Automated Design Workflow for Structural Nodes of Space Frame Structures

Conference Paper

Author(s):
Beutler, Patrick; Biedermann, Manuel; Hofmann, Urs; Rosenbauer, Ralph; Meboldt, Mirko

Publication date:
2022

Permanent link:
https://doi.org/10.3929/ethz-b-000554264

Rights / license:
Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International

Originally published in:
Procedia CIRP 109, https://doi.org/10.1016/j.procir.2022.05.272
Abstract

Large-scale structures are often built using space frames or trusses. These can become complex if the geometry of the space frame structure is customizable and if they do not contain any recurring components. Consequently, the design effort required, and the production cost of these unique components become large. Further, designing and creating these structures is challenging for novice designers as these processes require large design know-how. This challenge can be addressed with a knowledge-based engineering (KBE) approach. Design processes for specific, repeating tasks are analyzed and broken down into individual steps, which can be automated as a rule-based design procedure. This work introduces an automated design workflow that follows a KBE approach. The workflow automates the design process for irregular space frame structures. The workflow takes a mesh geometry of the space-frame structure as an input, and decomposes the mesh to nodes, bars, and panels. The workflow then generates and exports the fabrication data that are required for production: STL files for the three-dimensional node geometries that are additively manufactured with multi jet fusion, cutting lines for the planar panels that are produced with a laser-cutter, and length values that are used to cut straight bars. The work describes in detail the rules applied for the automation of the design process. This automated design workflow enables novice designers to create space frame structures with reduced design effort. The workflow is applied on a horse sculpture to demonstrate the manufacturability of the parts and the assembly of the space frame structure.

© 2022 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0)
Peer-review under responsibility of the scientific committee of the 32nd CIRP Design Conference

Keywords: Space Frame Structures, Knowledge Based Engineering, Additive Manufacturing, Automated Design
structures becomes challenging due to their non-repeating geometries. Additive manufacturing (AM) offers great potential in the production of complex components and enables cost-efficient manufacturing of small lot sizes [5-7]. AM thus offers the possibility of producing such individualized nodes. Wang et al. developed additively manufactured, structural nodes for an architectural application [8]. In their design, the material volume was minimized to further reduce the manufacturing costs. Raspall et al. designed a functional space frame structure with laser powder bed fusion (L-PBF) printed nodes connecting aluminum bars for architectural applications [4].

Even though AM offers great design freedom, the effort involved in designing unique components remains very high [9]. Knowledge-Based Engineering (KBE) is an approach to capture design know-how, transfer it to procedural design rules, and automatically design part geometries [10-12]. The use of KBE is particularly suitable for applications in which recurring design elements are to be generated automatically [13]. Further, it is necessary to identify repetitive design tasks and to formulate the design steps as procedural design rules [14].

Figure 2: Detail view of a node with panels and bars. (Photo: Andreas Marx)

The design effort for space frame structures can therefore be significantly reduced by automatically generating the geometries of their components. Within this work, the components are defined as shown in Figure 2: Nodes are produced with multi-jet fusion (MJF), panels are made of laser-cut cardboard and wooden rods are used for the bars. This work describes a digital workflow that generates all these components of a space frame structure automatically. The input of the workflow is a mesh geometry of the space frame to be manufactured. The output is the fabrication data for the individual components: STL files for the printed nodes, laser cutting paths for the laser-cut panels, and length tables for the bars.

This work presents an automated design process from a draft in the form of a mesh geometry to a physical prototype. Section 2 describes the overall workflow as well as the design rules that are used according to the KBE approach. A demonstrator in the form of a horse sculpture is created using the presented workflow and realized as a physical object shown in Section 3. The paper concludes with a discussion of the advantages, limitations, and potential future improvements to the workflow.

### Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C_{lf}</td>
<td>Connectivity matrix between lines and faces</td>
</tr>
<tr>
<td>C_{lp}</td>
<td>Connectivity matrix between lines and points</td>
</tr>
<tr>
<td>C_{fp}</td>
<td>Connectivity matrix between points and faces</td>
</tr>
<tr>
<td>d</td>
<td>bar diameter [mm]</td>
</tr>
<tr>
<td>d_k</td>
<td>Distance from node point to start point of bar k [mm]</td>
</tr>
<tr>
<td>f_n</td>
<td>face n</td>
</tr>
<tr>
<td>l_k</td>
<td>line k</td>
</tr>
<tr>
<td>n</td>
<td>normal vector</td>
</tr>
<tr>
<td>p_i</td>
<td>point i</td>
</tr>
<tr>
<td>r_k</td>
<td>radial vector of the bar k</td>
</tr>
<tr>
<td>r</td>
<td>bar radius [mm]</td>
</tr>
<tr>
<td>S_k</td>
<td>starting point of bar k</td>
</tr>
<tr>
<td>u_k</td>
<td>direction unit vector of the bar k</td>
</tr>
</tbody>
</table>

### 2. Automated Design Workflow

Figure 3 provides an overview of the steps of the digital workflow developed in this work. It takes a mesh geometry of the space frame structure and a bar diameter as an input. First, the geometry in the mesh is decomposed, and the information on the geometry is stored in matrices (see Section 2.1). Then the nodes are analyzed and the bar lengths are calculated (see Section 2.2). Finally, the fabrication data is created and directly exported as STL files for the nodes, laser cutting lines for the panels, and length tables for the bars (see Section 2.3).

Figure 3: Overview of the automated design workflow.

The workflow has been developed in the CAD system *Rhinoceros 3D*® and its visual programming interface *Grasshopper*®. Extension plugins for *Grasshopper* that were used are mentioned in the next sections.
2.1. Decomposition

It is assumed that the input mesh consists of long edges with respect to the chosen bar diameter, and has large angles between adjacent edges. These two conditions are required to avoid intersections in the resulting node geometries. The input mesh is decomposed into the simple geometric elements points, lines, and surfaces, and the relationships of these elements within the mesh are stored in matrices. This decomposition simplifies the analysis and the creation of geometries in a later step. The mesh vertices and edges are transformed into points \( p_0 \ldots p_i \) and straight lines \( l_0 \ldots l_k \) respectively (Figure 4). Both lines and points are stored in separate lists maintaining their accessibility through their ID number. All geometries in the subsequent steps except the face panels \( f_0 \ldots f_m \) are generated with the help of these points and lines.

The relationship between the line and point elements must also be extracted to store all the information of the mesh. This is done by defining three connectivity matrices. The \( C_{pl} \) matrix with dimension \([i \times k]\) defines the relationship between the points (row indices) and the edges (column indices). The values in the matrix are defined as:

\[
C_{pl}[i,k] = \begin{cases} 
-1 & \text{point } i \text{ is the starting point of line } k \\
0 & \text{no relationship} \\
1 & \text{point } i \text{ is the end point of line } k 
\end{cases}
\]

All the information of the mesh is accessible through the combination of the connectivity matrices and the lists containing the geometry elements of lines and points.

2.2. Node Analysis

The design of the structural nodes needs to avoid potential collisions between bars that join in one node. To do so, all lines \( k \) that join point \( i \) are evaluated using the connectivity matrix. This condition is fulfilled if \( C_{pl}[i,k] = 1 \). A direction unit vector \( \mathbf{u}_k \) is calculated based on the line objects for all the lines that fulfill this condition. The vectors are calculated such that they point away from the point under consideration. Consequently, the direction of the direction unit vectors \( \mathbf{u}_k \) must be reversed if point \( i \) is the endpoint of the corresponding line \( k \) (\( C_{pl}[i,k] = 1 \)).

The start length \( d_k \) is calculated for each bar. The start length \( d_k \) is defined as the distance between a node point \( i \) and the starting point of the physical bar. It must be defined such that geometric intersections between any bars are avoided. This is done by calculating the angle \( \alpha_k \) between one direction unit vector \( \mathbf{u}_k \) and all other direction unit vectors from the same node \( i \). If the smallest angle \( \alpha_{k,min} \) for \( \mathbf{u}_k \) is larger than or equal to \( 90^\circ \), the start length \( d_k \) is set to the radius of the bar. This bar is not critical for any collisions and hence the vector \( d_k \cdot \mathbf{u}_k \) points to the starting point of bar \( k \).

