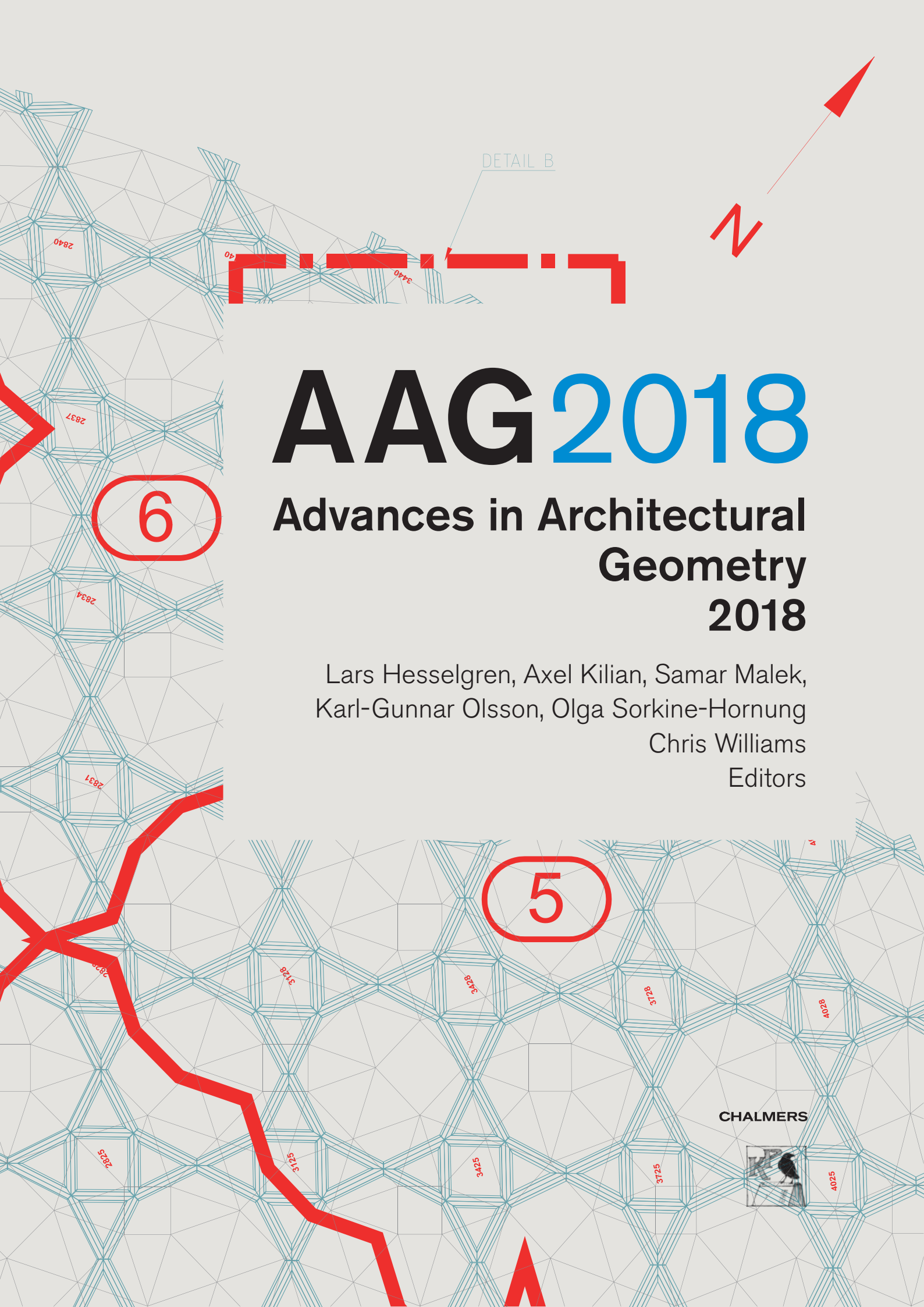


AAG2018

Advances in Architectural Geometry 2018

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

AAG2018

Printed in Vienna 2018
Published by Klein Publishing GmbH (Ltd.)
kpv@gmx.at
ISBN 978-3-903015-13-5

Download open access:
https://research.chalmers.se/en/publication/504188
www.chalmers.se
info@chalmers.se

Cover illustration by Goswin Rothenthal, Djordje Spasic and René Ziegler, Waagner-Biro Stahlbau AG; Ateliers Jean Nouvel, architect and TDIC owner, “Oasis of Light – Dome – Outer Cladding”

Host:

CHALMERS

Supported by:

City of Gothenburg

Gold sponsor:

wsp

Silver sponsor:

AUTODESK.

Bronze sponsors:

waagner biro

white

Semrén+Månsson

Johanneberg Science Park

KUKA

FOJAB arkitekter

SWECO

AAG2018

Advances in Architectural Geometry 2018

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

Editors

CHALMERS

Foreword

Architecture is the physical realization of geometry, and even today architecture poses aesthetic and practical problems for architects, engineers, mathematicians and computer scientists to overcome using traditional and new mathematical methods. It is 60 years since Computer Numerical Control (CNC) machines were first introduced, but now they are becoming commonplace in the building industry enabling complex designs to be fabricated enabling buildings to be constructed which almost rival medieval stereotomy.

There has always been debate in architecture as to the relative merits of simplicity verses complexity, structural honesty verses ornamentation and the advent of computers in the design process has if anything confused this issue, especially now that sustainability is a major concern. Some architects and engineers mistakenly believe that decisions can be left to computational optimization, whereas the more one knows about a subject the more one understands that a true optimum is a chimaera.

Since its first edition, organized by Helmut Pottmann in 2008, the aim of the Advances in Architectural Geometry symposium has been to provide a platform for interdisciplinary debate through contributions of both technical and theoretical nature. It is hoped that this volume will continue this debate and in particular encourage young architects, engineers and computer scientists to enter the world of Euclid, Hypatia, Descartes, Gauss and Ada Lovelace.

Lars Hesselgren, Karl-Gunnar Olsson
Conference Co-chairs

Axel Kilian, Samar Malek,
Olga Sorkine-Hornung, Chris Williams
Scientific Co-chairs

Jonas Runberger, Ahmad Abdul Sater
Workshop chair, Coordinator

Morten Lund, Linnea Jansson
Conference moderator, Coordinator

Toni Kotnik, Mats Ander, Erica Hörteborn
Communication chair, Coordinators

Contents

Adapting architectural form to digital fabrication constraints

Guy Austern, Gershon Elber, Isaac Guedi Capeluto, Yasha Jacob Grobman

Guy Austern
guyaustern@gmail.com
Technion – Israeli Institute of Technology, Israel

Gershon Elber
gershon@cs.technion.ac.il
Technion – Israeli Institute of Technology, Israel

Isaac Guedi Capeluto
arrguedi@technion.ac.il
Technion – Israeli Institute of Technology, Israel

Yasha Jacob Grobman
yasha@technion.ac.il
Technion – Israeli Institute of Technology, Israel

Keywords:
Digital fabrication, rationalization, design method, optimization

Abstract

Today, architects can conceive an almost infinite variety of complex 3D geometries using modelling software. However, the realization of these geometries is still limited by the capabilities and constraints of the fabrication machinery. The modelling platforms used to design the geometries are still not able to evaluate fabrication parameters such as feasibility and machining time. This makes architects are dependent upon fabricator input for introducing fabrication considerations into the creative process, so that they can rationalize their designs.

To bridge this gap, we present a near real time method for the analysis and evaluation of the fabrication potential of molds for complex geometries. Our method can approximate the feasibility, material use, and machining time required for the fabrication of molds for concrete façade elements. The method was developed for mainstream mold fabrication techniques such as cutting and assembly of sheet materials as well as milling of volumetric material. It was further expanded to include robotic hot wire cutting, a state of the art mold fabrication technique.

The method described provides numeric and graphic evaluation results for complex geometries in a few seconds. Its high speed makes it useful for interactive, fabrication aware design and for computational optimization. In this paper, we outline the need for such a method, explain its main algorithms and show case studies where it was used for design rationalization.

1. Introduction

Contemporary Computer Aided Design (CAD) software enables architects to design a seemingly unlimited range of geometries. However, the Computer Aided Manufacturing (CAM) techniques used to fabricate these shapes are not omnipotent and have many constraints related to the physical capabilities of the fabrication machinery or material properties (Kilian 2006). Design for Manufacturing (DFM) is a common practice in industrial design, aimed at accommodating machining constraints at an early design stage (Liu and Yang 2001). In architecture, accommodating practical fabrication constraints is often referred to as design rationalization. Pottman et al. (2015) review contemporary rationalization

practices and advocate the introduction of simple simulations of fabrication constraints into architectural modeling systems. In a more recent review, Austern et al. (2018) stress the importance of providing simple, parametric rationalization methods which can be performed within the architectural design process.

In this paper, we present the results of a research aimed at developing a computational method for analyzing geometry at the preliminary stage of the design process. The method described predicts the feasibility of a geometry and the resources needed to fabricate it with the digital fabrication techniques most often used in the industry. The method was developed into a near real time analysis plugin for the Rhino/Grasshopper (RH/GH) parametric environment. A typical analysis of a single surface ranges from a fraction of a second to few seconds, depending on the complexity of the surface and the number of sampling points chosen. This is significantly faster than traditional CAM methods which require specialized technicians and software to simulate fabrication, a method which usually takes about an hour. The near real time operation of the method makes it possible to incorporate fabrication evaluation into the architectural design process.

In the background chapter of this paper, we review rationalization in contemporary architectural practice and research as well as mold making techniques in the industry, illustrating the gaps in this field. In the methods chapter, we describe the main features of our suggested analysis method, detailing its innovations. In the results section, we discuss validation and show use examples. Ultimately, we show how the method can be used as a fitness criterion for an optimization algorithm.

2. Background

2.1 Rationalization in the architectural practice

The practice of designing buildings to accommodate available construction techniques is not new. From Roman temples to Gothic Cathedrals, architects traditionally designed with the practicalities of the building trade in mind. In the previous century, the works of Nervi, Isler, and Dieste pioneered the use of simple construction techniques for achieving complex building geometries. In contrast, works like Utzon's Sidney Opera

and Le Corbusier's Philips Pavilion exhibited complex designs which could not be built as the architect originally intended. These designs had to be simplified and disciplined in order to be realized (Fischer 2012), a practice referred to today as rationalization.

The contemporary use of the term design rationalization is attributed to Glymph, an architect working in Gehry Partners, who described the practice of introducing "rules of constructability" into Gehry's free-form designs (Lindsey 2001). Whitehead further developed the term, describing projects he developed in Foster + Partners as either Pre or Post Rationalized, depending on the timing in which geometries were translated into constructible forms (Whitehead 2005). Ceccato, working at Zaha Hadid Architects highlighted the importance of Co-Rationalization, the intensive use of parametric modeling tools throughout design sequence, in contemporary architectural practice (Ceccato 2011). With the widespread acceptance and availability of parametric environments over the last decade, this type of rationalization has been found to be the one most commonly performed in the architectural practice (Austern et al. 2018).

2.2 Rationalization in academic research

Shortly after the millennium, one of the main research directions in this field focused on developing methods for adapting geometry to different types of realization constraints. The largest branch of this research direction deals with the rationalization of glazed facades towards constraints related to glazing systems, often focusing on the creation of Planar Quad meshes with certain favorable properties. These studies rely heavily on advanced geometry and mathematics and are reviewed in (Pottmann 2010; Pottmann et al. 2015). Recent work in this field has highlighted the importance of real-time evaluation of design geometry as a basis for interactive design and computational optimization (Deng et al. 2015; Tang et al. 2014).

Other research focuses on finding methods for accommodating constraints related to digital fabrication techniques. Manahl et al. (2012) describe a method for translating free-form geometry into plates producible by 3-axis milling. Dritsas et al. (2013) describe a method of optimizing free-form shells so that its pieces can be optimally packed on sheets of material for 2.5D milling. Flory et al. (2013) describe methods which translate geometry into ruled surfaces, which are relatively easy to build using a variety of techniques, including robotic hot wire cutting.

Brander et al. (2016) describe methods related to designing with flexible hot blades. Louth et al. (2017) describe how complex structural geometries can be rationalized so that they can be assembled from laser kerf-cut, bent sheet material. However, all of these methods are focused on a single fabrication process, and none of them enable comparing different digital fabrication techniques on the same geometry. An exception to this is the work of Eigensatz et al. (2010), who propose a façade paneling algorithm capable of discerning between the geometry of the panels (such as flat, single curved or double curved). However, this method does not differentiate between specific fabrication techniques, instead classifying all complex double curved panels into a single category.

2.3 Mold making techniques in the industry

The presented research focuses on rationalizing geometry towards the constraints of mold fabrication, focusing on molds for concrete. Concrete, the most common building material of our era (CSI 2009), is exceptionally suited for producing free-form geometry due to its initial, liquid state. In order to realize to the full range of geometries that can be produced from concrete, complex 3D molds are required. In the industry, these molds are usually manufactured from CNC cut plywood sheets, such as the ones developed by Designtoproduction for SANAA's Roles learning center (Scheurer 2010). Alternatively, molds are milled from Expanded Polystyrene (EPS) using 3-axis, 5-axis or robotic milling setups. This is a precise technique but it is highly time consuming, as was reported by TailorCrete project- a joint EU research into 3D formwork (Andersen et al. 2016). Another technique which has recently gained acceptance into the industry is Robotic hot wire cutting of EPS, which has been successfully used by ODICO in building scaled projects (Sondergaard and Feringa 2017).

Due to the large waste of materials and machine time incurred by all of these molds, many have attempted to devise alternative fabrication techniques. Reusable, flexible molds are a promising technology reviewed in (Hawkins et al. 2017). Despite the many benefits of this technology, the review shows how geometric constraints, modelling issues and uncertainty regarding its benefits still prevent the industry from adopting this technique. A different solution to the problem is the use of 3D printing in concrete to do away with the formwork completely (Khoshnevis 2004; Lim et al. 2012). This technique is not fully integra-

ted into the industry yet, and currently used mostly to create extruded geometries, due to the material properties of concrete (Labonnote et al. 2016).

To adapt their design to any one of the mentioned techniques, architects need information regarding its specific fabrication performance. In practice, evaluating the fabrication parameters of a geometry involves two different stages: a) Manual preparation for fabrication, typically performed by a CNC operator, using 3D modeling platforms and dedicated CAM software. b) A machining simulation performed by the CAM software, which calculates all the points along the machining toolpath. Both of these operations are highly time consuming, and not achievable by designers within typical 3D modelling environments. This means that in order to rationalize a geometry designers must send it over to the fabricator for appraisal. The typical tendering structure of architectural projects means that this is usually not possible until after the tender has been issued (Celento 2010). At this point the design has already been finalized and major changes due to the rationalization very hard to implement.

In the following section, we will describe a method for evaluating geometries in relation to different fabrication techniques, so that they can adapt them to the chosen fabrication technique. The method was initially developed for different types of milling setups, the most common fabrication technique in contemporary industry. It was later expanded to cover robotic hot wire cutting. The modular structure of the method allows future expansion to include any technology accepted by the industry. Currently, the proposed method can evaluate the fabrication parameters of a single surface in less than a second, providing the designer with near real time feedback which he can incorporate into the design process.

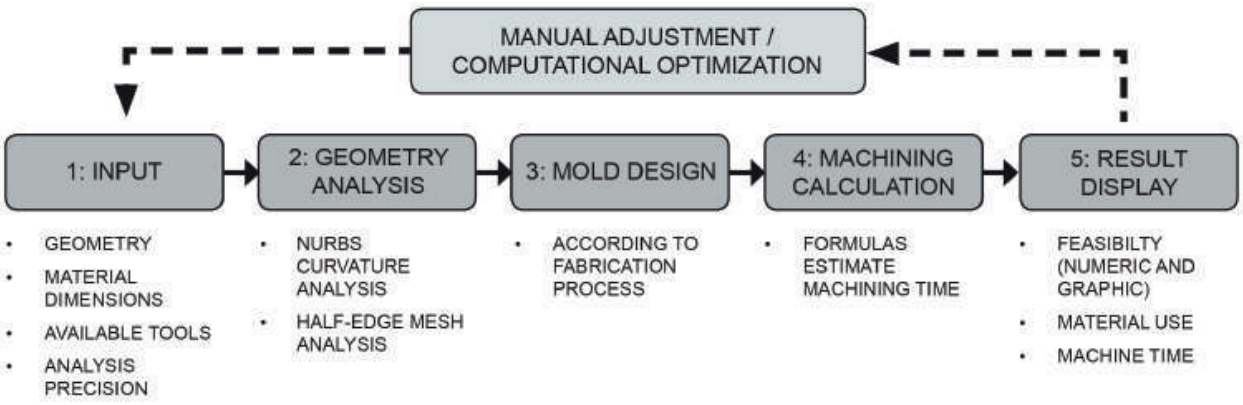


Figure 1: Structure of the proposed method.

3. Methods

The first stage of the proposed method uses the Rhino/Grasshopper (RH/GH) interface to input a NURBS geometry and key fabrication/material settings. Following, a dual analysis of the input geometry is performed: once as a NURBS surface and once as a half-edge mesh. Then, the geometry is translated into molds, according to the material and fabrication parameters. In the next stage we introduce an innovative approach to estimating machining time by using mathematical approximations to predict the tool path length. This saves significant computational resources in comparison to simulating machine behavior, the typical procedure in the CAM industry. Finally, the results are displayed both numerically and graphically, detailing the feasibility, machining time and material use for any of the chosen fabrication techniques. The general structure of the proposed method is described in Figure 1.

3.1 Stage 1: Input and initialization

At the first stage of the proposed method, it accepts single surfaces, multiple surfaces or poly-surfaces from the user in NURBS form. Fabrication parameters such as design tolerance, available tools, material types, and dimensions are set in the GH interface. The user can also choose the orientation strategy: No orientation, best fit (for minimizing material use), common normal (for mostly flat surfaces) or average normal (best vertical tool access for complex geometries). The user then sets the sampling precision, which is automatically adjusted for surface size and complexity. A major premise behind our method is that analyzing the entire surface is not always necessary and that an approximation based on a point sample can estimate realistic machining behavior in a fraction of the calculation time.

3.2 Stage 2: Geometry analysis

In this stage we use IRIT – a freeform geometric modelling environment geared toward development and research (Elber 2016), to provide an efficient differential analysis of the NURBS surface. The rhino NURBS representation is converted into an IRIT NURBS by a C# plugin we developed, which invokes efficient C based IRIT functions directly from the GH environment. This provides us a set of computationally efficient curvature related values: normals, curvatures, Gaussians, radii and principal directions for all our sampling points in less than 1/100th of the time

required to achieve this by the existing Rhino curvature engine.

The surface is also translated into a half-edge mesh representation whose vertexes are the sampling points described above. This representation is interrogated to provide further information regarding the local conditions around the sampling points. Directed, half edge meshes were introduced by Campagna et al. (1998) for improving computation speed at the cost of memory and implemented for RH/GH by (Piker 2017). The result is a dataset which includes the neighboring indexes, positions, influence areas, and normals for each of the sampling points.

3.3 Stage 3: Mold design

In a traditional design flow, the translation of the architectural form into molds is manually performed by a CAD operator at the construction stage. To be able to estimate the required fabrication resources, the suggested method includes a near real time, automated mold design module which can produce single sided “open” molds, or double sided closed molds. Figure 2 shows the results of automatic operations performed on the same geometry to achieve open molds for different fabrication techniques.

For 2.5 axis sheet material cutting and assembly, we design a “waffle” structure to support the mold face, unroll it and nest it onto sheets. For milling, the geometry is situated within blocks of EPS material, and its borders extruded diagonally to ensure milling access. The intersection between the extrusion and the EPS geometry is the actual milling volume. For Hotwire operations, we approximate the NURBS geometry using piecewise developable ruled surfaces, extending the surfaces towards the material boundaries results in a model representing the hot wire material removal operation.

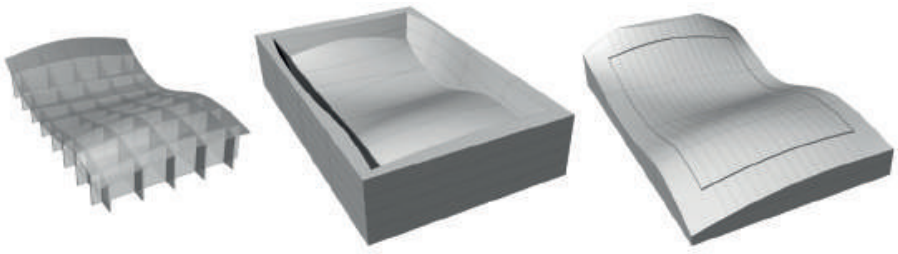


Figure 2: Different open molds automatically prepared by the algorithm for the same input geometry: cutting and assembly of sheet material (left), milling in EPS (center) and hot-wire material removal (right).

3.4 Stage 4: Calculating feasibility and resources of different fabrication techniques

At the core of our suggested method, we use computationally efficient mathematical formulas to estimate the fabrication resources needed for each fabrication technique. The algorithms presented below were either developed for this research or adapted from existing research and practice. They have never been used together for estimating machining resources by means of a point sample.

3.4.1 "2.5 axis" cutting and assembly

The feasibility of continuous bent sheet material molds is based on two material related properties: bendability and deformability. Bendability is defined here as the minimal radius in which a material can bend in one direction. For example, it is known that laminated plywood can be bent to a radius about two hundred times its thickness (Certiwood 2012). Using our already calculated curvature radius it is possible to discover areas of a geometry which are too curved to be bent with plywood of a given thickness. Deformability is defined here as a material's ability to stretch and compress so that it can achieve bending in two directions simultaneously. Experiments suggest that the local Gaussian curvature of sampling points on the surface is a strong indicator of the ability to deform a flat sheet into its shape. In our experience , there are limits to both the Gaussian and the primary local curvatures, beyond which deformations are difficult to achieve. The precise values of these limits are related to material properties such un-isotropic stretching and bending, as well as layering and thickness. They are beyond the scope of the current work and will be further explored in the future.

After combing bendability and deformability to obtain a feasibility indicator, the mold is unrolled and nested onto material sheets. Efficient, quasi-optimal nesting is achieved using the "Pack-Rat" GH plugin (Chatzikonstantinou 2017), a rectangle packing algorithm inspired by the extreme points concept introduced in (Crainic, Perboli, and Tadei 2008). Eq.1 shows how we approximate machining time(M), by multiplying the total length of the geometry's edges(E) by the number of machining passes necessary, which by default is set at the material thickness(T)/ tool diameter(D). Then, we multiply the tool path length by the appropriate feed rate(F), arriving at an approximation of the machining time for each sheet. The feed rate is calculated using information from a

commercial “feeds and speeds” calculator developed by www.CNCcookbook.com. This calculator provides a reliable, industry accepted standard based on material type, cut depth and width, and the tool diameter.

$$M_s = E_{mm} * \frac{T_{mm}}{D_{mm}} * F_{mm/s} \quad (1)$$

3.4.2 “3 axis” milling of EPS molds

For 3-axis milling spindle accessibility, we use the Z access as an initial indication and double check accessible areas with a simplified spindle model. The tooltip access calculation uses the principle NURBS curvature radius of the surface at a given point to predict the maximal diameter of tool which can fit into concave areas. In the case of compound poly-surfaces, the curvature based calculation does not suffice as an access measure as it cannot detect tooltip clashes along the intersection curves. For points on a poly-surface, the tooltip is positioned along the normal of the surface or the mesh before checking for collisions. However, this will not work in points on intersections between two surfaces, as the normal in these cases is arbitrarily related to either one of the surfaces, or averaged between them. Here, searching along the normals of all of the neighboring vertices in the half edge mesh structure will guaranty finding the right tool tip position without expensive trial and error.

The milling time calculation is separated into two parts. For the rouging tool path length, we contour the milling area derived in the mold preparation stage and divide the area of the contours by the roughing tool diameter. We can do this because for 2D shapes almost all toolpaths have approximately the same length (B. H. Kim and Choi 2002).

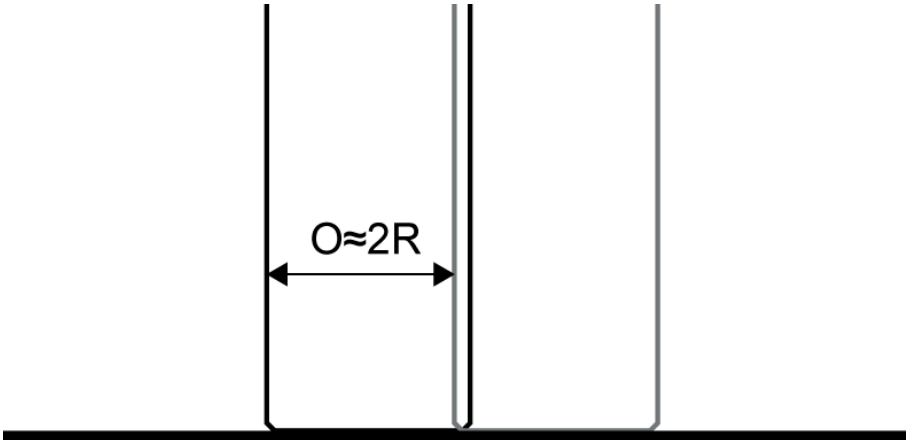


Figure 3: Flat end finishing in 3 axis milling.

For finishing flat, horizontal regions we use the full practical distance allowed by a flat tool, as shown in figure 3. To ensure overlap, the practical offset step is slightly smaller then twice the tool radius (R). This gives us the exact shape where the scallop height h=0.

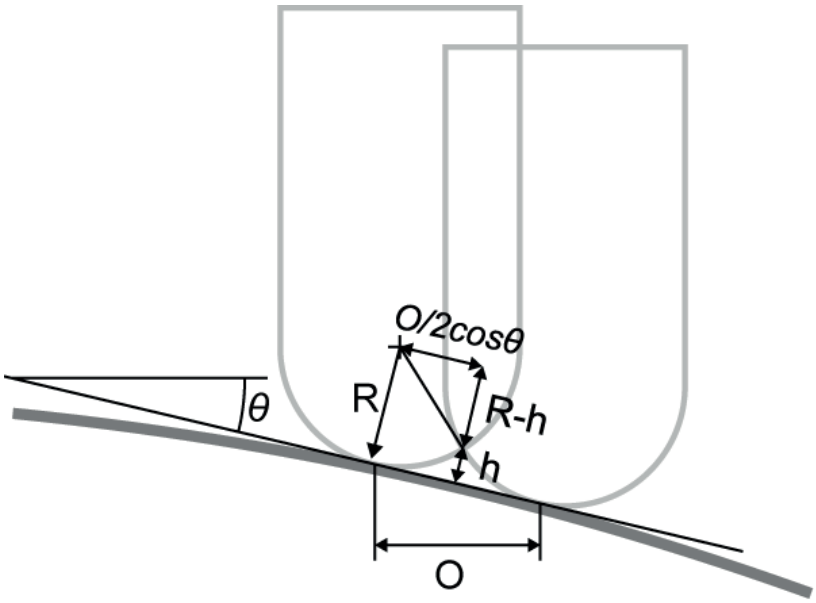


Figure 4: Offset step (O) in ball end 3-axis finishing.

For any other type of surface, we use ball end finishing. As shown in figure 4. the offset step in this case is determined by three factors: the tool radius (R) which we already calculated, the surface inclination (θ), and the required precision or scallop height (h) which is set by the user. The offset step can be developed from a formula suggested by (Han and Yang 1999), taking into account that (h2-->0):

$$O = \cos\theta \times 2\sqrt{R^2 - (R-h)^2} = \cos\theta \times 2\sqrt{2Rh + h^2} \approx \cos\theta \times 2\sqrt{2Rh} \quad (2)$$

After calculating each sampling point on the surface, we divide the vertex influence area by the offset, summing the results to estimate the minimal Iso-scallop tool path length. Now, we can again use information from the feeds and speeds calculator to arrive at the machining time.

3.4.3 "5-7 Axis" milling of EPS molds

For more advanced milling setups (including robotic milling), we use the following: For accessibility calculations, our algorithm considers a radial

3D array of approaches to every point, until one is located, or all allowed approaches disproved. In areas where the access direction is not close to the surface normal, this is the most computationally expensive part of our algorithm.

For approximating machining time, we use the same type of roughing calculation we did for 3 axis milling. For finishing, we use tilted flat end milling, due to its ability to adapt the practical radius of the tool to the surface curvature. This type of milling, also called curvature matched machining, is more effective at surface finishing than ball end milling (Jensen 1993). Coupled with a gauging detection algorithm such as (Y. J. Kim et al. 2015) local gauging can be completely eliminated. For determining the path length of this type of milling, we calculate the local step offset in relation to the geometry of a minimally tilted flat end mill. As we are ready to accept lower precision in favor of near real time operation, we use the following abstractions in our calculations:

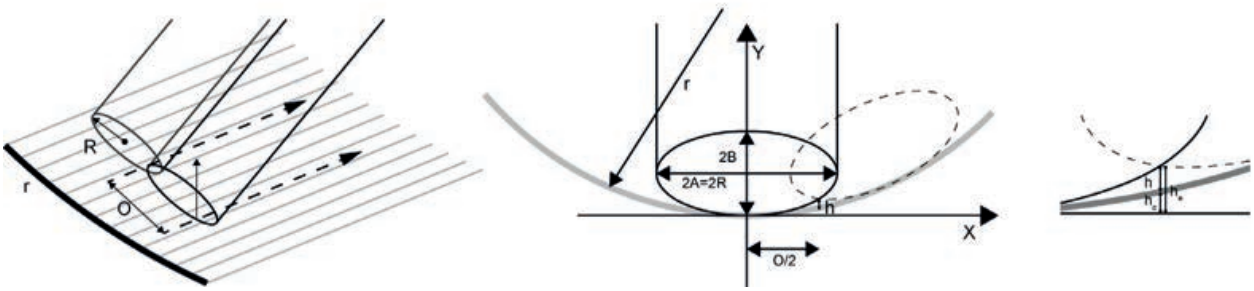


Figure 5: curvature matched milling of concave surfaces.

In concave areas we assume that the cutting direction is orthogonal to the largest principal curvature, because it is potentially more effective. Furthermore, the cutting tool is tilted along the same principal curvature direction to match the tool curvature to the surface curvature at the contact point. Figure 5 center shows a section orthogonal to the cutting plane thru the surface normal. (O) is the offset step, (h) the allowed scallop height, (R) is the tool radius, (A) and (B) ellipse constants, and (r) the normal curvature radius of the surface in the direction orthogonal to the tool path (i.e. figure 5 center). At the scallop point X=O/2, figure 5 right shows that h is roughly the difference between the distance of the tangent plane at the contact point and the cutter ellipse (he) and

the distance of the tangent plane and the surface (h_c). For the distance between the tangent plane and the ellipse $Y = h_e$, while having $A = R$:

$$(\frac{O/2}{R})^2 + (\frac{B - h_e}{B})^2 = 1 \quad (3)$$

For the distance between the tangent plane and the original surface, $A=B=r$, $Y=h_c$:

$$(\frac{O/2}{r})^2 + (\frac{h_c}{r} - 1)^2 = 1 \quad (4)$$

Solving, we find that:

$$h_e = B(1 + \sqrt{1 - \left(\frac{O}{2R}\right)^2}) \quad (5)$$

$$h_c = r(1 + \sqrt{1 - \left(\frac{O}{2r}\right)^2}) \quad (6)$$

Expanding $h = h_e - h_c$ and assuming $O/A << 1$ and $O/r << 1$, we find that the fourth order tailor expansion is:

$$h = h_e - h_c \approx \left(\frac{B}{8R^2} - \frac{1}{8r}\right) O^2 + \left(\frac{B}{128R^4} - \frac{1}{128r^3}\right) O^4 + \cdots \quad (7)$$

The quadratic term can be eliminated by choosing $B=R^2/r$. We are left with:

$$h = \frac{1}{128r}(\frac{1}{R^2} - \frac{1}{r^2})O^4 \quad (8)$$

And for $R << r$ we find that

$$O = \sqrt[4]{128hrR^2} \quad (9)$$

And since we must have $O < 2R$

$$O = \min\left(2R, 2\sqrt[4]{8hrR^2}\right) \quad (10)$$

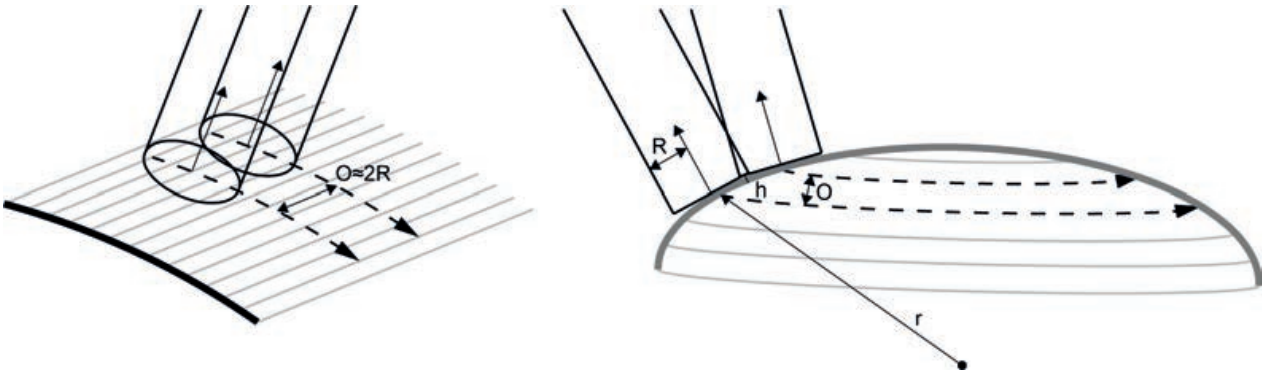


Figure 6: curvature matched milling of convex surfaces.

For parabolic single curved areas, we assume the milling direction is orthogonal to the vanishing principle curvature direction and use the full practical distance allowed by the tool, achieving $h=0$, see figure 6 left. In convex double curved surfaces, we assume that the milling direction is orthogonal to the principal curvature direction. Hence, we use the same development we used in ball end milling of flat surface (eq. 2), only now we exchange the role of the tool and the surface as shown in figure 6 right, arriving at:

$$O = \min(2R, 2\sqrt{2rh}) \tag{11}$$

3.4 Robotic hot-wire cutting

For determining hot-wire accessibility we begin by disqualifying points with a concave double curvature, as a straight wire cannot access them. Then, we rotate a line around the remaining sampling points until access is discovered. We use the secondary curvature direction calculated in the analysis stage as a starting direction, as experiments have shown that this saves us significant search time. For path length approximation, we use a piecewise developable approximation of the original geometry, which is based on (Elber and Fish 1997). The resulting surfaces are extended towards the material boundaries and their length multiplied by the hotwire speed. The algorithm used is based on the UV directions of the surface and does not take into account the access direction. In the future, a ruled surface approximation method which takes accessibility into account should be developed.

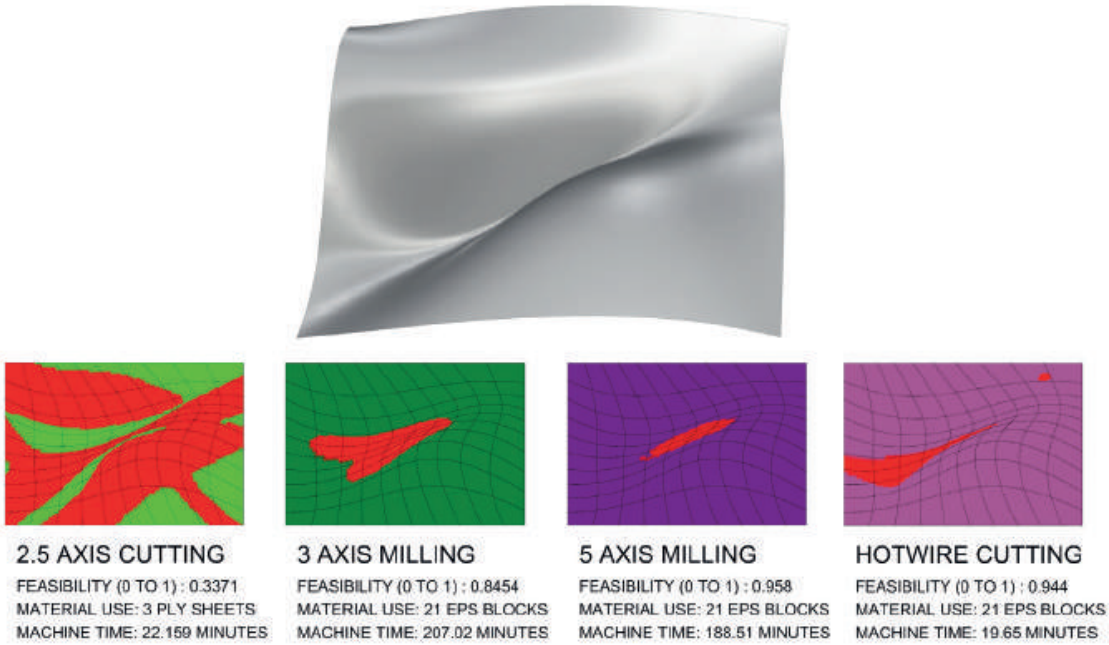


Figure 7: Analysis result, input geometry (top) and evaluation of different fabrication techniques (bottom).

