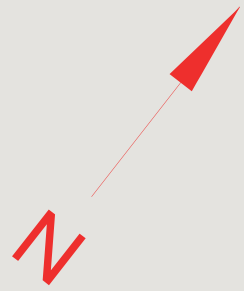


DETAIL B



6

AAG2018

Advances in Architectural Geometry 2018

Lars Hesselgren, Axel Kilian, Samar Malek,
Karl-Gunnar Olsson, Olga Sorkine-Hornung
Chris Williams
Editors

5

CHALMERS



Computational design of robotically assembled spatial structures

**A sequence based method for the generation
and evaluation of structures fabricated with
cooperating robots**

Stefana Parascho, Thomas Kohlhammer, Stelian Coros,
Fabio Gramazio, Matthias Kohler

Stelian Coros
stelian.coros@inf.ethz.ch
Computational Robotics Lab, ETH Zurich, Switzerland

Stefana Parascho
parascho@arch.ethz.ch

Thomas Kohlhammer
kohlhammer@arch.ethz.ch

Fabio Gramazio
gramazio@arch.ethz.ch
Gramazio Kohler Research, ETH Zurich, Switzerland

Matthias Kohler
kohler@arch.ethz.ch
ETH Zurich, Switzerland

Keywords:

Spatial structures, sequence based design, cooperative robotic
assembly, multi-robotic fabrication, computational design,
reciprocal structures, design through optimisation, fabrication
aware design, digital fabrication, assembly sequence

Abstract

Cooperative robotic fabrication enables the development of new types of spatial structures, provided that assembly sequence and robot path-planning is considered in the design process early on. This paper presents a design strategy for a lightweight steel structure assembled by two robots. The developed structure describes a novel typology of spatial structures and consists of steel tubes that form spatial configurations through their three-dimensional aggregation. The bars are joined notch-free through welding and without additional connecting elements. Besides fabrication-driven constraints, the design process is informed by functional, geometric and structural parameters. The paper presents the development of a novel connection system and the resulting dependencies for the geometric and structural system, as well as a four-step computational design method that allows to explore a large area of the design space of such structures. Optimisation methods are employed to solve the complex dependencies of the presented structures and find a valid design.



Figure 1: Multi-robotic assembly of spatial structures.

1. Problem statement

The introduction of robotic manufacturing methods in architecture and construction has augmented the range of the design possibilities that are currently available. Particularly processes that require assembly profit from the robot's capacity to precisely hold, move and position an element in three-dimensional space. Through the use of industrial robotic arms for the placement of discrete elements, it has become possible to build bespoke structures with elements of non-standard dimensions, which can be freely placed in numerically defined positions and orientations (Helm et al., 2017). This greatly increases the design space of spatial structures, allowing more geometric freedom than manually assembled structures. However, the robotic assembly procedure also introduces new constraints, such as robot reachability and sequencing. In addition, the higher geometric complexity requires advanced computational methods in order to handle the large number of dependencies during the design process.

Addressing these new possibilities, this paper presents a design method for a new typology of spatial metal structures consisting of steel bars (round hollow profiles) that are assembled by two robots and an implementation of a corresponding computational design tool in Python. The bars have individual lengths and are welded manually after being robotically positioned. The assembly method relies on the use of two robotic arms, which alternately place elements in space (Parascho et al., 2017) such that while one robot places a new element, the other one serves as support for the already built structure (Mirjan, 2016) ([Fig. 1](#)). This results in a fabrication method that does not require additional support structures or scaffolds. Furthermore, the alternating placing of elements prevents the accumulation of tolerances as both the supporting robot and the one placing an element serve as a reference.

The design process is based on this sequential fabrication procedure in order to ensure successful fabrication. Related research projects where robotic fabrication directly informs the design process applied constraints in particle spring models (Parascho et. al., 2015) or predefined the potential design scope through constraining the assembly logic to, for example, layer-based systems (Apolinarska, 2016). However, these strategies do not prioritise the fabrication sequence and describe it as either a pre-defined order or a post-rationalisation step. Defining the assembly sequence directly in the design process leads to a reinter-

pretation of fabrication as a main driver for the design. This enables the design process to explore buildable geometries while generating them and not constrain the solution space artificially beforehand to predict a feasible design space.

In addition to fabrication, the design requires to consider other factors, such as geometric rules, structural behaviour and functionality. Strategies to simultaneously address multiple of these parameters in the design process are difficult to identify due to the large number of parameters and their different nature (discrete, continuous, binary). One possible method implies the use of optimisation to improve material efficiency, robot reachability or stability. However, problems of discrete nature require different optimisation methods than continuous ones. As a result, this research proposes a combination of methods to negotiate between the individual design problems.

2. Design procedure

The design procedure follows four steps which individually address one or more of the different design constraints (Fig. 2). The following presents an overview of these steps which are described in detail in the subsequent sections. The design is generated by adding bars one by one, defining the order which will later reflect the fabrication sequence. Each bar must fulfil following requirements: a) its position must allow to connect to two existing bars of the already defined structure, b) its position must guarantee stability during assembly and in the structure's final state and c) the robot has to reach its placement pose without collisions.

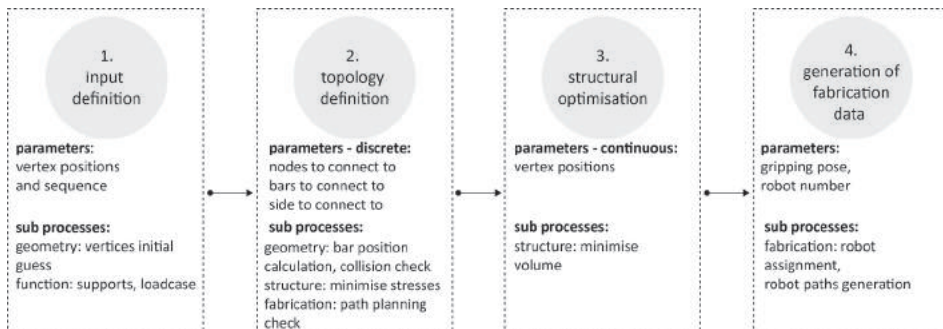


Figure 2: Design workflow describing involved sub-processes and variables to be defined or changed in each step.

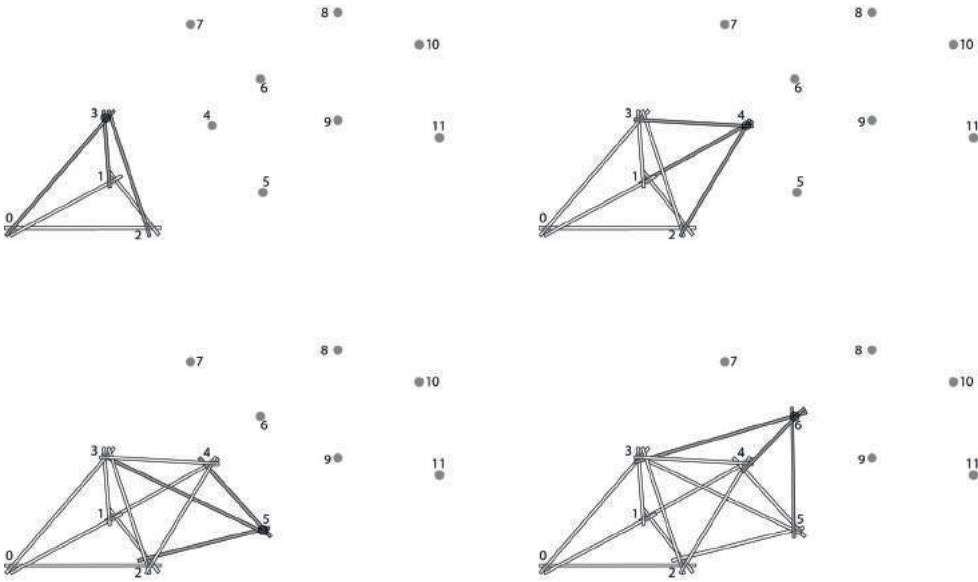


Figure 3: Design workflow, step 2: topology definition. A new vertex is chosen and connected to the existing structure via three bars. The input vertices are pre-defined and connections are chosen based on structural considerations.