Figure 4: Mesh decomposition: points, lines, and faces.

The distinction between the starting point and the endpoint of a line is later needed for the design generation of the node geometries.

The \( C_{pl}[i \times k] \) and the \( C_{fl}[k \times m] \) matrix are built up similarly, and store the relationship between point indices \( i \) (rows) and face indices \( m \) (columns), and line indices \( k \) (rows) and face indices \( m \) (columns) respectively. These matrices only save the information on the relationship between the geometrical elements.

\[
C_{pl}[i,m] = \begin{cases} 
0 & \text{point } i \text{ is no corner of face } m \\
1 & \text{point } i \text{ is a corner of face } m 
\end{cases}
\]

\[
C_{fl}[k,m] = \begin{cases} 
0 & \text{line } k \text{ is not on the boundary of face } m \\
1 & \text{line } k \text{ is on the boundary of face } m 
\end{cases}
\]

In the example in Figure 4, the point \( p_1 \) is the endpoint of the line \( l_0 \) and a corner point of the face \( f_0 \). Further, line \( l_0 \) is related to face \( f_0 \). Consequently, the values found in the connectivity matrices are \( C_{pl}[1,0] = 1 \), \( C_{fl}[1,0] = 1 \) and \( C_{fl}[0,0] = 1 \). The following matrices show all three connectivity matrices for the example shown in Figure 4. The values explained above are marked with bold, red letters.

\[
C_{pl} = \begin{bmatrix} 
-1 & 0 & -1 & 0 & \cdots \\
1 & -1 & 0 & 0 & \cdots \\
0 & 1 & 1 & 1 & \cdots \\
0 & 0 & 0 & 0 & \cdots \\
\cdots & \cdots & \cdots & \cdots & \cdots 
\end{bmatrix}, \\
C_{fl} = \begin{bmatrix} 
1 & 1 & \cdots \\
1 & 0 & \cdots \\
0 & 1 & \cdots \\
0 & 0 & \cdots \\
\cdots & \cdots & \cdots 
\end{bmatrix}, \\
C_{fl} = \begin{bmatrix} 
1 & 0 & \cdots \\
1 & 1 & \cdots \\
0 & 1 & \cdots \\
0 & 0 & \cdots \\
\cdots & \cdots & \cdots 
\end{bmatrix}
\]

All the information of the mesh is accessible through the combination of the connectivity matrices and the lists containing the geometry elements of lines and points.

Figure 5: Example for the calculation of start lengths of lines 1 and 5.

Otherwise \( (0^\circ < \alpha_{k,min} < 90^\circ) \), the pair of bars is critical and the starting points of these two bars must be calculated. The calculation steps for the calculation of the start lengths \( d_k \) are described in the following. These steps are described using the example in Figure 5 and the two direction unit vectors \( \mathbf{u}_1 \) and \( \mathbf{u}_5 \) in node 11:
1. Calculation of the normal vector \( \mathbf{n} \) through the cross product between the direction unit vectors \( \mathbf{u}_1 \) and \( \mathbf{u}_5 \). Vector \( \mathbf{n} \) defines the plane that is spanned by the two-direction unit vectors.

2. Calculation of the radial vectors \( \mathbf{r}_1 \) and \( \mathbf{r}_5 \) through the cross product of the normal vector \( \mathbf{n} \) and each direction unit vector \( \mathbf{u}_1 \) and \( \mathbf{u}_5 \). The radial vectors radial vectors \( \mathbf{r}_1 \) and \( \mathbf{r}_5 \) are then scaled in their length with the corresponding bar radius \( r_1 \) and \( r_5 \).

3. Vector combination of the two radial vectors to a vector \( \mathbf{S}_1\mathbf{S}_5 \) which connects the two starting points \( S_1 \) and \( S_5 \).