3.5 Stage 5: Result display

Figure 7 shows the results of applying our method to a complex 3D surface. The graphic component of the analysis shows unfeasible areas in red, while other colors denote feasible areas, according to the fabrication technique. Each graphic is accompanied by 3 numeric scores representing feasibility, material use and machine time. Running on a regular laptop, the results in figure 7 were achieved in just under a second for the surface, sampled at 1500 points. The speed of the analysis allows the designer to interactively adjust the geometry until it satisfies the requirements of the chosen fabrication strategy.

4. Result calibration and case study

To calibrate our predictions, we prepared a large amount of samples for milling in a professional CAM software used by the 3 axis CNC in our lab. So far, we have seen a good correlation between the predicted results and results obtained by a full scale CAM machining simulation . Figure 8. Left shows a series of single surfaces which were prepared for milling using ALPHACAM 2014 and compared to the predictions of our method.

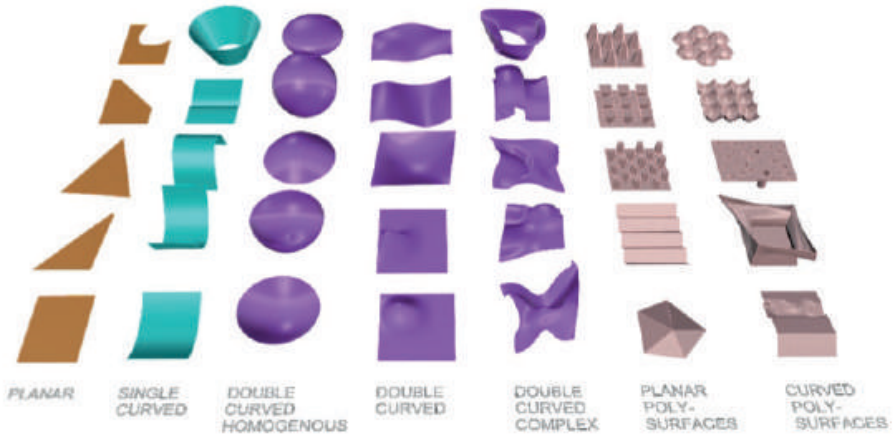


Figure 8: Some of the shapes used for testing our method: planar and single curved surfaces(left), double curved surfaces(center), and poly-surfaces (right).

In our benchmark test, the diferences between the estimated machine time and the actual milling time ranged between -/+ 5% with an average of 1.1%. The prediction time of our method was between one to three seconds for all the single surface geomtries (for a ~1500 point sample on the surface). In comparison, preparing the geometry for fabrication in the CAM software and running a machining simulation averaged at 35 minutes for the same geometries.

In a different set of tests, a series of complex poly-surfaces, shown on the right of figure 8, were analysed with our method. These predictions took slighlty longer (3–5 seconds for a 1 000 point sampling size) but achieved similar precision to the single surfaces in terms of fabrication time. Additionally, our feasbilty prediction was correct for over 99% of the analysed points, compared to both human judgement and the predictions of proffessional CAM software.

In the future, we need to further test our method by comparing its predictions with data obtained from the robotic fabrication setup currently being installed in our lab, and with data from commercial mold fabrication companies.

We have also tested a possible use for our method by manually rationalizing complex façade tiles, designed to increase turbulence on the building surface according to (Grobman and Elimelech 2015; Hershcovich et al. 2017). As shown in figure 9, we used our method to analyze the original design, indicating areas which could not be fabricated using a 3 axis CNC mill with an 8 mm round finishing tool. Manually adjusting these areas eliminated the problems and shortened the fabrication

time by 30%. We milled the mold for this prototype, cast concrete into it and demolded the result to demonstrate an entire design fabrication process assisted by our method.

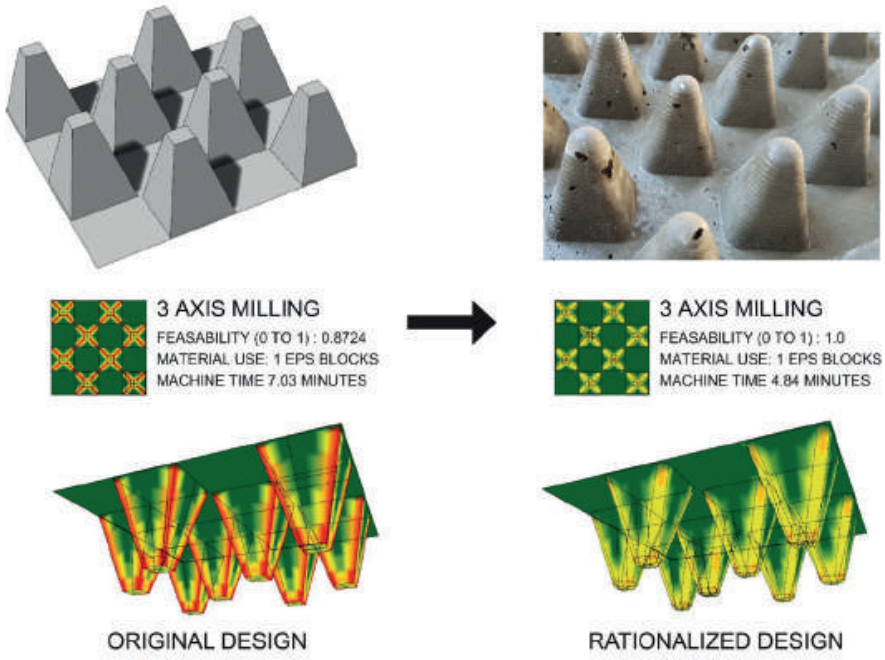


Figure 9: Manual Rationalization process – original geometry (left), rationalized and fabricated result on (right).

5. Future development: Fitness criterion for optimization

To study the potential of our method as an optimization criterion, we used it to adapt a complex geometry to different fabrication techniques. We used two different "black box" solvers available for the RH/GH environment: "OPOSSUM", a machine learning, model-based solver (Wortmann 2017), and "GOAT", a gradient-based solver developed by www.rechenraum.com. These general solvers are ideally suited for use with our method as they rely on a single fitness criterion – any one the feasibility or resource score evaluated. Using general purpose solvers demonstrated the flexibility of our method, which can be easily adapted to any type of optimization method available to the designer.

In our case study we balanced the feasibility score for the different techniques with the change from the original surface, measured as the average distance from the original surface, normalized by the diagonal

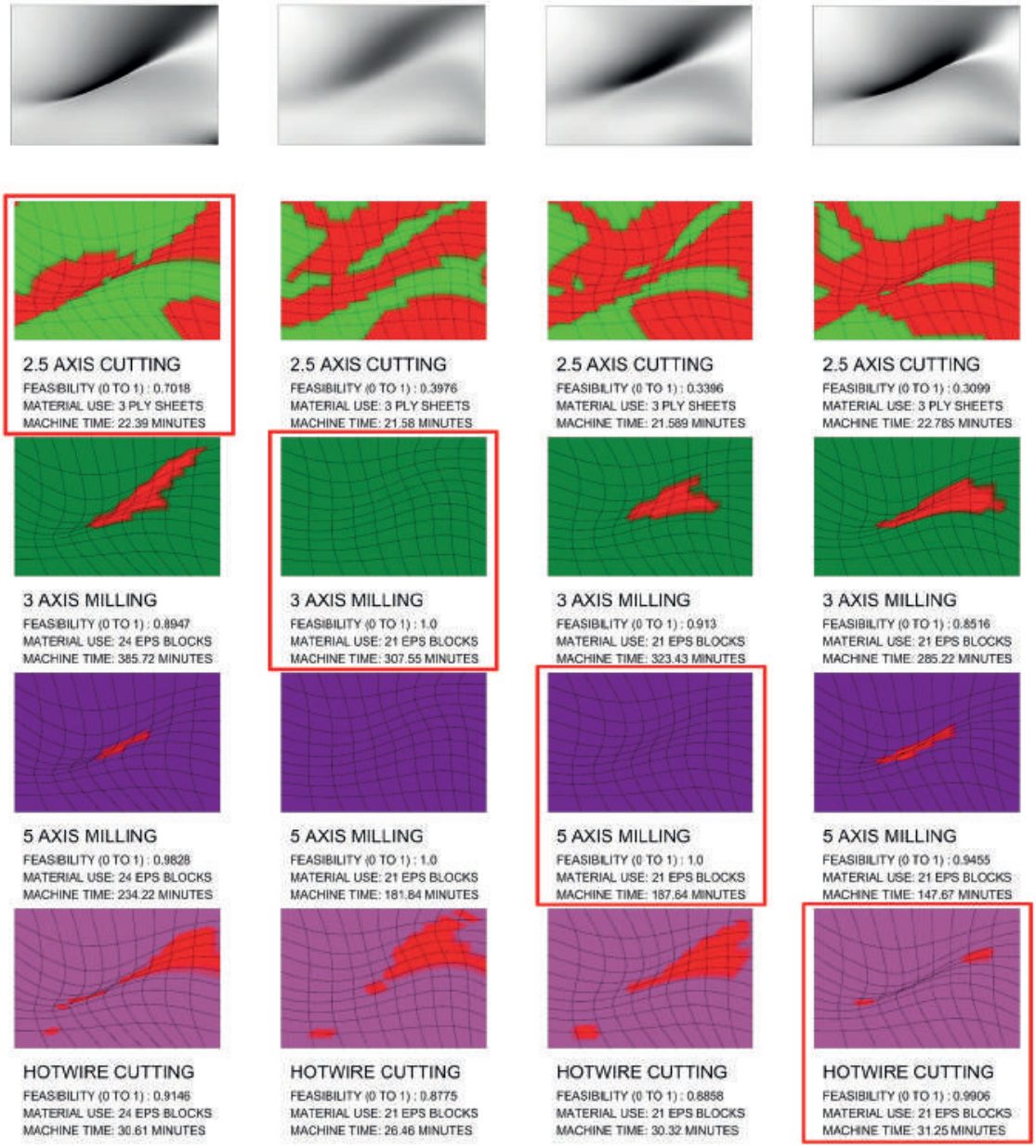


Figure 10: Optimization case study, each column is optimized towards a different fabrication technique.

size. Thinking of the panels as a part of a series, the borders of the geometries were further constrained to prevent any movement outside of the Z axis.

We conducted thirty optimization attempts on five different surfaces, limiting the running time to 10 minutes and the solvers to 1 000 iterations. In these tests we were able to improve the feasibility score by an average of 8% , while the change in in the geometry was kept at 1%. Figure 10 shows the best results of the different optimization runs performed on the surface shown in Figure 7. Each column represents a different optimization, targeted at a specific fabrication technique.

6. Conclusions

In this paper, we have outlined the development of a unique analysis method, aimed at evaluating the fabrication of molds for concrete façade elements. The main contribution of the suggested method is in providing architects with fabrication related information which was never made available to them. This will improve their digital craftsmanship (McCullough 1996), helping them to achieve more complex designs within practical limitations. The proposed method can be used to determine the best fabrication technique for a given design. Alternately, it can be used as an expert system, helping architects as well as fabricators to rationalize geometry towards a specific fabrication technique. We have also shown that due to its relatively high speed, our method has the potential to be used as the basis for an optimization mechanism which automatically adjusts geometries towards specific fabrication goals.

References

Andersen, Thomas Juul, Leal Da Silva, Ricardo Wilson, and Lars Nyholm Thrane. 2016. "Lessons from the TailorCrete Project." *Concrete International* Vol. 38 (Issue 3): p54–61.

Austern, Guy, Isaac Guedi Capeluto, and Yasha Jacob Grobman. 2018. "Rationalization Methods in Computer Aided Fabrication: A Critical Review." *Automation in Construction* 90: 281–93. https://doi.org/10.1016/j.autcon.2017.12.027.

Brander, David, Andreas Bærentzen, Kenn Clausen, Ann-Sofie Fisker, Jens Gravesen, Morten N. Lund, Toke B. Nørbjerg, Kasper Steenstrup, and Asbjørn Søndergaard. 2016. "Designing for Hot-Blade Cutting: Geometric Approaches for High-Speed Manufacturing of Doubly-Curved Architectural Surfaces." In *Advances in Architectural Geometry 2016*. Zurich: vdf Hochschulverlag AG. https://www.researchgate.net/publication/308074868_Designing_for_hot-blade_cutting.

Campagna, Swen, Leif Kobbelt, and Hans-Peter Seidel. 1998. "Directed Edges—A Scalable Representation for Triangle Meshes." *Journal of Graphics Tools* 3 (4): 1–11.

Ceccato, Cristiano. 2011. "GALAXY SOHO - Large Scale Cladding Construction in CHina." In Fabricate: *Making Digital Architecture,* edited by Ruairi Glynn and Bob Sheil. The Bartlett School of Architecture, UCL, London: Riverside Architectural Press. http://www.jstor.org/stable/j.ctt1tp3c6d.32.

Celento, David. 2010. "Innovate or Perish: New Tecnnologies and Architecture's Future." In *Fabricating Architecture, Selected Readings in Digital Design and Manufacturing*, edited by Robert Corser. New York: Princeton Architectural Press.

Certiwood. 2012. "Plywood Design Fundamentals." http://www.canply.org/pdf/main/plywood_designfund.pdf.

Chatzikonstantinou, Yannis. 2017. "Packrat." Yannis Chatzikonstantinou. 2017. http://yconst.com/software/packrat.

Crainic, Teodor Gabriel, Guido Perboli, and Roberto Tadei. 2008. "Extreme Point-Based Heuristics for Three-Dimensional Bin Packing." *Informs Journal on Computing* 20 (3): 368–384.

CSI. 2009. "The Cement Sustainability Initiative - Recycling Concrete." World Business Council for Sustainable Development. http://www.wbcsdcement.org/pdf/CSI-RecyclingConcrete-FullReport.pdf.

Deng, Bailin, Sofien Bouaziz, Mario Deuss, Alexandre Kaspar, Yuliy Schwartzburg, and Mark Pauly. 2015. "Interactive Design Exploration for Constrained Meshes." *Computer-Aided Design* 61: 13–23.

Dritsas, Stylianos, Raul Kalvo, and Andres Sevtsuk. 2013. "Packing Optimization for Digital Fabrication." In *Computation and Performance – Proceedings of the 31st ECAADe Conference*, 1:655–64. Faculty of Architecture, Delft University of Technology, Delft, The Netherlands. http://cumincad.scix.net/cgi-bin/works/Show?_id=ecaade2013_011&sort=DEFAULT&search=Dritsas%2c%20S%2e%2c%20Kalvo%2c%20R%2e%20%20Sevtsuk%2c%20A%2e%20&hits=1794.

Eigensatz, Michael, Mario Deuss, Alexander Schiftner, Martin Kilian, Niloy J. Mitra, Helmut Pottmann, and Mark Pauly. 2010. "Case Studies in Cost-Optimized Paneling of Architectural Freeform Surfaces." In Advances in *Architectural Geometry 2010*, edited by Cristiano Ceccato, Lars Hesselgren, Mark Pauly, Helmut Pottmann, and Johannes Wallner, 49–72. Springer, Vienna. http://link.springer.com/chapter/10.1007/978-3-7091-0309-8_4.

Elber, Gershon. 2016. "The IRIT Modeling Environment - Home Page." 2016. http://www.cs.technion.ac.il/~irit/.

Elber, Gershon, and Russ Fish. 1997. "5-Axis Freeform Surface Milling Using Piecewise Ruled Surface Approximation." *Journal of Manufacturing Science and Engineering* 119 (3): 383–87. https://doi.org/10.1115/1.2831117.

Fischer, Thomas. 2012. "Geometry Rationalization for Non-Standard Architecture." *Architecture Science* 5: 25–47. https://www.researchgate.net/publication/273320668_Geometry_Rationalization_for_Non-Standard_Architecture.

Flöry, Simon, Yukie Nagai, Florin Isvoranu, Helmut Pottmann, and Johannes Wallner. 2013. "Ruled Free Forms." In *Advances in Architectural Geometry* 2012, 57–66. Springer, Vienna. https://doi.org/10.1007/978-3-7091-1251-9_4.

Grobman, Yasha J, and Yosie Elimelech. 2015. "Microclimate on Building Envelopes: Testing Geometry Manipulations as an Approach for Increasing Building Envelopes' Thermal Performance." *Architectural Science Review*, 1–10.

Han, Z., and D.c.h. Yang. 1999. "Iso-Phote Based Tool-Path Generation for Machining Free-Form Surfaces." *Journal of Manufacturing Science and Engineering, Transactions of the ASME* 121 (4): 656–64.

Hawkins, Will J., Michael Herrmann, Tim J. Ibell, Benjamin Kromoser, Alexander Michaelski, John J. Orr, Remo Pedreschi, et al. 2017. "Flexible Formwork Technologies – a State of the Art Review." *Structural Concrete* 17 (6): 911–35. https://doi.org/10.1002/suco.201600117.

Hershcovich, C, R van Hout, V Rinsky, M Laufer, and Yasha Grobman. 2017. "Microclimate on Building Envelopes: Wind Tunnel and Computational Fluid Dynamic Analysis of Basic and Complex Geometries." In *2017 Proceedings of the Symposium on Simulation for Architecture and Urban Design*. Toronto. http://www.simaud.org/proceedings/.

Jensen, Charles Gregory. 1993. "Analysis and Synthesis of Multi-Axis Sculptured Surface Machining." https://docs.lib.purdue.edu/dissertations/AAI9403717.

Khoshnevis, Behrokh. 2004. "Automated Construction by Contour Crafting – Related Robotics and Information Technologies." *Automation in Construction*, The best of ISARC 2002, 13 (1): 5–19. http://www.sciencedirect.com/science/article/pii/S0926580503000736.

Kilian, Axel. 2006. "Design Innovation through Constraint Modeling." *International Journal of Architectural Computing* 4 (1): 87–105. https://doi.org/10.1260/147807706777008993.

Kim, Bo H., and Byoung K. Choi. 2002. "Machining Efficiency Comparison Direction-Parallel Tool Path with Contour-Parallel Tool Path." *Computer-Aided Design* 34 (2): 89–95. https://doi.org/10.1016/S0010-4485(00)00139-1.

Kim, Yong Joon, Gershon Elber, Michael Barto , and Helmut Pottmann. 2015. "Precise Gouging-Free Tool Orientations for 5-Axis CNC Machining." *Computer-Aided Design* 58: 220–229. https://doi.org/10.1016/j.cad.2014.08.010.

Labonnote, Nathalie, Anders Rønnquist, Bendik Manum, and Petra Rüther. 2016. "Additive Construction: State-of-the-Art, Challenges and Opportunities." *Automation in Construction* 72 (December): 347–66. https://doi.org/10.1016/j.autcon.2016.08.026.

Lim, S., R. A. Buswell, T. T. Le, S. A. Austin, A. G. F. Gibb, and T. Thorpe. 2012. "Developments in Construction-Scale Additive Manufacturing Processes." *Automation in Construction* 21 (January): 262–68.

Lindsey, Bruce. 2001. *Digital Gehry*. 1st ed. Birkhäuser Basel.

Liu, C. Richard, and Xiaoping Yang. 2001. "Design for Manufacturing." In *Handbook of Industrial Engineering*, edited by Gavriel Salvendy, 1311–31. John Wiley & Sons, Inc. https://doi.org/10.1002/9780470172339.ch50.

Louth, Henry, David Reeves, Shajay Bhooshan, Patrik Schumacher, and Benjamin S Koren. 2017. "A Prefabricated Dining Pavilion: Using Structural Skeletons, Developable Offset Meshes and Kerf-Cut Bent Sheet Materials." In *Fabricate 2017: Rethinking Design and Construction*, edited by Achim Menges, Bob Sheil, Ruairi Glynn, and Marilena Skavara, 58–67. Stuttgart, Germany: UCL Press. http://www.jstor.org/stable/j.ctt1n7qkg7.12.

Manahl, Markus, Milena Stavric, and Albert Wiltsche. 2012. "Ornamental Discretisation of Free-Form Surfaces: Developing Digital Tools to Integrate Design Rationalisation with the Form Finding Process." *International Journal of Architectural Computing* 10 (4): 595–612. https://doi.org/10.1260/1478-0771.10.4.595.

McCullough, Malcolm. 1996. *Abstracting Craft : The Practiced Digital Hand*. Cambridge, MA : MIT Press.

Piker, Daniel. 2017. *Plankton: A C# Half-Edge Mesh Data Structure, and Components for Using This in Grasshopper/Rhino*. C#. meshmash. https://github.com/meshmash/Plankton.

Pottmann, Helmut. 2010. "Architectural Geometry as Design Knowledge." *Architectural Design* 80 (4): 72–77. https://doi.org/10.1002/ad.1109.

Pottmann, Helmut, Michael Eigensatz, Amir Vaxman, and Johannes Wallner. 2015. "Architectural Geometry." *Computers & Graphics* 47: 145–164. https://doi.org/10.1016/j.cag.2014.11.002.

Scheurer, Fabian. 2010. "Materialising Complexity." *Architectural Design* 80 (4): 86–93. https://doi.org/10.1002/ad.1111.

Sondergaard, Asbjorn, and Jelle Feringa. 2017. "SCALING ARCHITECTURAL ROBOTICS CONSTRUCTION OF THE KIRK KAPITAL HEADQUARTERS." In *Fabricate 2017: Rethinking Design and Construction*, edited by Achim Menges, Bob Sheil, Ruairi Glynn, and Marilena Skavara. Stuttgart, Germany: UCL unversity press.

Tang, Chengcheng, Xiang Sun, Alexandra Gomes, Johannes Wallner, and Helmut Pottmann. 2014. "Form-Finding with Polyhedral Meshes Made Simple." *ACM Transactions on Graphics* 33 (4): 1–9. https://doi.org/10.1145/2601097.2601213.

Whitehead, Hugh. 2005. “Laws of Form.” In *Architecture in the Digital Age: Design and Manufacturing,* edited by Branko Kolarevic, New Ed edition, 30–54. New York: Taylor & Francis.

Wortmann, Thomas. 2017. “Opossum: Introducing and Evaluating a Model-Based Optimization Tool for Grasshopper.” In *Proceedings of the 22nd International Conference of the Association for Computer-Aided Architectural Design Research in Asia,* edited by P. Janssen, P. Loh, A. Raonic, and M. A. Schnabel. Hong Kong. http://papers.cumincad.org/data/works/att/caadria2017_124.pdf.

Aligning principal stress and curvature directions

Davide Pellis, Helmut Pottmann

Davide Pellis
davidepellis@gmail.com
Vienna University of Technology, Austria

Helmut Pottmann
pottmann@geometrie.tuwien.ac.at
Vienna University of Technology, Austria

Keywords:
Architectural geometry, computational design, freeform architecture, principal meshes in equilibrium, structural optimization

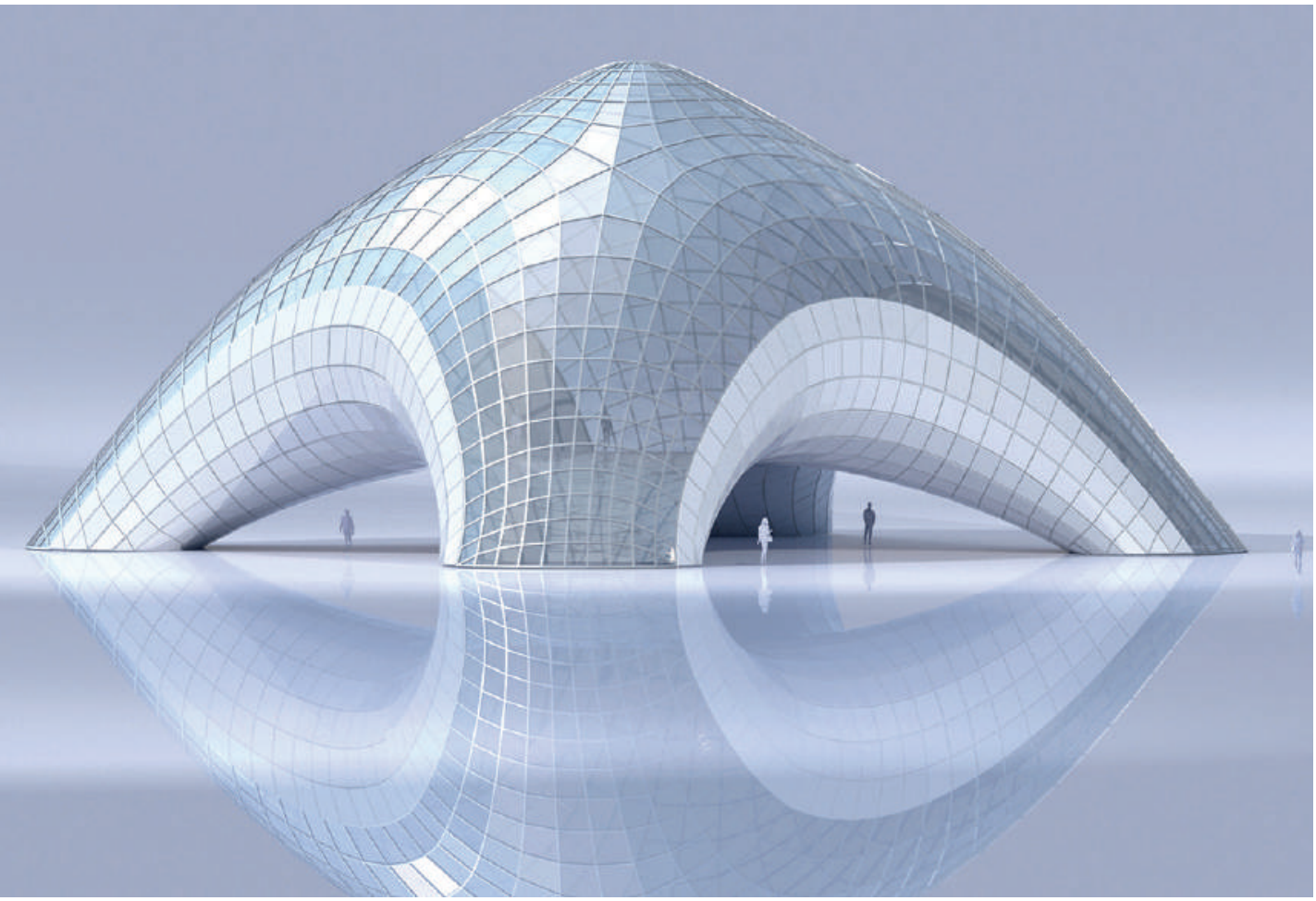
Abstract

Designing freeform architectural surfaces with due regard to economic and feasibility factors is a challenging task. Rationalizing such surfaces by means of quadrilateral meshes following principal curvature lines has proven to be beneficial for manufacturing reasons, such as planar cladding panels and simplified substructure connections. On the other hand, for structural efficiency, it is convenient to ensure static equilibrium in the load bearing structure through axial forces only. It turns out that both of these goals can be reached for surfaces in membrane equilibrium where principal stress and curvature directions coincide. In this paper, we present a method for the optimization of a given shape towards stress and curvature alignment, within a workflow for the design of principal meshes in equilibrium. Our method can be applied to shapes without any geometric or topological limitation.

1. Introduction

Motivation. Principal meshes are discrete versions of principal curvature parametrizations of surfaces. This kind of meshes is particularly suitable in architecture for several reasons. First of all, faces are planar and then easily manufacturable. Moreover, the network lines are as orthogonal as possible and admit the disposition of prismatic beams that meet in the nodes with minimized geometric torsion, reducing significantly the complexity of connections. On the other hand, the edges of these meshes are charged to bear the loads within the structure. It is well known that the most effcient manner of bearing loads in a framework is through strictly axial forces. This allows the beam cross section to be used to the highest capacity and at the same time to offer the highest stiffness. Principal meshes in axial force equilibrium provide an appealing solution for the discretization of freeform architectural surfaces. However, principality and equilibrium of quad meshes turn out to be often conflicting goals. It is of interest then to provide computational tools that enable to embody geometric and static optimization since the earliest design stages.

Previous work. A significant step in mesh optimization for equilibrium comes from thrust network analysis, introduced by Block and Ochsendorf (2007). Vouga et al. (2012) provide a differential geometric



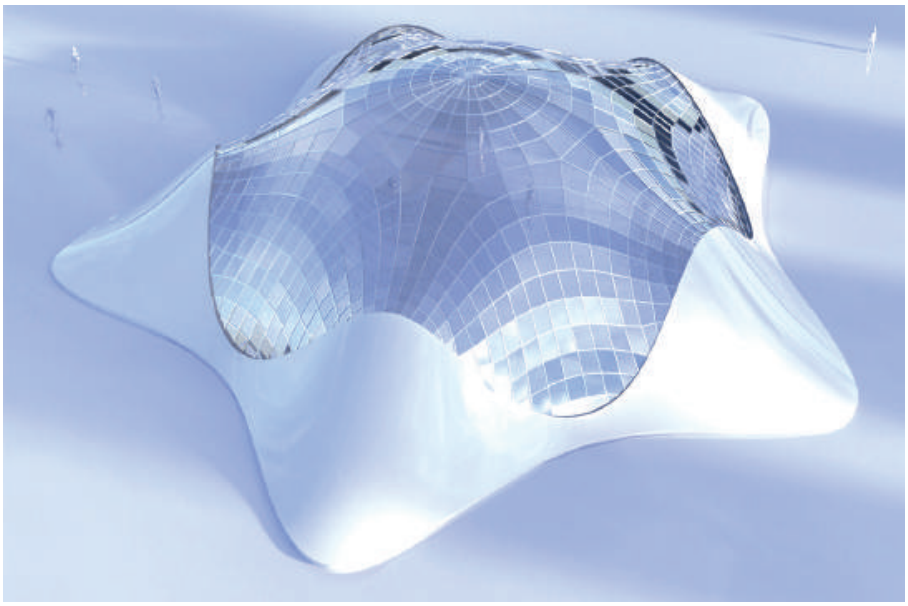


Figure 1: Architectural surfaces discretized with principal meshes in equilibrium, achieved thanks to stress and curvature alignment. Cladding can be realized with flat panels, and the substructure with prismatic beams and torsion-free nodes. At the same time, structural bending effects are minimized. The top mesh discretizes a non-height field shape.

understanding of this approach and use it for the design of planar quad meshes in equilibrium. An effcient optimization of quad meshes for equilibrium and face planarity is provided by Tang et al. (2014), but the success of this method is strongly dependent on the initial mesh connectivity. Schiftner and Balzer (2010) propose a method for planar quad-remeshing of given surfaces, initialized by principal stress lines. However, the effectiveness of this method is limited, since for a general surface, planarity of quads and the alignment with principal stress directions are often conflicting goals. A first attempt to directly design principal meshes in equilibrium was made by Sun (2016), fixing the mesh combinatorics in advance. Unfortunately, this approach rarely yields good convergence of optimization.

The design of principal meshes in equilibrium is addressed in (Kilian et al., 2017), thanks to the alignment of principal curvature and stress directions and a subsequent quad-remeshing. However, the applicability of this method is restricted to height field shapes. We refer to the latter paper for further literature. For an introduction to principal meshes in architecture, we refer to (Pottmann et al., 2015).

Our contribution. It turns out that principal meshes in equilibrium are a discretization of membrane surfaces with coincident stress and curvature directions. In this paper, we propose a method for the optimization of a given shape towards stress and curvature alignment, within a workflow for the design of principal meshes in equilibrium. Relying on a discrete-continuous analog, we discretize a given surface with a triangular mesh, enforcing the equilibrium on its edges, and we estimate an equivalent membrane stress and curvature. We overcome in this way the shape limitations of (Kilian et al., 2017). Once the shape is optimized for stress and curvature alignment, we generate a quad mesh along the resulting directions that is post-optimized for force equilibrium and face planarity. Thanks to our initialization, we can expect convergence with minimal changes. It is noteworthy that mesh connectivity and geometry are both part of our solution.

2. Membranes and gridshells

This paper deals with meshes in equilibrium under axial forces and subject to vertical loads. We consider self-weight, dead, and static live loads

lumped in forces and applied in the vertices. The resulting framework structure is a *gridshell truss:* a system of straight beams, with axes corresponding to the edges of the mesh, connected together and to the supports with frictionless pin-joints. In this paper, we refer to this model as *gridshell*. We point out here that this kind of structure, depending on its geometry, connectivity and support conditions, might be a mechanism in equilibrium. However, even if in an actual gridshell the nodes are manufactured as rigid joints for stability and safety reasons, the use of a truss model in the design stage is strongly beneficial for minimizing bending effects.

Let us consider now a refinement process that increases the density of a gridshell truss. From a mechanical point of view, at the limit of refinement the gridshell will tend to a *membrane*: a surface-like continuum that cannot support out of plane bending, and with mechanical properties derived at each point from the thickness in the normal direction. At the same time, the axial forces in the beams will tend to the *membrane stress*. For a detailed description of gridshells approaching membranes, we refer to (Mitchell, 2013).

In this paper, we rely on a discrete-continuous analog based on this refinement process to describe principal meshes in equilibrium. In the following of this section, we introduce the computational setting of continuous and discrete equilibrium, namely membranes and gridshells.

2.1 Membrane equilibrium

Let us consider a membrane $\mathscr{M}$ given by a regular surface. Away from points with a vertical tangent plane, we parametrize the surface locally as a height function $\mathscr{M}(x,y) = (x,y,z\,(x,y))$. If we consider only vertical loads, it is convenient to express equilibrium in the global coordinate system (x,y,z), with a vertical z axis. Let $\bar{S}$ be the tensor representing the xy projection of the membrane stresses. The horizontal and vertical equilibrium, respectively, are expressed by

$$\mathrm{div}(\bar{S}) = \mathbf{0}, \quad \mathrm{div}(\bar{S}\nabla z) = \rho.$$

Here divergence of a matrix is applied to its columns, and $\rho\,(x,y)$ is the vertical load per unit xy area (Angelillo and Fortunato, 2004; Vouga et al., 2012).

The horizontal component of the equilibrium can be expressed by the existence of an Airy stress potential $\phi(x,y)$. The stress tensor $\bar{S}$ is then given by the adjoint Hessian of ϕ:

$$\bar{S} = \widetilde{\nabla^2}\phi = \begin{pmatrix} \phi_{,yy} & -\phi_{,xy} \\ -\phi_{,xy} & \phi_{,xx} \end{pmatrix},$$

where with comma we denote partial derivatives, and with over-tilde the adjoint matrix operation. Let $\mathbb{I}$ be the first fundamental form of $\mathscr{M}(x,y)$. The *principal stresses* are then given by the eigenvalues of $\Delta^{1/2}\mathbb{I}^{-1}\nabla^2\phi$, where $\Delta = \det(\mathbb{I})$. The corresponding eigenvectors define the principal stress directions of $\mathscr{M}(x,y)$. For further details we refer to (Kilian et al., 2017).

We can consider the Airy stress potential as a surface $z = \phi(x,y)$ in *isotropic space*. This is a 3D space with a preferred direction along the z axis, and where distances are measured in the xy plane. The Hessian of ϕ plays here the role of shape operator, and its eigenvalues are the *isotropic principal curvatures*. These correspond to the eigenvalues of $\bar{S}$ along the swapped eigenvectors. For an introduction to isotropic geometry, we refer to (Pottmann and Liu, 2007).

2.2 Gridshell equilibrium

Let M be a three dimensional gridshell truss, with members corresponding to the edges of a mesh. We consider vertical loads applied in the vertices $\mathbf{v}_i = (x_i, y_i, z_i)$ and support conditions given along the boundary. The force $\mathbf{f}_{ij}$ exerted by the oriented bar $\mathbf{e}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ on the vertex $\mathbf{v}_i$ can be expressed as $w_{ij}(\mathbf{v}_i - \mathbf{v}_j)$, where w_{ij} is the axial force per unit bar length or *force density*, and where positive values of w_{ij} indicate compression. Let A_i be the area of influence at $\mathbf{v}_i$. If the system is in equilibrium, at each unsupported vertex $\mathbf{v}_i$ the horizontal and vertical force balance gives, respectively,

$$\sum_{j\sim i} w_{ij}(\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j) = \mathbf{0}, \quad \sum_{j\sim i} w_{ij}(z_i - z_j) = \rho_i A_i, \tag{1}$$

where $\bar{\mathbf{v}}_i$, $\bar{\mathbf{v}}_j$ are the xy projections of the points $\mathbf{v}_i$, $\mathbf{v}_j$, and ρ_i is the vertex-wise load per unit area. With $j \sim i$ we denote all the vertices j connected with the vertex i.