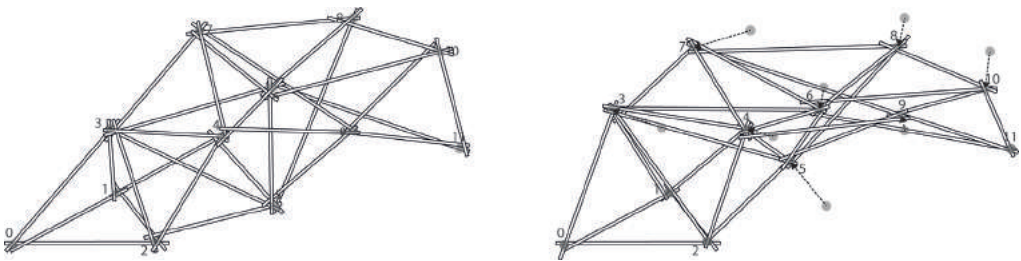


Figure 4: Design workflow, step 3: structural optimisation. Locations for vertices are refined to minimise the material usage for a given load case.

In short, the generation process can be summarised as follows: In step 1 a set of points (which will be referred to as vertices) is defined that describes a goal-geometry by being distributed in a given boundary geometry. This input set includes the points' sequence, pre-defined support points and a given load case (position and magnitude of one or more point load vectors). In step 2 the topology is established: for each consecutive vertex, three bars are created that connect it to the already

defined structure, forming a stable configuration (Fig. 3). Due to the chosen connection logic, each bar has to touch at least two existing bars (see Section 3). The two bars to connect to are chosen such that the stresses in the structure are minimised (see Section 6.1). This process, run through all input vertices, defines the connectivity between elements of the structure, which remains constant in further steps. In step 3 the structure is optimised for structural behaviour by refining the positions/coordinates of the input vertices via an optimisation process (Fig. 4) (see Section 6.2). Finally, in step 4 fabrication data, including final poses for bars and robotic paths, is generated. The design and analysis tools are implemented in python using the COMPAS library (Van Mele et al., 2017) and are thus CAD independent. Visualisation of the results is done in Rhinoceros 3D (McNeel, 2015).

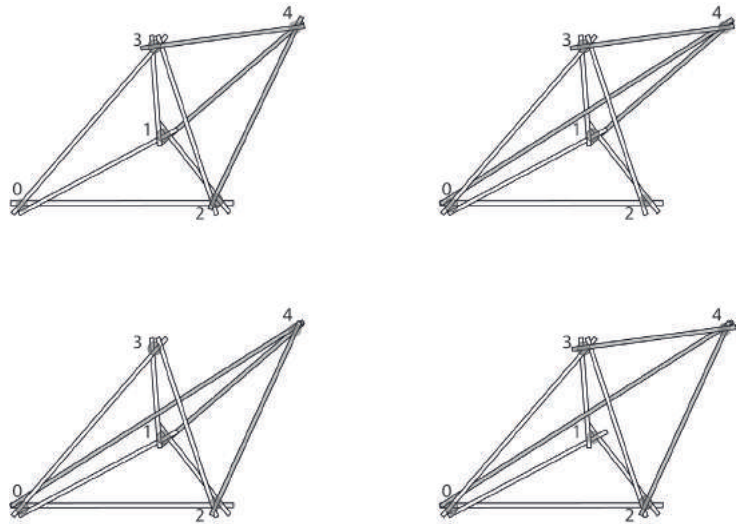


Figure 5: Example of options for three connections from a new vertex to the existing structure.

3. Geometric system

The chosen design strategy is based on the sequential definition of the bars' positions in space. In addition to fabrication feasibility, this ensures that geometric dependencies that require knowledge of previously placed bars are fulfilled. The final structure consists of groups of three bars

which form stable sub-structures and lead to a structurally determinate system. These groups will be referred to as three-bar-groups. Since the system is not restrained to a regular geometry, multiple options of connections are possible for a vertex (**Fig. 5**). The choice of connections is performed in the topology definition step which is described in Section 6.1.

3.1 Node configuration

Fabrication efficiency and structural performance of spatial structures are strongly influenced by the chosen connection system. Standardised systems use identical connection elements and same-lengths bars, for example the Mero system (Chilton, 2000), which leads to a simple fabrication process but limits the design to regular space frames. For differentiated space frame structures individual elements can be produced but require precise prefabrication and lead to an increased logistic effort in their assembly.

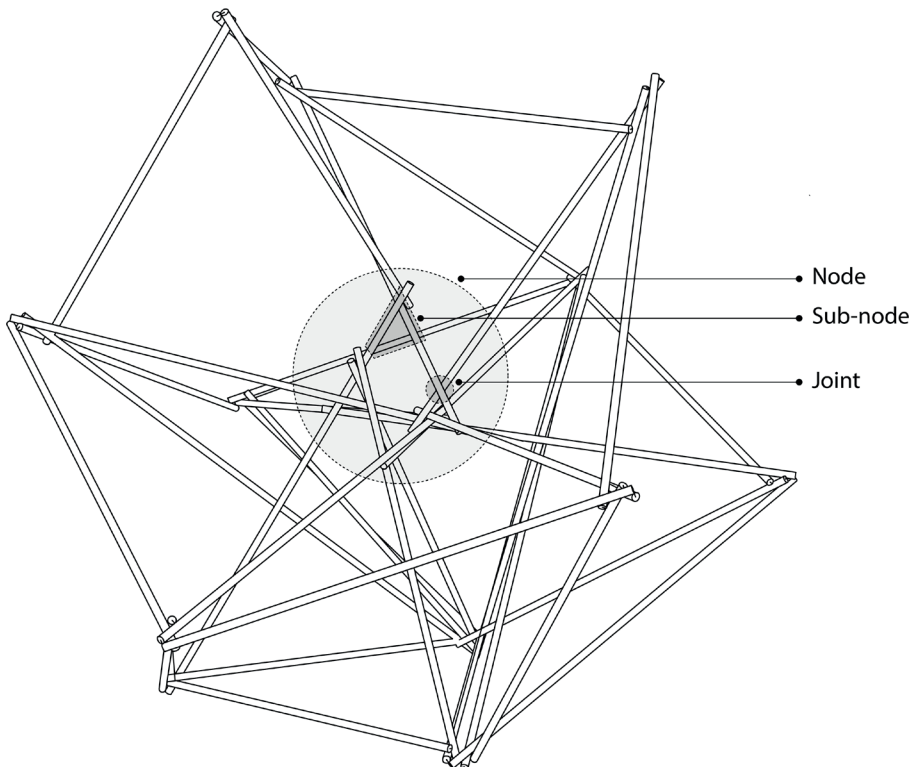


Figure 6: Definition of node, sub-node and joint. A node develops out of every vertex once bars are generated around it.

The presented research addresses these limitations of prevailing connection systems through the proposal of a novel node for spatial structures, which can potentially be fully integrated in the robotic fabrication process and does not rely on additional prefabricated elements. In the context of this paper the term *node* has been defined to include all connections that topologically come together in one vertex point. A *sub-node* represents all connections forming a reciprocal configuration between three or more bars in a node, while a *joint* refers to a single connection between two bars (Fig. 6).