4. The calculation of the lengths to the starting points can now be defined as a linear combination problem:

   \[ d_1 \cdot \mathbf{u}_1 + d_5 \cdot \mathbf{u}_5 = \mathbf{S}_1\mathbf{S}_5. \]

   This equation is solved for \( d_1 \) and \( d_5 \).

This calculation is repeated for all nodes and all bars that connect them. A new matrix stores the calculated length values for every line at its start and end for later accessibility.

2.3. Fabrication Data

All information needed to generate the fabrication data for the nodes, bars, and panels is available after decomposition and analysis. 3D bodies are needed for the node geometries. To create them, outer bodies (Figure 6A) are generated, which are connected by a Boolean union and from which the inner bodies (Figure 6B) are subtracted by a Boolean subtraction.

![Figure 6: Creation of a node geometry through a boolean operation](image)

First, the outer bodies are created (Figure 6A). These are cylindrical bodies, which later hold the bar ends; ribs, which give additional stiffness to the nodes and are used to fix the panels; and numbers of the node and bar IDs. The length of the cylindrical bodies is defined by the starting length \( d_k \) and an overlap length. The overlap length specifies how large the overlap between the bar and the nodal geometry will be, and is set to twice the bar diameter so that the bar is adequately supported. The creation of rib and numbers depends on planes, which are calculated from the direction vectors of the bars analogous to the analysis in Section 2.2.

Second, the inner bodies are generated (Figure 6B). The inner bodies consist of a guide sleeve with a limit stop for the bars and fixing holes for fixing the panels and the bars with screws. The final node geometry is created by a Boolean subtraction step which subtracts the inner bodies from the outer bodies (Figure 6C). The STL files can be exported to a directory and named with their number ID with Rhino Common. The panel geometries are created by applying an offset to the faces of the original mesh geometry using Weaverbird. The offset corresponds to the radius of the outer bodies of the nodes. Then the contour lines of the panels are used to create surfaces that are flattened. The flattened contours of the panels are transferred to sheets with specified dimensions and nested with the OpenNest plugin. The contours are numbered with the panel IDs to simplify identification. The sheets can then be exported as PDF files and used for production.

For the calculation of the bar lengths, the start lengths \( d_k \) at the beginning and the end of the bar are subtracted from the initial bar lengths. These calculated bar lengths are then exported together with their ID number into a CSV file.

3. Demonstrator

The digital workflow presented can automatically generate fabrication data for irregular space-frame structures. This section shows the end-to-end implementation of the workflow to the physical fabrication and assembly of a demonstrator. This demonstrator aims to show the potential of the workflow for a selection of irregular space frame structures and to prove that the generated data can be used for fabrication. It has no direct industrial use case for the time being. Since the input of the workflow is a mesh geometry of the space frame to be built, any STL geometry can be directly converted into a space frame construction.

Figure 7 shows how the input geometry of a mesh is converted into the output geometries using a mesh of a rhinoceros [15], a horse [16], and a duck [17]. All three
example geometries contain around 200 unique nodes. Furthermore, the structures are scaled so that their height is around two meters. The bar diameter is set to 12 mm for all three examples.

The workflow is set up such that the decomposition (Section 2.1) and analysis (Section 2.2) steps are performed for the whole structure simultaneously. Since these steps contain simple calculation steps, they usually last less than one second. The generation of the node geometries requires more computationally intensive operations, such as Boolean operations with surfaces, and is therefore performed sequentially. The average time needed to create and export one node as an STL file is around 5 seconds. One limitation to the mesh geometry input is found during the creation of these three examples: input meshes should not contain any mesh panels with very sharp corners (equal to small angle α). The nodes in such a corner become very large and can create collisions with adjacent nodes.

To test the whole process chain, the geometry of the horse is processed from the input mesh geometry to the physical prototype. Inexpensive materials and readily available manufacturing methods are used to create this structure (Figure 8): the bars are realized with round wooden rods with a diameter of 12 mm, the panels are made of cardboard, which is cut to shape with a laser cutter, and the nodes are made using the AM process Multi Jet Fusion (MJF). The resulting horse sculpture is 2.6 m long, 0.65 m wide and 2.4 m tall.