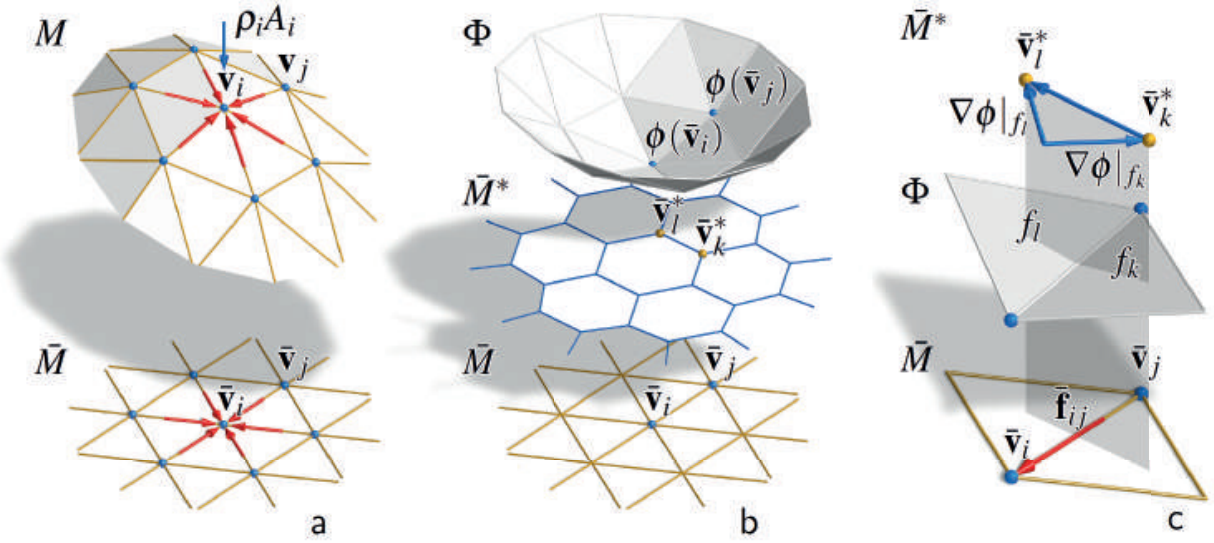


Figure 2: Gridshell equilibrium. **(a)** A portion of a gridshell M under vertical load and its projection into the xy plane $\bar{M}$. **(b)** The horizontal equilibrium yields a force dual mesh $\bar{M}^*$ with edges given by the forces acting in the correspondent primal edges of $\bar{M}$ rotated by 90 degrees, as shown by Maxwell (1872). We can construct an Airy polyhedron Φ with face gradients given by the coordinates of the corresponding dual vertex of $\bar{M}^*$. **(c)** By construction, the magnitudes of the forces in the bars $\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j$ are given by the isotropic angles between the adjacent faces on Φ. The isotropic angle can be seen as the change in slope between two faces of Φ when traversed orthogonally to $\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j$.

Let us now consider the projection of the structure in the xy plane, denoted as $\bar{M}$. Let $\bar{\mathbf{f}}_{ij}$ be the xy projections of the forces $\mathbf{f}_{ij}$. Since $\bar{M}$ is a 2D system in horizontal equilibrium under boundary loads (given by the xy projections of the support reactions), the forces $\bar{\mathbf{f}}_{ij}$ acting on each vertex $\bar{\mathbf{v}}_i$ can be arranged in a planar closed cycle. We can build thus a reciprocal diagram $\bar{M}^*$, combinatorially dual to $\bar{M}$, whose edges are given by the forces acting in the corresponding primal edge. For convenience, we represent this dual diagram rotated by 90 degrees clockwise in the xy plane, as shown in Figure 2b.

We can now construct a polyhedral stress potential $\Phi = (x, y, \phi(x, y))$, whose edges and vertices coincide in the xy projection to the primal truss $\bar{M}$, in the following way. Let us denote as f_k the faces of Φ, and let $\bar{\mathbf{v}}_k^* = (x_k^*, y_k^*)$ be the corresponding dual vertices of $\bar{M}$. Hence, each face f_k of Φ lies on a plane with gradient $\nabla\phi|_{f_k} = (x_k^*, y_k^*)$. The closure of

each face of $\bar{M}^{\star}$ ensures the closure of the polyhedron Φ turning around the corresponding primal vertex. This construction is uniquely defined up to vertical translations and shearing. For further details we refer to (Fraternali, 2010) and (Vouga et al., 2012).

Let f_k, f_l be the faces of Φ meeting at the oriented edge with projection $\bar{\mathbf{e}}_{ij}$, as shown in Figure 2c. The force $\bar{\mathbf{f}}_{ij}$, by construction, is given by $J(\nabla\phi|_{f_l} - \nabla\phi|_{f_k})$, where $J = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$ is the 90 degrees counter-clockwise rotation matrix in the xy plane. Denoting the xy unit edge vector as $\hat{\bar{\mathbf{e}}}_{ij} = (\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j)/\|\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j\|$, the quantity

$$\beta^{is}(\bar{\mathbf{e}}_{ij}) = J(\nabla\phi|_{f_l} - \nabla\phi|_{f_k}) \cdot \hat{\bar{\mathbf{e}}}_{ij} \tag{2}$$

is the signed isotropic angle between the faces f_l and f_k. Positive values of $\beta^{is}(\bar{\mathbf{e}}_{ij})$ indicate compression in the bar $\mathbf{e}_{ij}$. Note that $\beta^{is}(\bar{\mathbf{e}}_{ij}) = \beta^{is}(\bar{\mathbf{e}}_{ij})$.

3. Principal meshes in equilibrium

In this section we describe principal meshes in equilibrium under vertical loads. In Section 3.1, we show that these meshes are discretizations of special surfaces in membrane equilibrium, where principal directions of stress and curvature coincide. In Section 3.2, we first show how a membrane can be conveniently discretized with a triangular gridshell, enforcing the equilibrium on its edges. Then, we describe how to evaluate an equivalent stress tensor on a triangular gridshell and how to align the resultant principal directions with those of curvature. Finally, in Section 3.3, we outline our workflow to design principal meshes in equilibrium.

3.1 Principal stress and curvature alignment

Principal meshes are a discretization of the network of principal curvature lines of a continuous surface. A principal mesh in equilibrium, from a mechnanical point of view, is a gridshell with a quad combinatorics. At the limit of refinement, this gridshell will tend to a principal network of curves on a continuous surface in membrane equilibrium.

Let $\mathscr{M}(x,y)$ be a membrane under vertical load, parametrized as a height field surface over the xy plane, as described in Section 2.1. Let

us then consider the principal network of curves of $\mathscr{M}(x,y)$, defined at each point by two tangent vectors $\mathbf{a}_1$ and $\mathbf{a}_2$, and let $\bar{\mathbf{a}}_1$ and $\bar{\mathbf{a}}_2$ be their xy projections. We are now looking for simple conditions which express that the principal network is in equilibrium.

First, for principal curve networks, the vectors $\mathbf{a}_1$, $\mathbf{a}_2$ follow principal curvature directions. These directions are orthogonal on the surface. With $\mathbb{I}$ as first fundamental form of $\mathscr{M}(x,y)$, we can express the orthogonality condition of $\mathbf{a}_1$, $\mathbf{a}_2$ as

$$\bar{\mathbf{a}}_1^T \mathbb{I} \bar{\mathbf{a}}_2 = 0. \tag{3}$$

Secondly, as seen in Section 2.2, if a gridshell is in equilibrium under vertical loads, its xy projection must admit an Airy polyhedron Φ with planar faces. At the limit of refinement, the polyhedron Φ will tend to a continuous stress surface $z = \phi(x,y)$. For a quadrilateral gridshell, the corresponding Airy polyhedron is a quad mesh with planar faces. It is well known that a planar quad mesh, at the limit of refinement, will converge to a network of *conjugate curves* on a surface (Liu et al., 2006). We can then state the following condition: *a quad network on a surface is in horizontal equilibrium under vertical load if it is vertically projected onto a conjugate curve network on the corresponding Airy stress surface*. The condition for the directions $\mathbf{a}_1$, $\mathbf{a}_2$ to be vertically projected onto conjugate directions of $\phi(x,y)$ is expressed by

$$\bar{\mathbf{a}}_1^T \nabla^2 \phi \, \bar{\mathbf{a}}_2 = 0. \tag{4}$$

As shown in (Kilian et al., 2017), Equations (3) and (4) imply that the vectors $\bar{\mathbf{a}}_1, \bar{\mathbf{a}}_2$ are eigenvectors of $\mathbb{I}^{-1}\nabla^2\phi$. Since the principal stress directions on $\mathscr{M}(x,y)$ are given by the eigenvectors of $\Delta^{1/2}\mathbb{I}^{-1}\nabla^2\phi$, we can see that the only directions in horizontal equilibrium and orthogonal on the membrane are the principal stress directions. Therefore, we can state the following important fact:

Proposition 1. Principal meshes in equilibrium under vertical loads are discrete representations of membrane surfaces where principal stress and principal curvature directions agree. There, they follow these principal directions.

3.2 Estimating stress and curvature

As seen in Section 2.1, at each point of a membrane we find three unknown stress components and three equilibrium equations. Membranes are then statically determinate in the sense that, given the loads and the boundary tractions, the stress tensor is uniquely determined; the existence of a solution depends only on the membrane geometry. Let us now consider a triangular gridshell forming a closed polyhedron Γ of genus zero, and with loads applied in its nodes. Denoting by v its number of vertices and e its number of edges, Euler's formula shows that $3v = e + 6$. Since we have one unknown axial force per edge and three equilibrium equations per vertex, the solution is uniquely determined up to rigid body motion; the existence of the solution depends on the geometry of the polyhedron. The same is true for a portion of Γ, given the force reactions of the remaining part. Triangular gridshells can therefore reproduce the statical determinacy of membranes, see (Pavlovi, 1984). In the following, we express membrane behavior of a surface through the equilibrium of a gridshell triangulation.

In the continuous membrane, the projected stress tensor $\bar{S}$ and the isotropic shape operator $\nabla^2\phi$ are related by $\bar{S} = \widetilde{\nabla}^2\phi$. We are now searching for a discrete analog of the isotropic shape operator defined for triangle meshes, and at first look at the Euclidean counterpart. For that, we use the normal cycle approach by (Cohen- Steiner and Morvan, 2003). One computes an extended shape operator W (3×3 matrix with two eigenvectors in principal curvature direction and the third eigenvector, with eigenvalue close to zero, orthogonal to the surface) as follows. Selecting a vertex $\mathbf{v}_i$ and a surrounding region R_i of area A_i, W is found by

$$W(\mathbf{v}_i) = \frac{1}{A_i}\sum_{j\sim i}\beta(\mathbf{e}_{ij})\left\|\mathbf{e}_{ij}\cap R_i\right\|\hat{\mathbf{e}}_{ij}\hat{\mathbf{e}}_{ij}^T. \qquad (5)$$

Here $\beta(\mathbf{e}_{ij})$ is the signed Euclidean angle between the two normals of the faces adjacent to the edge $\mathbf{e}_{ij}$, $\mathbf{e}_{ij}\cap R_i$ is the portion of the edge $\mathbf{e}_{ij}$ intersecting the region R_i and $\hat{\mathbf{e}}_{ij}$ is the unit edge vector, given by $(\mathbf{v}_i - \mathbf{v}_j)/\left\|\mathbf{v}_i - \mathbf{v}_j\right\|$. The eigenvalues of $W(\mathbf{v}_i)$, associated with the two eigenvectors lying in the tangent plane at $\mathbf{v}_i$, will give an estimation of principal curvatures along the swapped tangent eigenvectors. To obtain a discrete isotropic shape operator, we have to replace Euclidean

quantities by isotropic ones. This means that lengths and areas are measured in the xy plane and the Euclidean angle $\beta(\mathbf{e}_{ij})$ is replaced by the signed isotropic angle $\beta^{is}(\bar{\mathbf{e}}_{ij})$, given by Equation (2). Setting $\bar{\mathbf{e}}_{ij} \cap \bar{R}_i = \bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j$, we can estimate the 2×2 adjoint Hessian of Φ at $\bar{\mathbf{v}}_i$ as

$$\widetilde{\nabla}^2 \phi(\bar{\mathbf{v}}_i) = \bar{S}(\bar{\mathbf{v}}_i) = \frac{1}{\bar{A}_i} \sum_{j \sim i} J(\nabla\phi|_{f_l} - \nabla\phi|_{f_k})(\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j)^T.$$

Observing that $J(\nabla\phi|_{f_l} - \nabla\phi|_{f_k}) = \bar{\mathbf{f}}_{ij} = w_{ij}(\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j)$, we can estimate the stress tensor directly through force densities as

$$\bar{S}(\bar{\mathbf{v}}_i) = \frac{1}{\bar{A}_i} \sum_{j \sim i} w_{ij}(\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j)(\bar{\mathbf{v}}_i - \bar{\mathbf{v}}_j)^T. \tag{6}$$

To estimate the principal curvature directions on the triangulated surface we use again Cohen-Steiner Equation (5). For sufficiently smooth meshes, we can make the approximation $\beta \approx \sin\beta$. With $\mathbf{n}_{f_k}$ and $\mathbf{n}_{f_l}$ as the unit normals of the left and right faces of the edge $\mathbf{e}_{ij}$, we can then estimate the 3×3 extended shape operator as

$$W(\mathbf{v}_i) = \frac{1}{A_i} \sum_{j \sim i} (\mathbf{n}_{f_l} \times \mathbf{n}_{f_k})(\mathbf{v}_i - \mathbf{v}_j)^T. \tag{7}$$

Let κ_1 and κ_2 be the eigenvalues of W corresponding to the two eigenvectors in the tangent plane of M. We can ensure the alignment of two vectors $\mathbf{a}_1, \mathbf{a}_2$ with principal directions at each vertex $\mathbf{v}_i$ by requiring

$$W\mathbf{a}_1 = \kappa_1 \mathbf{a}_1, \quad W\mathbf{a}_2 = \kappa_2 \mathbf{a}_2.$$

3.3 Workflow

We have now the elements to design principal meshes in equilibrium. In particular, we solve this problem: *Given an initial surface subject to gravitational load and its support conditions, find a quadrilateral mesh in force equilibrium with edges aligned along principal curvature directions that is close to the initial design surface*. Our procedure can be summarized in the following steps:

Step 1. Given an input surface as a triangular mesh and the support conditions, we optimize the mesh geometry in order to align the

equivalent stress and curvature directions as described in Section 3.2, while keeping the vertices as close as possible to the input shape. The development and implementation of this step is the main contribution of this paper.

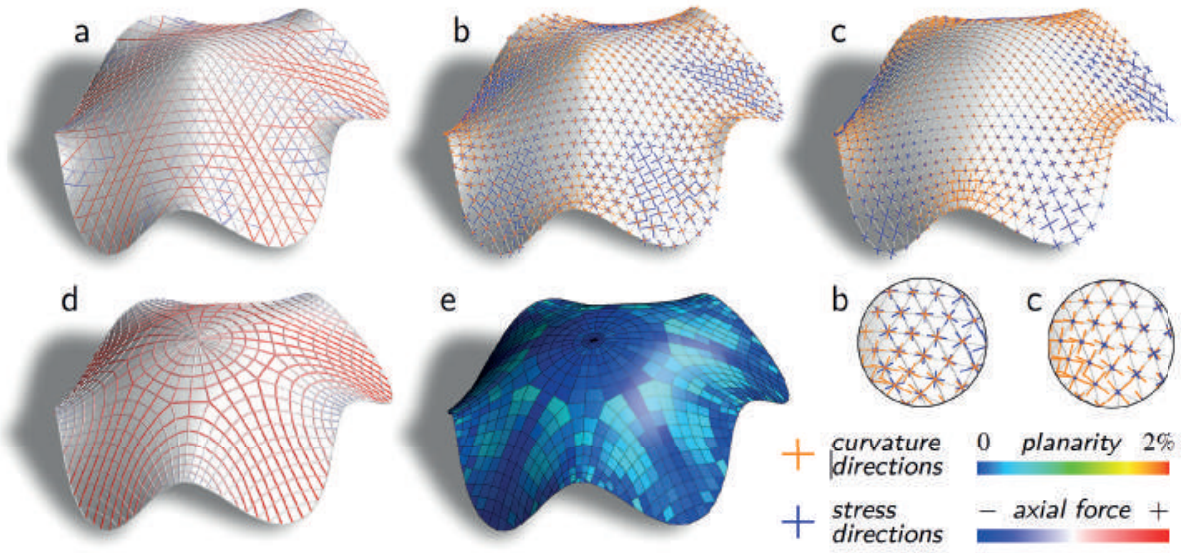


Figure 3 : Design workflow. (a) An initial shape is given as triangular mesh and the equilibrium is enforced on the edges. (b) The estimated curvature and stress directions, in general, are not aligned. (c) After our optimization, we reach the alignment with a change in the shape. (d) We remesh the resulting shape with mixed integer quadrangulation along the computed directions. After a post-optimization, the structure is in equilibrium under axial forces, (e) and panels are close to planar. According to a finite element analysis, the ratio of internal elastic work w_a due to axial forces in the final structure is 0.95. The stress and curvature directions are scaled according to their anisotropy, given by the difference between the two eigenvalues. A possible application of this design is depicted in Figures 1 and 7.

Step 2. We use the resulting directions as guide for a quadrilateral remeshing of the optimized mesh. At this purpose we use *mixed integer quadrangulation* proposed by (Bommes et al., 2009). In this step, the density of the mesh can be chosen according to fabrication and design considerations.

Step 3. The obtained quadrilateral mesh is subject to post-optimization for equilibrium and planarity of faces, while applying some fairness to the network curves to guarantee aesthetic quality. For this purpose, we use the method of (Tang et al., 2014). Thanks to *step 1,* we can expect convergence with minimized conflict between planarity and equilibrium.

4. Implementation

In this section we briefly describe the implementation of *step 1*, described in the workflow Section 3.3. Starting from a given triangular mesh M^0 with specified support conditions, we find a mesh M where principal stresses and principal curvature directions are aligned, as close as possible to M^0.

Main variables and constraints. For a mesh M^0 with v vertices and e edges, being s and c respectively the number of vertices that are mechanically supported, and fixed during the optimization, the main variables of the problem are:

» the position of the vertices $\mathbf{v}_i$ of M $(3(v-c)$ variables)

» the force densities $w_{ij} = w_{ji}$ $(e$ variables)

» the components of the stress tensor $\bar{S}_{11}$, $\bar{S}_{22}$ and $\bar{S}_{12}$ $(3v$ variables)

» the components of the extended shape operator $W_{11}, W_{22}, W_{33}, W_{12}, W_{23}$ and W_{13} $(6v$ variables)

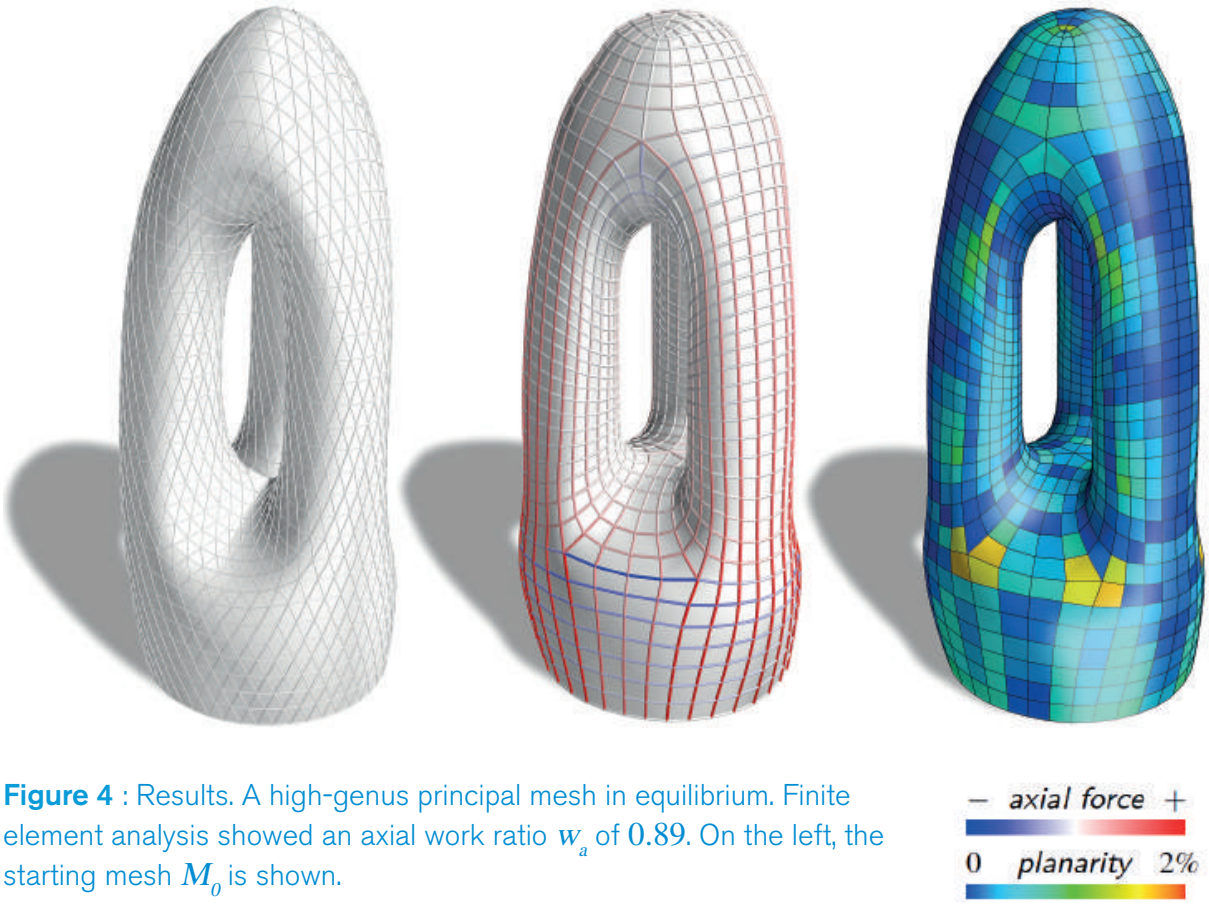


Figure 4 : Results. A high-genus principal mesh in equilibrium. Finite element analysis showed an axial work ratio w_a of 0.89. On the left, the starting mesh M_0 is shown.

» the tangent eigenvalues $\boldsymbol{\kappa}_1$ and $\boldsymbol{\kappa}_2$ of the extended shape operator ($2v$ variables)

» the directions $\mathbf{a}_1$ and $\mathbf{a}_2$ at $\mathbf{v}_i$ ($6v$ variables).

The main constraints are:

» the equilibrium at unsupported vertices $\mathbf{v}_i$ through Equation (1) ($3(v-s)$ equations)

» the connection of the stress tensor components with force densities through Equation (6) ($3v$ equations). Since we are interested only in principal directions, we can omit $\bar{A}_i$ from the equations.

» the connection of the curvature components with face normals through Equation (7) ($6v$ equations). As for the previous point, we omit A_i.

» the normalization of directions: $\mathbf{a}_1^T\mathbf{a}_1 = 1$ and $\mathbf{a}_2^T\mathbf{a}_2 = 1$ ($2v$ equations)

» the tangency of directions (tangency is guaranteed together with principal direction alignment, see below): $(\mathbf{a}_1 + \mathbf{a}_2)^T\mathbf{n}_i = 0$ (v equations)

where $\mathbf{n}_i$ is the vertex normal at $\mathbf{v}_i$. The target functions are given by the alignment equations of the vectors $\mathbf{a}_1$, $\mathbf{a}_2$ with stress and curvature directions, as seen in Sections 3.1 and 3.2. We have then:

» conjugacy on the Airy surface: $\bar{\mathbf{a}}_1^T\bar{\bar{S}}\bar{\mathbf{a}}_2 = 0$ (v equations)

» principal direction alignment: $W\mathbf{a}_1 = \kappa_1\mathbf{a}_1$ and $W\mathbf{a}_2 = \kappa_2\mathbf{a}_2$ ($6v$ equations).

For proximity to the starting surface, we minimize the distance between the points $\mathbf{v}_i$ and the tangent plane of their closest vertex $\mathbf{v}^0{}_j$ of M^0. We point out here that the projected stress tensor $\bar{S}$ is not properly defined for surface points with a vertical tangent plane. To avoid noise in the solution, we remove the target functions of Airy conjugacy on vertices $\mathbf{v}_i$ where the z coordinate of the normal $\mathbf{n}_i$ is in the range $\pm 10^{-2}$.

Counting degrees of freedom. Subtracting the number of constraints from the number of variables, and keeping fixed during the optimization the supported vertices (then $s = c$), we find $5v + e$ degrees of freedom. The target functions of alignment yield $7v$ equations. Conside-

ring that on a triangle mesh we have $e \approx 3v$, we are left with approximately v degrees of freedom. This allows us to ask for closeness to the reference shape as a soft constraint.

Solver. For the optimization, we use the *guided projection* method of (Tang et al., 2014). This method works best for systems of quadratic constraints. To reduce the degree of the main constraints when higher than two, we introduce secondary variables that are quadratic functions of the main ones; then, these functions are added as constraints. Let us rearrange all the variables, in number of V, in the vector $\mathbf{x} \in \mathbb{R}^V$. Let then $\varphi_n(\mathbf{x}) = 0,\ n = \{1, \ldots, N\}$, be the equations given by the constraints and the target functions. It is possible to add more or less importance to a specific constraint or target function by multiplying the corresponding equations by a weight ω_n. The system is solved iteratively. At each iteration k, given the current variable vector $\mathbf{x}^k$, each equation is linearized with a 1st order Taylor expansion:

$$\varphi_n(\mathbf{x}) \approx \varphi_n(\mathbf{x}^k) + \nabla\varphi_n(\mathbf{x}^k)^T(\mathbf{x} - \mathbf{x}^k) = 0.$$

The linearized system of weighted equations can be rearranged in matrix form as $H\mathbf{x} = \mathbf{r}$, with $H \in \mathbb{R}^{N \times V}$ and $\mathbf{r} \in \mathbb{R}^N$. To guarantee mesh quality and smoothness during the optimization, we add a fairness energy; we define it at each vertex $\mathbf{v}_i$ as the squared norm of the distance between $\mathbf{v}_i$ and the barycenter of its connected vertices $\mathbf{v}_{j \sim i}$. The total fairness energy can be written in matrix form as $\|K\mathbf{x} - \mathbf{s}\|^2$.

Fig.	v	e	iterations	time (s)
3	681	1960	12	13.8
4	1941	5784	15	78.4
5a	606	1760	15	16.7
5b	1140	3302	14	31.9
5c	1089	3136	13	28.3

Table 1: Optimization times and corresponding number of iterations for stress and curvature alignment, relative to the presented results. Values refer to triangular meshes with v vertices and e edges. The algorithm has been implemented in Python and tested with an Intel Core i7-6700HQ CPU with 2.60 GHz and a 15.9 GB RAM memory.

Additionally, the distance from $\mathbf{x}^k$ is used as a regularizer. The successive variable vector $\mathbf{x}^{k+1}$ is found by solving

$$\|H\mathbf{x}-\mathbf{r}\|^2+\delta^2\|K\mathbf{x}-\mathbf{s}\|^2+\varepsilon^2\left\|\mathbf{x}-\mathbf{x}^k\right\|^2\rightarrow\min,$$

with $\delta, \varepsilon \in (0,1)$ as weights. The iteration stops when a desired accuracy is achieved, or when no more improvement is gained. For further details on *guided projection*, we refer to (Tang et al., 2014).

5. Results and discussion

The proposed workflow has been tested on some sample architectural surfaces. Results are shown in Figures 3 to 5. The presented examples were subject to a uniform load per unit surface area and supported along the boundary. Optimization times of *step 1* are given in Table 1. To evaluate the quality of the result, we relied on the following two criteria.

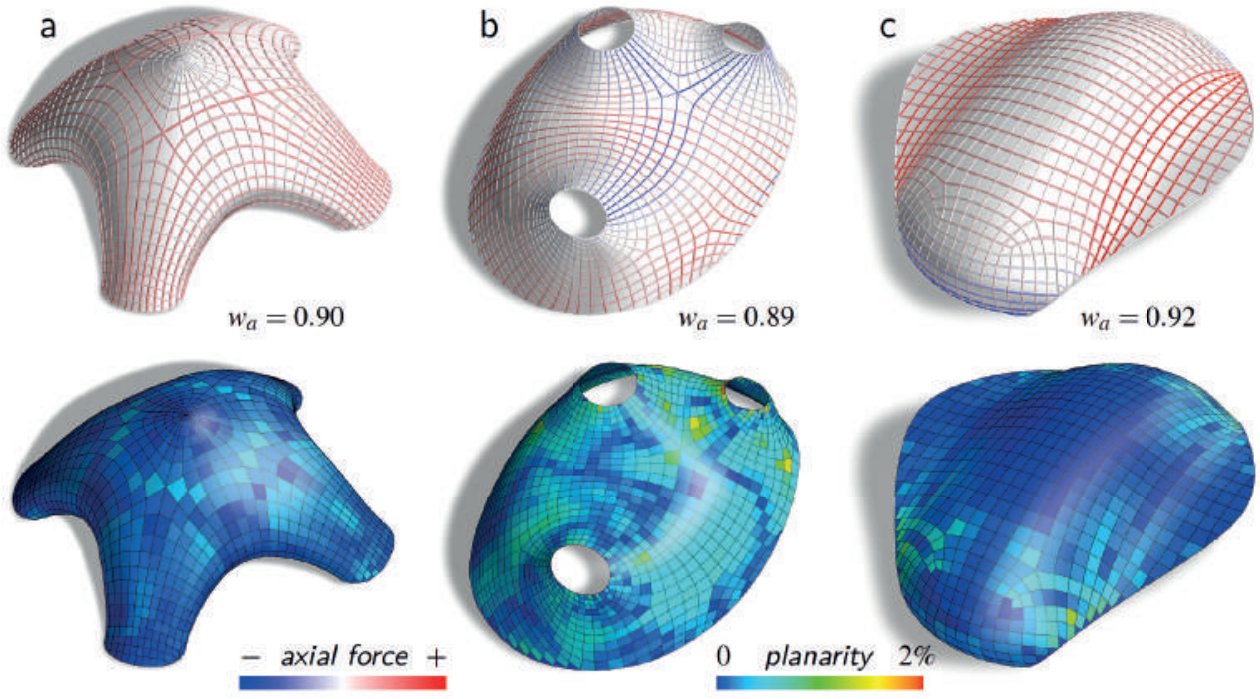


Figure 5: Results. Principal meshes in equilibrium achieved with our method. Meshes (**a**) and (**c**) discretize non-height field shapes. All boundaries are supported. The gridshell structures are in axial equilibrium under a homogeneous vertical load per unit surface area. Axial forces, planarity error and axial work ratios w_a are shown.

Convergence of post-optimization. As seen in Section 3.3, the quad mesh is post-optimized for equilibrium and planarity with the method of (Tang et al., 2014). In this step, we let the supported vertices glide along the corresponding boundary. We estimated planarity error of quadrilateral faces as the distance between the two face diagonals divided by their mean length. Regarding equilibrium, we evaluated the error per vertex as the norm of equilibrium Equation (1) divided by the mean vertex load magnitude. We considered the post-optimization converged when it reached a maximum planarity error below 2 % and a mean equilibrium error below 1 %. In the test samples we achieved convergence in less than ten iterations, noticing small changes in the mesh.

Finite element analysis. In actual gridshells the structure is dimensioned according to finite element analysis. It is of interest to evaluate the effectiveness of our optimization in this way as well. For this purpose, we modeled the final grid shell with steel S235 Timoshenko beam elements, connected together with rigid joints. Area loads were lumped in the nodes. The size of the cross section was chosen constant, according to resistance verification. To evaluate the equilibrium hypothesis, we computed the ratio of internal elastic work due to axial force in the beams over the total elastic work made by external loads. Axial work ratios w_a, found for our results, are shown beside Figures 3 to 5.

Limitations. Not all shapes achievable with our method own a stress-curvature network suitable for the extraction of architectural meshes. Indeed, the network layout may yield a mesh with a large variation of cell size, numerous or bad positioned singularities, or more generally, the resulting mesh may not possess the desired aesthetic qualities.

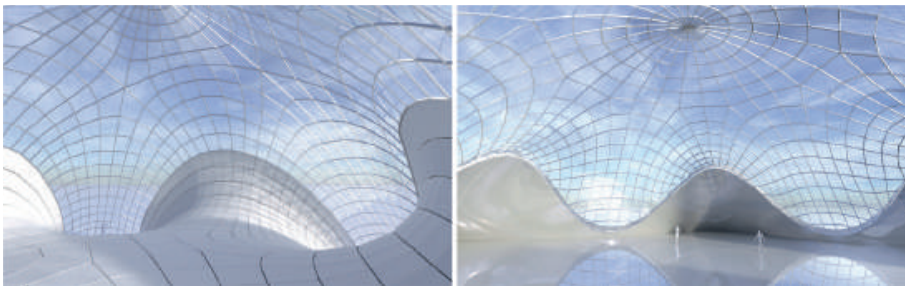
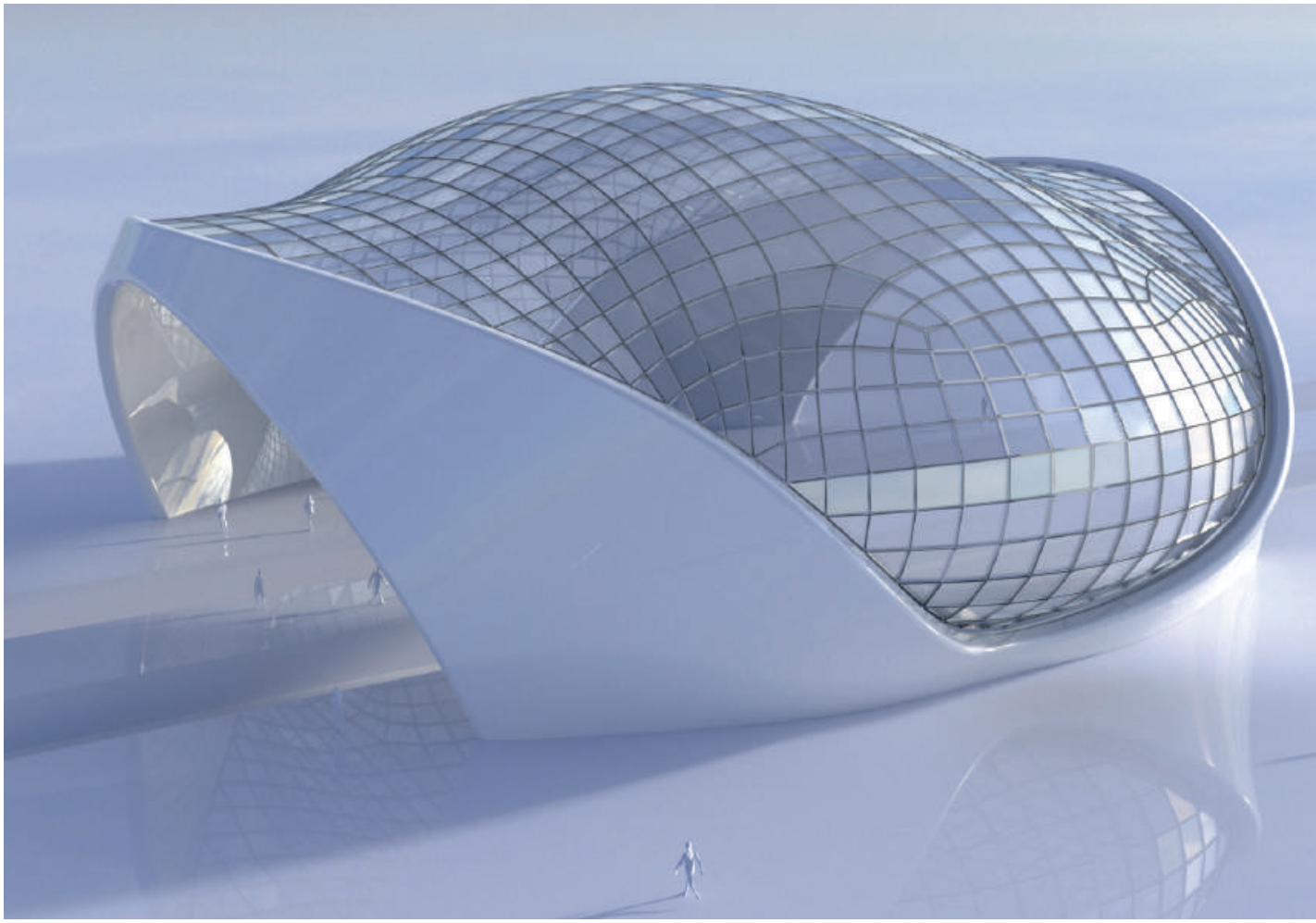


Figure 7: Architectural applications. Interior views of the meshes shown in Figures 5a (on the left) and 3 (on the right). Exterior views are shown in Figure 1.



