng* 2004;193:469–96.
- [58] Wang SY, Lim KM, Khoo BC, Wang MY. An extended level set method for shape and topology optimization. *J Comput Phys* 2007;221:395–421.
- [59] Wang SY, Tai K. Bar-system representation for topology optimization using genetic algorithms. *Eng Comput* 2005;22:206–31.
- [60] Veenendaal D. Evolutionary optimization of fabric formed structural elements [Master thesis]. Faculty of Civil Engineering and Geosciences, TU Delft University; 2008.
- [61] Hassani B, Hinton E. A review of homogenization and topology optimization III—topology optimization using optimality criteria. *Comput Struct* 1998;69:7.
- [62] Hajirasouliha I, Pilakoutas K, Moghaddam H. Topology optimization for the seismic design of truss-like structures. *Comput Struct* 2011;89:702–11.