In the proposed geometric system, a node is composed of a cluster of joints, which connect no more than two bars at a point. However, this node configuration reduces the stiffness of the overall structure through introducing bending moments in the bars. To counteract this effect, the stiffness of the node is increased through connecting each bar additionally in a second point to another existing bar, leading to closed reciprocal sub-nodes (Fig. 7).

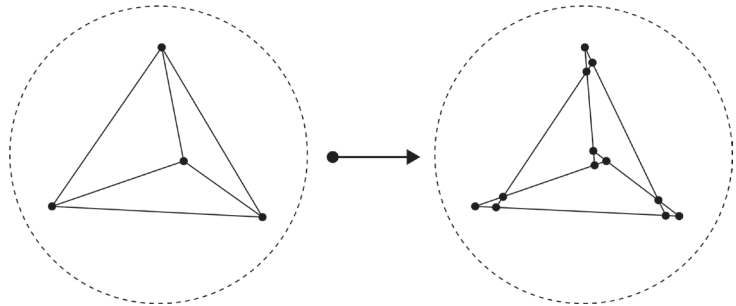


Figure 7: Geometric development of node in an aggregation of 6 bars.

In order to generate the reciprocal sub-nodes in the design definition, one needs to find the possible solution space for newly added bars that fulfil the geometric constraints of a node. Visualising all possible angles of attachment for a tangent line to two given bars allows to identify areas where no solution exists (Fig. 8). This leads to discontinuities in the descriptive function which need to be taken into account in the input and topology definition process (steps 1 and 2) as well as in the optimisation process (step 3) (see Sections 6.2, 6.3). For steps 1 and 2, if vertices are located in areas where no solution exists, a correction process is performed which moves the vertex to the closest feasible point of the solution space.

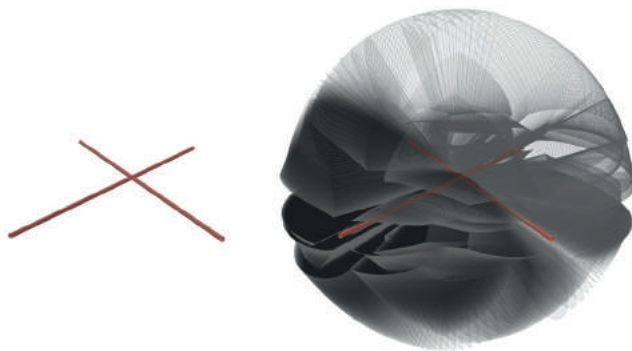


Figure 8: Visualisation of the solution space for a bar tangent to two existing bars. The volume shows all possible angles of attachment for a new bar and two fixed existing ones.

The very vast, but still locally constrained solution space additionally shows the necessity of implementing computational methods in order to be able to explore the entire geometric design space of the developed system.

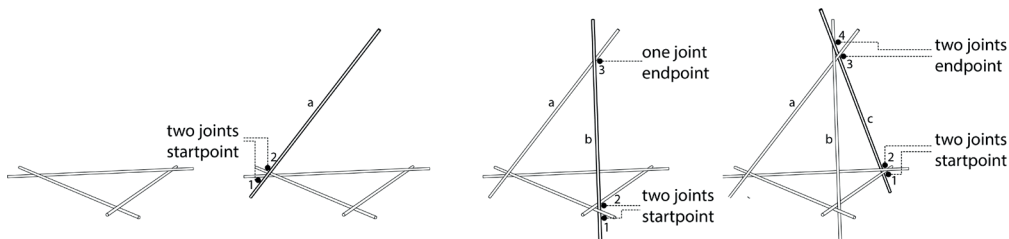


Figure 9: Sequential adding of three bars onto an existing structure leading to two joints, three joints and four joints.

3.2 Calculation of bar positions

The described node configuration leads to various geometric dependencies within the system. A bar needs to be tangent to two, three or four existing bars accordingly, depending on whether the considered bar is the first, second or third to be added among the three new bars of a vertex (**Fig. 9**). For any bar connecting to two existing bars, four solutions can be found depending on which side it attaches to (**Fig. 10** right). These options are used to either react to collisions or if a robotic path cannot be found (see Sections 4 and 6). The position fulfilling the geometric constraints of two, three or four tangent connections is found by the calculations shown in the following three cases.

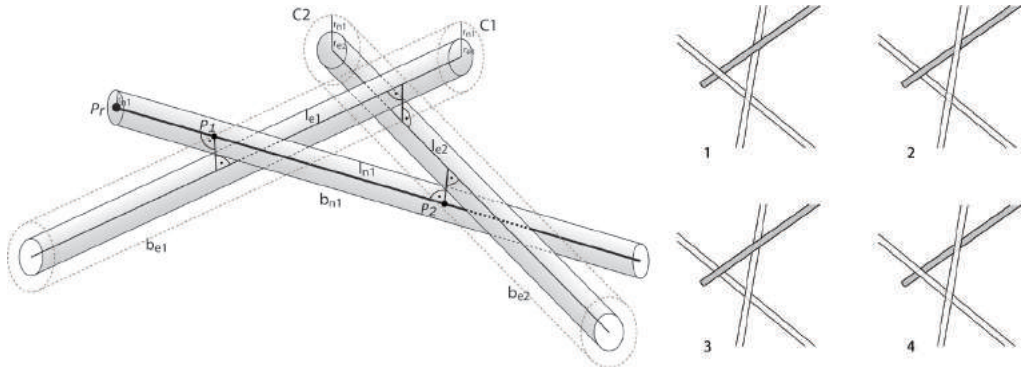


Figure 10: Dependencies between three tangent bars (left) and four possible solutions for one bar tangent to two other bars (right).

Case 1: For the first bar of a three-bar-group b_{n1} its centreline l_{n1} has to be found, such that it touches the two other bars b_{e1} and b_{e2} passing through a given vertex point P_r (Fig. 10, left). This can be described through calculating a line which is tangent to two cylinders C_1 and C_2 defined by the axes of the existing bars l_{e1} and l_{e2} and a radius equal to the sum of the existing bars' radius r_{e1} or r_{e2} and the radius of the bar to be added r_{n1} . This problem is mathematically determinate and can be solved as follows (Fig. 11, left): the line l_{n1} is calculated at the intersection of the planes p_1 and p_2 that pass through the given point P_r and are tangent to the two cylinders defined by the given bars axes and the determined radii.

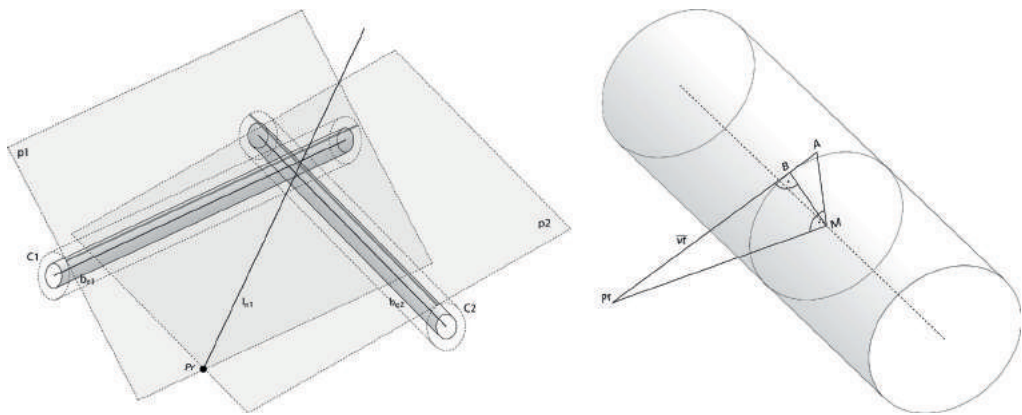


Figure 11: Case 1. Calculation of the first bar tangent to two existing bars b_1 , b_2 (left). Calculation of a vector v_1 tangent to a cylinder through a given point P_r (right).