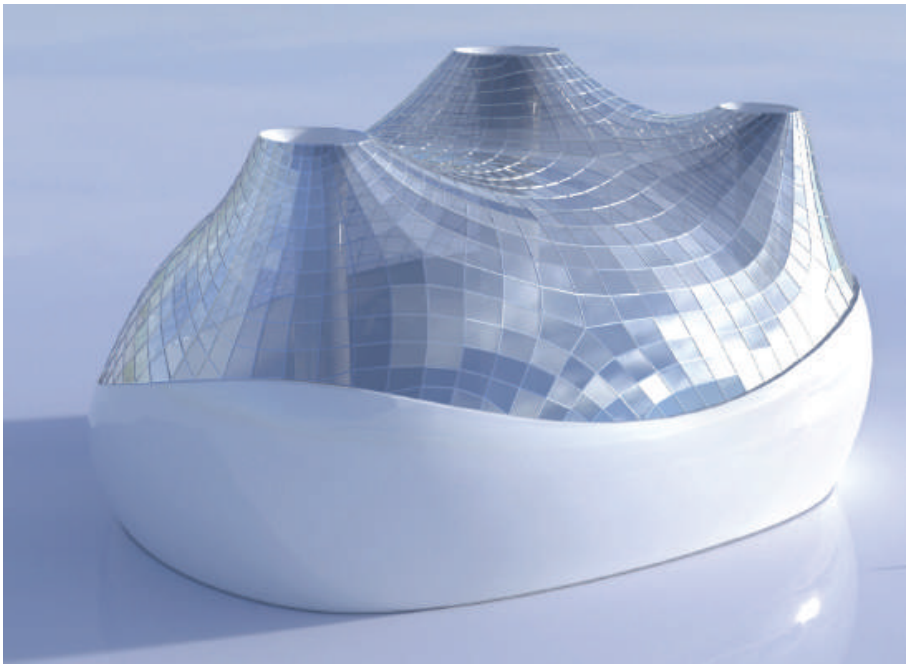


Figure 6: Architectural applications. Steel-glass gridshells achievable with the meshes shown in Figures 5b (bottom) and 5c (top). Face planarity errors below 2 % are compatible with cladding through flat glass panels.

6 Conclusion

We have introduced a method for the design of principal meshes in equilibrium. Thanks to the optimization of a membrane surface towards stress and curvature alignment, we can achieve a quad mesh connectivity that can be post-optimized for equilibrium and planarity of faces with low conflict. In our implementation, the use of a discrete equilibrium on the edges of a triangular mesh allows us to avoid a parametrization of the surface and the explicit use of an Airy stress function. We have overcome in this way the limitations to height field shapes of (Kilian et al., 2017). We provide then a tool for the design of a wide range of architectural freeform shapes that, taking into account geometric and static factors, can reduce realization costs.

Acknowledgments

This research was supported by SFB-Transregio programme Geometry and Discretization (Austrian Science Fund grant no. I 2978), and by the project “Geometryand Computational Design for Architecture and Fabrication” at Vienna Universityof Technology.

References

Angelillo, M. and A. Fortunato (2004). Equilibrium of masonry vaults. In M. Fremond and F. Maceri (Eds.), *Novel Approaches in Civil Engineering*, pp. 106–111. Oxford: Springer.

Block, P. and J. Ochsendorf (2007). Thrust network analysis: A new methodology for three-dimensional equilibrium. *Journal of International Association for Shell and Spatial Structures* 48(3), 167–173.

Bommes, D., H. Zimmer, and L. Kobbelt (2009). Mixed-integer quadrangulation. *ACM Transactions on Graphics 28*(3), 77:1–77:10.

Cohen-Steiner, D. and J. M. Morvan (2003). Restricted Delaunay triangulations and normal cycle. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry*, New York, pp. 312–321. ACM.

Fraternali, F. (2010). A thrust network approach to the equilibrium problem of unreinforced masonry vaults via polyhedral stress functions. *Mechanics Research Communications 37*(2), 198–204.

Kilian, M., D. Pellis, J. Wallner, and H. Pottmann (2017). Material-minimizing forms and structures. *ACM Transactions on Graphics 36*(6), 173:1–173:12. Proc. SIGGRAPH Asia.

Liu, Y., H. Pottmann, J. Wallner, Y.-L. Yang, and W. Wang (2006). Geometric modeling with conical meshes and developable surfaces. *ACM Transactions on Graphics 25*(3), 681–689. Proc. SIGGRAPH.

Maxwell, J. C. (1872). On reciprocal figures, frames, and diagrams of forces. *Transactions of the Royal Society of Edinburgh 26*, 1–40.

Mitchell, T. (2013). *A Limit of economy of material in shell structures*. Ph. D. thesis, University of California, Berkeley.

Pavlovi, M. (1984). A statically determinate truss model for thin shells: One-surface analysis (membrane hypothesis). *International Journal for Numerical Methods in Engineering 20*(10), 1841–1861.

Pottmann, H., M. Eigensatz, A. Vaxman, and J. Wallner (2015). Architectural geometry. *Computers and Graphics 47*, 145–164.

Pottmann, H. and Y. Liu (2007). Discrete surfaces in isotropic geometry. In R. Martin, M. Sabin, and J. Winkler (Eds.), *Mathematics of Surface XII*, Volume XII, pp. 431–363. Springer.

Schiftner, A. and J. Balzer (2010). Statics-sensitive layout of planar quadrilateral meshes. In C. Ceccato et al. (Eds.), *Advances in Architectural Geometry 2010*, pp. 221–236. Springer.

Sun, X. (2016). *Discrete Curvature Theories and Applications*. Ph. D. thesis, King Abdullah University of Science and Technology, Thuwal.

Tang, C., X. Sun, A. Gomes, J. Wallner, and H. Pottmann (2014). Form-finding with polyhedral meshes made simple. *ACM Transactions on Graphics 33* (4), 70:1–70:9.

Vouga, E., M. Höbinger, J. Wallner, and H. Pottmann (2012). Design of selfsupporting surfaces. *ACM Transactions on Graphics 31*(4), 87:1–87:11.

Assembly of shells with bi-stable mechanism

Yu-Chou Chiang, Sina Mostafavi, Henriette Bier

Yu-Chou Chiang
Chiang.YuChou@gmail.com
Robotic Building, Delft University of Technology, Netherlands

Sina Mostafavi
S.Mostafavi@tudelft.nl
Robotic Building, Delft University of Technology, Netherlands

Henriette Bier
H.H.Bier@tudelft.nl
Robotic Building, Delft University of Technology, Netherlands

Keywords:
Bi-stable mechanism, reconfigurable assembly, shell structure, free-form construction, programmable material

Abstract

Shell structures achieve stability through double curvature, which brings about construction challenges. This paper presents a strategy to design and assemble a panelized shell with a bi-stable mechanism aiming to make the assembly process more efficient. The developed prototype has two states of flat and three-dimensional stable configuration. This reconfiguration is achieved by reconfiguring the flattened surface of a shell into a three-dimensional structure using embedded bi-stable joints. In order to apply this approach on free-form double curved shells, a workflow to translate a shell into its flattened state is developed. Discrete components are connected using bi-stable joints, where each joint has two stable states. Once the joints are mechanically activated, they guide the adjacent components contracting and rotating into the three-dimensional configuration. Initial explorations indicate that an edge of a shell will turn into an isosceles trapezoid in the flattened configuration while a node of a conical mesh will turn into a cyclic quadrilateral in the flattened configuration. The flattening process is demonstrated using a free-form vault, while scaled physical porotypes are 3D printed with PLA and tested. Future studies require exploration into applications with construction materials at larger scales.

1. Introduction

Double curved surfaces give shell structures their structural integrity and architectural expression, while being at the same time challenging with respect to construction. Between the 1920s and 1960s, numerous site-cast concrete thin-shells were built, then, because the geometry was confined to regular forms (e.g., sphere and paraboloids) and it required labor-intensive formwork, the construction of concrete shell declined (Meyer & Sheer 2005). Since the 2000s, with the increasing demand on free-form architectures, thin-shell structures in the forms of grid-shells and tessellated roofs have attracted the attention of architects, engineers, and geometricians (Pottmann et al. 2015).

Today, several technologies facilitate the revival of the thin-shell structure, including structural form-finding algorithms, numerical geometry optimization, and computational controlled machinery (Van Mele et al. 2016). By applying these methods, production of the geometrically

complex components becomes manufacturable within a moderate budget. However, assembly remains labor-intensive work. Both formworks for cast-in-place concrete and falseworks for on-site assembly demand considerable labor to assemble. To improve the assembly process, it is necessary to explore methods addressing this problem.

This research introduces an alternative way of assembling shell structures. The goal is to develop a methodology to decompose a shell structure into panels connected by bi-stable joints in a flat configuration. Then, through reconfiguration, the bi-stable joint will guide adjacent panels contracting and rotating into the pre-programmed position. As a visualized example, Figure 1 shows the overall process of translation and reconfiguration from flat to double-curved of a dome-like shell.

In this paper, the focus is on the geometrical aspect of the bi-stable mechanism. Further material tests, actuating strategy, static and dynamic analysis will be explored at building scale.

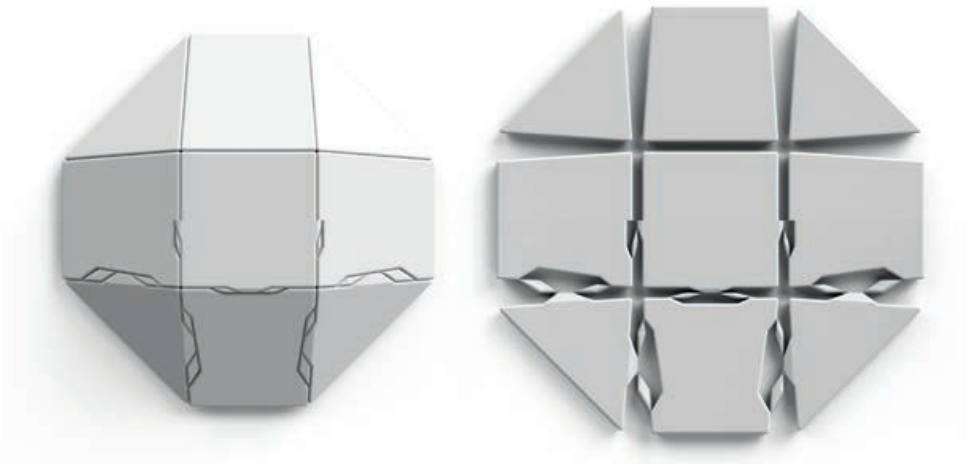


Figure 1: Top views of a dome are showcasing the four phases of the proposed workflow. The workflow starts from a panelized shell (top left), followed by its flattened configuration (top right) and the installation of bi-stable joints (bottom right). Once the assemblage is mechanically activated, it turns to the completed assembly (bottom left).

1.1 Outline

Related assembly approaches and the development of reconfigurable bi-stable mechanisms are introduced in the background section. Then, in the methods section, mechanical and geometrical details of the established bi-stable mechanism are presented. A novel bi-stable mechanism

capable of out-of-plane rotation is proposed. The criteria for unrolling a shell into its flattened state are identified. Some physical prototypes applying these methods are shown in the result section. In conclusion, the findings are summarized, and the future research is proposed.

2. Background

In order to enhance the stagnating productivity in the architecture, engineering and construction sector (McKinsey Global Institute, 2017), assembly, especially on-site assembly, need to take advantage of automation.

From a holistic view, assembly (or assembly-aware design) is not merely combing separate components together; it is a constructional strategy to discretize an assemblage into manufacturable components then aggregate the components into the completed assemblage. There are numerous research projects investigating the interrelation between design and assembly. By identifying the difference between the various schemes, an interesting approach to carry out assembly stood out, reconfigurable assembly.

Reconfigurable assembly could be defined as a constructional strategy to aggregate components in a simple and manufacturable configuration, and then reconfigure the assemblage into the desired complex configuration. For instance, A 3D-printed straight line can be reconfigured into a wavy curve or a set of polygons on a flat plan can be reconfigured into a polyhedron (Tibbits, McKnelly, Olguin, Dikovsky, & Hirsch, 2014; van Manen, Janbaz, & Zadpoor, 2017). Potentially, the reconfigurable mechanisms can reduce the number of independent components in a system (Tibbits et al., 2014); an object can be stored and transported in the compact configuration, then be deployed to the serving configuration (Haghpanah, Salari-Sharif, Pourrajab, Hopkins, & Valdevit, 2016).

The previous published shape-reconfigurable mechanisms are achieved through expanding, contracting materials or architected materials. The first approach mainly relies on special materials that are capable of expanding or contracting when the environment is changing. For example, a folding mechanism can be created by layering two materials, which have different expansion rate in water, ensuring that a box can be reconfigured from a flat 3D-printed object (Tibbits et al., 2014). This type

of approach requires a special expanding material and the corresponding environmental change to activate the expansion

The other approach involves designed porosity within the constituent material, and such porosity makes the material bendable and stretchable, which are termed as architected shape-reconfigurable materials (Konaković et al., 2016; Rafsanjani & Pasini, 2016). Inspired by the research of "Beyond Developable" (Konaković et al., 2016), where researchers developed algorithms to translate a flat auxetic material to a double curved surface. The stretchable mechanisms open up an insight of how to approximate a double curved surface. To be noted that, this auxetic mechanism has a certain flexibility, therefore, it is vulnerable to deform when external forces are applied.

Conversely, there is a type of architected materials can stably maintain it reconfigured shape, termed bi-stable or multi-stable mechanisms (Haghpanah et al., 2016; Rafsanjani & Pasini, 2016). It provides an interesting reconfiguration feature: the scale factor between the two stable states can be engineered, and the two stable states have the mechanical strength to resist external forces. However, for the time being, the published bi-stable mechanisms are limited to in-plane or two dimensional reconfigurations.

Inspired by the development of the auxetic mechanism and the bi-stable mechanisms, this research investigates the application of bi-stable auxetic mechanisms in flat-to-curved reconfigurable shell structures.

3. Methods

In the method section, three aspects of the workflow are discussed. Considering the overview of the processes introduced in Figure 1, illustrating the panelization of a shell, this section addresses the methods with emphasis on the technical details of the bi-stable mechanism. In the first sub-section, the basic in-plan translation of the bi-stable mechanism is presented. Then, geometric features of the proposed out-of-plane rotation of bi-stable mechanism are introduced. The third sub-section discusses the panelization process for flattening a double-curved surface. By applying these processes, a given double curved shell can be flattened into a bi-stable reconfiguring mechanism.

3.1 Mechanical features of the in-plane translation of bi-stable mechanisms

A basic bi-stable mechanism is represented by a unit where two beams are connected by a hinge and binned at two supports (Haghpanah et al., 2016; Huang & Vahidi, 1971). The idealized structure unit is depicted in Figure 2 (top). When the unit is subjected to an external force at the center of the hinge, the two inclined beams compress against each other. More specifically, the external force is balanced by the vertical components of the axial compressions of the two inclined beams. The axial compressions make the beams shortened, resulting in the downward displacement of the center hinge. In the case that the external force is small and removed afterward, the beams will spring back to the original state, noted as the first stable state in Figure 2. In a case that the external force makes the axial forces of the beams either reach the Euler buckling critical load or the beams are too flat to provide effective vertical components, the mechanism arrives its critical state and consequently, it snaps-through. After the external force is removed, and all the material spring back to its original length, the mechanism rests in the alternative configuration, noted as the second stable state in Figure 2. Although the mechanical deformation of the material does happen during the reconfiguration, geometrically, the two stable states can be simply regarded as two possible solutions of the circle-circle intersection.

The load-displacement response of the mechanism in two scenarios is illustrated in Figure 2. Cases with different beam thickness are presented. The red case has thicker beams, reconfigures without Euler buckling and it has a smooth load-displacement response curve, indicated in red in the diagram. The blue case with thinner beams has a higher tendency of buckling. The sharp turns in the blue curve indicate the start and the end of buckling. The critical states of the two cases are indicated with the circles in the diagram. It is noteworthy that the critical state of the blue case is reached easier than the other, because of the Euler buckling; less force or less displacement is required.

In the reconfiguration, two features are important for the application. First, the critical force can be programmed with tuning the stiffness of the inclined beams. Secondly, the displacement is controlled by the following formula:

$$\delta = 2 \cdot L \cdot \sin \alpha \tag{1}$$

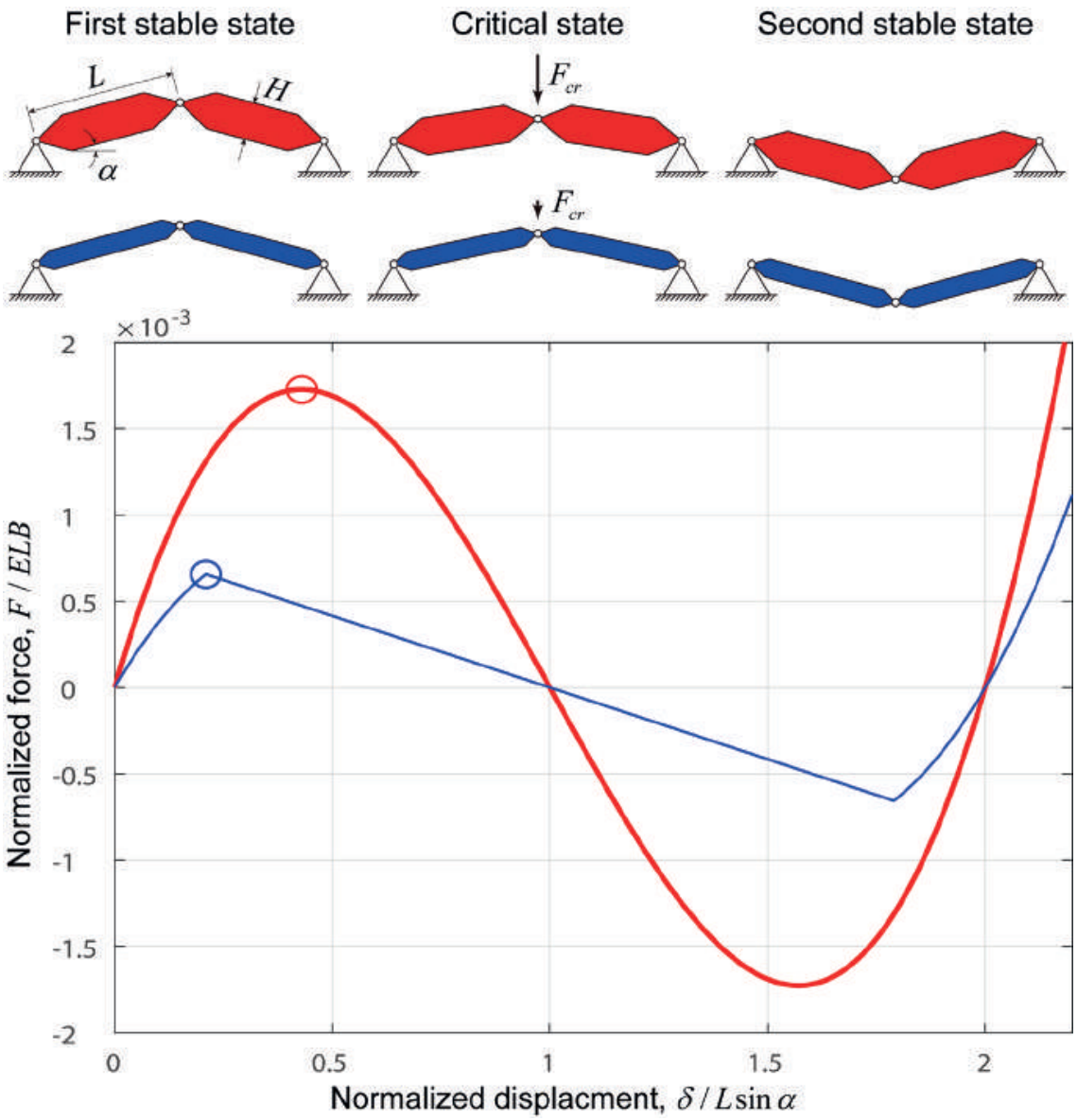


Figure 2: Basic bi-stable mechanisms and their load-displacement response: The first and second stable states have normalized displacement of 0 and 2 respectively. The critical states are indicated with the circles. E elastic modular of the material, B width of the beam, δ displacement of the center hinge.

The displacement formula can be applied to create a difference of displacements on top and bottom surface to create a bending mechanism. For more detail, the geometrical features of such mechanism are explored and discussed in the following sub-section.

3.2 Geometric features of the basic and proposed bi-stable joints

The previously published projects on bi-stable joints mainly focus on in-plane or two-dimensional translations. To create a bi-stable mecha-

nism capable of out-of-plane bending and bring adjacent panels together, further adaptation and exploration are necessary. Starting from the basic unit mentioned in the previous sub-section, the inclined beams are turned into a pair of the bi-stable connector to link two panels. One of the panels is connected to the center hinge, and the other one to the supports. Subsequently, the parallel contraction is achieved on condition that the lengths of the two connectors at the top and bottom surface are the same. An example of panels linked by a pair of connectors is shown in Figure 3a. When a connector has a shorter length than the other, an in-plane rotation can be achieved (**Fig. 3b**). In order to trigger a reconfiguration and make the mechanism allow for bending, the displacements at the top and bottom surface should be different. The contracting displacement, as shown in Equation (1), is related to rotating angle and length of the rotating arm. To avoid torsion, the rotating angle shall remain constant in a single object. Conversely, it is possible to enlarge or shorten the rotating arms. In Figure 3c, the rotating arms at the top surface (i.e. L_{aT} and L_{aB}) are smaller than the counterparts at the bottom. Subsequently, the bending is achieved. After the two bi-stable connectors with different distances of contraction are created, an in-plane rotation can also be integrated (**Fig. 3d**).

To be noted that, during the reconfiguration, each element rotates around its axis, and undergoes a temporary deformation as suggested in the previous sub-section. When the elements spring back to their original length, the system reaches the second stable state. Figure 4 shows the rotation of each element relative to the panel in the back (gray). During the reconfiguration, the red and blue connectors rotate around the physical hinges (the dash-dotted axes in red and blue). Meanwhile, the green panels rotate around the green axis.

In addition to the orientation of the axes and the hinges, the contact surfaces between the elements need to be designed. Considering the fact that the axes line of the hinges (red and blue colors) have to have a point intersection with the rotation axis line of the panels (green color), a mapping method is proposed to transform a lateral surface of a cone around the rotation axis to a rectangle (**Fig. 5**). On the rectangular image, it is possible to design various patterns then translate the pattern to physical cases. One of the pattern and the result is shown in Figure 5. As a constraint, the void space between the two panels has to be an isosceles trapezoid.

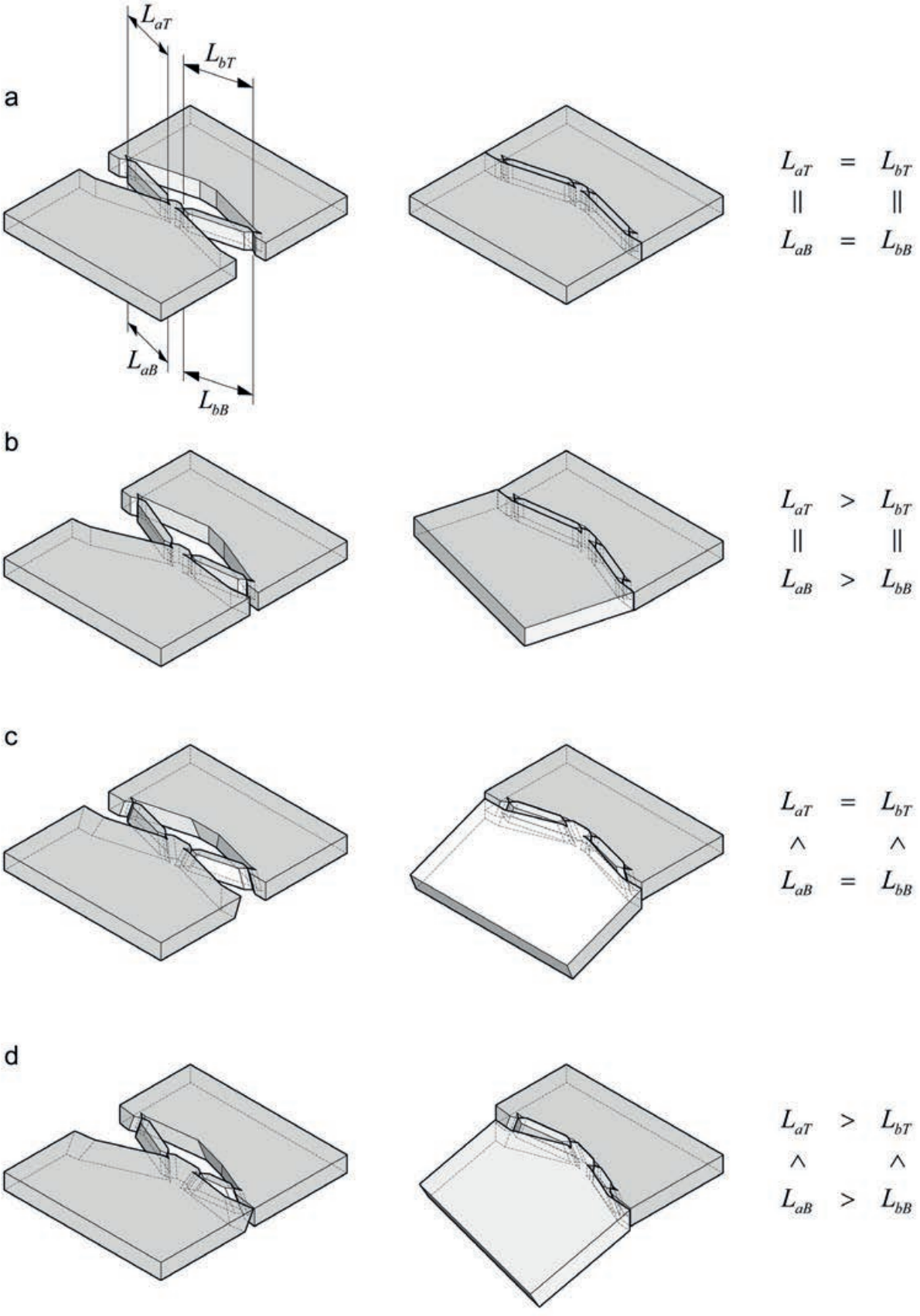


Figure 3: Joints with different degrees-of-freedom. Flattened states and curved states are shown in the left and middle column respectively. The different arrangements of the rotating arm (right column) can result in different degrees-of-freedom in the reconfiguration.

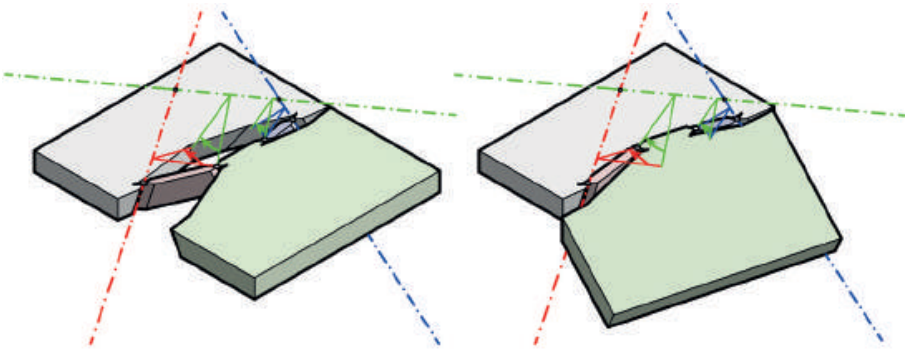


Figure 4: The geometric detail of the reconfiguring process: The flattened stable state (left) can be reconfigured into the curved stable state (right). The dash-dotted lines indicate the rotation axes; the colored arrowheads suggest the rotating directions of the corresponding elements.

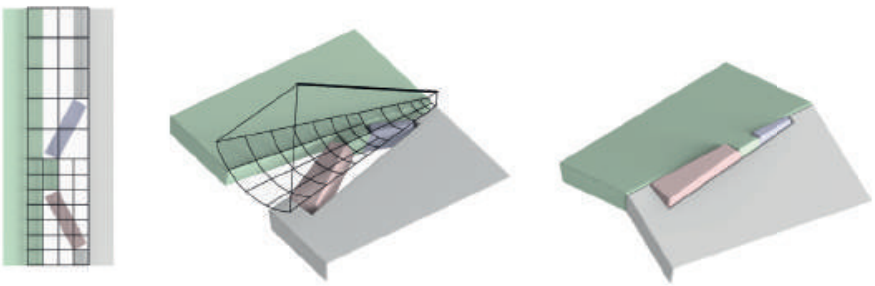


Figure 5: The conformal mapping grids show how to transform the design pattern on the rectangular(left) to the conical surface (middle). The reconfigured result is shown on the right.

This sub-section described the process of transforming an edge connected to two panels to the flattened configuration with the bi-stable mechanism. Two features affect the following process. These features are results of the adjacent panels rotating around the same axes. First, the rotation axis must be located on the bisector planes of the angle between the two panels (in the curved state). Secondly, the void between the panels in the flattened state has to be an isosceles trapezoid. In the next subsection the unrolling of a shell to a flattened configuration, with integrated reconfigurable joints is explained.

3.3 Panelization and unrolling of a panelized shell into the flattened state

As mentioned, the rotation axis has to be in the bisector plane of the dihedral angle between adjacent panels. Furthermore, the rotation axis dictates the orientation of the hinges and the interface between panels and connectors. For a node, each surrounding edge introduces one in-

terface with such orientation. One may prefer all the interfaces intersect each other at the same axis. In such case, one of the findings of the research is that the co-axis of every node dictates the tessellation to be conical meshes (Liu, Pottmann, Wallner, Yang, &Wang, 2006).

The constraint of the isosceles trapezoid and the preference of the conical mesh determine that a node has to be a cyclic polygon in the flattened state as shown in Figure 6. The panels are shaded in gray, and the symmetric axis of the trapezoid illustrated in dash-dotted lines. The proving of the cyclic quadrilateral is separated in two parts.

The first part of the proving explains that, in the flattened state, the vertex of a panel (**Fig. 6**) have to be on an arc if the two neighbor panels are fixed. Considering a node surrounded by four panels, in a flattened state, and the location of the second panel (P_2) is not determined yet while the other P_1 and P_3 are fixed. By the round angle around

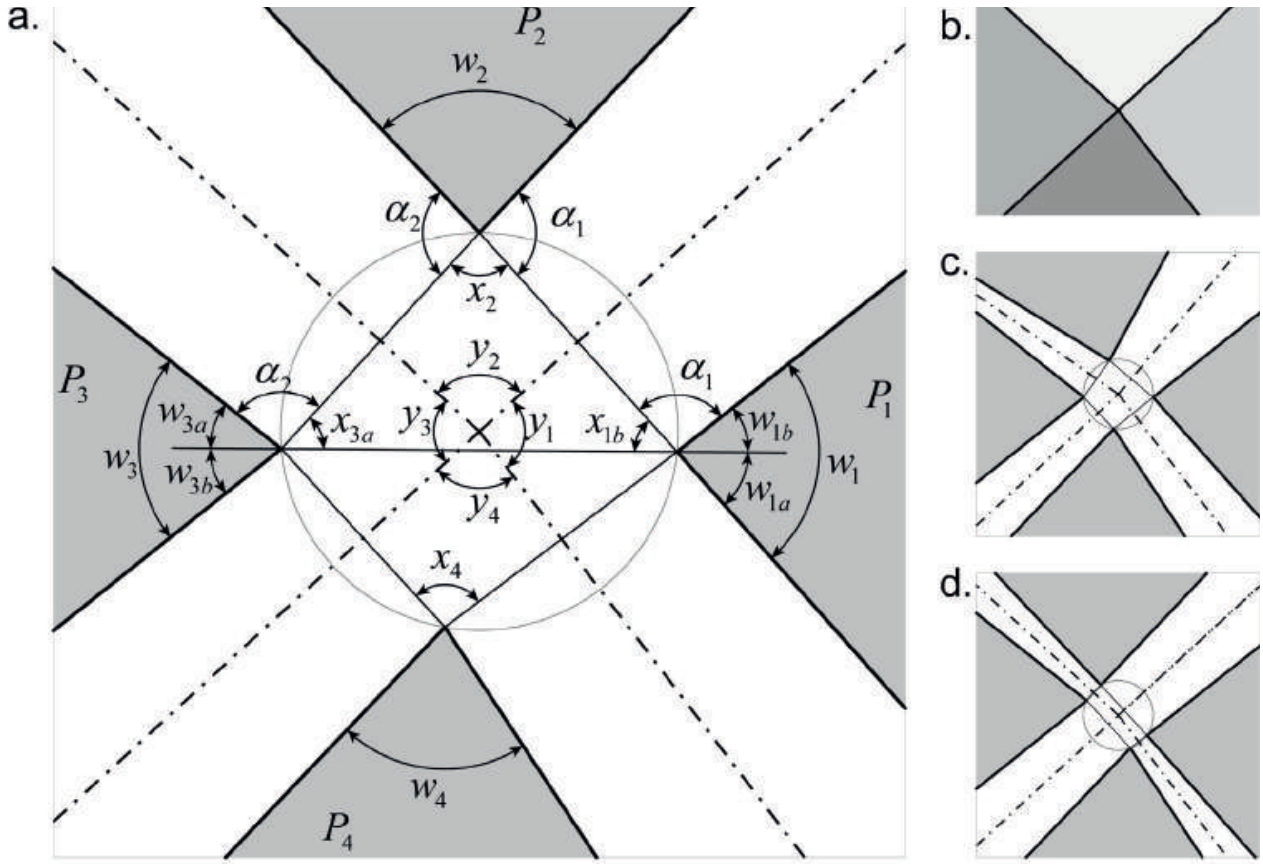


Figure 6: Different flattened states of a node from a conical mesh. The shell state is shown in b, the a, c and d display the flattened states in different configurations. The dash-dotted lines indicate the symmetric axes of the trapezoids. These axes intersect at the center of the circumcircle of the quadrilateral formed by the vertices.

the vertex of P_2, it can be derived that $w_2+\alpha_1+\alpha_2+x_2=2\pi$. By the straight angles around the vertices of P_1 and P_3, it can be formulated as $w_{1b}+\alpha_1+x_{1b}=\pi$ and $w_{3a}+\alpha_2+x_{3a}=\pi$. While the triangle between these vertices implies that: $x_{1b}+x_2+x_{3a}=\pi$. With these equations, the variable angles (i.e. α_1, α_2) can be eliminated. Then the following equation is derived: $w_2-w_{1b}-w_{3a}+2x_2=\pi$. In an alternative format, it can be expressed as: $x_2=\pi/2+(w_{1b}-w_2+w_{3a})/2$. This equation implies that the vertices of the P_2 must locate on the arc between P_1 and P_3, no matter what degrees the angles α_1 and α_2 have.

The second part of the proving explains that the arcs of the vertices of P_2 and P_4 complete a circle. Given: $x_4=\pi/2+(w_{1a}-w_4+w_{3b})/2$. Therefore, the summation of the opposite angles in the quadrilateral can be expressed as it follows:

$$x_2+x_4=\pi+\frac{w_1-w_2+w_3-w_4}{2} \tag{2}$$

Considering the node is originated from a conical mesh, which means that the angles of the four panels should meet the condition (Liu et al., 2006):

$$w_1+w_3-w_2-w_4=0 \tag{3}$$

Therefore, Equation 2 can be updated as:

$$x_2+x_4=\pi \tag{4}$$

Equation (4) implies the arc for P_2 and the arc for P_4 complete a full circle. This feature constrains the quadrilateral to be cyclic.

