

Procedural Concept Design with Computer Graphic Applications for Light-Weight Structures using Blender with Subdivision Surfaces

Martin Denk^{1,*}, Johannes Mayer¹, Harald Völkl¹, Sandro Wartzack¹

¹ Engineering Design, Friedrich-Alexander-Universität Erlangen-Nürnberg, 91058 Erlangen, Germany

* Corresponding Author:

Martin Denk
Friedrich-Alexander University Erlangen-Nürnberg, KTmfk
Paul-Gordan-Straße 5
91052 Erlangen
☎ 0 9131 85-23218
✉ denk@mfk.fau.de

Abstract

In today's concept development, it is important to develop customer requirements in advance via prototypes with little effort. Some of these requirements can be pre-tested through virtual product development, although creating them with computer aided design (CAD) software can be time-consuming. In the context of this work, tools commonly used in animation design are adapted to the early phase of virtual product development. This is primarily to limit the time effort of prototyping in product development. However, the models must be led to non-intersecting 3D manifolds to ensure a transfer into computer aided engineering (CAE) or possible manufacturing. Therefore in this work, so-called procedural designs based on computer graphic applications are described and examined for their suitability for CAE.

Keywords

Subdivision Surfaces, Concept Design, Light Weight, Procedural, Parametric

1. Motivation

In today's product development, the number of issues related to the development of a product is constantly increasing, so that different types of functions have to be integrated into the design at the same time [1]. In order to meet customer requirements as early as possible, but also to perform feasibility studies, the use of prototypes is necessary [2]. The virtual creation of a suitable prototype, especially with classical CAD software packages in industrial design, can be very time-consuming. Much faster development times for the creation of 3D models can be found in the gaming and design industries [3]. Tools such as geometry nodes [3] or the more flexible surface description in animation programs per se [4] make it possible to design airplanes, bridges, cars, plants, animals, and even entire landscapes intuitively and quickly. Applying these design methods to lightweight structures such as stringer-shell combinations of airplane fuselages, the arrangement of the ABC column of cars, and even biologically inspired designs such as the leaves of a tree can greatly accelerate concept development. Based on the reconstructed models, further techniques such as shape optimization [5], [6], manual adjustment [7] or verification with CAE software can be applied.

While so-called Non-Uniform Rational B-Splines (NURBS) are predominant in industrial design, Subdivision Surfaces [8] can be found in the entertainment and animation industry. Both descriptions are defined using control grids or in other words control polygon meshes. In contrast to Subdivision Surfaces, the topology of the control meshes in NURBS cannot be defined arbitrarily [8], [9], so geometric bodies can be constructed without constraints for such Subdivision Surfaces [8]. This freedom of modeling is one of the reasons why this surface description is so popular in the entertainment domain. For this reason, this thesis develops procedures for creating virtual CAD models with 3D computer graphics software based on such subdivision surfaces (Catmull-Clark). The obtained results are virtually verified by the transferability to CAD software, the application of FE simulations or the manufacturability using path planning in additive manufacturing.

2. State of the Art

Freeform surfaces are usually defined and controlled by so-called control points, whose control points are connected by a control mesh (control polygon mesh). Depending on the choice of the freeform type, restrictions in the respective structure of this control mesh have to be considered. One of the most common freeform surfaces are so-called Non-Uniform-Rational-B-Spline Surfaces in tensor product notation. The control mesh is created as a grid, so that the number of control points in the u -direction and the number of control points in the v -direction must not differ. Such a surface can be generated by tensor multiplication (tensor product spline) with

$$P(u, v) = M_u T_u(u) M_v T_v(v) C_{u,v}$$

where T_u , T_v are the nominal basis, $C_{u,v}$ are the control points, and M_u, M_v are the coefficient matrix (often referred also as geometry matrix) in u and v (typically $M_u = M_v$). Note in particular that the control points $C_{u,v}$ are gridlike arranged along a local coordinate system. Finding suitable grids in a polygonal surface is a subfield of geometric parameterization [10]. However, due to a large number of complex topologies, it is often necessary to use multiple surface segments and thus local coordinate systems u, v . Extensions of these NURBS by so-called T-splines [11] or hierarchical B-splines can counteract this topological limitation, although again, in contrast to the subdivision surface, no completely free choice of node topology is possible.

While this kind of freeform modeling with NURBS as well as their extensions is common in engineering due to the exact surface description as well as the surface quality in terms of continuity (C^2), so-called subdivision surfaces are used in animation programs. The following figure shows the use of control points of a NURBS surface as well as the subdivision surface. Also, the first iteration as well as the convergence (9 iterations) of the subdivision surface is shown.