The planes p_1 and p_2 are calculated through Pythagoras relations in the following triangles (Fig. 11, right): AMP_r resulting from the vertex point P_r , M (the projection of P_r onto the bar's axis (l_{e1} or l_{e2})) and A (the point resulting from the intersection of the perpendicular through M to P_rM and the tangent to the cylinder P_rA) and AMB described by the Points A , M and B (the intersection between the radius MB perpendicular to the line P_rA). The sought tangent plane is defined by the point P_r , the vector v_i and the vector of the bar's axis (l_{e1} or l_{e2}).

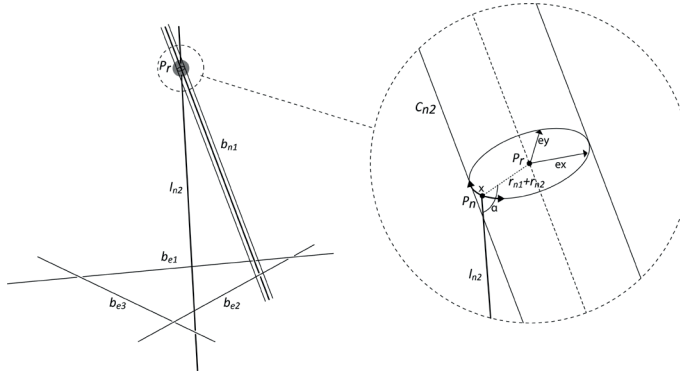


Figure 12: Case 2. Calculation of the second bar tangent to two bars at the base and the already defined first bar at the top.

Case 2: The axis of the second bar b_{n2} can be found by searching for a point on the circumference of a cylinder C_{n2} (Fig. 12) defined by the axis of the element b_{n1} and the radius $r = r_{n1} + r_{n2}$ such that the resulting line l_{n2} is tangent to C_{n2} . P_r describes the input vertex point through which the axis of the first bar b_{n1} passes and t the parameter between 0 and 1 on a circle perpendicular to the axis of the first bar and with a radius r equal to the sum of the radii of the existing first bar and the bar to be placed. From each resulting P_n on the circumference of C_{n2} a line l_{n2} tangent to the two given bars b_{e1} and b_{e2} can be calculated. A search method is needed to find such point P_n that this line is tangent to the first bar, i.e. the angle $\alpha = 90^\circ$. This is achieved by minimising the function $f(x) = |90 - \alpha|$. P_n is expressed through x and its position is calculated in relation to P_r through the coordinate system described by e_x and e_y .

Case 3: The axis of the third bar b_{n3} can be found by searching through points on a plane p_m (Fig. 13) which has been defined perpendicular to the vector connecting the base vertex position in a point M and

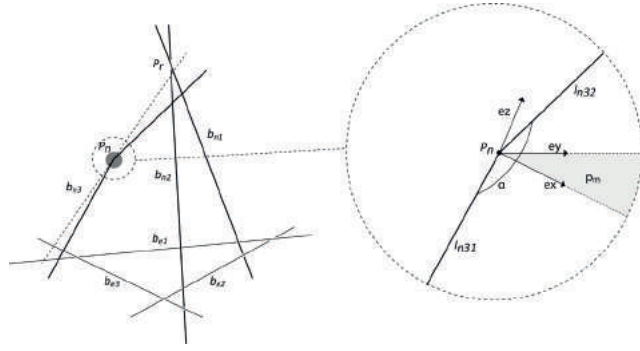


Figure 13: Case 3. Calculation of the third bar tangent to two bars at the base and the already defined first and second bars at the top.

the new vertex position. The vectors e_x and e_y define a new two-dimensional coordinate system with origin in M where e_x and e_y describe the orientation of the plane p_m and e_1 and e_2 the coordinates of a point P_n in this new coordinate system. Through the resulting point P_n two tangent lines l_{n31} and l_{n32} can be calculated, one to the two bars b_{n1} , b_{n2} and one to the already calculated first two bars of the group (b_{n1} , b_{n2}). The goal of the search is to find the point P_n on the plane p_m in which the two calculated tangents are collinear, i.e. the angle $\alpha = 180^\circ$. This is found through minimising a function $f(e_1, e_2) = 180 - \alpha$.

Both searches for case 2 and case 3 have been implemented using SciPy's optimisation library (Jones et al., 2001) and its minimisation function `fminbound()` which relies on Brent's method for finding a local minimum of a scalar function (Brent, 1973).

The developed connection system has been tested for feasibility in a physical prototype, where a structure consisting of thirty-three bars, including a central node with fourteen elements was designed and assembled with two robotic arms of the Robotic Fabrication Laboratory (RFL) at ETH Zurich (Fig. 14). In theory a node can be infinitely expanded to incorporate more bars, as long as physical collisions between the bars are avoided. In practice, the maximum number of bars in a node is strongly dependent on the attachment angles of the bars and the chosen connection bars and sides. The prototype additionally served for identifying fabrication challenges such as tolerances resulting from the robotic set-up. These ranged up to ± 3 mm and were dealt with by slightly forcing the elements until they are tangent to their neighbours. It was also shown that tolerances do not add up over time, as the robot positioning a new bar serves as a reference for the structure at every step.



Figure 14: Physical test of a structure in which 14 bars come together in one node.

4. Fabrication feasibility evaluation

The described geometric system is a direct result of the robotic fabrication procedure, allowing two robots to cooperatively assemble spatial structures while ensuring stability and simple connection of the elements. However, the cooperative assembly method strongly depends on the sequence of placing elements, which determines the buildability of the structure. Beside assembly sequence, reachability and trajectory planning need to be taken into account during the design process.

The chosen strategy is to evaluate the buildability at two steps: first, during the topology definition process (step 2) and second, after the structural optimisation process (step 4). The main goal of these evaluations is to identify whether bar positions are reachable by the robotic arms, and whether collision-free robot trajectories can be found to place each bar in the given sequence. For this purpose, a path planning method relying on random sampling algorithms is integrated into the computational set-up and used to search for feasible paths (Gandia et al., 2018). The path planning algorithm requires a starting configuration and a final pose to be reached as an input and results in a list of joint values describing the collision-free movement of the robot. Three parameters can influence the success of this procedure and need to be defined during the design process: 1. the robot assigned to place the bar, 2. the gripping position and orientation on the bar and 3. the final pose to be reached by the robot. The assignment of the robot placing the bar is performed within a three-bar-group such that it ensures the stability of the structure throughout the placing process. It is based on the logic that the first and third bar of a three-bar group can be placed by any of the two robots while the second bar needs to be placed by the other robot than the one that placed the first bar. An initial assignment is performed by approximating which robot has better reachability, but is changed if no path can be found for the placement of the bar. In order to find collision-free paths, different gripper positions and orientations can be tested until a feasible one is found. The path planning process does not calculate a trajectory to the final position of a bar, but to a translational and rotational offset pose that guarantees that a linear robot movement towards the final position does not encounter collisions. This is done to induce more flexibility into the path planning process since this pose can be adjusted if no path is found. In both fabrication evaluations the following process is performed: For each bar, a path is searched for and

if a collision-free one cannot be found parameters are changed from local ones to global design ones (Fig. 15).