Since the quadrilateral is cyclic, the symmetric axes of the trapezoids intersect at the center of the circumcircle. The angles between the symmetric axes (e.g., y_1 in **Fig. 6**) meet a condition similar to conical mesh: $y_1+y_3-y_2-y_4=0$. It can be derived from $y_2=\pi-x_2$ and $y_4=\pi-x_4$. Therefore, $y_2+y_4=2\pi-x_2-x_4$. Similarly, $y_1+y_3=2\pi-x_1-x_3$. Since $x_2+x_4=\pi$, and $x_1+x_3=\pi$, it can be concluded that:

$$y_1+y_3-y_2-y_4=0 \tag{5}$$

Equation (4) is similar to the condition displayed in Equation (3).

As it is shown in dash-dotted lines in Figure 7, the symmetric axes of the trapezoids surrounding a panel form a polygon. The polygons can be seen as the extensions of the panels. In the three-dimensional conical mesh all the nodes surround by four angles meet Equation (3), and the same condition makes the extended flattened polygons in the two-dimensional configuration hold the condition of Equation (5).

These identified features help the development of further algorithmic design methods. A preliminary result of unrolling of a free-form conical mesh is displayed in Figure 7.

4. Results, reflection, and future exploration

Following the discussion on mechanical and geometrical features of the bi-stable joint, panelization, and flattening, this section describes the validation of the proposed methods through physical prototyping. The prototypes were produced using fused filament 3D printing with polylactic acid (PLA). The hinge between connectors and panels are fabricated as compliant hinges. Although the bi-stable mechanism does not have to be manufactured with compliant hinges, it is one of the most convenient methods to combine the hinge mechanism with additive fabrication. Currently, the detail design scheme for the compliant hinge is under further investigation. Some of the results demonstrate that the 3D-printing of compliant hinges is a promising design to production method for prototyping. Figure 8 shows one of the first set prototypes of the bi-stable mechanism applied on bending reconfiguration, while Figuer 9 displays the reconfiguration process of the 3D-printed saddle surface.

By extending the principles of the edge-based bi-stable mechanism as shown in Figure 8 and Figure 9, a node-based bi-stable mechanism can also be achieved. Figure 10 shows an ongoing exploration of designing a bi-stable auxetic flat-to-curve reconfigurable mechanism, which is achieved by applying the same design principles.

When comparing widely explored origami mechanisms, which require thin materials, the proposed mechanisms are compatible with thick materials. By introducing the tilted hinges, the thick materials can undergo flat-to-curved reconfiguration. The property of thickness-insensitiveness allows engineers to thicken any identified week part to reduce the local stresses.

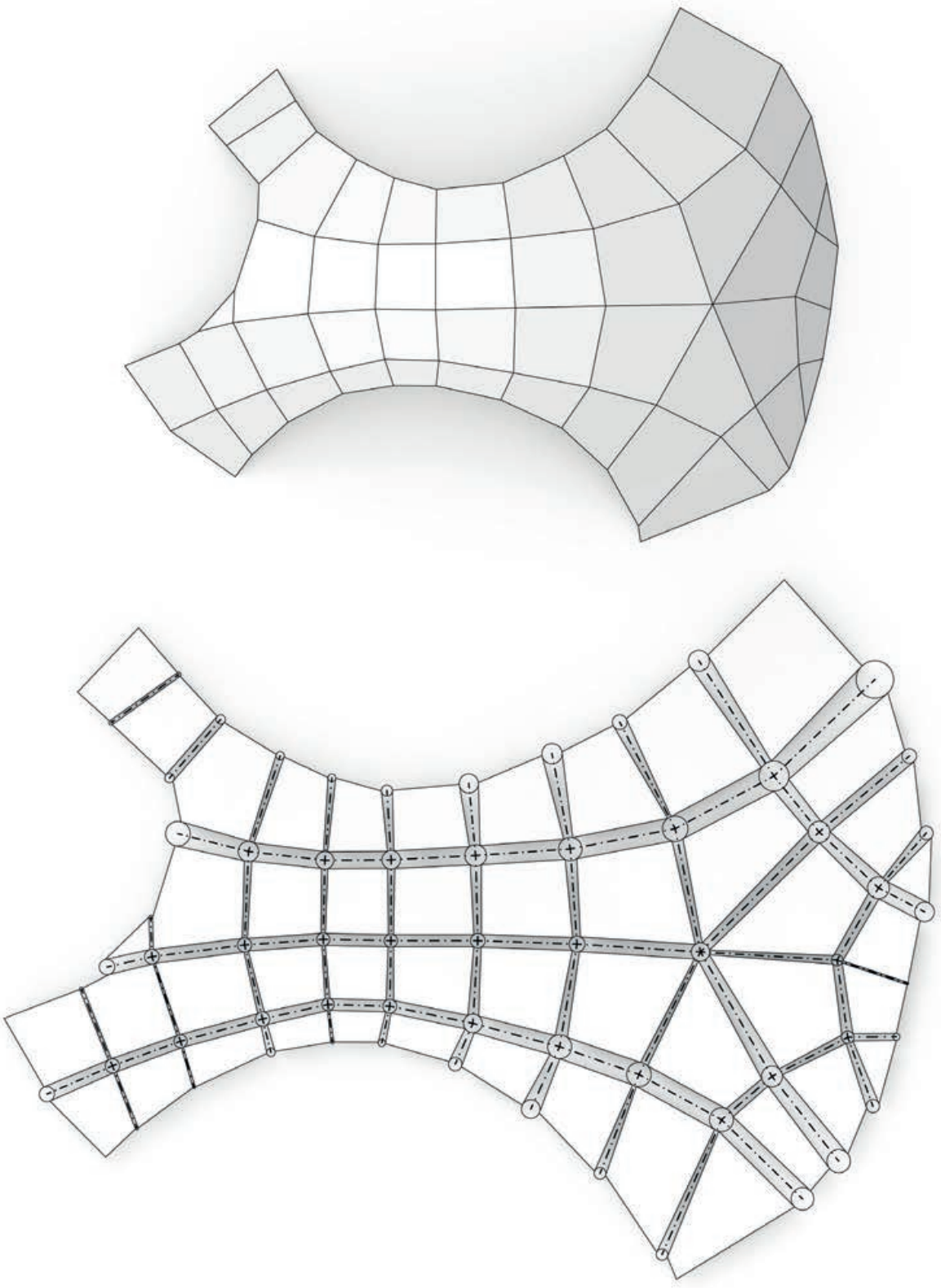


Figure 7: Top views of a free-form vault: The conical mesh (top) is translated into a flattened configuration (bottom). A node in the shell configuration will transform into a cyclic quadrilateral in the flat configuration while an edge will turn into an isosceles trapezoid. The circles in the figure denote the circumcircle of the quadrilaterals while the dash-dotted lines represent the symmetrical axes of the trapezoids.

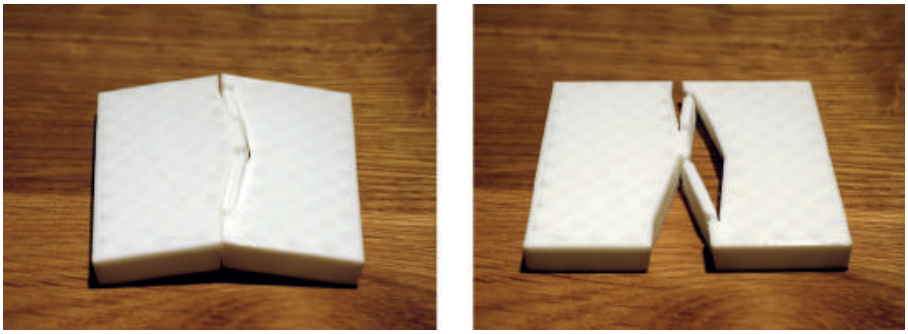


Figure 8: Test of the bi-stable joint, which is 3D printed with 5 mm thick PLA panels and 0.5 mm width compliant hinges. Pushing the panels (left), the bi-stable joint is mechanically activated and reconfigures to the curved shape (right).

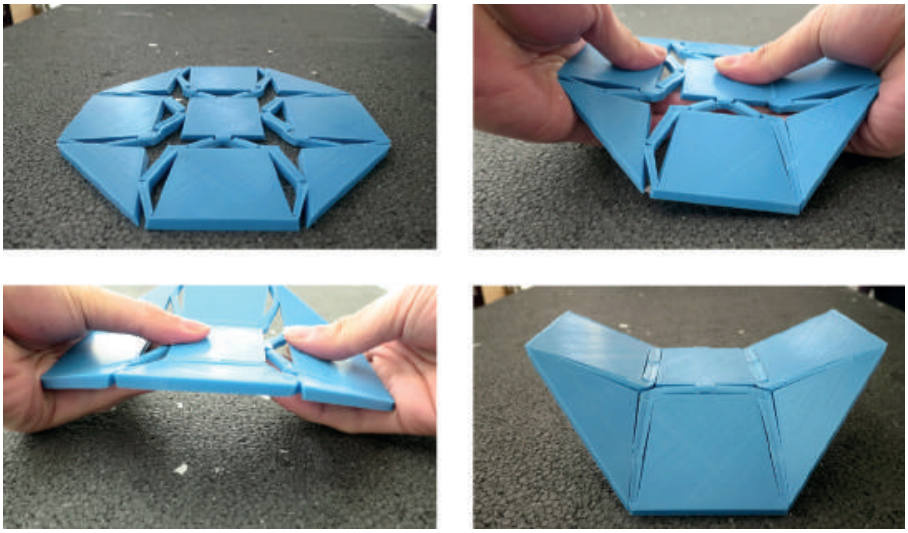


Figure 9: Reconfiguration sequences of a saddle surface produced with 3D printed PLA. It proves the concept of applying the bi-stable mechanism to a double curved shell. The sequence starts from the nine panels in the flat configuration (top left), pushing the joints one-by-one, it gradually takes the shape of the final configuration.

The exploration has identified that solution exists, under geometrical constraints and under the preliminary assumptions of zero-stiffness hinges and flexible materials. For future geometrical studies, the authors will explore and develop relevant methods to systematically translate freeform surfaces into the titled cutting patterns for the bi-stable auxetic mechanism. For the future material investigation, the authors will investigate materials to fabricate the hinges and the panels. For the structural representative prototypes, the authors will add extra weight to the scaled physical prototypes for both statical and dynamical tests to compensate scale factor.

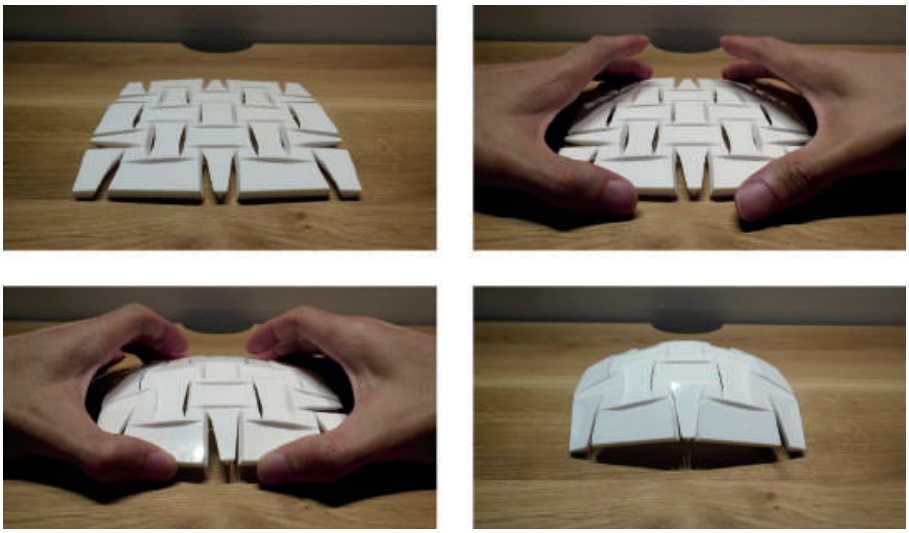


Figure 10: From flat to sphere reconfiguration with bi-stable auxetic mechanism can also be achieved with the shown design principles. By squeezing the mechanism, 25 panels connected by 16 smaller rotating node connectors are reconfigured into the pre-programed position in one step. Video of the reconfiguration can be accessed via https://youtu.be/4GcG_AurBQk.

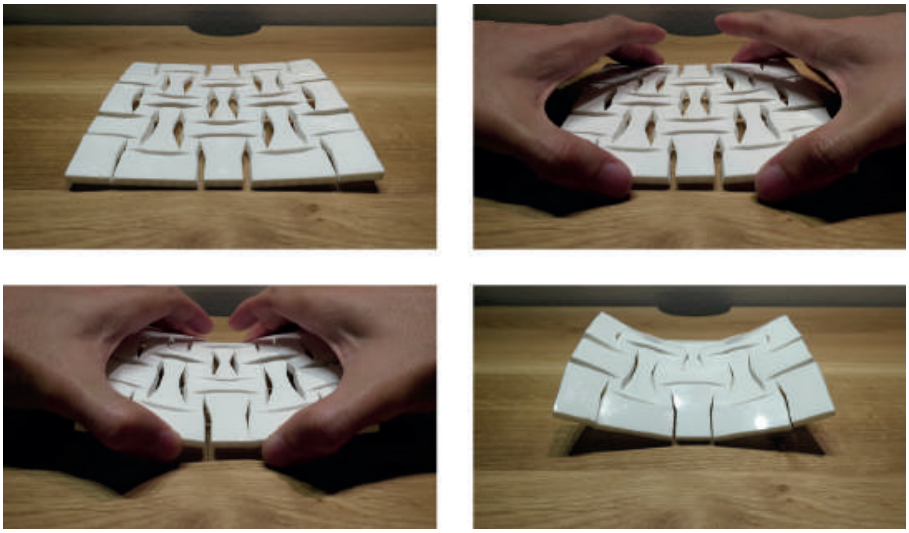


Figure 11: Flat to saddle reconfiguration with bi-stable auxetic. Same as the reconfiguration of the spherical surface, by squeezing the mechanism all the components are reconfigured into the pre-programed position in one step. Video of the reconfiguration can be access via https://youtu.be/WWHXlySTkfI.

5. Conclusion

In this research, the presented method of developing bi-stable mechanism is introduced as an approach for the assembly of shell structures.

The workflow to translate a three-dimensional curved shell structure into a flattened surface is explored, and specific geometric design constraints are discovered and proved. Validating the method, several reconfigurable shell topologies are designed and physically prototyped with bi-stable mechanisms.

In cases of any given planar quadrilateral meshes, as a necessary condition, the spaces between the edges of the flattened panels are isosceles trapezoids. While in cases that the curved surface is a conical mesh, the spaces between the flattened panels not only met the necessary condition of being isosceles trapezoid but also, as a sufficient condition, vertices of all neighboring panels are on a circle. These necessary and sufficient conditions later can be further integrated in an assembly-aware parametric modeling. The novelty of the proposed method is that it allows out-of-plane or three-dimensional reconfiguration. Consequently, the derived principles can be applied to more complex free form morphologies.

To implement the proposed reconfigurable assembly at building scales, further investigations need to address structural aspects and material properties. In macro scale design of reconfigurable shells, integrating the structural analysis in the form-finding process can inform the overall morphology of the planar mesh surfaces. In micro or material scale, further studies can address mechanical properties of the connector elements, considering fatigue for passive joints and controlled actuation for active systems. This is important as the critical force to activate the reconfiguration can be adjusted by tuning the stiffness. The strength of different bi-stable joints can be mechanically tested. The set of produced prototypes shows that in some cases the sequence of activating the bi-stable joints is important (**Fig. 9**). While in some cases as it is tested in the prototype with an auxetic property, there is no sequence and the reconfiguration happens at once (**Fig. 10** and **Fig. 11**).

Acknowledgments

The research is supported by Delft University of Technology (TU Delft) and Dessau Institute of Architecture. The first author would like to express the gratitude to Taiwanese Ministry of Education for granting the scholarship. The prototyping is sponsored by the Science Center of TU Delft.

References

Haghpanah, B., Salari-Sharif, L., Pourrajab, P., Hopkins, J., & Valdevit, L. (2016). Architected Materials: Multistable Shape-Reconfigurable Architected Materials. *Advanced Materials*. https://doi.org/10.1002/adma.201670255

Huang, N. C., & Vahidi, B. (1971). Snap-through buckling of two simple structures. *International Journal of Non-Linear Mechanics*, 6(3), 295–310. https://doi.org/10.1016/0020-7462(71)90011-4

Konakovi , M., Crane, K., Deng, B., Bouaziz, S., Piker, D., & Pauly, M. (2016). Beyond developable. *ACM Transactions on Graphics*. https://doi.org/10.1145/2897824.2925944

Liu, Y., Pottmann, H., Wallner, J., Yang, Y.-L., & Wang, W. (2006). Geometric modeling with conical meshes and developable surfaces. *ACM SIGGRAPH 2006 Papers on – SIGGRAPH '06*, 681. https://doi.org/10.1145/1179352.1141941

McKinsey Global Institute. (2017). *MGI-Reinventing-construction-A-route-to-higher-productivity-Full-report*. McKinsey & Company. Retrieved from https://www.mckinsey.com/industries/capital-projects-and-infrastructure/our-insights/reinventing-construction-through-a-productivity-revolution

Rafsanjani, A., & Pasini, D. (2016). Bistable auxetic mechanical metamaterials inspired by ancient geometric motifs. *Extreme Mechanics Letters*. https://doi.org/10.1016/j.eml.2016.09.001

Tibbits, S., McKnelly, C., Olguin, C., Dikovsky, D., & Hirsch, S. (2014). 4D PRINTING AND UNIVERSAL TRANSFORMATION. In *ACADIA 2014 Design Agency: Proceedings of the 34th Annual Conference of the Association for Computer Aided Design in Architecture* (pp. 539–548).

van Manen, T., Janbaz, S., & Zadpoor, A. A. (2017). Programming 2D/3D shape-shifting with hobbyist 3D printers. *Mater. Horiz.*, 4(6), 1064–1069. https://doi.org/10.1039/C7MH00269F

Van Mele, T., Mehrotra, A., Mendez Echenagucia, T., Frick, U., Ochsendorf, J. a., DeJong, M. J., & Block, P. (2016). Form finding and structural analysis of a freeform stone vault. *Proceedings of the IASS Annual Symposium 2016 "Spatial Structures in the 21st Century,"* 1–10.

Beyond the basket case: A principled approach to the modelling of kagome weave patterns for the fabrication of interlaced lattice structures using straight strips.

Phil Ayres, Alison Grace Martin, Mateusz Zwierzycki

Phil Ayres
phil.ayres@kadk.dk
CITA, Denmark

Alison Grace Martin
alisonmartin57@gmail.com
Independant 3D Weaver, Italy

Mateusz Zwierzycki
mateuszzwierzycki@gmail.com
The Object, Poland

Keywords:
Kagome, triaxial weaving, basket weaving, textiles, mesh topology, mesh valence, mesh dual, fabrication, constraint modelling, constraint based simulation, design computation

Abstract

This paper explores how computational methods of representation can support and extend *kagome* handcraft towards the fabrication of interlaced lattice structures in an expanded set of domains, beyond basket making. Through reference to the literature and state of the art, we argue that the instrumentalisation of *kagome* principles into computational design methods is both timely and relevant; it addresses a growing interest in such structures across design and engineering communities; it also fills a current gap in tools that facilitate design and fabrication investigation across a spectrum of expertise, from the novice to the expert.

The paper describes the underlying topological and geometrical principles of *kagome* weave, and demonstrates the direct compatibility of these principles to properties of computational triangular meshes and their duals. We employ the known Medial Construction method to generate the weave pattern, edge 'walking' methods to consolidate geometry into individual strips, physics based relaxation to achieve a materially informed final geometry and projection to generate fabrication information. Our principle contribution is the combination of these methods to produce a principled workflow that supports design investigation of *kagome* weave patterns with the constraint of being made using straight strips of material. We evaluate the computational workflow through comparison to physical artefacts constructed ex-ante and ex-post.

1. Introduction

The term "weaving" covers a broad range of textile production methods. Common to all is the principle of material interlacing to generate local systems of friction-based reciprocity. This imbues resulting artifacts with robustness through structural redundancy, resilience through friction-based junctions, efficient use of material and potent aesthetic qualities. These attributes have long been exploited in a diverse range of use arenas, through craft-based tacit knowledge or engineering-based explicit knowledge, to produce lightweight artifacts with emergent properties that offer advantage beyond those of the constituent materials.

1.1 Kagome

Kagome represents a particular class of weave which, in many ways, is conceptually closer to braid. Where conventional weave is defined as the interlacing of two distinct sets of yarns (warp and weft) at right angles to each other, braid is defined as the interlacing of three or more distinct sets of yarns (or "weavers") at oblique angles to each other [1]. In *kagome*, the geometrical archetype arranges these three sets as a regular trihexagonal tiling with a vertex configuration $(3.6)^2$ and p6 symmetry.

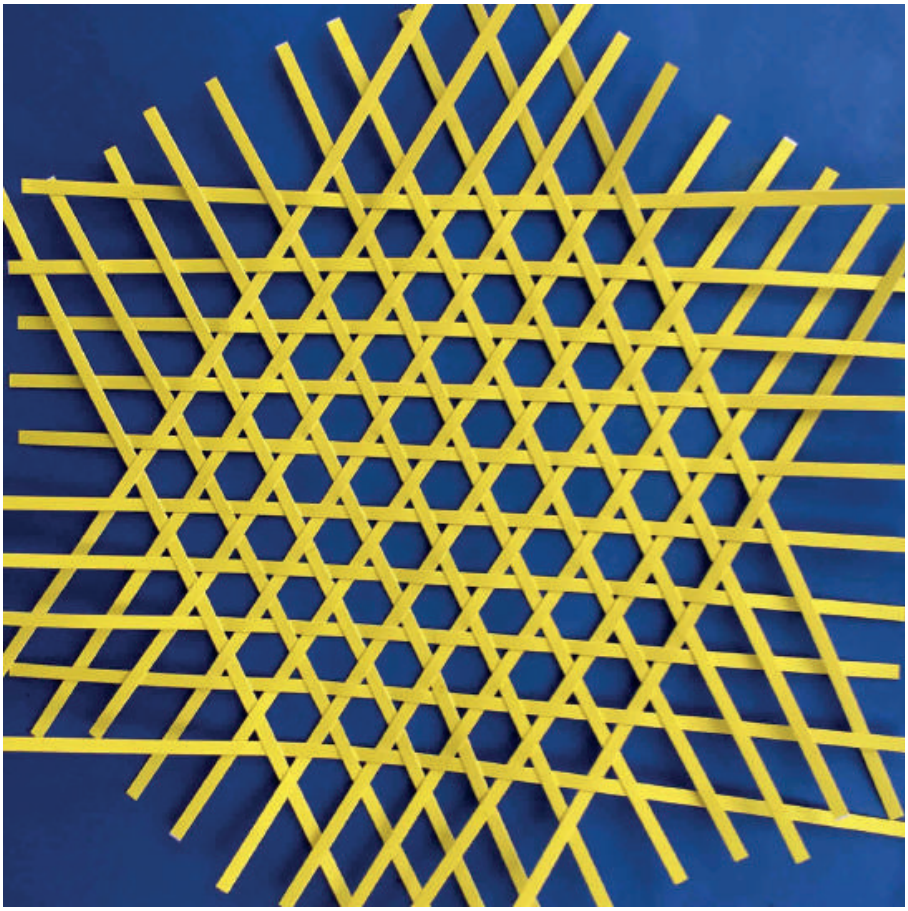


Figure 1: A regular planar sparse *kagome* weave comprising three distinct sets of weavers. The underlying pattern is a trihexagonal tiling.

The physical properties of these lattices are determined by the interplay between combinatorics (valences of vertices and faces, connectivity, and topology), geometry (vertex positions) and material attributes (mechanical and geometric). Tacit understanding of this interplay allows the crafts person to fabricate close approximations of arbitrary design targets.

1.2 Motivation

Kagome represents a highly principled method for producing complex curved geometries with a single mesh structure, without the necessity of joinery or the fabrication of nodes. The self-bracing capacity, greater shear resistance (compared to biaxial weave), ability to rigorously control geometry, high redundancy, ability to locally repair and potent aesthetic qualities, make triaxially woven structures an attractive target for investigation across a diverse range of design and craft practices, including architecture. However, without means for visualisation and interrogation, complex design targets can remain challenging for experts to strategise and realise (keeping account of the number of weavers, their crossings and potential self-crossings, calculation of material requirements, assessing discretisation due to material lengths, etc.), and remain out of reach for those without a tacit craft understanding.

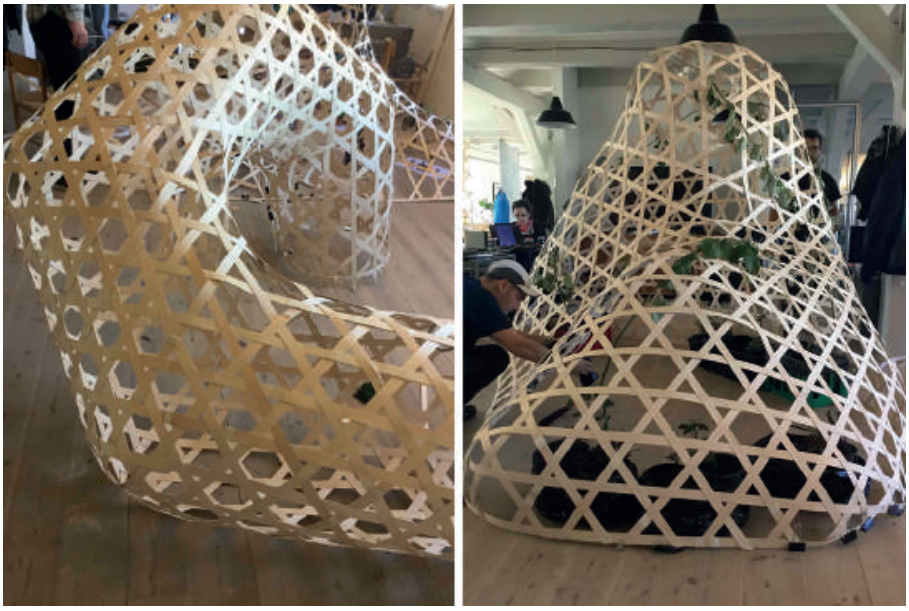


Figure 2: Triaxially woven structures produced using straight maple strips. Regular (left) and arbitrary (right) geometries are clearly governed by the interplay between introduced topological defects, material stiffness and material geometry.

By intersecting the underlying principles governing the interplay of topology and geometry in triaxial systems with computational representation, a platform for expanded exploration of these systems can be established. This holds relevance to a wide variety of current and emerging domains of application.

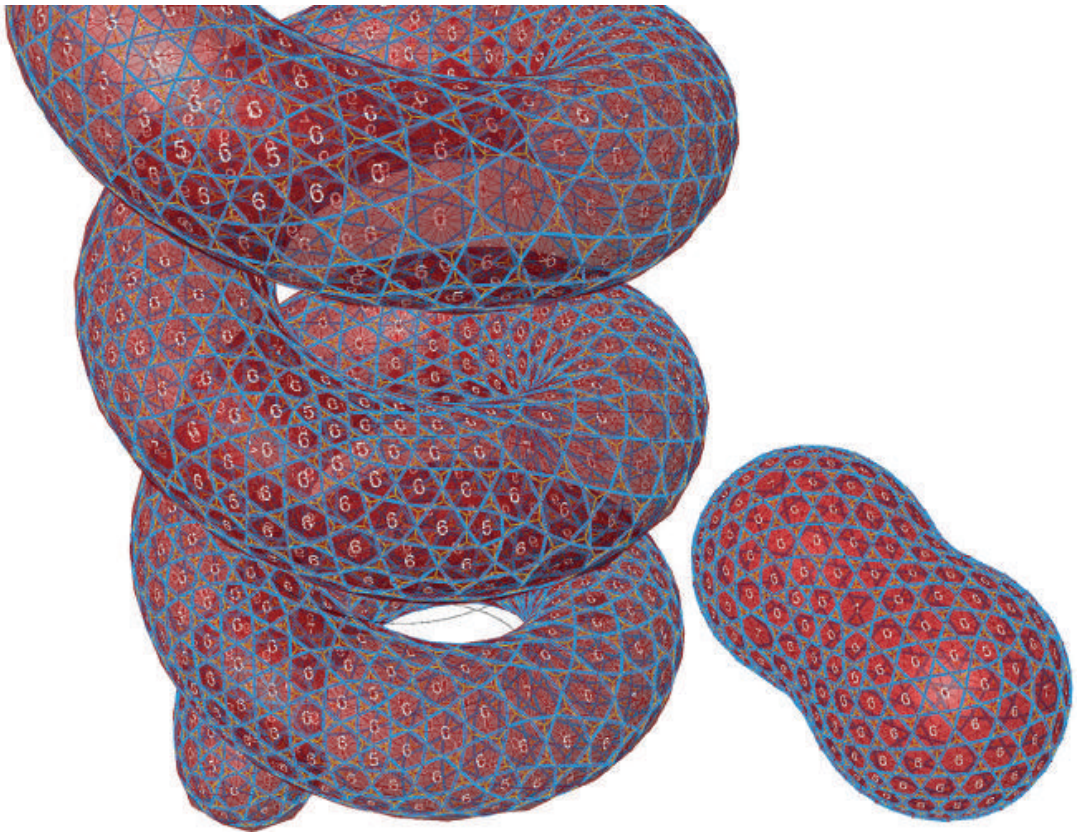


Figure 3: Two pre-relaxed *kagome* patterns approximating design geometries. The weave topology is directly derived from inherent properties of the design mesh (valence) and the weave pattern directly derived from geometric attributes of the mesh dual (connected edges and their lengths).

In this paper, we present a method for generating weave patterns with the constraint that they be fabricated from straight strips of material. Our motivation for working with straight strips relates to supporting the future exploration of *kagome* applications at scales "beyond the basket", where efficient use of material becomes a poignant issue. We address key representational challenges including the generation of appropriate topology, or mesh valence, to achieve a design target, together with the relaxation of the mesh to simulate material performance – both of which hold influence over final shape. In addition, we demonstrate the extraction of fabrication instruction and the physical making of computationally developed design targets. We position this work in connection with the literature to: 1) differentiate it from related approaches (specifically related to the use of geodesics); 2) identify the open challenge that our work addresses; 3) cite computational methods that we build upon. Finally, we discuss our contribution, identify its limits and offer trajectories for future work.

2. Topological principles governing kagome geometry

The archetypal *kagome* lattice is a woven version of a tri-hexagonal tiling; the weavers in one direction incline at an angle of 60° to those of the other two directions, and the lattice, consisting of equilateral triangles and regular hexagons, will cover an infinite flat plane (**Fig. 1**).

2.1 Single curvature

Single curvature of the *kagome* lattice is easily achieved by bending the plane, creating a developable surface. If the axis of curvature exists across the centre points of opposite edges in the unit hexagon, one set of weavers will act as arches perpendicular to this axis. If the axis of curvature exists across opposing vertices of the unit hexagon, one set of weavers will act as beams parallel to this axis. Limits on the radius of curvature are dependent on the mechanical properties of the material.

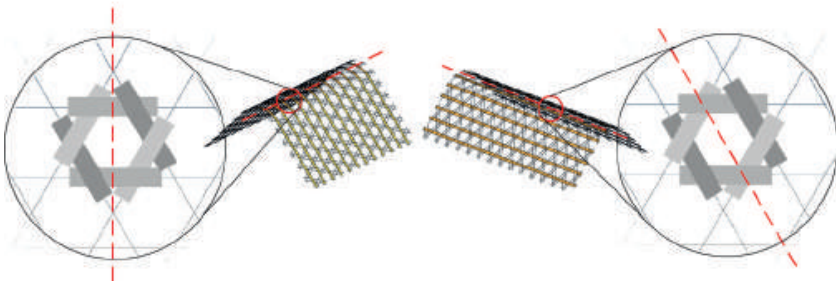


Figure 4: Single curvature is easily achieved in the regular triaxial lattice and can follow any line of hexagonal symmetry – across opposite edge centres privileges arches, across opposite vertices privileges beams.

2.2 Double curvature

Breaking topological symmetry of a regular trihexagonal tiling by the introduction of geometric singularities will induce double curvature [2]. These topological defects, or "lattice disclinations", are the mechanisms that introduce in-plane strains and result in out-of-plane deformation [3]. Positive Gaussian curvature results from the introduction of <6 sided cell. Figure 5 shows physically woven examples in which a single cell has been substituted; firstly with a pentagon, then a quadrilateral and finally a triangle. Of note is the way in which deformation out-of-plane becomes more pronounced as edges are removed from the substituted polygon. Figure 6 shows physically woven examples of negative Gaussian

curvature resulting from cell substitution with a polygon of side >6; firstly a heptagon, then octagon and finally a nonagon. Here, it is the increase in sides of the substituted polygon that results in a more pronounced curvature. Despite changes in topology through the introduction of disclinations, the vertex valence of the materialised lattice is maintained at v4 throughout, corresponding to the local crossing of two weavers.

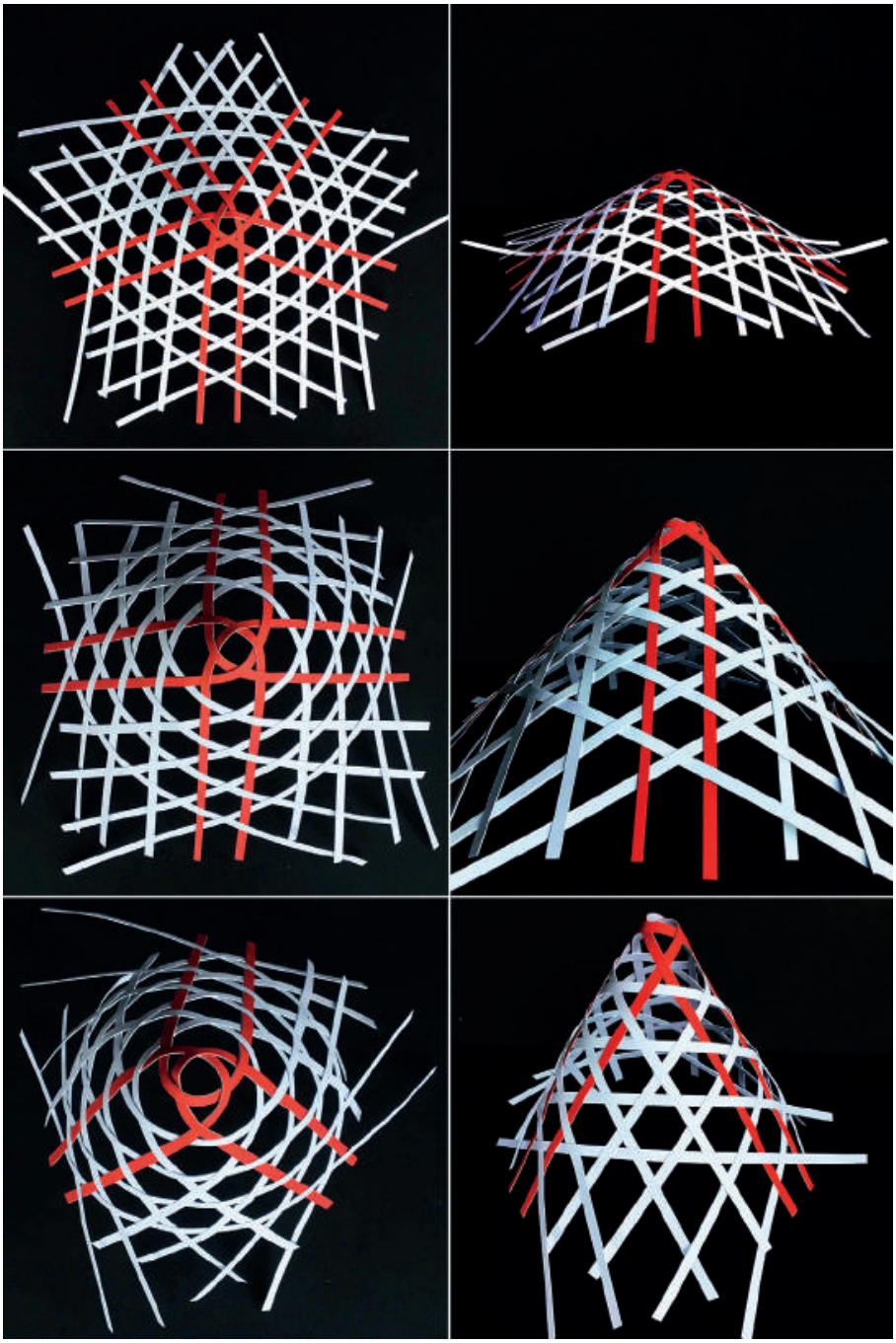


Figure 5: Introducing disclination in the regular lattice by substituting a <6 edge count polygon produces positive Gaussian curvature. From top to bottom, each row decreases an edge – pentagon; quadrilateral; triangle.