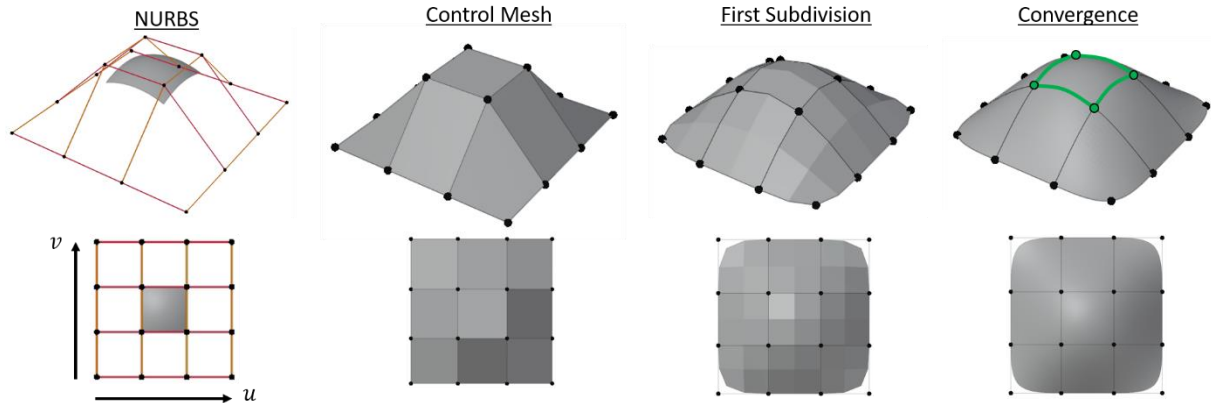


Figure 1: Comparison of Catmull-Clark-Subdivision surfaces and NURBS

While freeform surfaces represent the surface exactly, subdivision surfaces approximate the surface as a polygon mesh by a multiple subdivisions and repositions of the involved faces [9], [12]–[14]. Both the type of displacement and the generation of new subdivision surfaces can be done by different methods such as "Loop subdivision" [13], "Catmull-Clark subdivision" [12] or "Doo-Sabin" subdivision [14]. While some methods are restricted to triangulated control networks [13], others can consider arbitrary control networks [15]. Again depending on the algorithm, the continuity of surfaces at infinite subdivision can be distinguished from C^1 for e.g. "Doo-Sabin" or C^2 for Catmull-Clark subdivision surfaces [15]. However, the authors of [9] recommend the choice of Catmull-Clark subdivision surfaces since they lead to simple but also visually reasonable freeform surfaces. It should be noted, however, that C^2 -continuity only occurs at nodes with valence 4. For all other nodes, only C^1 continuity can be guaranteed. If every interior control point at the node has a valence of exactly 4, Catmull-Clark subdivision surfaces can reproduce the freeform surface very well [8]. The following figure shows a mesh topology consisting of nodes with valence 3,4,5,6, which can no longer be represented as pure NURBS surfaces.

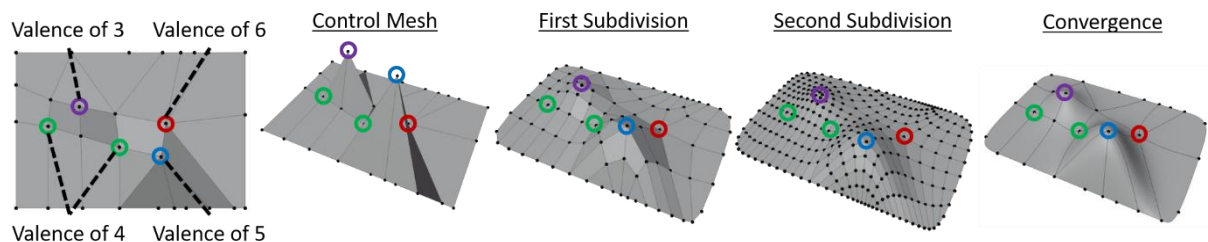


Figure 2: Representation of the subdivision surfaces of different nodal valences

For CAD, these control grids can now be adapted to produce sharp edges, for example by giving a node more weight [16], which can be used to sharp for example the edges of a cylinder. This surface description can also be used for concept development, so that a complete aircraft outer contour can be modeled and then analyzed with CFD [4]. Similarly, individual parametric dependencies can also be defined directly on the subdivision surfaces [17]. To further assist

in the creation of such designs, sketches can be used in addition to the direct definition of individual control points [18]. Simple hand sketches can be used to draw rough contours, which can then be transferred to the corresponding control grids.

While the manual creation of control grids and their parameterization is the classical way of modeling, there are currently efforts to derive such parametric subdivision surfaces directly from 2D and 3D objects. In [19], for example, automated centerline models were recognized directly from colored 2D sketches with subsequent transformation into subdivision surfaces. In this context, organic freeform surface models can be generated via a colored assignment of lines with associated cross sections. In addition to manual construction, such parametric control meshes can also be built using procedural modeling [20]–[22]. In this case, a geometry is parametrically described with various operators from basic elements and then built up so that, for example, rail profiles [21] or truss structures [22] can be generated. Such models can be created, for example, using so-called geometry nodes in Blender [23], Sverchok nodes [23] in Blender or Grasshopper in Rhinoceros 3D [21]. Another option is to convert existing 3D objects directly into a parametric free surface format. In [7], [24] subdivision surfaces were obtained directly from 3D images and polygon meshes parametrically bound to so-called curve and surface skeletons. With these concepts, e.g., results from a topology optimization can be transformed directly into a parametric freeform surface like subdivision surfaces [7], [19], [24] or NURBS [25], but also primitive surface descriptions from Constructive Solid Geometry (CSG) [26]–[28] as well as likewise implicit geometry consisting of convolutional surfaces [29].