![Figure 8: Fabricated nodes, panels, and bars for the horse space frame](image)

The assembly process is tested in advance with a prototype of five nodes forming three panels. The manufacturing tolerances of the MJF components were adapted to the wooden bars and the assembly of three panels was tested. Based on this simplified test, the dimensions of the inner bodies of the nodes were defined for the final parts. Some adaptations are made to the presented workflow to simplify the assembly of the horse space frame. Since the head consists of many detailed elements, it is made from three large MJF parts, which are then glued and screwed together. For this purpose, the bars are modeled with a reduced diameter and directly united with the automatically generated nodes of the head. This adaptation reduces the number of components and thus also the assembly effort. To provide the structure with better stability, the feet are made of several layered MDF boards, which are also made with a laser-cutter. Figure 9 shows the assembled horse sculpture. The detailed features of the head and feet that deviate from the presented workflow are easily recognizable.

![Figure 9: Realized physical prototype of an automatically generated space frame structure (Photo: Andreas Marx)](image)

4. Discussion

This work introduced a digital workflow for the automatic creation of fabrication information for highly irregular space frame structures. The design rules according to which the geometries are generated are described in detail. This workflow allows the creation of such structures, or different variants of a single structure, in a short time. This efficiency gain in the design process saves time and thus enables faster design iterations in the development process. Designers can now concentrate entirely on the design of the input mesh geometry, which makes their work easier. Further, such automated design approaches can also help novices to get started as most of the design know-how is integrated into the workflow. The general approach of the presented workflow can be transferred to other tasks and applications while adhering to its three main steps: (1) The input data is converted to an accessible form by matrices and other data formats. (2) The converted data are processed in a calculation step that contains procedural design rules. (3) The geometries are generated in the last step based on the data from the previous steps.

Currently, the developed workflow is a proof of concept whose functionality has been tested on a demonstrator. At its current stage, the workflow has some limitations. To make the workflow suitable for technical applications, its functions must be further expanded. This can be realized step by step due to its modular algorithms. The following options for
extending and improving the workflow can be implemented:

- The digital workflow does not yet include an automated examination of the suitability of the mesh. It is assumed that the input mesh fulfills the requirements of long edges and angles. An algorithm for automated detection and a re-meshing algorithm could be implemented in the future.
- Application of a structural optimization algorithm, which chooses a separate bar diameter for each bar. Consequently, the workflow must be adapted to cope with individual bar diameters.
- The material and manufacturing processes may be changed. The AM parts of the nodes could be designed to ensure manufacturability with other AM processes than MJF. For example, the overhang constraint for designs produced with L-PBF or fused deposition modeling (FDM) could be considered in the design process.
- Currently, bar elements are defined as straight circular bars. The workflow can be extended by various approaches, for example by allowing a variety of cross-sectional profiles or by having the bar elements follow predetermined non-straight contours.
- The connection of the individual bars to the nodes can be rethought. In this work, nodes with lateral screws were used for fixation.

The next step is to extend the digital workflow specifically for a technical application and to further investigate its potential in a case study. Further, the workflow will be adapted for other tasks and applications.

5. Conclusion

A digital workflow was presented for the automatic generation of fabrication information for highly irregular space frame structures. Starting from a mesh geometry, the mesh is first decomposed in its components (points, lines, panels) and then the manufacturing information (STL files, cutting lines, bar lengths) is automatically generated according to specific design rules. The overall structure is composed of AM node geometries (manufactured with MJF), round bars, and laser-cut cardboard panels. The feasibility of a structure generated with the workflow was demonstrated on a physical prototype of a horse sculpture. In the next step, the workflow should be transferred into a technical application and its functionalities should be extended accordingly. Further, similar workflows should be developed for other design tasks and applications.

Acknowledgments

The authors thank everybody who contributed to this work by giving feedback, supporting during the production and assembly of the demonstrator, and taking photos or videos: Noemi Bearth, Daniel Omidivarkarjan, Jonas Conrad, Andreas Marx, Kordian Caplazi, David Shapira, Thushwee Srikanthan, Lisa Vuillemin and Kai von Petersdorff-Campen.

References