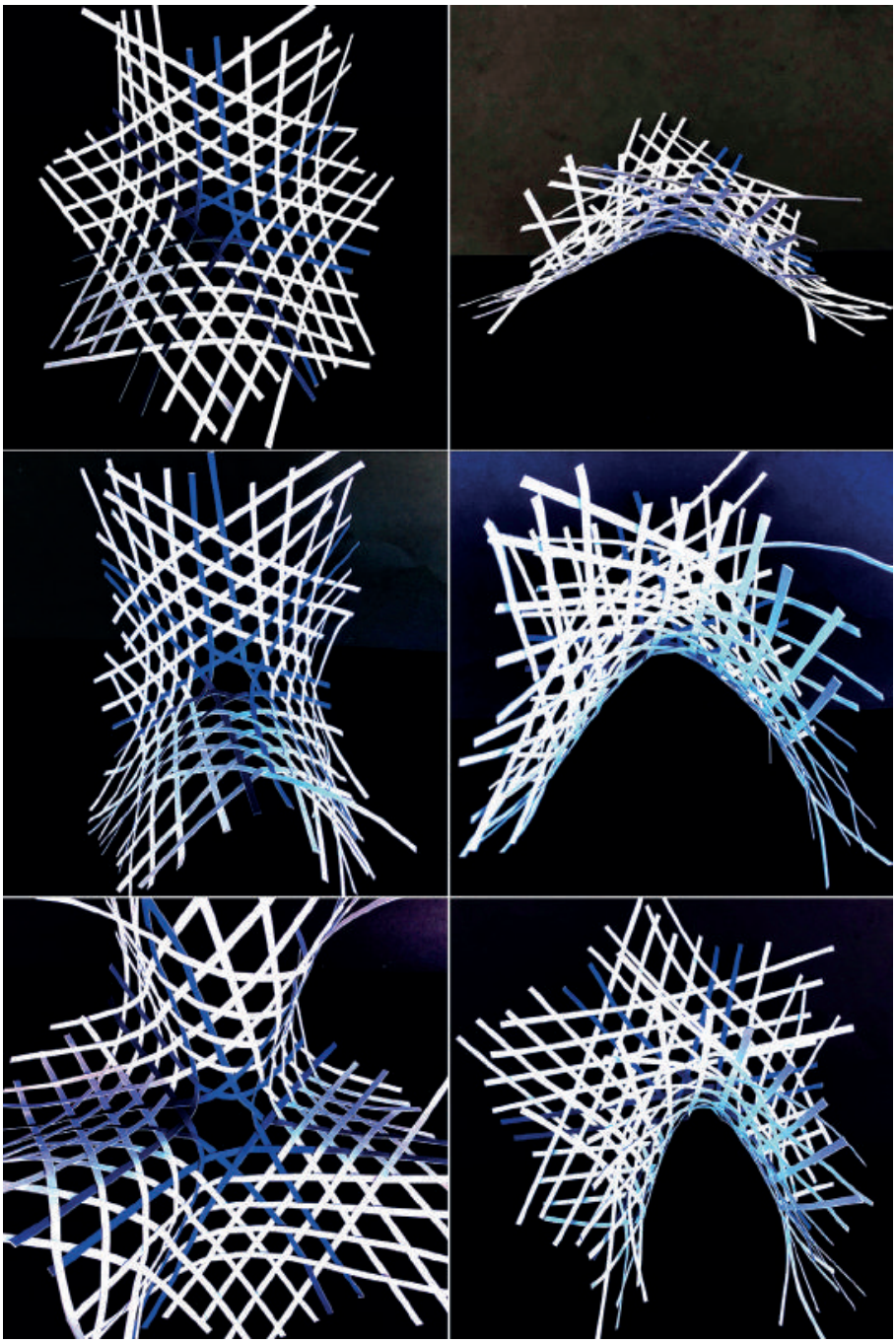


Figure 6: Introducing disclination in the regular lattice by substituting a >6 edge count polygon produces negative Gaussian curvature. From top to bottom, each row increases an edge – heptagon; octagon; nonagon.

Weaving disclinations provides the means to locally distort the lattice, causing a controlled deformation of the surface out of plane. Strategic combinations of disclinations, informed through tacit knowledge, allow the crafts person to realise specific and diverse design intent (**Fig. 7**). However, in an inexhaustible space of possible combinations, enlisting computation becomes a relevant tool for exploring, searching and navigating this space.

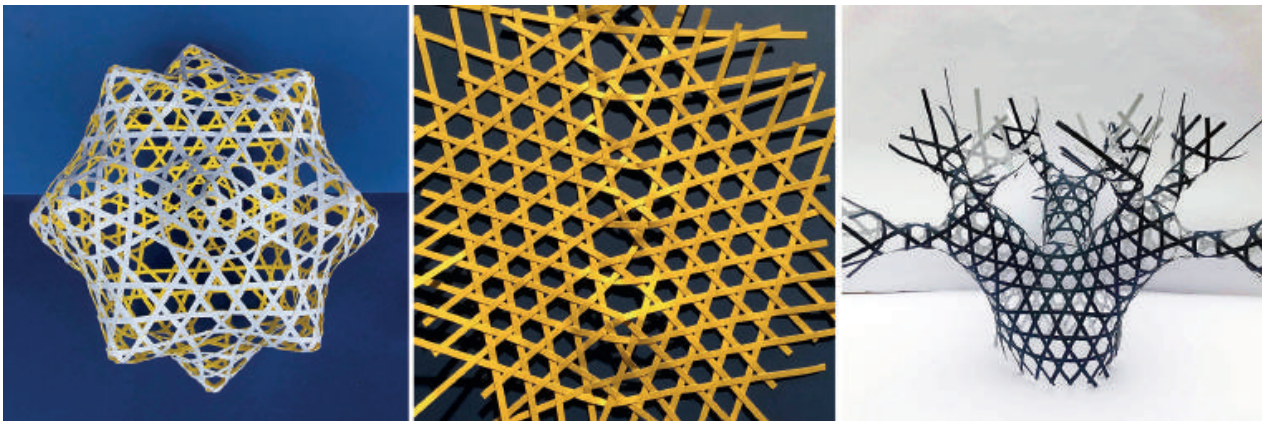


Figure 7: A diverse variety of artefacts demonstrating results from strategic combinations of disclinations.

3. State of the art

In this section, we highlight relevant literature restricted to computational representation of weave patterns and related computational methods with a particular focus towards architectural design. We briefly cover methods for establishing and refining mesh topologies, approaches to weave in general, approaches to *kagome* representation in particular and provide a summary that identifies the open challenge that we address.

3.1 Mesh topology and refinement

With a focus on mesh representations that have relevance to architecture, Schiftner et al. provide a method for refining triangular design meshes such that the incircles of mesh faces form a packing – a CP mesh [4]. This class of mesh is directly related to the *kagome* pattern, which can be produced by connecting the centres of tangent incircles. As precise CP meshes are rare, an optimisation algorithm is used to refine a mesh towards an approximation of the design target. The mesh is generated by producing an isotropic centroidal Voronoi diagram which is iteratively relaxed using Loyd's algorithm. However, this leads to random placement of singularities which is undesirable if aiming to achieve regular geometries. Use of the mesh operators, edge collapse, edge flipping and edge splitting is a common method for locally refining the topology of mesh as described in Narain et al. [5] and allows approximate locating of required valence in the required position.

3.2 Approaches to weave pattern representation in general

Computational representation of weave patterns in general have been well studied, however, the majority of these relate to biaxial weaving or braiding. In most cases, the representation task is approached using the tiling method described by Mercat [6] in which a predefined tile dictionary defining local weaver geometry and crossings can be applied to a quad mesh. This has been applied in the context of arbitrary manifold design meshes [7], and with specific focus on braided structures [8, 9]. In these cases, the principled approach to representation, which considers interlacing and constraints related to fabrication, provides workflows and tools for realising complex morphologies that are directly producible. However, these tools operate with quad meshes which are less suited to the *kagome* representation task. In another approaches, modelling proceeds through direct manipulation of explicit geometry [10]. This is not deemed to be a viable approach for the task considered here, considering the opportunity for exploiting the close affinity between the data structures of triangular meshes and *kagome* pattern principles, and the culture of use surrounding meshes for design expression.

3.3 Approaches to kagome pattern representation in particular

Within architectural design specifically, approaches for defining *kagome* patterns tend to exhibit shortcomings by either: 1) only considering a topologically regular trihexagonal tiling; 2) exploring geometrical outcomes of fixed and predetermined topologies; 3) abstracting out the weaving principle such that the mechanical properties gained by interlacing are sacrificed, whilst maintaining the topology of the trihexagonal tiling.
In the first case (which is often coupled with the third case) complex geometries are achieved by a distortion of the regular grid rather than conforming to the principles for achieving curvature described in the section above [11, 12]. This can present significant challenges for fabrication strategies, junctioning methods and material use. In the second case, relaxation of pre-determined and fixed topologies can result in principled patterns, but impedes fluid design investigation due to a lack of “on-the-fly” topology editing methods.

Kagome patterns have also been explored as a derivative of a general approach to free-form surface segmentation using geodesic pattern

generation [13, 14]. The cited literature describes two approaches – *N*–patterns from level sets, and the use of a regular trihexagonal web of geodesics – but also identify limits in both cases. Pottman et al. acknowledge that the level set approach produces webs of curves that are as geodesic as possible, but deviations are inevitable in conditions of strong Gaussian variance [13]. Deng et al. point to the fact that true geodesic webs do not exist in general and that adequate surface approximation is not always possible [14].

In contrast to these geodesic methods, which operate from properties of a surface (which in practice is generally approximated by a mesh), our approach operates directly on properties of the mesh and form-finds the final geometry through a relaxation procedure. This models the actions of the local reciprocal systems, which, in practice, we find causes material strips to deviate from true geodesics due to induced torsions often arising in areas of pronounced double curvature. In short, the use of geodesics to derive *kagome* patterns cannot cover all cases that can be materialised in practice, whereas a principled *kagome* pattern can always be derived from a manifold triangular mesh [15].

The strong affinity between *kagome* lattice patterns and computational triangular manifold meshes have been described by Mallos and implemented in the context of a *kagome* design and fabrication tool [ibid]. However, to our knowledge, this tool does not implement a step that allows the consideration of *kagome* patterns resulting from straight members – a case that requires relaxation of the *kagome* geometry with specific relaxation constraints.

3.4 Identifying the open challenge

In summary, and in reference to the state-of-art presented here, we can state that whilst there exist a number of methods and algorithms related to the *kagome* representation task, to the best of the authors knowledge, a holistic computational approach that aids designers by coupling specific fabrication constraints with the principles for “real-time” exploration of arbitrary *kagome* topologies and geometries, remains an open challenge.

4. Computational approach

Our approach to achieve a principled and generalised method for *kagome* representation, of arbitrary geometries, makes use of various algorithms and methods described in the literature; we declare these below. The contribution of this paper is to draw these together to fulfill the representation task with a focus on fabrication using straight strips of material. The representation task has three stages:

1. topology generation
2. *kagome* pattern generation
3. relaxation to final geometry

4.1 Topology generation

Using the low-polygon modelling method [16], a coarse triangular mesh approximation of the desired geometry is created. In the example shown, the target geometry to model is a existing *kagome* "socket" condition comprising a regular planar face intersected by a singularly curved tube. The transition exhibits negative Gaussian curvature (**Fig. 8**).

The topology of the low-poly mesh is adjusted to establish the required valence structure. Adjustment is done using conventional mesh refinement operations; edge splitting, edge flipping and edge collapsing [5].

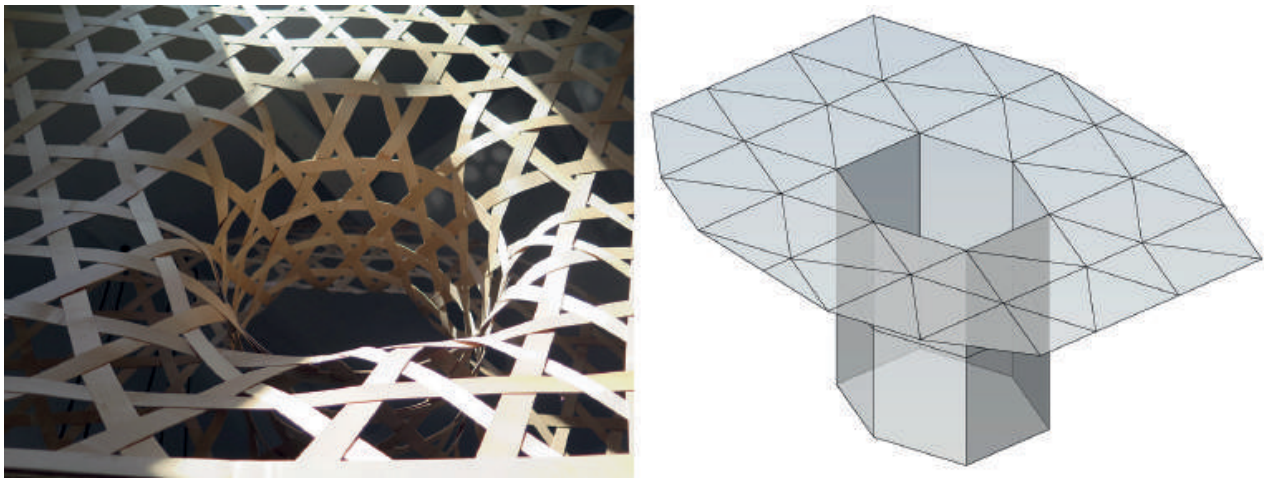


Figure 8: The target geometry to model is a detail of an existing *kagome* weave with negative Gaussian curvature (left). This is coarsely approximated with a low-polygon mesh (right).

Mesh valence of a regular planar tiling is 6, positive Gaussian curvature requires <6 (but >2) valence and negative curvature requires >6 valence. In this case, six valence 7 conditions around the rim of the transition and regular valence 6 conditions to the stem have been introduced. Once the refined valence structure is established, intermediary mesh operations such as relaxation (as in the case shown in **Fig. 9**) or mesh subdivision can be applied.

4.2 Kagome pattern generation

The mesh dual is obtained and decomposed into a data structure of individual vertices and their three connecting edges. A new vertex is then placed at the centre of each connecting edge and these three new vertices connected with a closed polyline. This operation essentially truncates the original vertex, creating a new facet that represents the triangular element in the *kagome* lattice. The operation is equivalent to the medial construction method described by Mallos [15]. At this point, the weave pattern is purely visual and contains no information about weaver continuity; all higher edge faces of the lattice are visually inferred from their surrounding triangles.

The list of truncated face polylines is now converted into a data structure that represents individual weavers. The polylines are exploded into individual linear elements and then "walked" to find connected segments that meet a criteria of minimum angular deviation. Once weavers have been identified, they are locally displaced in an alternating pattern (up/down) along the surface normal vector at crossing points to model interlacing. Once interlaced, each weaver is converted into a triangular mesh approximating the material strip width using the method described by Vestartas et al. [9]. At this stage, meshes may exhibit areas of intersection as can be seen in Figure 11 (right).

4.3 Relaxation to final geometry

The weaver meshes are relaxed using the constraint-based solver Kangaroo2 for Grasshopper. Additional constraints are added to ensure weavers relax into developable geometries approximating straight strips, and to prevent collisions and intersections between weavers – thus preserving the structure of interlacing. Having found the final geometry

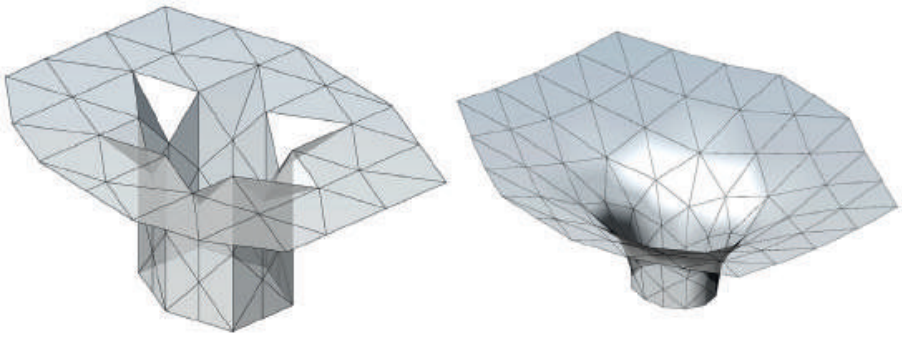


Figure 9: The mesh is refined by collapsing, splitting and flipping edges to modify the valence according to the required curvature (left). A preliminary relaxation has then been performed after adding an additional layer of outer triangles in the plane to encapsulate the valence 7 conditions (right).

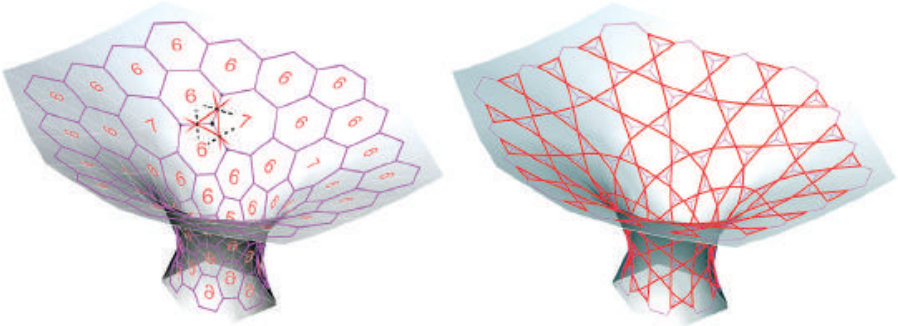


Figure 10: The mesh dual is obtained (left) and each vertex "truncated" to generate a visual kagome pattern (right). This pattern does not yet describe individual weavers.

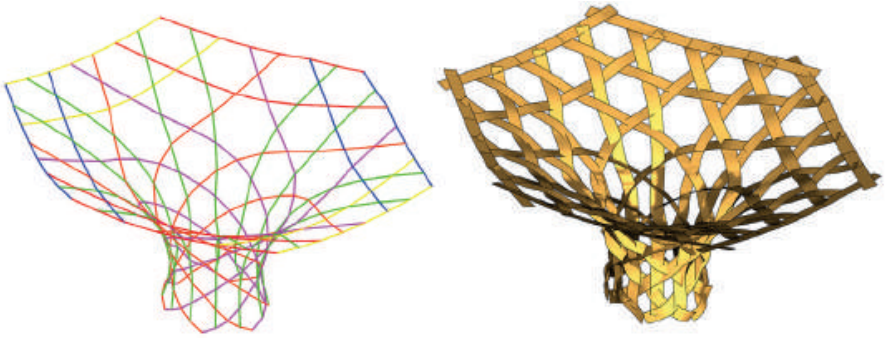


Figure 11: The edges of the kagome pattern are "walked" to construct individual weavers (left). Weavers are then displaced normal to the surface to model interlacing, and then meshed according to material geometry (right).

through relaxation, fabrication information can now be extracted (**Fig. 12**). Weaver lengths are easily determined, and being developable, projected as unrolled strips and marked with crossing points indexed with other weavers or self-intersections. Physical limits on material length can inform weaver discretisation, ensuring sufficient material cross-over for splicing.

5. Two cases

In this section we briefly present two case studies that examine relationships between a computational representation and a physical artefact – one constructed ex-ante and the other ex-post modelling. The first study demonstrates the use of our approach in the context of a simple fabrication exercise. The second study demonstrates the use of our approach in the context of computational design exploration.

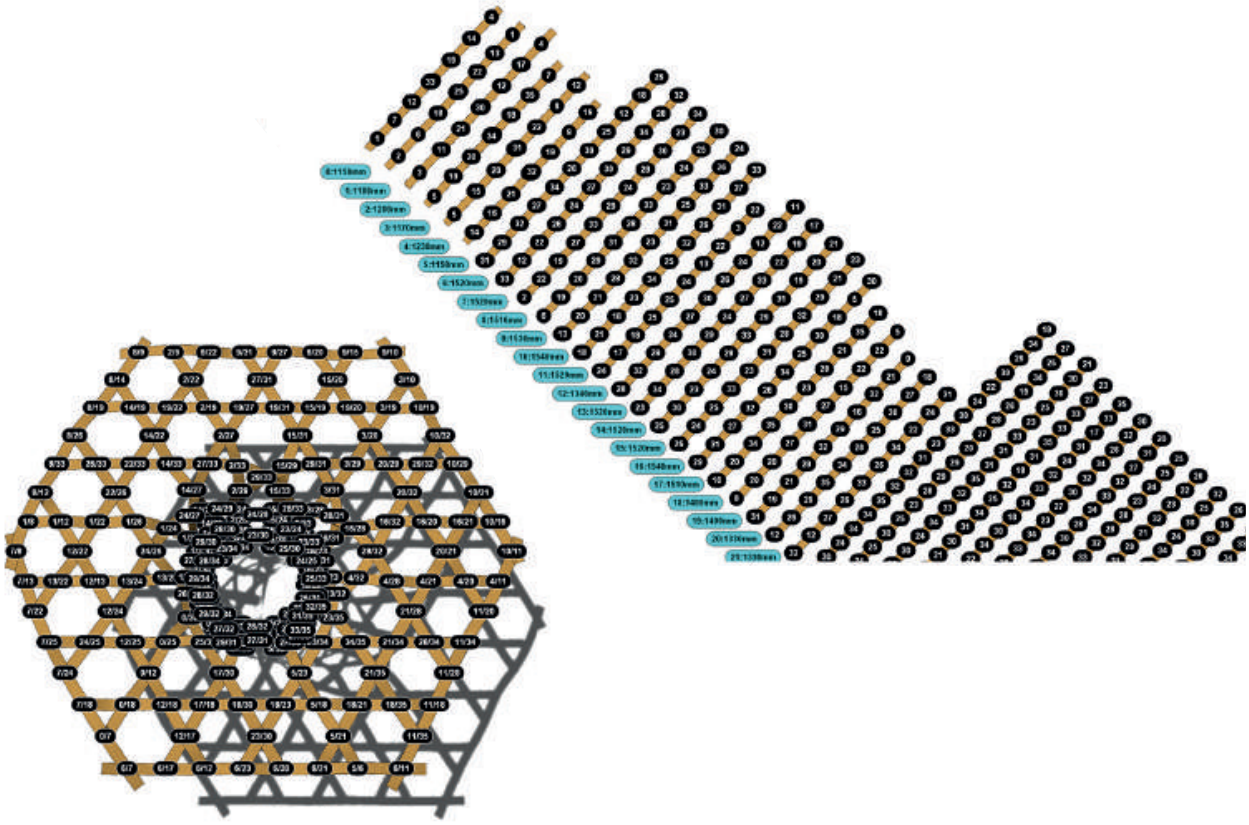


Figure 12: The modelled weavers are relaxed to ensure they correspond to straight elements and the final weave geometry is form-found. Fabrication information is then extracted and includes length of strips, strip ID's and strip crossing ID's. This information is applied to the weave representation (left) and as material layout (right).

5.1 Case 1: Stadium of revolution

In this first case, we aim to construct a physical weave from computationally generated fabrication information. A stadium of revolution, or "capsule" geometry, is defined as the design target. This geometry comprises a cylinder with single curvature and two hemispherical caps. Drawing upon the principles governing double curvature in *kagome* lattices, we expect the hemispherical portion to contain pentagonal "defects" to achieve local synclastic curvature. Each pentagon included in the mesh increases the aggregate angular deficiency by $\varpi/3$, therefore a triaxial mesh with 6 pentagons will make a hemisphere. The rest of the lattice can be achieved using a regular

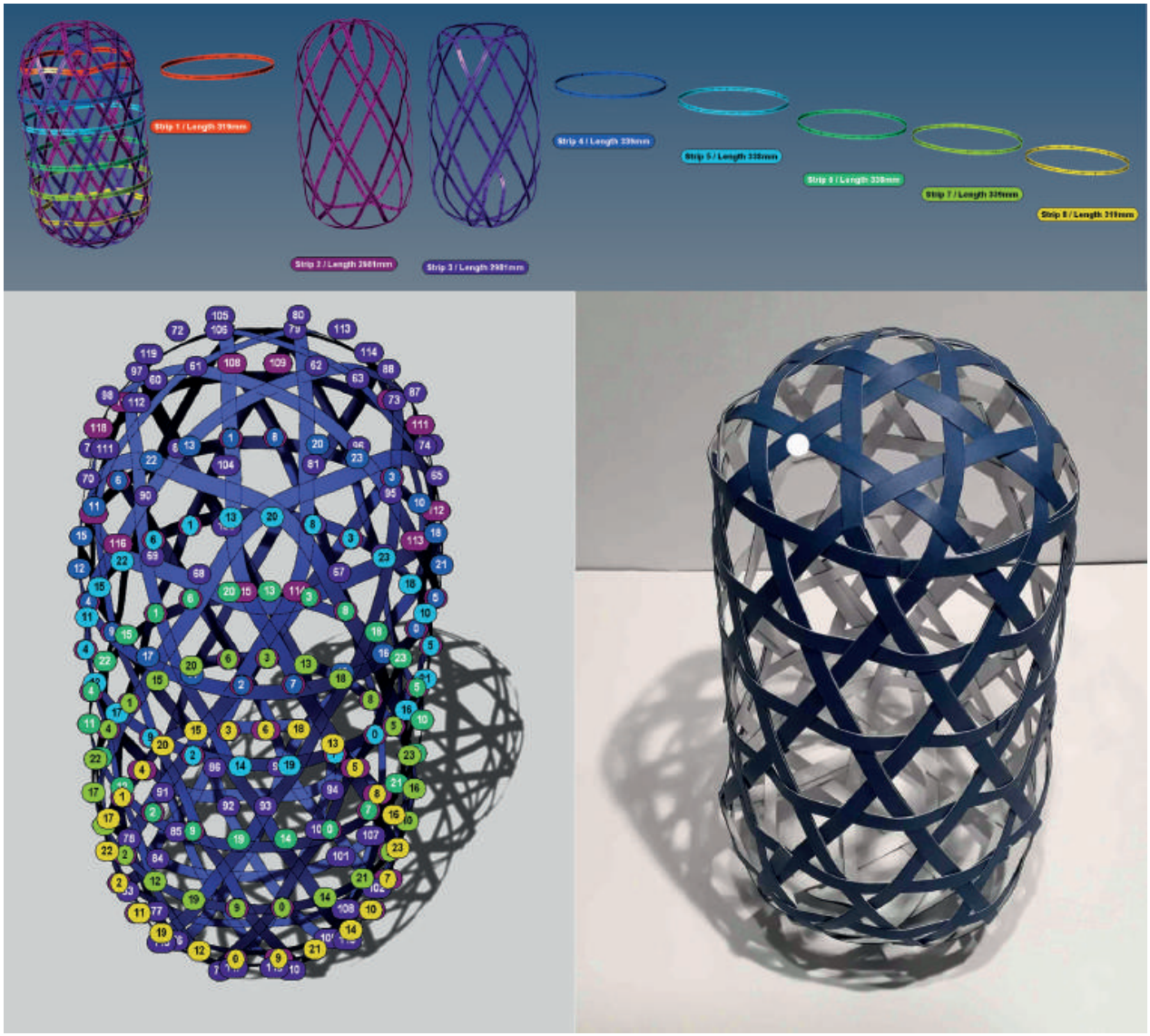


Figure 13: Extraction of fabrication information to produce a woven stadium of revolution.

hexagonal tiling. We follow the modelling steps described in section 4 to determine how many weavers, their respective lengths, crossings with other weavers and self-crossings. We see from this analysis that the woven figure comprises 6 simple rings of length cca. the circumference of the cylinder, and two longer weavers with multiple self-crossing points. This is verified with the physically weaving shown in Figure 13 (bottom right).

5.2 Case 2: The distorted helix

In this second case, the *kagome* helix is woven prior to any computational modelling. Rather than aiming towards verisimilitude of the model, we demonstrate how the relaxation step can provide exploratory insights through simulating the interplay of material behaviour and topology. The helix is modelled and the mesh refined, but in this case disclinations are randomly placed within the mesh. As the relaxation proceeds and weaver geometries straighten according to our fabrication constraints, and local sites of curvature emerge where hexagons have been substituted with synclastic curvature inducing pentagons, or anticlastic curvature inducing heptagons. In this case, we demonstrate how computation provides an accessible and fast (compared to physical weaving) exploratory tool to assist the designer in searching the inexhaustible space of possible disclination combinations, and potentially discovering novel aesthetic expressions.

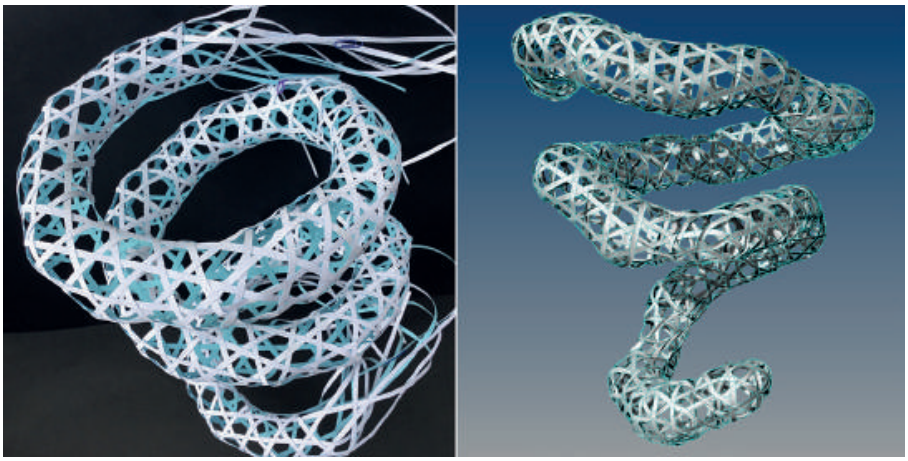


Figure 14: A physically woven helix with mesh disclinations placed to realise a regular geometry (left) compared to a simulation where disclinations have been randomly located (right). This shows the necessity for the relaxation step, but also suggests interesting geometric articulations and “organic” expressions of a corrupted ideal.

6. Towards architectural and structural applications

the instrumentalisation of a principled computational approach to *kagome* pattern generation and representation has broad applicability. Within architecture, hexagonal tiling patterns have been exploited to stunning spatial effect by Shigeru Ban in projects such as the Pompidou Metz and Nine Bridges golf club. However, in these cases, double curvature is achieved through a distortion of the regular hexagonal tiling. The resulting geometry is realised through complex shaping of stiff curved laminated members. In such a context, the application of *kagome* topology principles for achieving complex geometry could offer a more rational approach to geometry with the implication of greater efficiency in fabrication.

In the context of elastically bent structures, the attributes of mechanical performance arising from interlaced material and efficient spanning of space with straight strips of material have been demonstrated in the CODA Jukbuin Pavilion. In this case, the weave principle of material interlacing is maintained but double curvature is achieved through material bending behaviour rather than steered by topology – the design topology is a regular hexagonal tiling. This results in global curvature effects but denies the possibility of highly localised double curvature. Nevertheless, this work is of particular interest as it demonstrates the transfer of interlacing principles at architectural scale.

In framing a direction for future work, our emerging hypothesis is that architectural scale structures can be realised with full adherence to *kagome* weaving principles, including material interlacing. This hypothesis is supported by a comparative analysis of two hypothetical gridshells which shows that a *kagome* gridshell outperforms a quadrilateral gridshell for a very similar construction cost [17].

Our outlook is towards the use of elastically bent members rather than stiff curved laminated members. However, as we discuss above, we see *kagome* principles being applicable in both contexts - in the former, towards bending-active structures that adhere more closely to their basket antecedents; in the latter, towards rationalised approaches to geometry and fabrication. In the context of elastically bent structures, principle challenges revolve around structural capacity. Yet despite this challenge, the opportunities for material efficiency, a rationalised approach to free-form geometry and efficient fabrication minimising the use of connectors make this a compelling territory for further study.

6.1 Limits and future work

Where the work presented in this paper has limited itself to exploring the task of *kagome* representation and simulation with consideration to fabrication constraints, analysis of structural performance marks a necessary next step – especially if seeking to explore architectural applications. Preliminary investigations of model transfer to the structural analysis platform Autodesk Robot indicate that representational outcomes generated by the approach described are poised to be taken forward into this domain of analysis. In addition, the ability to computationally represent arbitrary *kagome* geometries and interrogate these from a fabrication perspective, presents the compelling opportunity of investigating robotic production.

7. Conclusion

This paper has presented a principled computational approach to the task of *kagome* representation in arbitrary triangular meshes. Following the literature, we have demonstrated the strong affinity between the principles governing *kagome* patterns and intrinsic topological features of computational meshes and geometric features of their duals. We have shown how design meshes can be manipulated to adjust the baseline valence 6 structure that governs planar *kagome* tiling, upwards and downwards to create sites of local double curvature. We have also shown how the *kagome* pattern itself is derived from the mesh dual by vertex truncation to the mid-points of connected edges – following the medial construction method.

We have extended the state-of-the-art by intersecting this method with physics based relaxation to allow simulation of the interplay between topology and notional mechanical properties of weaver material, thereby constraining results within the bounds of fabrication criteria – specifically that patterns can be made from straight strips of material. This constraint is seen to be a benefit for enticing transferability and use within domains where material saving can be a key issue, such as architecture.

FInally, the approach presented here contributes a method that can be computationally leveraged to explore and search the inexhaustible domain of possible *kagome* patterns, and opening the possibility of this search to be conducted by both the novice and the expert.

Acknowledgements

This work was partially supported by the European Union's Horizon 2020 research and innovation program under the project "flora robotica", FET grant agreement no. 640959. The authors gratefully acknowledge the participation of the 2017/18 master students enrolled at the Master of Architecture programme CITAstudio: Computation in Architecture, KADK in the "Triaxial Weaving & Bio-receptivity" workshop conducted over one week in September 2017. Five large scale woven structures were produced, two of which are shown in Figure 2. The authors also wish to thank the anonymous peer reviewers for their comments and suggestions to improve the quality of the paper.

References

[1] Rana, S. and Fangueiro, R., editors. 2015. Braided structures and composites: production, properties, mechanics, and technical applications. CRC Press.

[2] Martin, A. G. 2015. A basketmaker's approach to structural morphology. In *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium*.

[3] Callens, S. J. and Zadpoor, A. A. 2017. From flat sheets to curved geometries: Origami and kirigami approaches. *Materials Today*.

[4] Schiftner, A., Höbinger, M., Wallner, J. and Pottmann, H. 2009. Packing circles and spheres on surfaces. In *ACM Transactions on Graphics (TOG)*, Vol. 28, No. 5, 139.

[5] Narain, R., Samii, A. and O'Brien, J. F. 2012. Adaptive anisotropic remeshing for cloth simulation. *ACM transactions on graphics (TOG)*, 31(6), 152.

[6] Mercat, C. 1997 Les entrelacs des enluminures celtes. In *Dossier Pour la science*, 15.

[7] Akleman, E., Chen, J., Xing, Q. and Gross, J. L. 2009. Cyclic plain-weaving on polygonal mesh surfaces with graph rotation systems. In *ACM Transactions on Graphics (TOG)*, Vol. 28, No. 3, 78.

[8] Zwierzycki, M., Vestartas, P., Heinrich, M. K. and Ayres, P. 2017. High Resolution Representation and Simulation of Braiding Patterns. In *Acadia 2017.*

[9] Vestartas, P., Heinrich, M. K., Zwierzycki, M., Leon, D. A., Cheheltan, A., La Magna, R. and Ayres, P. 2018. Design Tools and Workflows for Braided Structures. In *Humanizing Digital Reality*, Springer, Singapore, pp. 671–681.

[10] Harnomo, F. I., and Indraprastha, A. 2016. Computational Weaving Grammar Of Traditional Woven Pattern. In *Parametricism Vs. Materialism: Evolution of Digital Technologies for Development [8th ASCAAD Conference Proceedings]*, pp. 75–84.