In order to make this surface description usable for CAE, it is necessary to take into account the manifold and the Jordan-Brouwer decomposition theorem [30], but also to integrate the sharpening of some edges and surfaces. This serves to completely eliminate ambiguities of a surface assignment as well as the self-intersection of the surface. Also, the valence of most nodes should be equal to four to get a smooth shape description. The following figure shows the violation of the two criteria as well as a way to sharpen local edges.

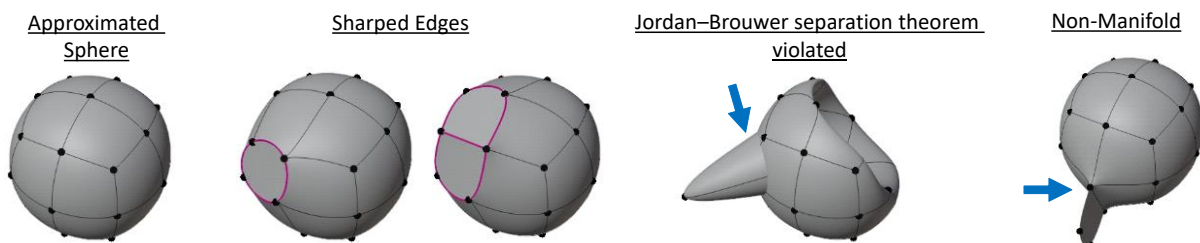


Figure 3: Approximated sphere, shaped edges and violations of the separation theorem and manifold

In contrast to NURBS, these criteria can be checked with simple test methods from the so-called mesh processing [10], e.g. the number of surfaces at a polygon edge can be evaluated to determine if it satisfies the manifold criterion. If these criteria are fulfilled, the subdivided surfaces can be transferred directly into the 3D CAD [31]. In the following, using so-called state diagrams and geometry node rules, it is shown how CAD models can be created in a few steps using tools from the Blender program.

3. Procedural Models with State Diagrams

The two state diagrams are defined as generally as possible and can be applied to geometrically similar cases. The following figure shows the creation of an aircraft fuselage based on a concept sketch over such a diagram. The outer skin of the aircraft is created as a surface model including the stringers. The aircraft segments are then multiplied and can

optionally be manually adjusted. The 3D geometry can then be derived by thickening the surface model. Since it is exclusively a facet model with constant normal vector, thickened models with constant thickness can be generated directly without violating the Jordan-Brower theorem for separation.

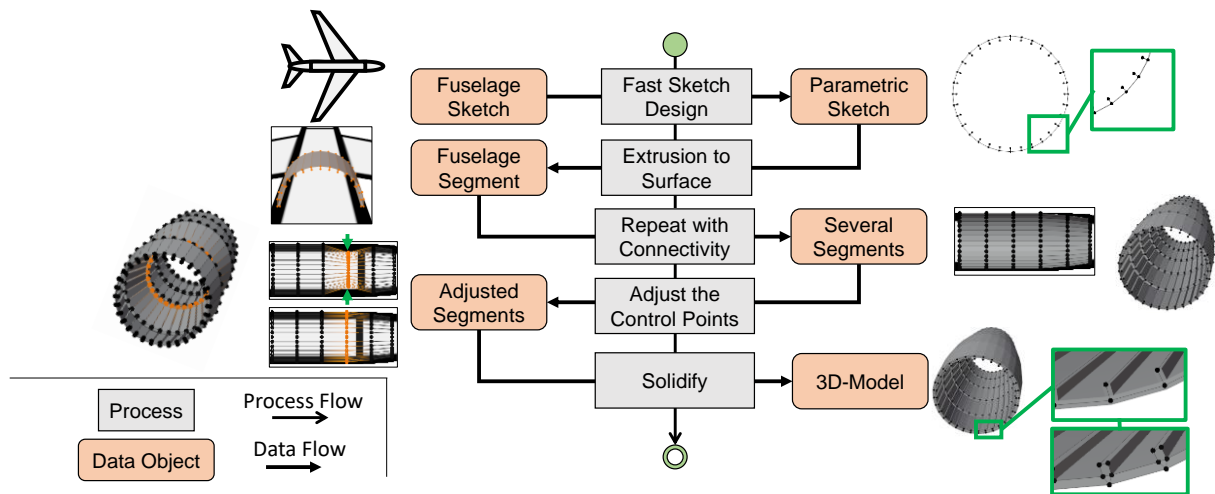


Figure 4: Procedure of the fast concept design of a fuselage based on a sketch

An alternative approach can be developed for frame-shaped circular structures. Here, a control grid is created as a parametric sketch along a profile line. A 3D object is then created by specifying a radius at each node of the sketch. The radius is approximated by a square control grid cross-section and extruded along the single line. Finally, the pattern can be repeated, e.g. by mirroring, and the local thicknesses can be adjusted. In contrast to the previous model, the 3D elements result directly from the extruded cross sections.