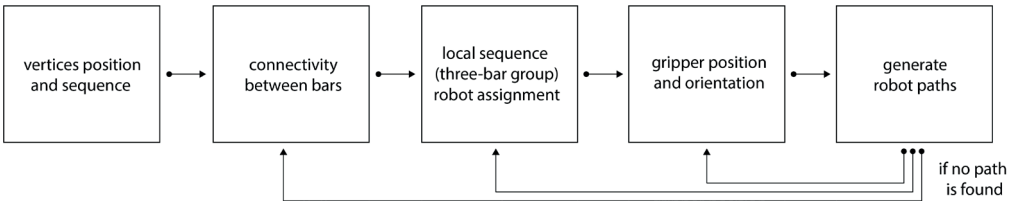


Figure 15: Workflow for the generation of robot paths. If no path is found for a bar parameters are changed from the right to the left.

Due to the low speed of the path planning process, requiring 30 to 60 seconds per bar, its use has not been fully automated in the computational geometry generation process. However, path planning checks are performed after the topology definition (step 2), in order to early identify situations where no path can be found, and in a final step before fabrication (step 4) in order to generate fabrication data.

5. Integrated structural analysis

The investigated geometric system shows high complexity in load bearing behaviour. On the one hand, this means the interplay of geometric parameters and structural performance is not obvious, thus strategies for geometric changes to improve the structural behaviour are difficult to define. On the other hand, complexity means the system is statically sensitive to changes in geometric configuration, hence slight geometric modifications of the structure may have a very large impact on its load bearing performance. A major reason for these behaviours is the reciprocity of the nodes, as for example shown by investigations of reciprocal frame structures in (Kohlhammer 2014) and (Kohlhammer et al., 2017). Due to this complexity, structural optimisation of the discussed system is a highly non-trivial problem and can only be solved through iterative tools. These require a fast structural analysis to evaluate a large number of parametric system states.

In consequence of this, the computational design environment includes algorithmic methods of structural analysis, which enable immediate feedback about the static performance of the system during the design and optimisation process. To establish a direct and seamless integration, the structural analysis is implemented in the same environment as the geometric design. Figure 16 shows the workflow of the developed structural analysis, which is divided in the three following steps: modelling, calculation and evaluation.

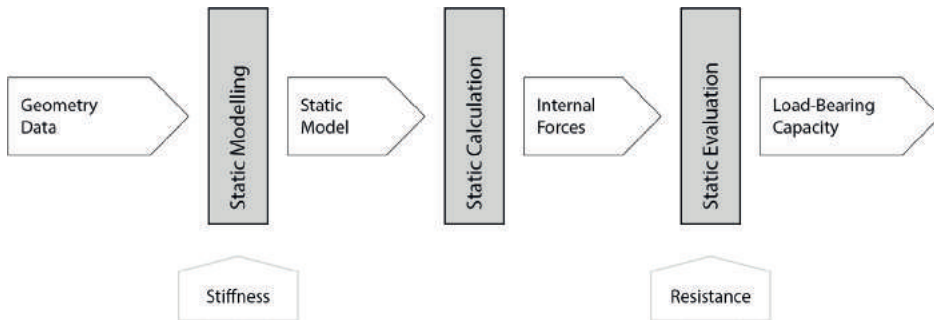


Figure 16: Workflow diagram of the integrated structural analysis. The structural analysis is used in step 2 and 3 of the design process.

5.1 Modelling

As a basis for the structural analysis, an appropriate static model with linear elements is generated. It is an abstraction of the real volumetric geometry and includes two types of elements: 1. bars which represent the steel rods and 2. connectors which represent the welded connections of two rods (Fig. 17). For each single element translational and rotational stiffness values have to be defined in order to emulate the real structural behaviour of the system. While for bars these values are defined by respective cross-section geometries and material properties, for connectors a specific mechanical model was assumed based on positions and geometries of the weld points which connect two steel rods. This model was verified by physical test series. The tests were performed on single nodes with two or four welding points and five force directions (compression, tension, shear and two rotations). In addition, the tests showed very little deviations of the values throughout a test series, meaning that the welds display a similar behaviour even if they are executed manually.

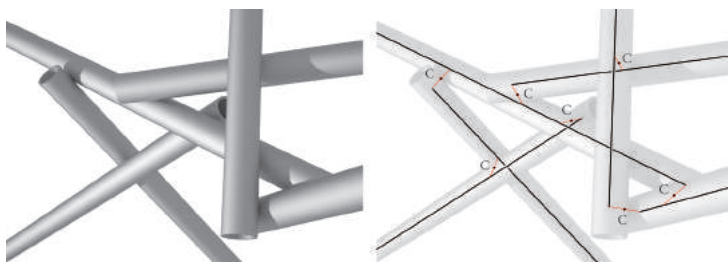


Figure 17: Geometry of part of the system (left) and corresponding linear static model (right).

5.2 Calculation

As a result of assumed load scenarios the inner forces and deformations of bars and connectors are calculated. Therefore, a direct interface to an FEM engine was established. In this case the finite element base engine (feb) of the static analysis software Karamba (Preisinger et al., 2015) was used, which is a fast and programmable finite element core.

5.3 Evaluation

In the final step of the analysis the calculated inner forces and deformations are evaluated for every bar and connector of the static model. The evaluation is based on the criterion of utilization u . In this research, u is defined as the ratio of an inner force to its corresponding maximum value which is here represented by the yield point. For bars this evaluation corresponds to the Swiss steel codes SIA 263. For connectors the calculated inner forces are transformed into a resulting force-vector F as well as a resulting moment-vector M . Both refer to the contact point C of two steel rods which is equal to the midpoint of the shortest distance line between the two axes of connecting bars (Fig. 17). Each component of $F (F_x, F_y, F_z)$ and $M (M_x, M_y, M_z)$ has a resistance, represented by the maximum possible value of the component. The connector-resistances result from the same mechanical model as the stiffness values and were also verified by test. As in general all inner-force-components exist simultaneously, resistance boundaries for force-interactions have to be assumed. Resistance values and interaction boundaries define a specific resistance graph (Fig. 18) for each connector. If this graph is displayed together with the existing force-vector, the utilization u of a connector can be visualized through the length of the vector in relation to its maximum possible length within the resistance graph. In addition to stresses

in the bars and connections, utilisations in regards to deformation and stability are calculated. However, these results have not been integrated in the design procedure, but will be included in the next iteration of the design tool.

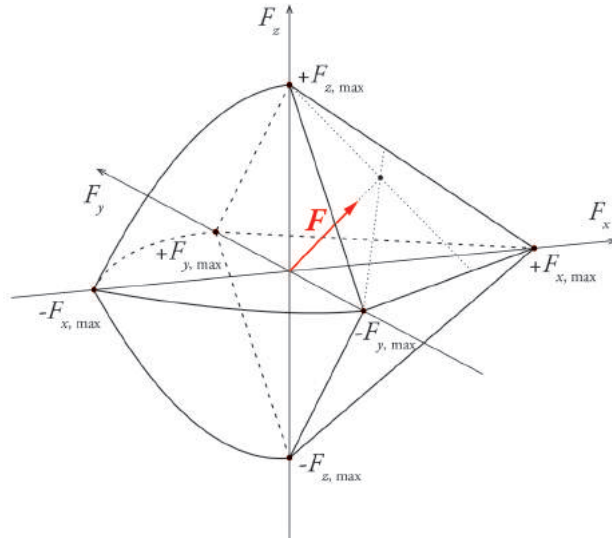


Figure 18: Example of a resistance diagram of a connector with existing force-vector and maximum graph.