[11] Digital [Sub]stance, *kagome* Weave Grasshopper Script (GH 0.8.0063). Download https://digitalsubstance.wordpress.com/subcode/ (Accessed 11.09.2017)

[12] Scheurer, F., Stehling, H., Tschümperlin, F. and Antemann, M. 2013. Design for assembly—digital prefabrication of complex timber structures. In *Beyond the Limits of Man*, Proceedings of the IASS 2013 Symposium, Wroclaw.

[13] Pottmann, H., Huang, Q., Deng, B., Schiftner, A., Kilian, M., Guibas, L. and Wallner, J. 2010. Geodesic patterns. *ACM Transactions on Graphics*, Vol. 29, No. 4, p. 43.

[14] Deng, B., Pottmann, H. and Wallner, J. 2011. Functional webs for freeform architecture. *Computer Graphics Forum,* Vol. 30, No. 5, pp. 1369–1378.

[15] Mallos, J. 2009. How to weave a basket of arbitrary shape. *ISAMA*, 2009, pp.13–9.

[16] Steed, P. 1998. The art of low-polygon modeling. *Game Developer*, pp.62–69.

[17] Mesnil, R., Douthe, C., Baverel, O. and Léger, B. 2017. Linear buckling of quadrangular and *kagome* gridshells: a comparative assessment. *Engineering Structures*, 132, pp.337–348.

Computational design of deployable auxetic shells

Mina Konaković-Luković, Pavle Konaković, Mark Pauly

Mina Konaković-Luković
mina.konakovic@epfl.ch
École Polytechnique Fédérale de Lausanne, Switzerland

Pavle Konaković
pavle.konakovic@ecal.ch
École cantonale d'art de Lausanne, Switzerland

Mark Pauly
mark.pauly@epfl.ch
École Polytechnique Fédérale de Lausanne, Switzerland

Keywords:
Auxetic structures, deployable shells, form finding, computational design, numerical optimization

Abstract

We propose an interactive computational design method for deployable auxetic shells. We realize deployable auxetics as triangular linkages that can be actuated with simple expansive mechanisms to assume a desired freeform target shape. The core feature of these structures is that the target shape is directly and uniquely encoded in the 2D linkage layout. As a consequence, the structure can be fabricated and assembled in the plane and automatically deployed to its 3D target configuration without the need for any scaffold, formwork, or other temporary support structure. We focus on automatic deployment via inflation or gravitational loading for which a rigorous theoretical analysis has been given in prior work. Our paper builds upon these results and presents optimizationbased direct manipulation tools to edit and adapt an auxetic linkage structure to effectively explore design alternatives. In addition, our solution enables simulation-based form-finding, where the desired target surface is interactively constructed using the deployment mechanism as a form-finding force. We present several design case studies that demonstrate the effectiveness of our approach and highlight potential applications in architecture.

1. Introduction

Architectural structures are commonly composed of multiple elements that are assembled on-site. Construction is executed by incrementally placing components at their target 3D location, using scaffolding or other support mechanisms to guide element positioning and maintain structural stability during intermediated stages of the assembly. Especially for intricate free-form geometry, the complexities of this process can pose severe challenges. Deployable structures offer an interesting alternative for construction. They typically can be assembled in a significantly simpler state and then deform to the desired target shape. A prominent example is elastic grid shells that can be assembled on the ground and mounted into a double-curved form, see Lienhard (2014).

We propose a computational design system for a new type of deployable structure based on a triangular auxetic linkage. Our structures can be fabricated and assembled in the plane and deployed to their target position using either inflation or gravity. No additional guiding

scaffold is required because the target shape is *directly encoded* in the planar assembly. The key concept is a spatially graded auxetic pattern, where individual triangular elements are scaled to *program* the maximal local expansion factor required to achieve the global target shape. Paired with an area-expanding deployment, such as air-inflated cushions or gravitational loading, this yields a simple and robust way to realize double-curved surface structures.

Deployable auxetics offer a number of benefits:

» *Form-defining deployment*: The double-curved target shape is automatically achieved via expansive deployment from a planar configuration. Inflation or gravitational loading (for height field geometry) can be used to maximally stretch the material everywhere, which then constrains the surface to the desired target configuration.

» *Simple fabrication*: The geometric simplicity of the auxetic linkage directly transfers to fabrication. Variable-sized triangles can easily be cut using CNC fabrication technology from a wide variety of approximately inextensible base materials, such as fabrics, wood, metals, or plastics. Mass fabrication of joints is possible since all node connections are identical.

» *Rich geometry*: Deployable auxetics admit a rich and well-defined design space, enabling new forms beyond the existing classes of structures deployable from planar rest states.

This paper complements the work of Konaković-Luković et al. (2018) who proposed a postrationalization process to find a deployable auxetic linkage for a fixed input design surface. While post-rationalization is an important design tool, it offers limited support for evaluating design alternatives or engaging in material- and constructionaware exploration. The functional and aesthetic properties of the resulting auxetic linkage are difficult to anticipate when designing the required reference geometry. In particular, the sizing of triangles and specific boundary alignment result from a global optimization that does not necessarily yield easily foreseeable results. It is therefore beneficial to provide direct manipulation tools to further edit and adapt the optimized structure to better meet the design goals. Our work introduces such direct editing operations. The presented computation-assisted design system allows for effective design space exploration of deployable auxetic structures and gives the designer full control of the final deployed surface geometry. In addition,

our approach provides tools for computational form-finding, where the desired target surface is interactively constructed using the deployment mechanism as a form-finding force.

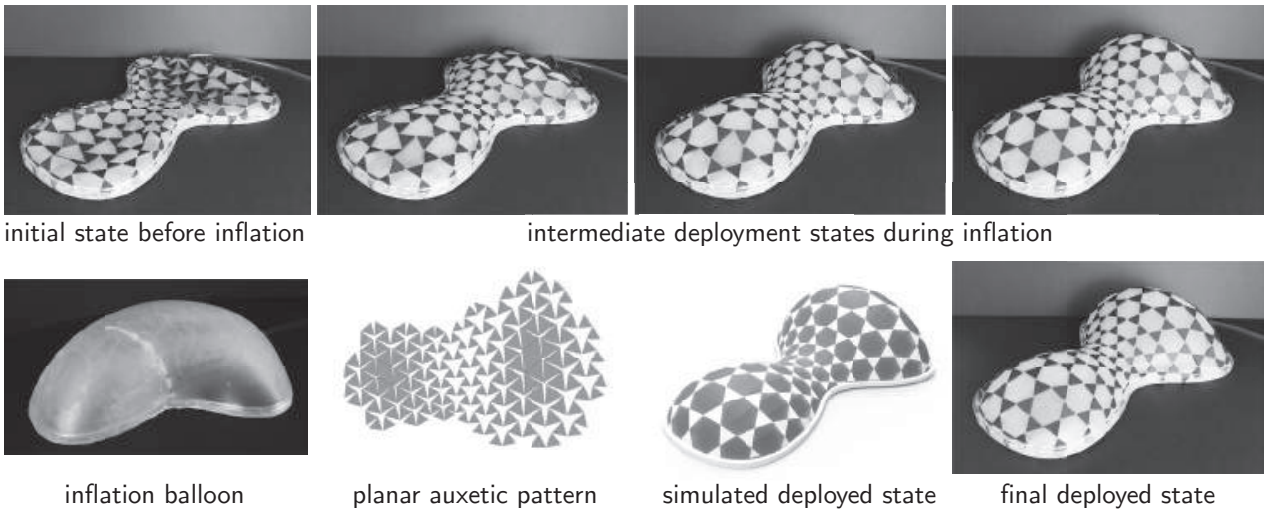


Figure 1: Physical prototype with inflation deployment, Konaković-Luković et al. (2018). The graded auxetic pattern has been laser cut, mounted onto a support frame and inflated with a generic rubber balloon.

2. Related Work

To put our work into context, we briefly review related work on deployable structures and auxetic materials. We refer the reader to Konaković-Luković et al. (2018) for additional discussion of prior work, particularly methods for material-aware post-rationalization in computer graphics and digital fabrication.

The concept of kinematic deployment is well studied in architecture. For large-scale structures, elastic grid shells are probably the most prominent example. Composed of interconnected elastic beams, an elastic grid shell achieves its desired target shape by active bending, Lienhard (2014). Common methods of erection include lifting with cranes or various types of scaffolding or mechanical formwork. Erection of elastic grid shells via inflation has been discussed in Quinn and Gengnagel (2014), where the authors identify a number of potential benefits in terms of safety, construction time, and cost. Form-finding for elastic grid shells is also an active topic in material science; see, for example, the recent work of Baek et al. (2017). Deployable structures are also

used for various building components. For example, Hannequart et al. (2018) investigate the use of shape memory alloys for deformable facade shading devices.

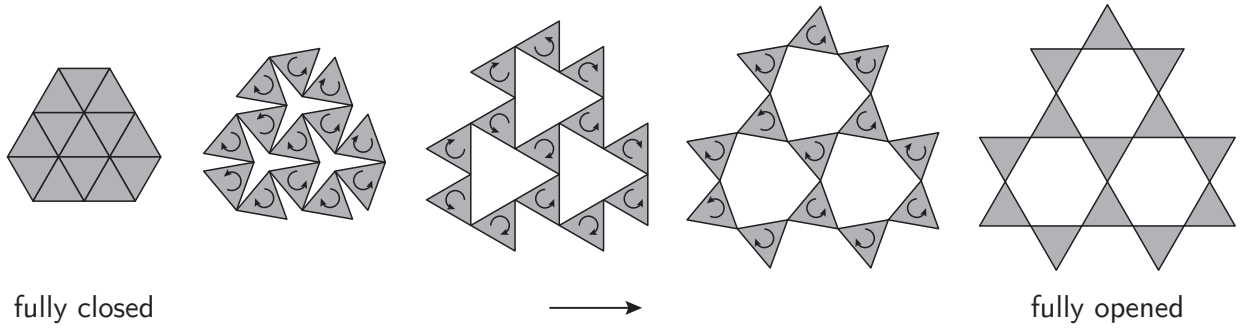


Figure 2: Our auxetic linkage is defined as a tri-hexagonal pattern. A uniform linkage can transition in the plane between a fully closed state (left) and a fully opened state (right) by rotating triangles around their connecting vertices. This expansion increases total area by a factor of four, which corresponds to a scaling of length by a factor two.

Auxetic meta-materials have been extensively studied in material science; see Saxena et al. (2016) for a comprehensive review. Konaković et al. (2016) proposed an optimization method for designing curved target surfaces that can be fabricated by deforming flat auxetic sheets. In this method, the auxetic structure is assumed to be spatially homogenous and have the same physical properties everywhere. Deforming such a 2D sheet material to the desired 3D shape is a complex manual process that requires a guiding surface or scaffold. Uniform auxetic materials have also been studied in Naboni and Sortori Pezzi (2016) to design bending-active grid shells. Spatially graded auxetics have been explored for freeform reinforced concrete components by Friedrich et al. (2018). They introduce an iterative evolutionary optimization process to find a planar pattern that conforms to a given target shape when expanded fully. The idea of optimizing the spatial layout of flat-produced patterns has also been studied by La Magna and Knippers (2018). They investigate how to induce controlled curvature through elastic bending of spatially graded cellular structures.

3. Programmable Auxetics

In this section, we describe the basic principles of deployable auxetic linkages and briefly review the post-rationalization approach presented in Konaković-Luković et al. (2018). Auxetic linkages are initially planar assemblies of rigid triangles that connect at hinge vertices in the specific arrangement shown in Figure 2. This arrangement allows the triangles to freely rotate around the hinge points to form openings, uniformly expanding the structure in all directions while resisting shear deformations. This uniform expansion behavior indicates the pattern has an effective Poisson's ratio of -1 (making it an *auxetic* structure) and offers a key advantage for architectural applications: it allows the linkage to be shaped into double-curved surfaces, unlike inextensible sheets of material, which can only bend into developable surfaces

As the linkage progressively expands, eventually its openings become regular hexagons, and its pattern of rigid triangles and holes forms a trihexagonal tiling known as a.

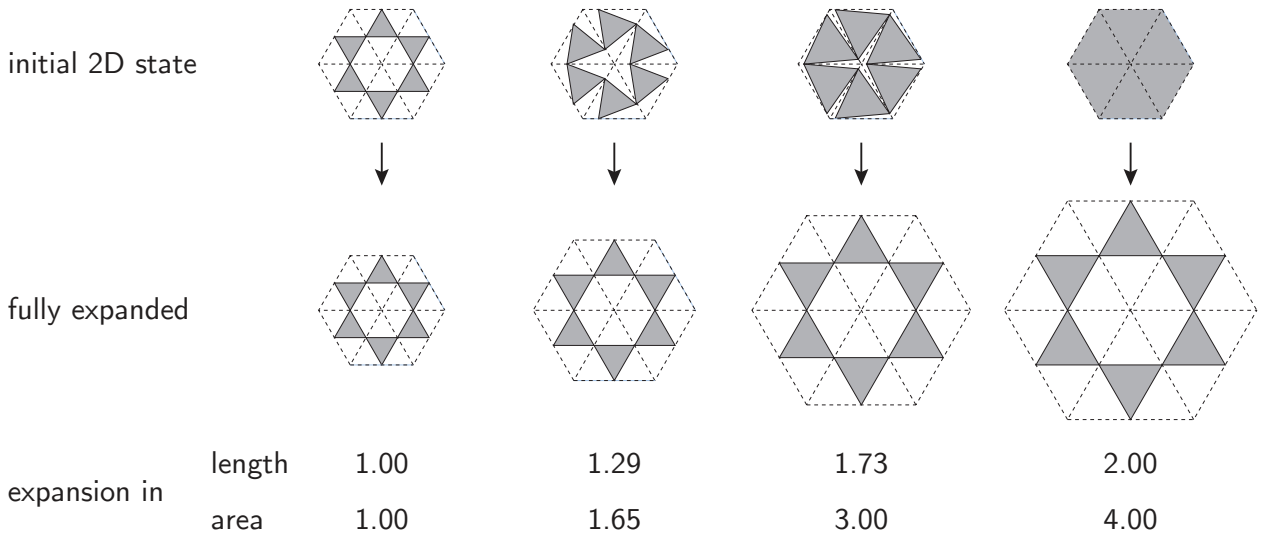


Figure 3: This allows controlling the deployed expansion factor within the range of one, when already fully opened in the rest configuration (left), to two in length resp. four in area, when fully closed in the rest configuration (right).

Kagomi lattice, see Grünbaum and Shephard (1986). In this fully opened configuration, the linkage has stretched from its closed configuration by a length scaling factor of two, and further expansion is blocked.

The *deployable auxetics* introduced in Konaković-Luković et al. (2018) leverage this fully expanded state as a mechanism for rapid deployment (**Fig. 1**). Observing that applying a specific spatially-varying

stretch λ to a flat sheet forces it to assume a unique shape (up to isometric deformation), the authors propose a spatially graded linkage that reaches its fully extended state exactly when stretched by λ. The key idea is to fabricate a planar linkage that is already partially opened: pre-opening the linkage by different amounts λ_{pre} at each point effectively programs a spatially varying maximum stretching factor $\lambda_{max} = 2/\lambda_{pre}$ (**Fig. 3**). If we program a planar linkage with the specific scaling field λ_{max} corresponding to some desired curved shape and subsequently apply an expansion-driven deployment process like inflation or gravitational loading, the process will automatically terminate when this scaling limit is hit; the resulting fully opened deployed linkage will form a trihexagonal tiling of the desired 3D surface. Note that pre-opening the linkage by different amounts in different regions requires varying the linkage's triangle sizes (**Fig. 3**).

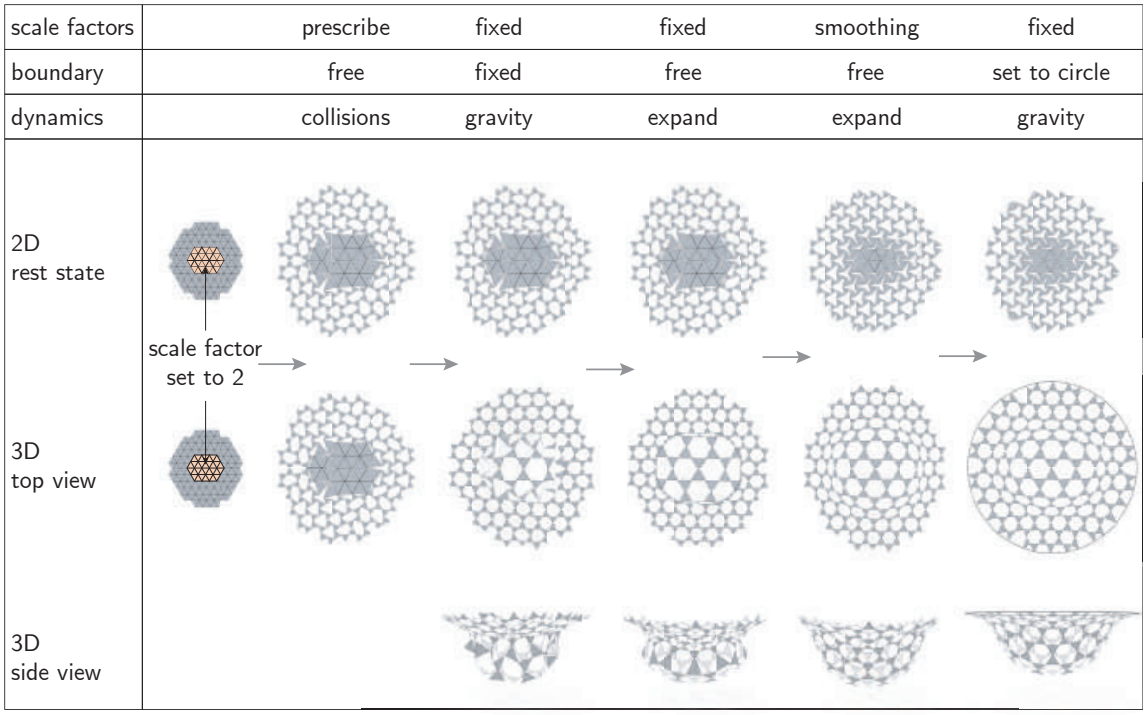


Figure 4: A simple form-finding example to illustrate our atomic editing operators and their effect on the auxetic structure. After prescribing scale factors, we resolve collisions which expands the material in the plane. Applying gravity forces pushes the linkage to a deployed state. However, when applying full expansion, we observe that the surface cannot be realized as a height field, mainly due to the sharp transition in scale factors. After smoothing the scale factors and letting the boundary evolve freely, we obtain a consistent height field surface. Finally, we show how to constrain the boundary onto a circle curve.

Konaković-Luković et al. (2018) have shown that a large and well-defined class of surfaces can be rationalized with deployable auxetics. Specifically, they prove that a stretchlimited surface can be deployed with inflation if and only if the target surface has positive mean curvature $\frac{1}{2}\nabla \cdot N$ everywhere (where N is the outward-pointing normal vector). Similarly, a *height field* surface can be deployed via gravity if and only if it has positive mean curvature. Surfaces not meeting these requirements can be projected to the nearest positively curved surface with a mean curvature flow process described in the paper. There are additional mechanical restrictions imposed by the linkage pattern: the range of length scale factors should fit between one and two.

The post-rationalization pipeline proposed in Konaković-Luković et al. (2018) builds on the close relationship between the auxetic linkage pattern and conformal maps. Like auxetic linkages, conformal maps permit uniform scale distortion but prohibit shearing deformations. Consequently, the map from the linkage's planar configuration to the deployed 3D surface is nearly conformal, and a conformal map from the plane to the desired curved surface can be approximated by a linkage (provided its conformal scaling factors fall within the permissible range). This motivates the use of a discrete conformal map to initialize a joint 2D/3D optimization to find the parameters of the auxetic linkage that best approximates the design surface when maximally stretched everywhere. For more details on this optimization, we refer the reader to their paper.

4. Design space exploration and form-finding

As discussed above, the desired target shape in the deployed state can be programmed into the auxetic structure by optimizing for suitable maximal expansion factors across the linkage, which in turn determine the spatial layout and sizing of linkage triangles. The indirect nature of this post-rationalization provides only limited support for exploring design alternatives or discovering new forms that are directly informed by the material and deployment mechanism. More direct manipulation is required to offer interactive design control in a tight feedback loop.

However, trying to manipulate the deployed geometry by directly displacing linkage vertices is not appropriate since the consistency of

the design cannot be easily maintained. Linkage vertices would need to be moved in a coordinated way to respect the complex global coupling imposed by the material structure and deployment mechanism, which becomes virtually impossible without computational support.

This is why we propose interactive, optimization-assisted design operators. Specifically, we allow the designer to directly modify the maximal scale factors of the linkage and impose design-specific geometric constraints. We then apply optimization to jointly determine the 2D rest shape and the 3D deployed shape. Since this optimization can be executed at interactive rates, the designer gets immediate feedback on her edits, while being freed of the complexities of maintaining consistency of the structure.

We found that the following editing operators yield an effective toolbox for design space exploration:

» *Prescribing scale factors*: We provide a painting interface where the designer can directly prescribe the desired maximal scale factors in the allowable range [1, 2]. Increasing scale factors allows the material to stretch more under deployment, while reducing scale factors locally shrinks the deployed surface.

» *Smoothing scale factors*: Sharp transitions in scale factors can lead to nonsmooth surface appearance and, in extreme cases, surface wrinkles. Spatially averaging the scale factors evens out these variations and generally leads to smoother deployed surfaces. Controlling the amount of scale factor smoothing yields different design alternatives.

» *Boundary control*: The user can directly edit the 3D boundary curves of the design and control the behavior of boundary linkage vertices, which can slide along boundary curves. Since the boundary has a strong influence on the overall shape of the deployed surface, we also allow boundary linkage triangles to deviate from equilateral shape, which can improve the overall surface quality.

» *Geometric constraints*: The user can further control the geometry of the deployed surface by imposing additional geometric constraints, for example on the planarity of certain edge curves, symmetry of selected vertices, or smoothness of the surface.

We also provide a separate form-finding optimization for the boundary curves. This can be helpful when the total area of the chosen linkage is not well-suited for the imposed boundary curve, e.g., when there is too

much material or too little for the surface to conform to the boundary. In such cases, we apply an expansion force on the linkage to fully expand the hexagonal openings and let the boundary vertices move freely to their preferred positions. Figure 4 illustrates how these design operators can be employed in an interactive form-finding design.

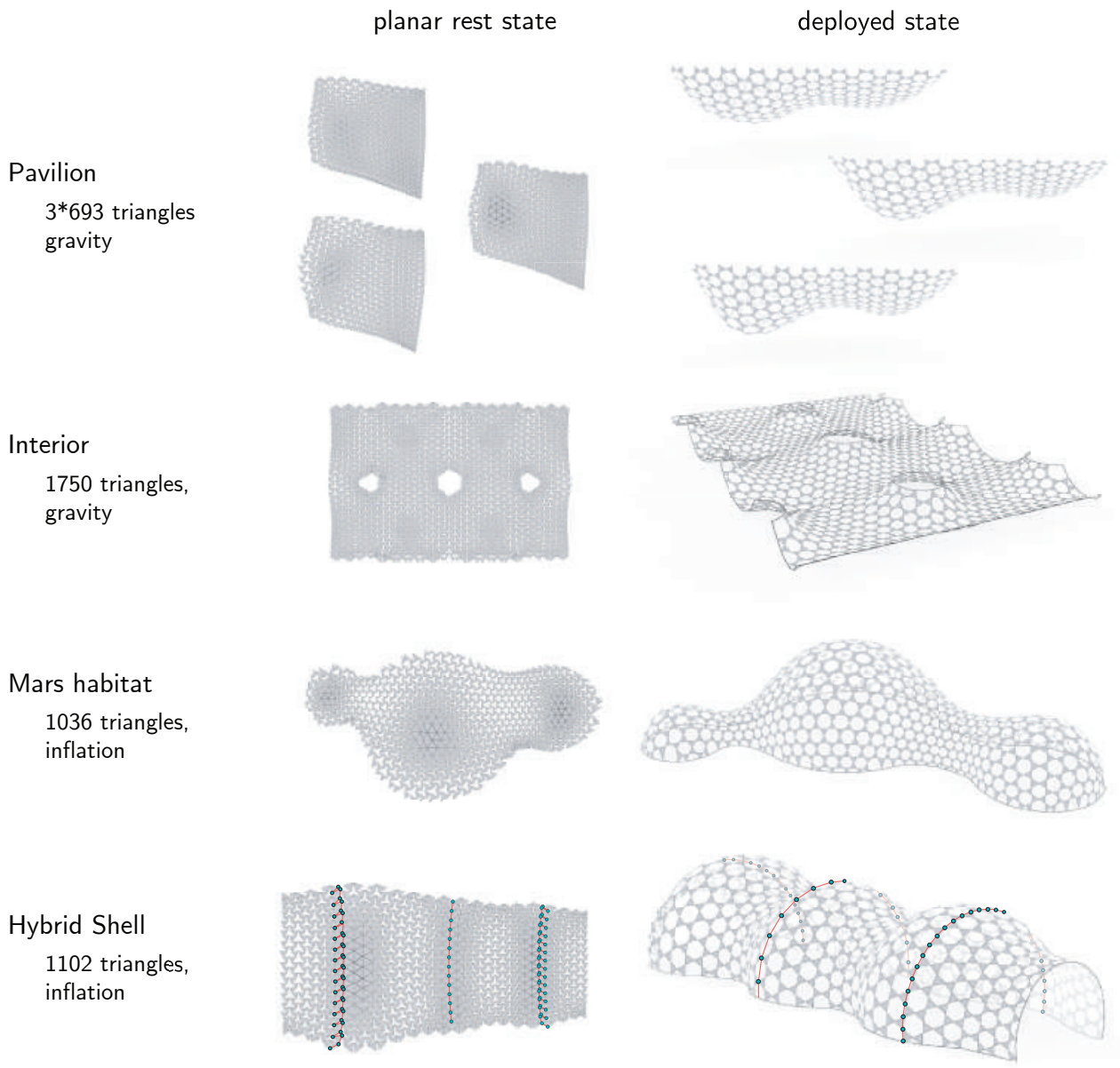


Figure 5: Four design examples shown in planar rest configuration and final deployed state. The number of auxetic linkage triangles and deployment method is indicated. In the bottom row, the highlighting shows three sets of vertices and edges that are each constrained to lie on a plane in the deployed 3D model to create planar support beams. See Figures 6 to 9 for detailed renderings.

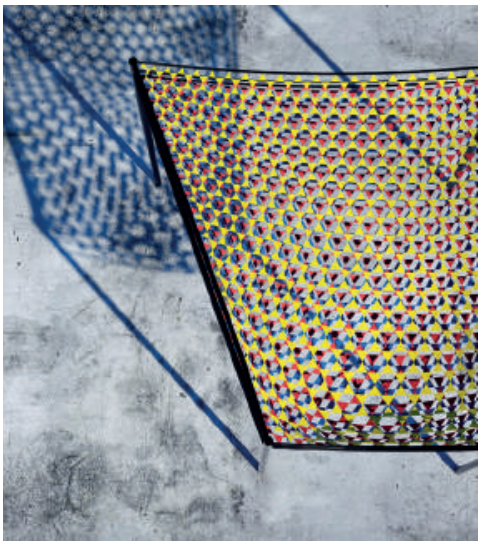


Figure 6: Multi-layer shading pavilion deployed by gravity.

4.1 Algorithm

Our interactive design system runs a constraint-based optimization algorithm to provide direct visual feedback on the flat and deployed state of the auxetic linkage. This optimization is based on the projective approach of Bouaziz et al. (2014); Deuss et al. (2015) that allows combining different geometric constraints to model the material behavior and the dynamics of the deployment mechanism.

Painting or smoothing scale factors provides constraint targets for the triangle edge lengths. We apply point-to-curve constraints to limit the movement of boundary vertices to the boundary curves. When optimizing for the boundary, we apply circle constraints on the hexagonal openings to expand the surface, as the maximal area is achieved when all hexagon vertices lie on a circle, see Niven (1981), page 236. Additional geometric constraints, e.g., planarity of user-selected edge curves in the deployed state, can easily be formulated on the linkage vertices. Gravitational deployment is modeled with a constant downward force, while inflation is approximated by outward-pointing normal forces. These forces are converted into geometric constraints in an implicit time integration solver as discussed in Bouaziz et al. (2014).

During editing, the constraint-based optimization solves for the linkage vertex positions in the flat 2D state and the deployed 3D state to provide immediate visual feedback on the performed edits. For more implementation details and an open-source library of the projection-based solver, we refer to www.shapeop.org.

5. Application case studies

We illustrate the potential of our computational design approach with a number of application case studies for deployable auxetic structures. The design process starts with an initial 2D triangular linkage, either obtained by the post-rationalization process of Konaković-Luković et al. (2018) or simply created as a uniform triangle pattern when designing from scratch. We then apply a series of editing operations as described above to explore design alternatives. The final output of this interactive form-finding process is a specific triangular linkage with spatially varying triangles that can be fabricated and assembled in the plane and deployed automatically to the desired target shape. Four example designs are

summarized in Figure 5 and described in more detail below.

Figure 6 shows a design study of a shading pavilion, realized as a linkage of inextensible fabric triangles that are connected with ring joints at the triangle vertices. As a hanging structure, the surface deploys under gravity to its desired double-curved target state. This design has been created in an interactive formfinding process from a uniform auxetic linkage that is subsequently manipulated using our design operators to create three design variations. These are combined in a multi-layer structure, which allows designing spatially varying opacity to optimize the shading performance of the structure for the anticipated use case.

Figure 7 shows another gravity-deployed structure in an interior space, with potential use cases of acoustic dampening or decoratively

Figure 7: Interior decorative cladding. This hanging structure has been optimized to align with the boundary constraints imposed by the ambient space. The designer controls the shape by interactively modifying scale factors.

masking of functional components such as AC pipes or wirings. This example shows how manipulating scale factors in combination with detailed boundary control offers effective ways to integrate a deployable auxetic structure into an existing space with precisely defined boundary constraints.

Figure 8 shows a speculative design study for habitats on Mars. Since the atmospheric pressure on Mars is 100 times lower than Earth's, the interior must be pressurized. This motivates the use of inflatable structures that can be efficiently erected from flat configurations, offering the additional benefits of low weight and compact storage. Our deployable auxetics offer a rich design shape space, so we can optimize the shape of the freeform domes to match interior space objectives.

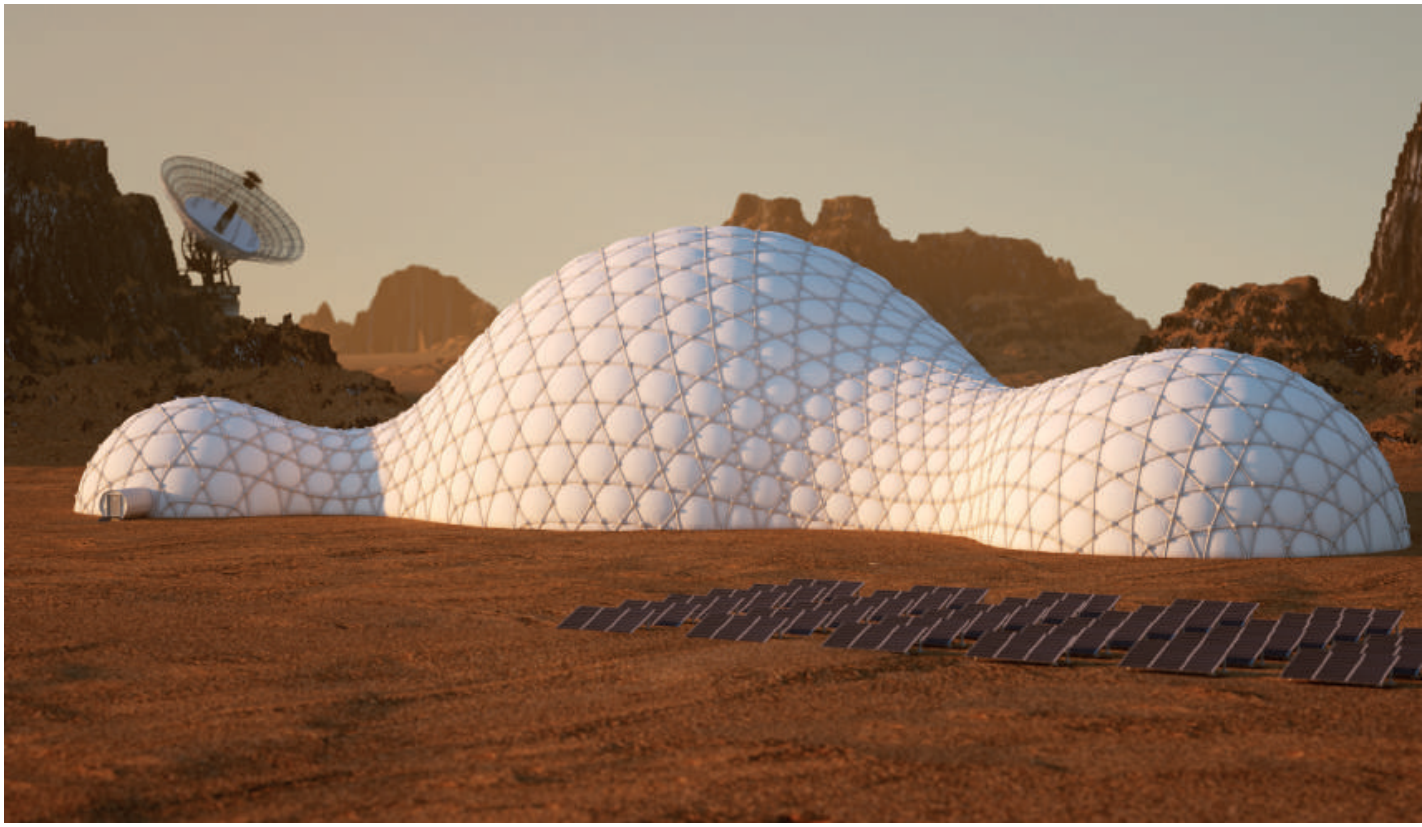


Figure 8: Inflatable freeform dome for a potential Mars habitat.

In Figure 9, we demonstrate how we can incorporate additional geometric constraints to optimize the design. In this example, we impose planarity constraints on selected edge and vertex curves of the auxetic linkage to form structural arches that can reinforce the inflated shell. The planarity of these arches significantly simplifies their fabrication.

Figure 9: A hybrid shell structure integrates planar support arches in the interior into a deployable auxetic surface.

6. Conclusion

We have shown how optimization-assisted shape exploration yields an effective method for designing deployable structures based on auxetic triangular linkages. By directly manipulating the form-defining geometric properties, i.e., the material scaling and the surface boundary, the designer obtains full control of the deployed shape while being shielded from the complexity of maintaining consistency between the 2D assembly state and the 3D deployed state. Automatic deployment via inflation or gravity allows transforming compact flat assemblies into freeform surfaces without the need of any supporting structures or complex construction process. Fabrication requires only 2D technologies such as sawing or laser cutting to produce the triangular panels. Despite this inherent simplicity, expressive freeform surfaces can be realized for a variety of different use cases.

A number of open questions offer numerous opportunities for future work. So far, we did not address questions of structural integrity in a systematic way, nor did we incorporate performative objectives into the optimization. For example, light transmission of the shading pavilion or the acoustic dampening for the interior cladding design study could be directly integrated into the form-finding method to yield a more informative shape exploration process. Another important topic for future work is the design of joint connections, in particular ones that lock into a stable state when deployed to the final target configuration. Finally, we see interesting research potential in exploring other expansive deployment mechanisms, for example based on material swelling, motorization, or pre-stressing.

Acknowledgements

We thank Julian Panetta and Keenan Crane for their feedback and advice. This research was supported by the National Centre of Competence in Research (NCCR) Digital Fabrication, funded by the Swiss National Science Foundation (NCCR Digital Fabrication Agreement 51NF40-141853).

References

Baek, C., A. O. Sageman-Furnas, M. K. Jawed, and P. M. Reis (2017). Form finding in elastic gridshells. Proceedings of the National Academy of Sciences.

Bouaziz, S., S. Martin, T. Liu, L. Kavan, and M. Pauly (2014, July). Projective dynamics: Fusing constraint projections for fast simulation. ACM Trans. Graph. 33(4), 154:1–154:11.

Deuss, M., A. H. Deleuran, S. Bouaziz, B. Deng, D. Piker, and M. Pauly (2015). Modelling Behaviour: Design Modelling Symposium 2015, Chapter ShapeOp—A Robust and Extensible Geometric Modelling Paradigm, pp. 505–515. Cham: Springer International Publishing.
Friedrich, J., S. Pfeiffer, and C. Gengnagel (2018). Locally varied auxetic structures for doubly-curved shapes. Humanizing Digital Reality: Design Modelling Symposium Paris 2017, 323–336.