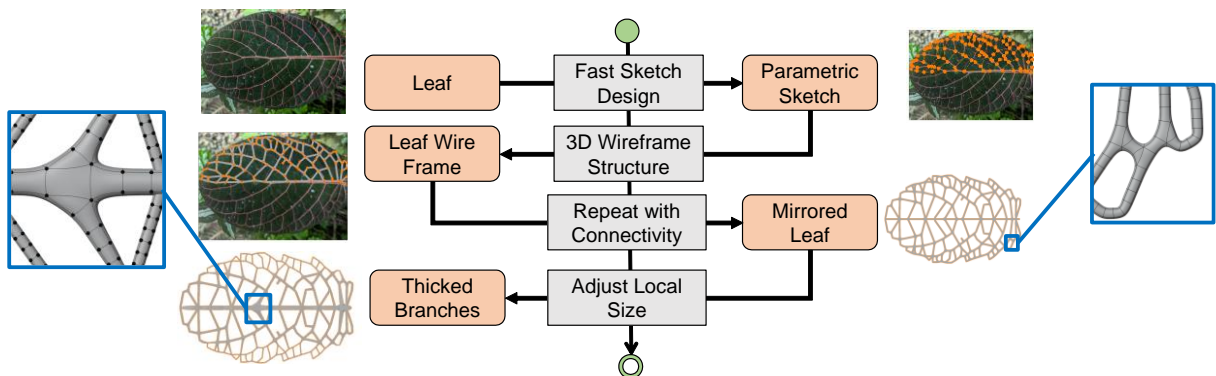


Figure 5: Procedure of the fast concept design of a leaf

The whole sketch can then be adjusted at the points of the skeleton lines. Both models show a systematic approach for the reconstruction of planar surfaces as well as for the reconstruction of frame structures. However, the current development of 3D models is visibly moving towards procedural parametric models. In this case, the entire model is built using individual graphical toolboxes that contain a large number of parameters. In the following, so-called geometry nodes are used in Blender.

4. Procedural Parametric Fast Concept Design with Geometry Nodes in Blender

Geometry nodes are elements that are coupled via nodes to change the geometry. The following figure shows the pipeline for creating a pipe with variable thickness profile along a spline, as well as an extended example with the inclusion of additional struts. The geometric body can now be modified in each of these individual toolboxes to change, for example, the radius distribution, the number of struts used and the mesh detailing. These parametrics can also be used to store the model in a knowledge-based manner so that this design logic can be applied directly to new splines.