6. Generation and optimisation

The design problem with its multiple constraints and parameters requires both continuous and discrete considerations. Discrete parameters are: the nodes to connect to from a new vertex, the bars to connect to in the nodes and the sides of the connections, while the continuous parameter describes the location of the vertices in space. To address this multitude of parameters, an algorithm was developed that treats topology generation and optimisation in a two-stage process (Fig. 2, steps 2 and 3). The number of possible combinations of discrete parameters increases drastically with the structure's overall number of vertices. For example, there are over 200 000 000 possible topologies for a structure with only 9 vertices. Treating the topology generation separately thus allows to decrease the dimension of the search space.

6.1 Topology definition

The goal of the topology definition process (Fig. 2, step 2) is to identify for every new vertex the three topological connections that induce the least amount of stresses on the bars. The main reason for this is that the subsequent structural optimisation process requires an initial guess which fulfils the given constraints, in this case, that bar stress utilisation values are not higher than 1.0. For this purpose, for every vertex in the structure's sequence, potential connection vertices to the already generated nodes are identified within a given distance and combinations of three such nodes are generated. These combinations are evaluated by calculating the three new bars' positions, generating their geometries, defining an approximated load case for the current structure's state, analysing the resulting structure's structural behaviour and evaluating the option through the total value of stress utilisations in the bars. The load case is defined by moving the final load case's force vector to the current vertex position and adding a moment vector that represents an approximation of the bending moment that the structure would experience at the vertex point in the final state. This moment vector is calculated as a vector connecting the current vertex and the final location of the force. For this step, only the connectivity between the nodes has been considered, while other discrete parameters are used to ensure the geometric integrity of the design. This is done through a collision check that is performed throughout the topology definition for every newly generated bar to identify intersections with the existing geometry. If collisions are found, the connection side and, if necessary, the bars to connect to are changed until a feasible solution is found.

A brute force approach was chosen to iterate through all connectivity options. The objective was defined as finding the option with the smallest total stress utilisations value in the bar elements and variable values have been limited to a list of potential node indices. As a result, the problem was formulated as follows:

$$\text{minimise} \quad f(x) = \sum u_b \quad u_b = \text{stress utilisation of bar}$$

where x describes a potential combination of 3 nodes to connect to. However, this topology definition process serves only as an approximation of an efficient structure, since its evaluation does not rely on the final positions of the vertices, which will be refined in the next step, and

structural analysis performed on a partial structure does not precisely represent its final behaviour, but merely an approximation.

6.2 Structural optimisation

To further improve the structural performance, the positions of bars are refined by allowing the input vertices to change position (Fig. 2, step 3). The topology established in step 2 remains unaltered. Vertices that serve as support points or desired fixed points are described as fixed vertices while all other vertices are defined as variables for an optimisation process. This allows to control how constrained a design is, depending on the input. Through the defined topology, instances of the design are recalculated and analysed using the developed FEM interface. As opposed to the topology definition problem, the optimisation problem in this case can be expressed as a continuous problem which allows for the use of gradient-based optimisation methods (Kraft, 1988). The python optimisation library pyOpt (Perez et al., 2012) with its Sequential Least Squares Programming solver (Kraft, 1988) is used for this problem. Since the purpose of the optimisation is to improve structural efficiency, decreasing material use was chosen as a goal. The objective function is thus formulated to minimise the total lengths of bars, while constraining the stress utilisations of bars to a limit value of 1.0, and thus prevent failure:

$$\text{minimise:} \quad f(x) = \sum l_b \quad l_b = \text{length of bar}$$

$$\text{constrained to:} \quad u_{b1,b2...bn} < 1 \quad u_b = \text{stress utilisation of bar}$$

6.3. Results

The proposed design process and optimisation were validated through modelling tests in which small structures were generated and optimised and compared to brute force approach results. The test models all have three supports on one side and a point load on the other end of the structure, representing a cantilevering structure. This describes an essential test case for spatial structures as it needs to withstand bending moments and thus requires structural height. The bars have a diameter of 25 mm and a thickness of 2 mm while their lengths vary between 800 mm and 1 800 mm.

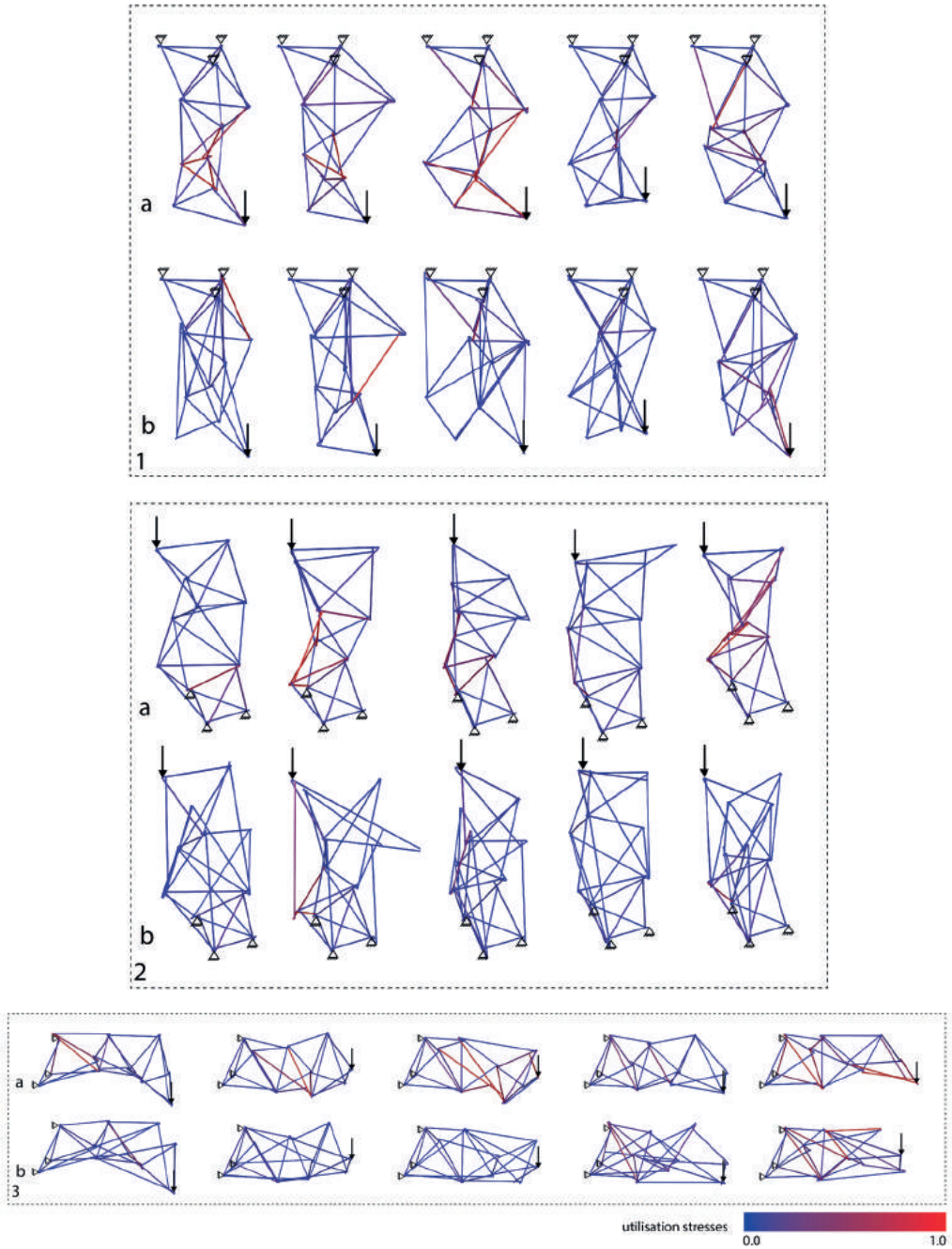


Figure 20: Test models for topology definition evaluation. 1. Tension loaded structures, 2. Compression loaded structures, 3. Bending loaded structures, a) structures with connections to closest nodes, b) structures resulting after optimisation.