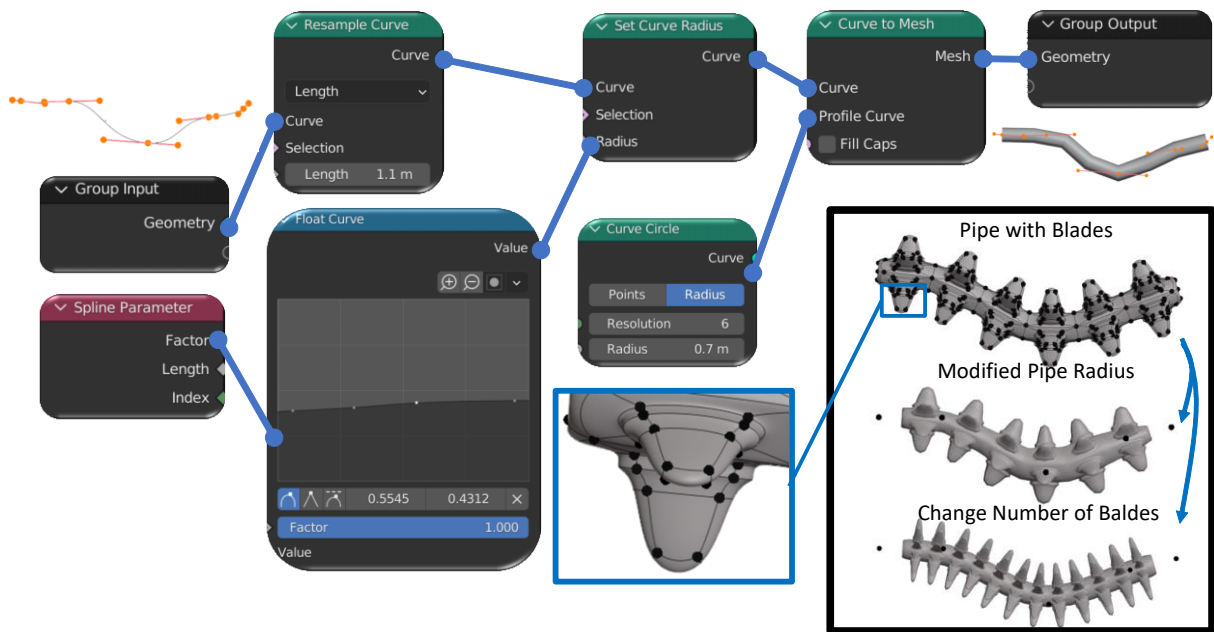


Figure 6: Geometry Nodes simple pipe example and results of considering blades

When creating these geometry nodes, care must be taken to ensure that the resulting geometry is closed and continuous and does not intersect itself (manifold and Jordan-Bower theorem). Otherwise, no meaningful 3D analysis can be performed in the CAE program. Such a condition can be ensured by the basis of a single pipe segment of the geometry. A single pattern must satisfy this condition so that the derived segments also take this property into account. Likewise, the linkage of the individual segments must be integrated into the structure. Furthermore, if only segments with nodes of valence 4 are used, a continuous C^2 geometry can be guaranteed for the entire geometry, as in this case for the pipe with blades.

In the following, it is now necessary to apply the concepts of geometry nodes as well as procedure models in order to directly create CAD models that can be directly analyzed by means of FEM, for example. If the 3D objects can be built directly as 3D FE models, then the Jordan-Bower theorem and the manifold of the underlying geometry are automatically correct.

5. Results

The following figure shows applying of the described process models for a honeycomb body as well for a result from a topology optimization of a heat exchanger (snowflake) [29]. The individual steps were applied directly to the problem in each case. The models were also reconstructed directly as 3D models.

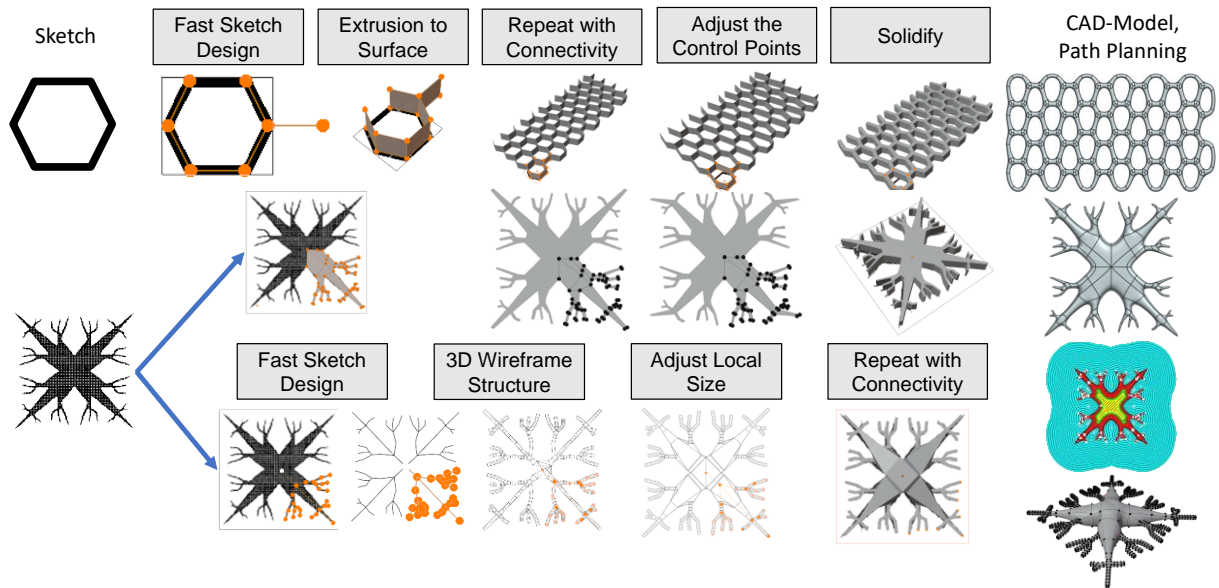


Figure 7: Results of applying the procedural state diagrams

In the case of the snowflake, additional path planning was carried out for subsequent prototype construction via FDM printing. Likewise, these models can now be virtually tested in a bending load case. The following figure shows the direct transfer of the honeycomb geometry into an FE model as well as its resulting displacement distribution.

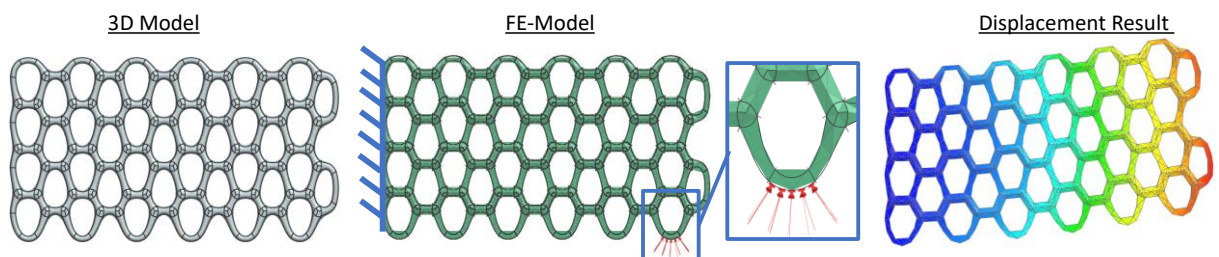


Abbildung 8: Importing the honeycomb sample in Siemens NX and applying a linear elasto static simulation

This use case demonstrates that the models derived from the described procedures can be verified automatically without modifications or repairs. Likewise, the models can now be evaluated or regenerated based on the geometry nodes. The following figure shows the creation of a 3D geometry based on a spiral Bezier curve drawn in Blender. The number of struts involved can be automatically adjusted in the model. The model can also be imported as a CAD component into Siemens NX and then built and evaluated as an FE model.

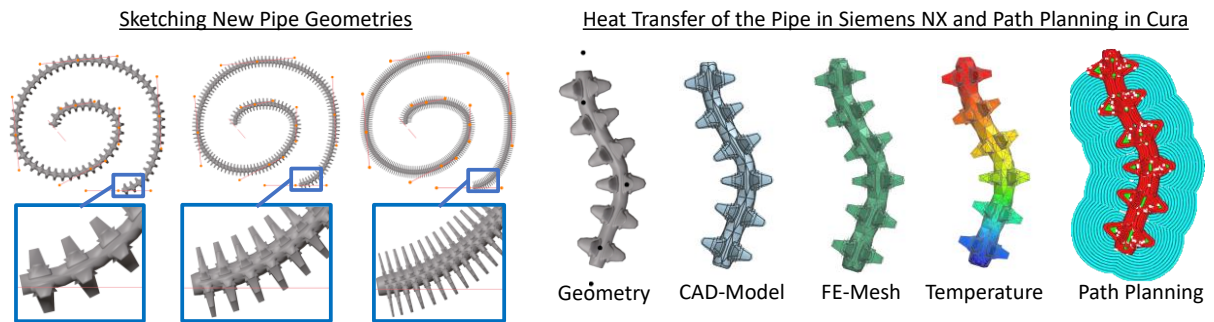


Figure 9: Different blade parameters and shape of the splines. Importing the pipe model in Siemens NX and applying a heat transfer simulation

Besides the possibility to use geometry nodes and the described state diagrams, the freeform surface components can be drawn without any procedure directly. Similar to the previous models, it is important to achieve a node valence of 4 to ensure curvature stability.

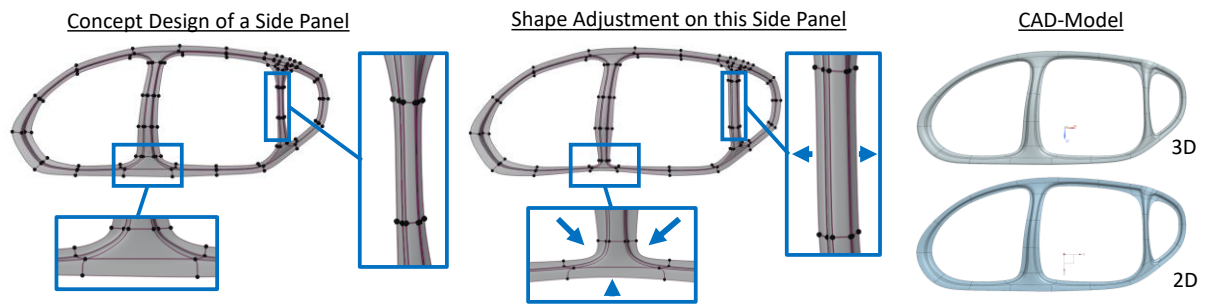


Figure 10: Side panel reconstruction and its derived 2D and 3D CAD-Models.

The results show a good transferability of the models from animation design to CAE. Models can be created directly in a few steps and variants can be generated by changing the control points. The junctions of the model merge seamlessly, and sharp edges are preserved where they are needed.

6. Discussion

The use of computer graphics software enables the rapid creation of 3D models. However, when creating the models, criteria such as maintaining the manifold and avoiding self-intersection must be considered. Similarly, the models in the animation program offer the possibility of user-friendly manual adjustment in the animation program through sketch-based parameterization. While the models themselves are parametric, common constraints such as parallelism or perpendicularity or simple dimension lines are often missing. These will also need to be integrated into the creation process in the future.

The polygon mesh format also restricts the user to freeform surfaces. While these can be converted to sharp edges, primitives such as cylinders can be accurately represented only by considering a large number of polygons. A direct coupling of computer graphics software from the entertainment sector with a CAD core could balance the advantages and disadvantages of both worlds, so that, for example, a direct evaluation of freeform surface models can be made.