The topology definition process (step 2) has been tested on models with 10 vertices and 27 bars. To verify the success rate of the proposed approach, a series of tests has been modelled and calculated and results between pre-defined topologies, always connecting to the closest three nodes (Fig. 20, a), and the calculated topologies (Fig. 20, b) were compared. Three different load cases were tested, by changing the orientation of the force vector, generating structures primarily loaded under tension (Fig. 20, 1), compression (Fig. 20, 2) and bending (Fig. 20, 3). For each load case, 5 different structures were modelled by modifying the input vertices' positions within a distance of 500 mm from their initial locations. This resulted in fifteen test structures of which fourteen showed improvements in the total added bar utilisation values of 11% (5.71 to 5.07) to 83% (17.35 to 2.89). Six structures started with a solution which included bars with utilisations higher than 1.0 of which four resulted in structures with no bar utilisations higher than 1.0 while the other two reduced the number of overloaded bars from four, respectively three, to one (Fig. 20). However, the success of the topology definition process is strongly related to the initial distribution of points and the given load case, as these must ensure that a solution with bars with stress utilisations lower than 1.0 exists. If this cannot be fulfilled, additional vertices have to be added in step 1 and the design has to be recalculated.

As a test case, one structure was generated which was later also used in the structural optimisation tests. For this specific case, improvements of 19.7% in utilisations (Fig. 21) were achieved.

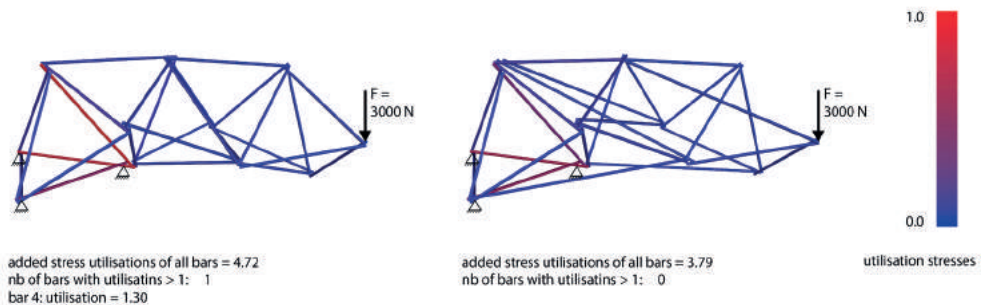


Figure 21: Example of topology generated by connecting to three closest vertices (left) and after the calculation process (right). Bars varying between 800 mm and 1800 mm in length, 25 mm of diameter and a thickness of 2 mm were used.

The structural optimisation tests were first performed on models with 4 vertices and 6 bars. Only one vertex was used as a variable to keep the model simple (Fig. 22). Since the geometric solution space defined by the connection system includes areas with no solutions, small discontinuities appear in the objective function (see Section 3.1). It is thus crucial to identify if the optimisation process is influenced by these discontinuities and, if so, to what extent. To do this, a second simplified geometric system was modelled, which does not include the reciprocal connection, but is built of bars connecting in one single point in a node. The behaviour of the optimisation processes for both models was compared and shows that solutions are consistently found for both systems. For validating both optimisation results, a brute-force process was implemented that iterates through 10 000 point locations and its results were compared to the optimisation results (Fig. 23). In both cases the optimised result shows a lower function value than the brute force approach. Additionally, in order to verify if the optimisation process reaches the function's minimum, a test was performed in which the optimisation process' resulting point position is used again as an initial guess for the same problem. Since the result changed only minimally (less than 10 mm) in 3 iterations, it is assumed that for this problem the optimisation reaches the minimum after the first iteration.

Finally, the method was applied to a larger structure consisting of 27 bars. The structure results from the topology definition test and serves as an initial guess for the structural optimisation process. Its first three vertices, which represent the supports, and its last vertex, where the point load is applied, are defined as fixed points, while the other vertices are set as variables. The method results in a 32 % decrease of material volume (Fig. 24) leading to a smaller material usage than the first uninformed guess (Fig. 21, left) while additionally ensuring that stresses in the bars do not exceed the material capacity.

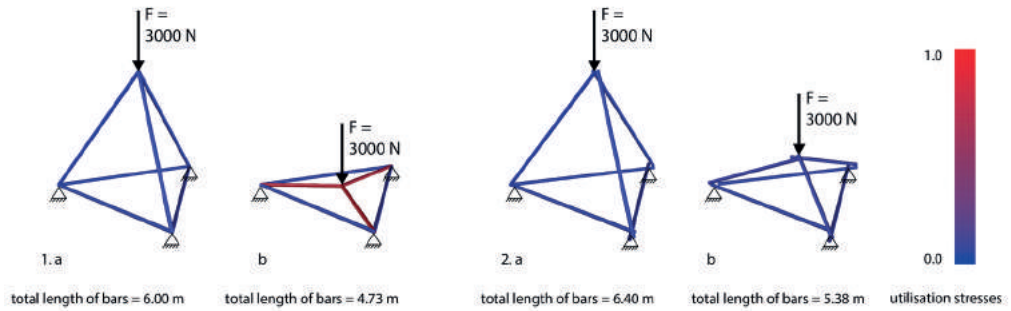


Figure 22: Optimisation results: 1. Simplified model: a) initial guess, b) optimisation result; 2. Model including connection: a) initial guess, b) optimisation result.

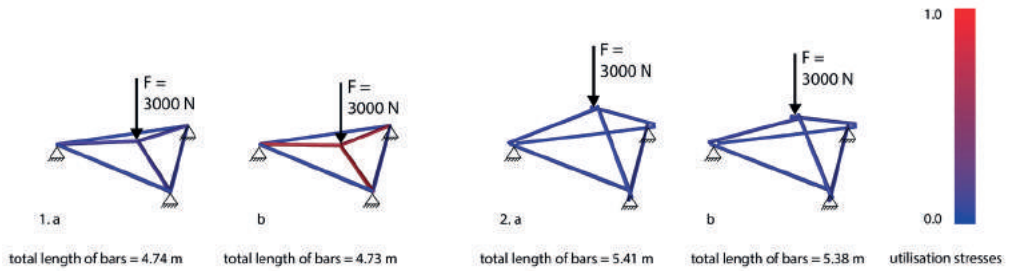


Figure 23: Optimisation results compared to brute force results: 1. Simplified model: a) brute force result, b) optimisation result; 2. Model including connection: a) brute force result, b) optimisation result.

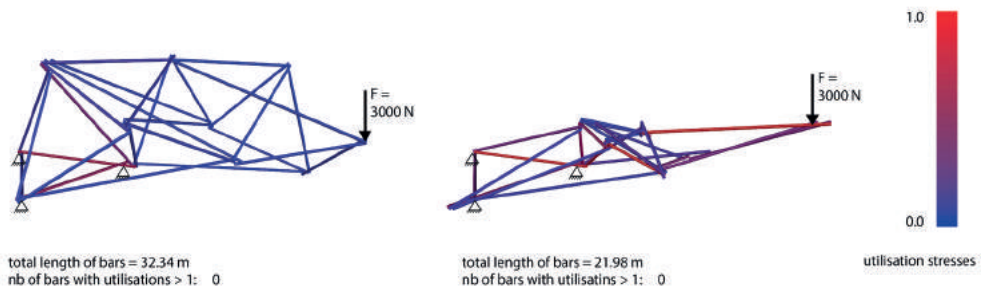


Figure 24: Example of structure before (left) and after (right) minimising material volume while constraining stress utilisations in bars.