The easiest way to create the geometry manually is to extrude profiles and manually join the individual facets to create a suitable freeform model covered in Figure 10. To speed up such a reconstruction, process models and predefined routines from computer graphics programs offer a drastic simplification shown in Figure 4, and Figure 5. For example, 3D models can be obtained directly from wireframe structures as freeform surfaces whose cross-section can be controlled by a local radius.

The most efficient variant is the use of so-called procedural toolboxes such as geometry nodes in Blender visualized in Figure 6. Here, the geometric bodies are created from individual existing modules. Each of these individual modules can contain a variety of parameters, so that, for example, the number of cooling struts can be set. The saved parametric design rule can also be used as its own module, so that the design can be saved directly. Overall, the use of computer graphics software is a good complement to conventional industrial design. These models are particularly suitable for rapid concept designs including verification calculations. However, the major drawback is that necessary primitives such as cylinders to generate holes are not integrated. These can be approximated as a polygon mesh, but the information such as the hole radius is not stored. However, these are essential for classical manufacturing processes. In the future, the above descriptions will be significantly extended so that CSG objects that provide information about polygon meshes can be directly integrated, but their parameters will be stored. The mapping of the primitive as a polygon mesh is necessary to create a seamless surface composite.

7. Summary

While the creation of e.g. automotive components in industrial CAD software requires a high effort, this investment is not made for models in the gaming and design area. This speed advantage can now be used in concept development or pre-development to be able to construct a large number of different variants, but also to prove them. In this work, such tools are used to reconstruct and parameterize a large number of models. In particular, the so-called geometry nodes in Blender provide a very efficient way to store design rules in the form of modules. The resulting bodies can then be loaded directly into industrial design programs and used for CAE simulations, for example. It is important that the geometry leads to a correct manifold and does not intersect itself (Jordan-Brouwer separation). It is also important to create polygon meshes where most of the nodes have a valence of four, if possible, in order to obtain surfaces that are continuous in their curvature. The biggest disadvantage of the described method is the still missing integration of primitive bodies and common CAD features. In the future, especially the integration of such primitive bodies and features shall be investigated, so that a hybrid format of CSG and subdivision surfaces is created.

Literature

- [1] M. Hoßfeld und C. Ackermann, Hrsg., *Leichtbau durch Funktionsintegration*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2020. doi: 10.1007/978-3-662-59823-8.
- [2] A. K. Kamrani und E. A. Nasr, *Engineering Design and Rapid Prototyping*. Springer Science & Business Media, 2010.
- [3] R. Lotter, *Taking Blender to the Next Level: Implement advanced workflows such as geometry nodes, simulations, and motion tracking for Blender production pipelines*. Packt Publishing Ltd, 2022.
- [4] K. Amadori und C. Jouannet, „Aircraft Concepts Modeling with Subdivision Surfaces“, in *54th AIAA Aerospace Sciences Meeting*, doi: 10.2514/6.2016-0775.
- [5] K. Bandara, T. Rüberg, und F. Cirak, „Shape optimisation with multiresolution subdivision surfaces and immersed finite elements“, *Comput. Methods Appl. Mech. Eng.*, Bd. 300, S. 510–539, März 2016, doi: 10.1016/j.cma.2015.11.015.