7. Conclusion

The paper presents a method for multi-informed design that combines fabrication, geometry, structure and material considerations. Additionally, the development of a complex geometric system resulted in a novel construction system with a high degree of freedom regarding possible topologies and therefore high differentiation potential regarding number, positions and length of bars. Describing the geometric relations computationally enabled the exploration of the new geometric system and a larger design space than standard spatial structures offer. Moreover, the sequence-based strategy combined with an integral fabrication feasibility evaluation allowed to generate designs that are easily buildable, despite their geometric complexity.

The digital and physical tests have shown that the initial design space is strongly defined by the fabrication set-up and material pre-conditions. For example, in a stationary set-up the robotic reach is very constrained, whereas the RFL's 36 axis set-up opens up a lot more geometric possibilities due to the robots' flexibility of movement. Similarly, the structural behaviour depends strongly on the chosen bar dimensions, determining whether failure will occur in the bars or in the connections. In the presented paper one scenario was investigated, however expanding it to different material systems and fabrication set-ups would strongly impact the potential design space.

Using optimisation methods allows for an efficient search of a design space but comes with own constraints and limitations. The topology definition's structural analysis (step 2) may lead to redundant bars that in a final configuration are not needed to transfer forces. In addition, the topology definition process usually increases the material volume in order to find a functional solution, which is then minimised throughout the optimisation process. Even though in most cases the resulting final material volume is smaller than the initial one, finding different strategies for the topology definition process could help the optimisation reach a better solution. In addition, the continuous structural optimisation is strongly dependent on the number of variables, constraints and fixed vertices. If the input scenario is geometrically strongly constrained (e.g. through many fixed vertices), it does not lead to a high improvement of the structural behaviour, whereas if the freedom is too high, it requires a long time for calculation. As a result, identifying correlations between the input scenario (number of variable and fixed vertices) and the optimisa-

tion result could improve the efficiency of the optimisation process. Regarding step 1 of the design workflow, the presented strategy for input generation allowed fast testing of multiple options to evaluate different inputs. However, its further development towards not pre-defining input points but generating them during the topology definition process could allow a more informed input and potentially improve the starting scenario for the structural optimisation.

Several other topics can be extended in future research. The implementation of a faster path planning method would allow a higher level of integration and more direct control over fabrication feasibility. Further research could expand the possibilities of the developed design method for different structures and geometric configurations, such as non-triangulated geometries. Even though optimisation methods lie at the core of the described methodology, the presented research attempts to address not individual design problems but the negotiation of multiple constraints and goals through a combination of methods. Generalising this approach into a flexible computational set-up which could integrate several solvers and allow different levels of parameter integration and geometric design definition could set the base for multi-variable, multi-objective design environments.

References

APOLINARSKA, ALEKSANDRA ANNA, RALPH BÄRTSCHI, RETO FURRER, FABIO GRAMAZIO AND MATTHIAS KOHLER. 2016. "Mastering the Sequential Roof: Computational Methods for Integrating Design, Structural Analysis, and Robotic Fabrication." In *Advances in Architectural Geometry 2016*, Sigrid Adriaenssens, Fabio Gramazio, Matthias Kohler, Achim Menges, and Mark Pauly, 240–258. Zürich: Hochschulverlag an der ETH Zürich, 2016.

BRENT, R. P. 1973. "Chapter 4: An Algorithm with Guaranteed Convergence for Finding a Zero of a Function", In *Algorithms for Minimization without Derivatives*, Englewood Cliffs, NJ: Prentice-Hall, ISBN 0-13-022335-2.

CHILTON, J. 2000. *Space Grid Structures*, Oxford, Architectural Press.

GANDIA, AUGUSTO, STEFANA PARASCHO, ROMANA RUST, GONZALO CASAS, FABIO GRAMAZIO AND MATTHIAS KOHLER, 2018. "Automatic Path Planning for Robotically Assembled Spatial Structures", submitted to RobArch 2018.

HELM, VOLKER, MICHAEL KNAUSS, THOMAS KOHLHAMMER, FABIO GRAMAZIO AND MATTHIAS KOHLER. 2017. "Additive robotic fabrication of complex timber structures." In *Advancing Wood Architecture*. 29–43. Routledge.

JANSEN, PETER W., AND RUBEN E. PEREZ. 2011. "Constrained Structural Design Optimization via a Parallel Augmented Lagrangian Particle Swarm Optimization Approach." In *International Journal of Computer and Structures*, 89(13–14):1352–1366. Elsevier.

JONES, E, E OLIPHANT AND P PETERSON. 2001. "Scipy: Open Source Scientific Tools for Python." <http://www.scipy.org/> (accessed 2018-03-18).

KOHLHAMMER, THOMAS. 2014. *Strukturoptimierung von stabförmigen Flächentragwerken mittels Reziproker Analyse*. Zurich: Dissertation ETH Zurich.

KOHLHAMMER, THOMAS, APOLINARSKA, ALEKSANDRA A., GRAMAZIO, FABIO, AND MATTHIAS KOHLER. 2017. "Design and structural analysis of complex timber structures with glued T-joint connections for robotic assembly." *International Journal of Space Structures* 32:199–215.

KRAFT, D. 1988. "A software package for sequential quadratic programming." Tech. Rep. DFVLR-FB 88-28, DLR German Aerospace Center, Institute for Flight Mechanics, Köln, Germany.

MCNEEL, 2015, Rhinoceros, available at <https://www.rhino3d.com/> (accessed 2018-03-18).

MIRJAN, AMMAR. 2016. *Aerial Construction: Robotic Fabrication of Tensile Structures with Flying Machines*. Zurich: Dissertation, ETH Zürich.

PARASCHO, STEFANA, AUGUSTO GANDIA, AMMAR MIRJAN, FABIO GRAMAZIO AND MATTHIAS KOHLER. 2017. "Cooperative Fabrication of Spatial Metal Structures." In *Fabricate 2017*, edited by Achim Menges, Bob Sheil, Ruairi Glynn and Marilena Skavara. 24–29. UCL Press.

PARASCHO, STEFANA, JAN KNIPPERS, MORITZ DÖRSTELMANN, MARSHALL PRADO, AND ACHIM MENGES. 2015. "Modular Fibrous Morphologies: Computational Design, Simulation and Fabrication of Differentiated Fibre Composite Building Components" in *Advances in Architectural Geometry*, edited by Philippe Block, Jan Knippers, Niloy Mitra and W. Wang. 24–45. Switzerland: Springer.

PEREZ, RUBEN E., PETER W. JANSEN AND JOAQUIM R. R. A. MARTINS. 2012. "pyOpt: A Python-Based Object-Oriented Framework for Nonlinear Constrained Optimization." In *Structural and Multi-Disciplinary Optimization*, Volume 45, Issue 1. 101–118. Springer.

PREISINGER, CLEMENS, AND BOLLINGER-GROHMANN-SCHNEIDER ZT GMBH. 2015. "Karamba 3d". <http://www.karamba3d.com/> (accessed 2018-03-18).

VAN MELE, TOM, ANDREW LIEW, TOMAS MENDEZ ECHENAGUCIA, MATTHIAS RIPPMANN AND OTHERS. 2017. "Compas: A framework for computational research in architecture and structures." <https://compas-dev.github.io/> (accessed 2018-03-18).