- [6] K. Bandara und F. Cirak, „Isogeometric shape optimisation of shell structures using multiresolution subdivision surfaces“, *Comput.-Aided Des.*, Bd. 95, S. 62–71, Feb. 2018, doi: 10.1016/j.cad.2017.09.006.
- [7] M. Denk, K. Rother, und K. Paetzold, „Fully Automated Subdivision Surface Parametrization for Topology Optimized Structures and Frame Structures Using Euclidean Distance Transformation and Homotopic Thinning“, in *Proceedings of the Munich Symposium on Lightweight Design 2020*, Berlin, Heidelberg, 2021, S. 18–27. doi: 10.1007/978-3-662-63143-0_2.
- [8] T. J. Cashman, „NURBS-compatible subdivision surfaces“, März 2010, doi: 10.2312/8238.
- [9] T. J. Cashman, „Beyond Catmull–Clark? A Survey of Advances in Subdivision Surface Methods“, *Comput. Graph. Forum*, 2012, doi: 10.1111/j.1467-8659.2011.02083.x.
- [10] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, und B. Lévy, *Polygon Mesh Processing*. AK Peters / CRC Press, 2010.
- [11] T. W. Sederberg, J. Zheng, A. Bakenov, und A. Nasri, „T-splines and T-NURCCs“, in *ACM SIGGRAPH 2003 Papers*, New York, NY, USA, 2003, S. 477–484. doi: 10.1145/1201775.882295.
- [12] E. Catmull und J. Clark, „Recursively generated B-spline surfaces on arbitrary topological meshes“, *Comput.-Aided Des.*, Bd. 10, Nr. 6, S. 350–355, Nov. 1978, doi: 10.1016/0010-4485(78)90110-0.
- [13] C. Loop, „Smooth Subdivision Surfaces Based on Triangles“, Master’s thesis, University of Utah, Department of Mathematics, Utah, USA, 1987.
- [14] D. Doo, „A subdivision algorithm for smoothing down irregularly shaped polyhedrons“, 1978, S. 157–165.
- [15] K. Vlachkova und P. Terziev, „A comparison of surface subdivision algorithms for polygonal meshes“, *AIP Conf. Proc.*, Bd. 1487, Nr. 1, S. 343–350, Okt. 2012, doi: 10.1063/1.4758977.
- [16] W. Ma, „Subdivision surfaces for CAD—an overview“, *Comput.-Aided Des.*, Bd. 37, Nr. 7, S. 693–709, Juni 2005, doi: 10.1016/j.cad.2004.08.008.
- [17] K. Amadori, C. Jouannet, und J. Andersson, „Parametrically Controlled Subdivision Surfaces for Conceptual Design“, in *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, doi: 10.2514/6.2017-4005.
- [18] M. Bein, S. Havemann, A. Stork, und D. Fellner, „Sketching subdivision surfaces“, in *Proceedings of the 6th Eurographics Symposium on Sketch-Based Interfaces and Modeling*, New York, NY, USA, Aug. 2009, S. 61–68. doi: 10.1145/1572741.1572753.
- [19] M. Denk, K. Rother, und K. Paetzold, „Beam-colored Sketch and Image-based 3D Continuous Wireframe Reconstruction with different Materials and Cross-Sections“, in *Entwerfen Entwickeln Erleben in Produktentwicklung und Design 2021*, Dresden, Germany, Sep. 2021, S. 345–354. doi: 10.25368/2021.33.
- [20] S. Bhooshan und M. El Sayed, „Use of sub-division surfaces in architectural form-finding and procedural modelling“, in *Proceedings of the 2011 Symposium on Simulation for Architecture and Urban Design*, San Diego, CA, USA, Apr. 2011, S. 60–67.
- [21] S. A. Biancardo, M. Intignano, N. Viscione, S. Guerra De Oliveira, und A. Tibaut, „Procedural Modeling-Based BIM Approach for Railway Design“, *J. Adv. Transp.*, Bd. 2021, S. e8839362, März 2021, doi: 10.1155/2021/8839362.
- [22] R. Ajouz, „Parametric design of steel structures“, *Steel Constr.*, Bd. 14, Nr. 3, S. 185–195, 2021, doi: 10.1002/stco.202100011.
- [23] G. Muioli, „Building a 3D Environment“, in *Introduction to Blender 3.0: Learn Organic and Architectural Modeling, Lighting, Materials, Painting, Rendering, and Compositing with Blender*, G. Muioli, Hrsg. Berkeley, CA: Apress, 2022, S. 167–228. doi: 10.1007/978-1-4842-7954-0_3.
- [24] M. Denk, K. Rother, und K. Paetzold, „Subdivision Surface Mid-Surface Reconstruction of Topology Optimization Results and Thin-Walled Shapes using Surface Skeletons“, *Proc. Des. Soc.*, Bd. 1, S. 2771–2780, Aug. 2021, doi: 10.1017/pds.2021.538.
- [25] A. Amroune, J.-C. Cuillière, und V. François, „Automated Lofting-Based Reconstruction of CAD Models from 3D Topology Optimization Results“, *Comput.-Aided Des.*, Bd. 145, S. 103183, Apr. 2022, doi: 10.1016/j.cad.2021.103183.
- [26] G. Yin, X. Xiao, und F. Cirak, „Topologically robust CAD model generation for structural optimisation“, *Comput. Methods Appl. Mech. Eng.*, Bd. 369, S. 113102, Sep. 2020, doi: 10.1016/j.cma.2020.113102.
- [27] J.-C. Cuillière, V. François, und A. Nana, „Automatic construction of structural CAD models from 3D topology optimization“, *Comput.-Aided Des. Appl.*, Bd. 15, S. 107–121, 2017.
- [28] A. Nana, J.-C. Cuillière, und V. Francois, „Automatic reconstruction of beam structures from 3D topology optimization results“, *Comput. Struct.*, Bd. 189, S. 62–82, Sep. 2017, doi: 10.1016/j.compstruc.2017.04.018.
- [29] M. Denk, K. Rother, und K. Paetzold, „Truss Parametrization of Topology Optimization Results with Curve Skeletons and Meta Balls“, *Proc. Des. Soc.*, Bd. 2, S. 363–372, Mai 2022, doi: 10.1017/pds.2022.38.
- [30] U. Reif und M. A. Sabin, „Old problems and new challenges in subdivision“, *J. Comput. Appl. Math.*, Bd. 349, S. 523–531, März 2019, doi: 10.1016/j.cam.2018.06.012.
- [31] M. Antonelli, C. V. Beccari, G. Casciola, R. Ciarloni, und S. Morigi, „Subdivision surfaces integrated in a CAD system“, *Comput.-Aided Des.*, Bd. 45, Nr. 11, S. 1294–1305, Nov. 2013, doi: 10.1016/j.cad.2013.06.007.