Grünbaum, B. and G. C. Shephard (1986). Tilings and Patterns. New York, NY, USA: W. H. Freeman & Co.

Hannequart, P., M. Peigney, J.-F. Caron, O. Baverel, and E. Viglino (2018). The potential of shape memory alloys in deployable systems—a design and experimental approach. Humanizing Digital Reality: Design Modelling Symposium Paris 2017, 237–246.

Konakovi -Lukovi , M., J. Panetta, K. Crane, and M. Pauly (2018). Rapid deployment of curved surfaces via programmable auxetics. ACM Trans. Graph. 37(4).

Konakovi´c, M., K. Crane, J. Panetta, and M. Pauly (2018). Rapid deployment of curved surfaces via programmable auxetics. ACM Trans. Graph. https://www. dropbox.com/s/gewfxwfoooyvfia/programmable_auxetics.pdf?dl=0.

La Magna, R. and J. Knippers (2018). Tailoring the bending behaviour of material patterns for the induction of double curvature. Humanizing Digital Reality: Design Modelling Symposium Paris 2017, 441–452.

Lienhard, J. (2014). Bending-Active Structures:Form-finding Strategies Using Elastic Deformation in Static and Kinetic Systems and the Structural Potentials Therein. Forschungsberichte aus dem Institut für Tragkonstruktionen und konstruktives Entwerfen der Universität Stuttgart. Universität Stuttgart Inst. f. Tragkonstr.

Naboni, R. and S. Sortori Pezzi (2016). Embedding auxetic properties in designing active-bending gridshells. XX Congress of the Iberoamerican Society of Digital Graphics.

Niven, I. (1981). Maxima and Minima Without Calculus. Number v. 6 in Dolciani Mathematical Expositions. Mathematical Association of America.

Quinn, G. and C. Gengnagel (2014). A review of elastic grid shells, their erection methods and the potential use of pneumatic formwork. WIT Transactions on The Built Environment.

Saxena, K. K., R. Das, and E. P. Calius (2016). Three decades of auxetics research – materials with negative poisson's ratio: A review. Advanced Engineering Materials 18(11), 1847–1870.

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 alternatingly 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.

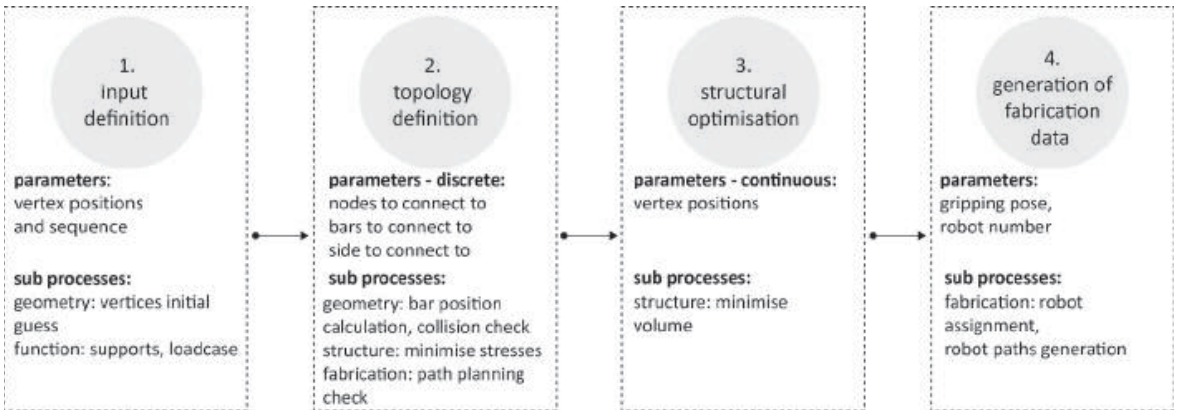


Figure 2: Design workflow describing involved sub-processes and variables to be defined or changed in each step.

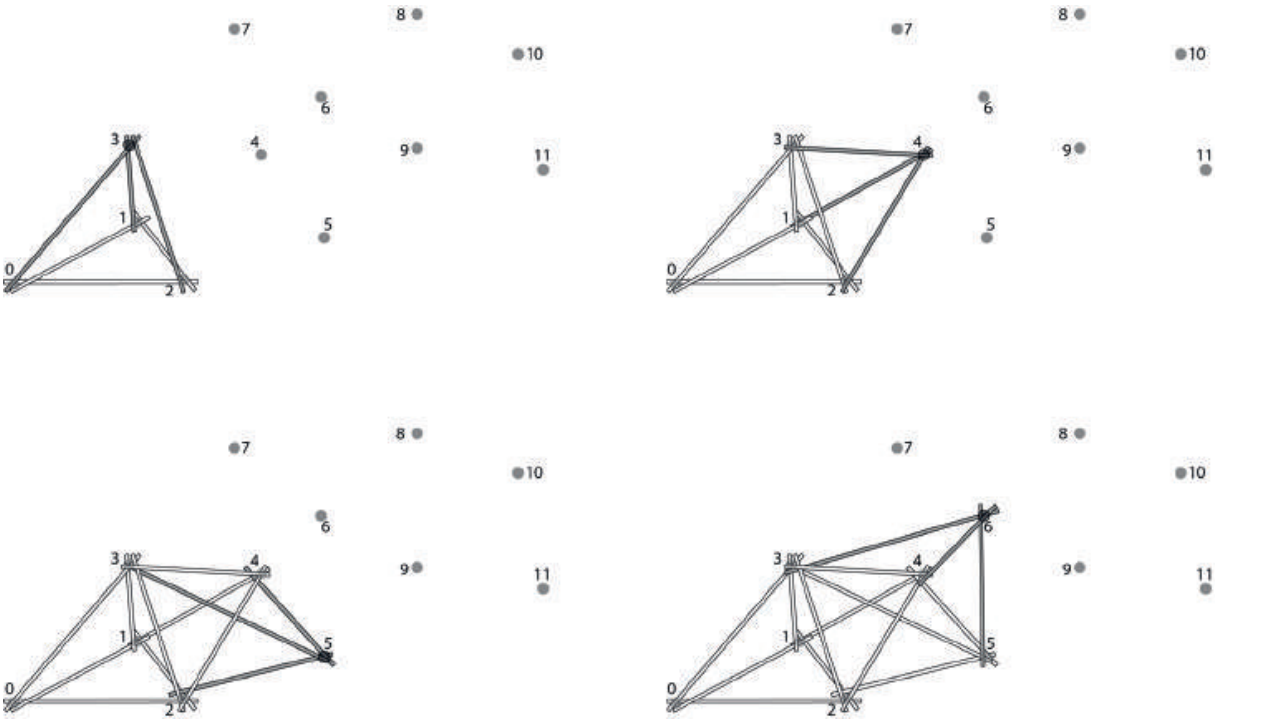


Figure 3: Design workflow, step 2: topology definition. A new vertex is chosen and connected to the existing structure via three bars. The input vertices are pre-defined and connections are chosen based on structural considerations.

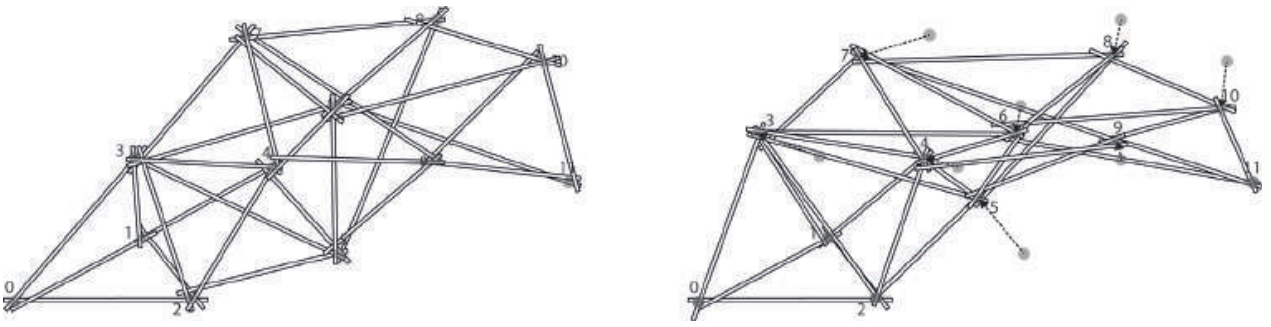


Figure 4: Design workflow, step 3: structural optimisation. Locations for vertices are refined to minimise the material usage for a given load case.

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).

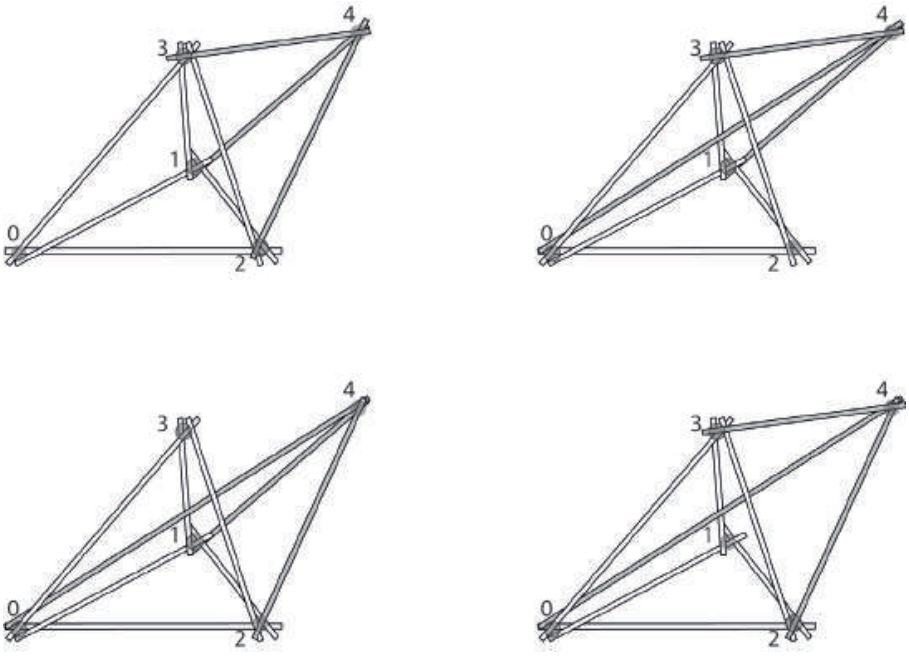


Figure 5: Example of options for three connections from a new vertex to the existing structure.

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.

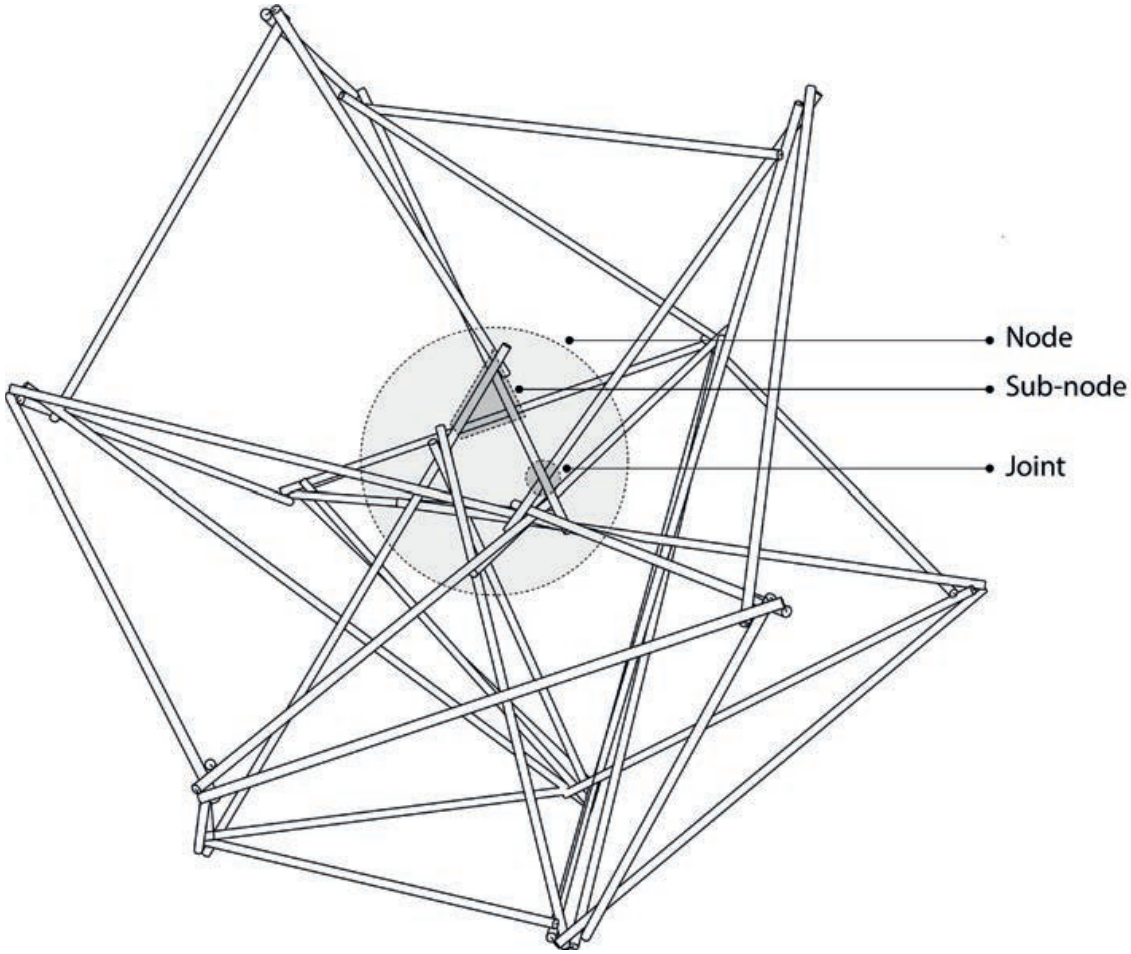


Figure 6: Definition of node, sub-node and joint. A node develops out of every vertex once bars are generated around it.

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**).

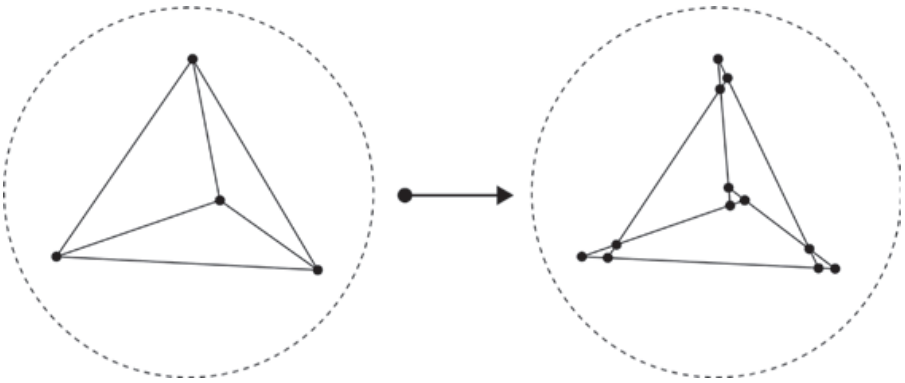


Figure 7: Geometric development of node in an aggregation of 6 bars.

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.

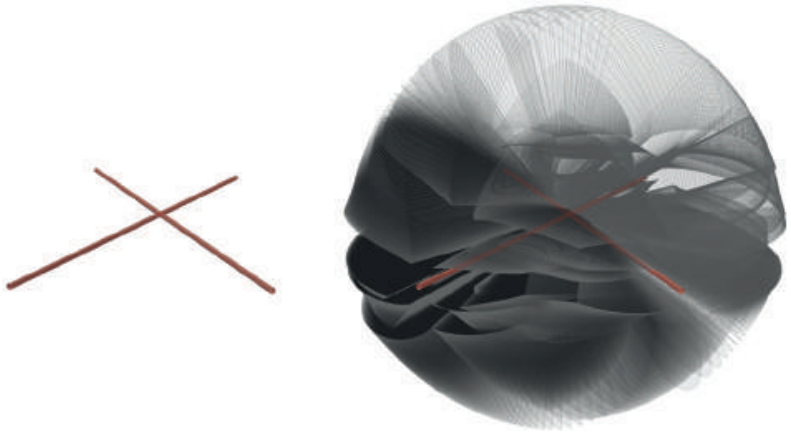


Figure 8: Visualisation of the solution space for a bar tangent to two existing bars. The volume shows all possible angles of attachment for a new bar and two fixed existing ones.

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.

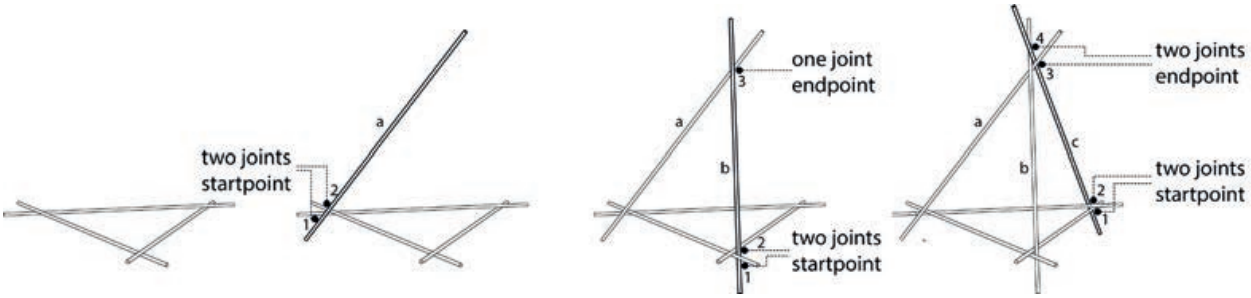


Figure 9: Sequential adding of three bars onto an existing structure leading to two joints, three joints and four joints.

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.

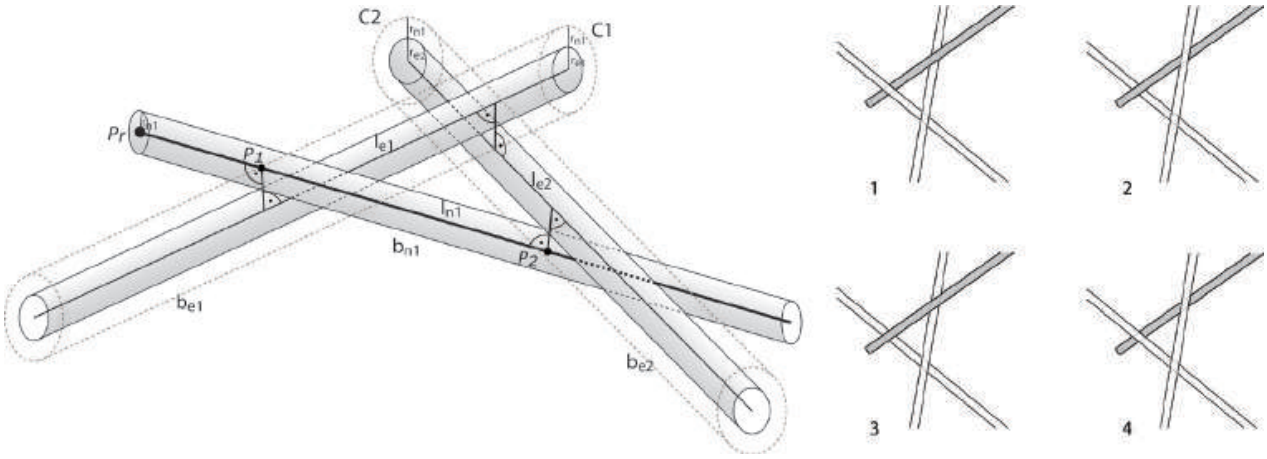


Figure 10: Dependencies between three tangent bars (left) and four possible solutions for one bar tangent to two other bars (right).

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.

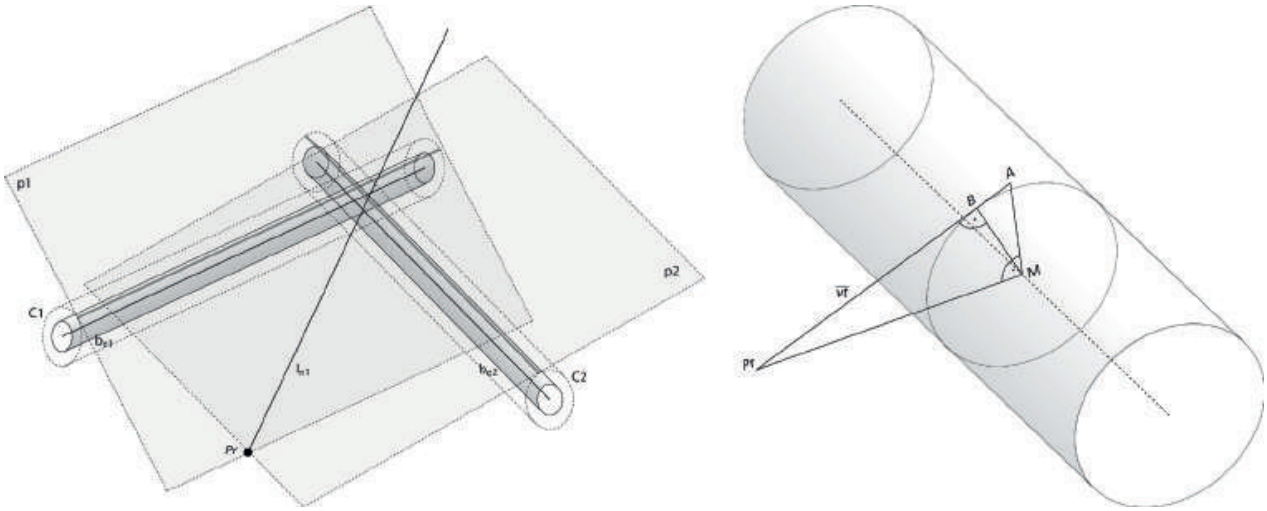


Figure 11: Case 1. Calculation of the first bar tangent to two existing bars b_1, b_2 (left). Calculation of a vector v_t tangent to a cylinder through a given point P_r (right).

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_t and the vector of the bar's axis (l_{e1} or l_{e2}).

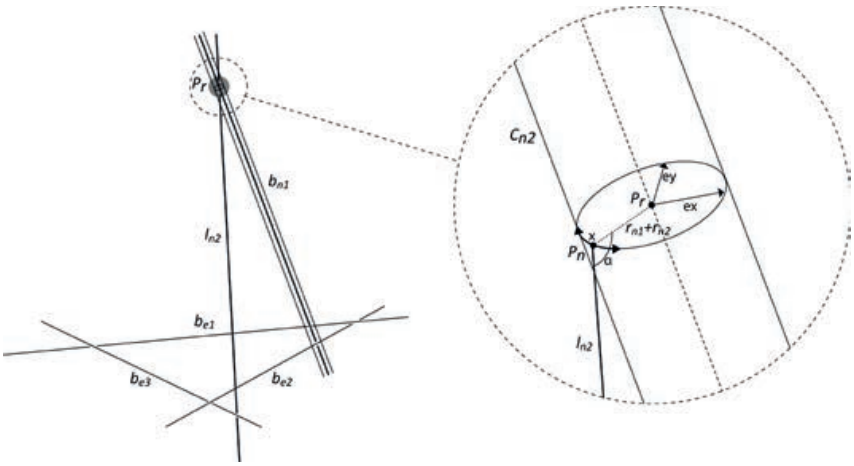


Figure 12: Case 2. Calculation of the second bar tangent to two bars at the base and the already defined first bar at the top.

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 $\alpha = 90°$. 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

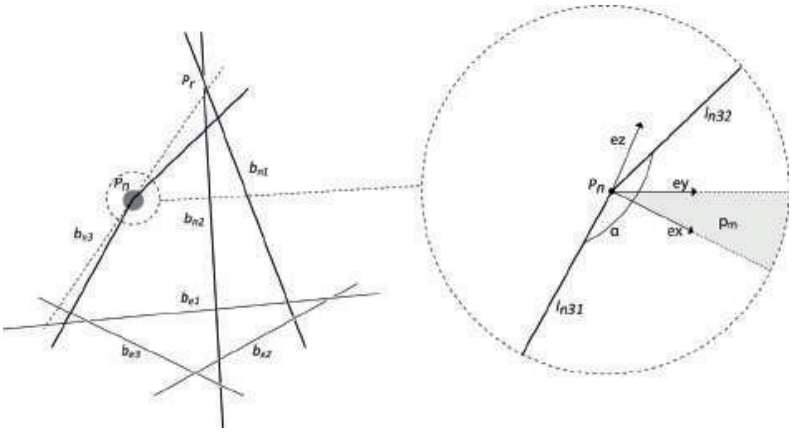


Figure 13: Case 3. Calculation of the third bar tangent to two bars at the base and the already defined first and second bars at the top.

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_{e1}, b_{e2} 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 $\alpha = 180°$. 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.

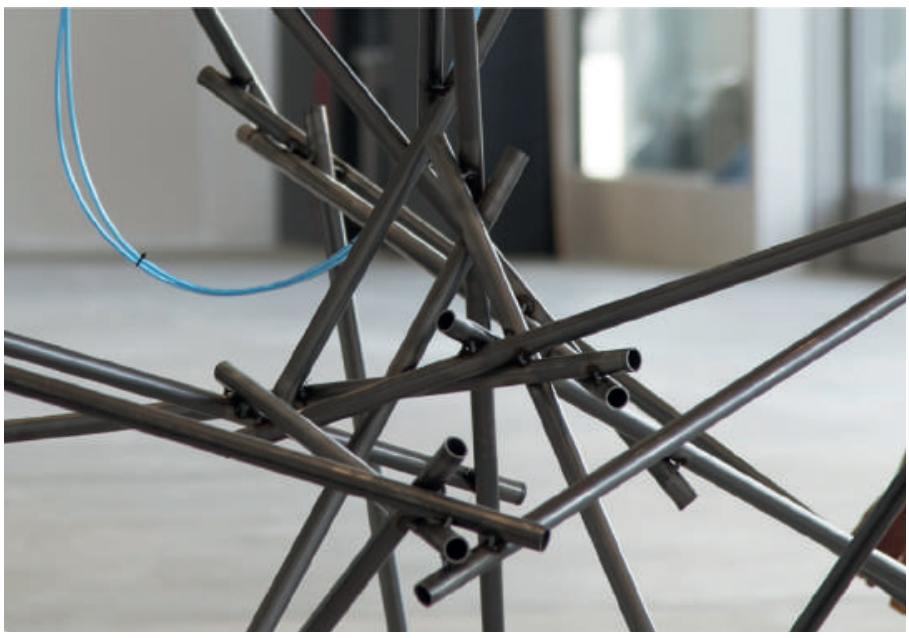


Figure 14: Physical test of a structure in which 14 bars come together in one node.

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**).

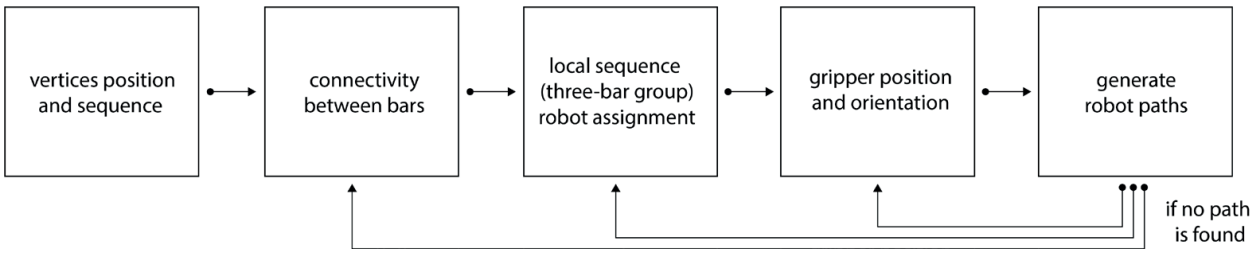


Figure 15: Workflow for the generation of robot paths. If no path is found for a bar parameters are changed from the right to the left.

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.

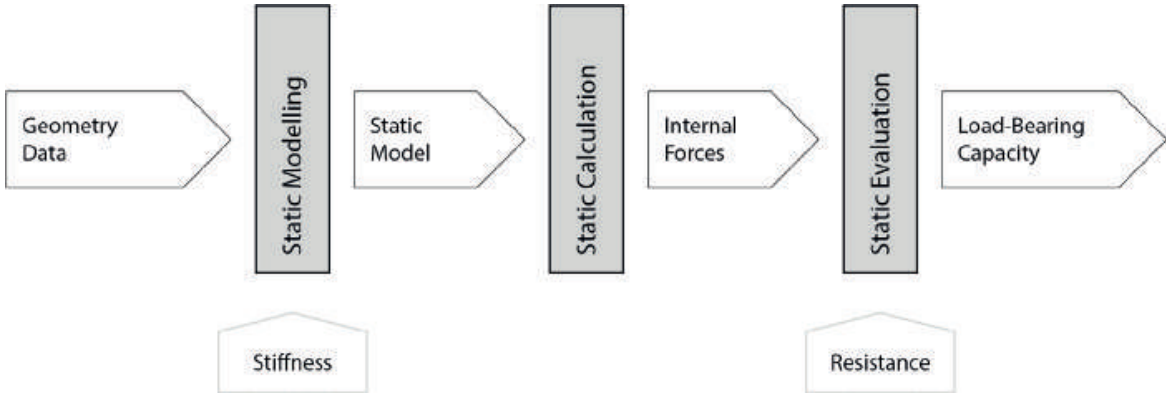


Figure 16: Workflow diagram of the integrated structural analysis. The structural analysis is used in step 2 and 3 of the design process.

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.

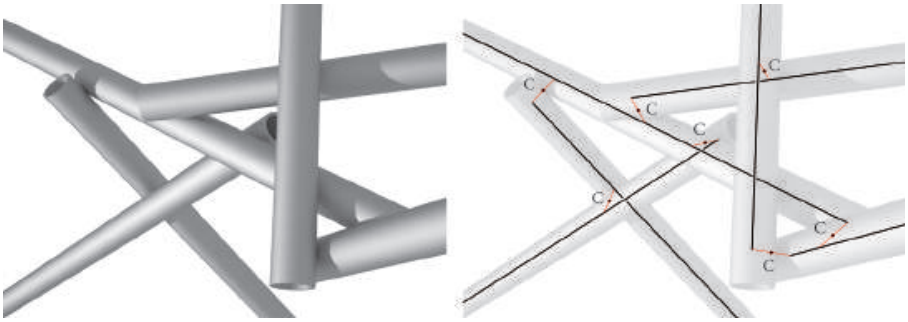


Figure 17: Geometry of part of the system (left) and corresponding linear static model (right).

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.

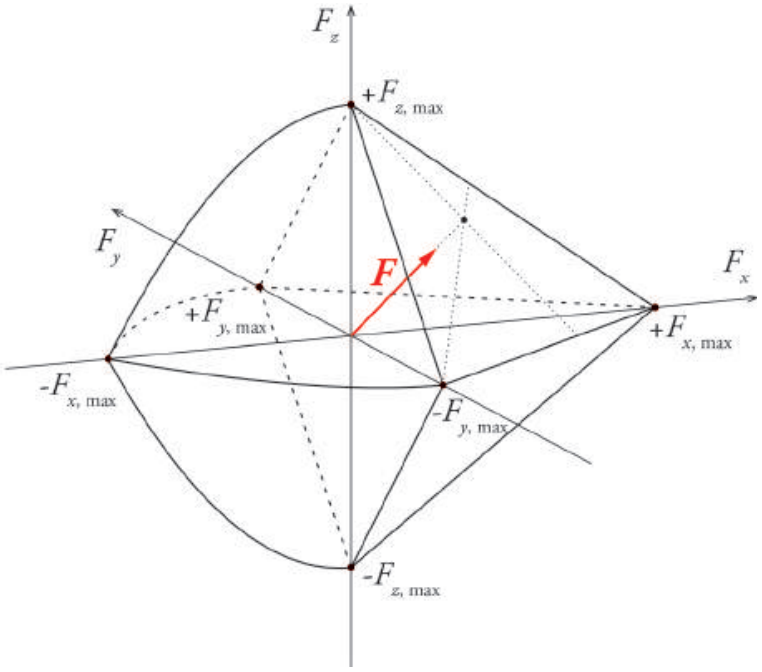


Figure 18: Example of a resistance diagram of a connector with existing force-vector and maximum graph.

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:

minimise $$f(x) = \sum u_b \qquad u_b = stress\ utilisation\ of\ bar$$

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:

minimise: $$f(x) = \sum l_b \qquad l_b = length\ of\ bar$$

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

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.

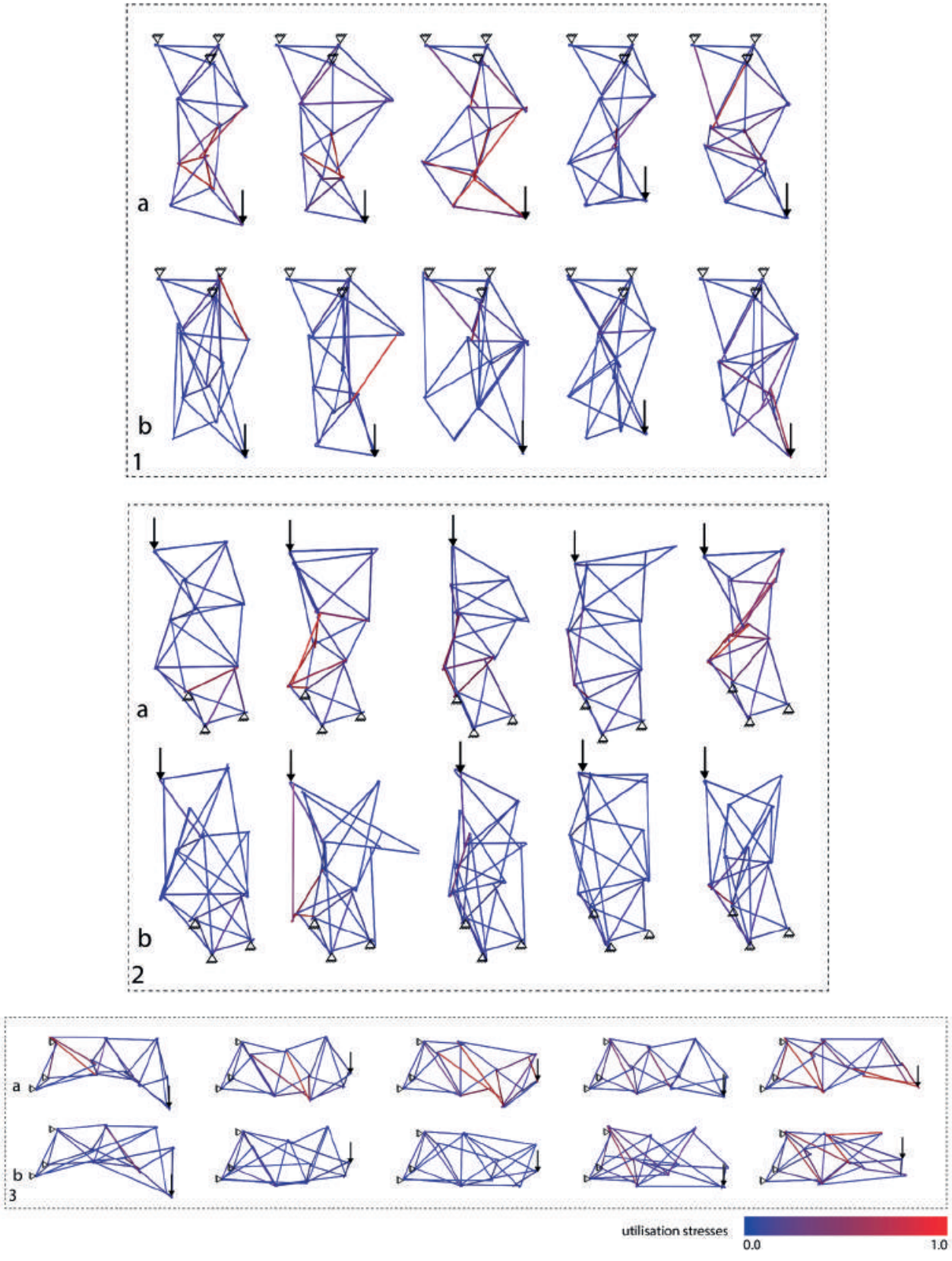


Figure 20: Test models for topology definition evaluation. 1. Tension loaded structures, 2. Compression loaded structures, 3. Bending loaded structures, a) structures with connections to closest nodes, b) structures resulting after optimisation.

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.

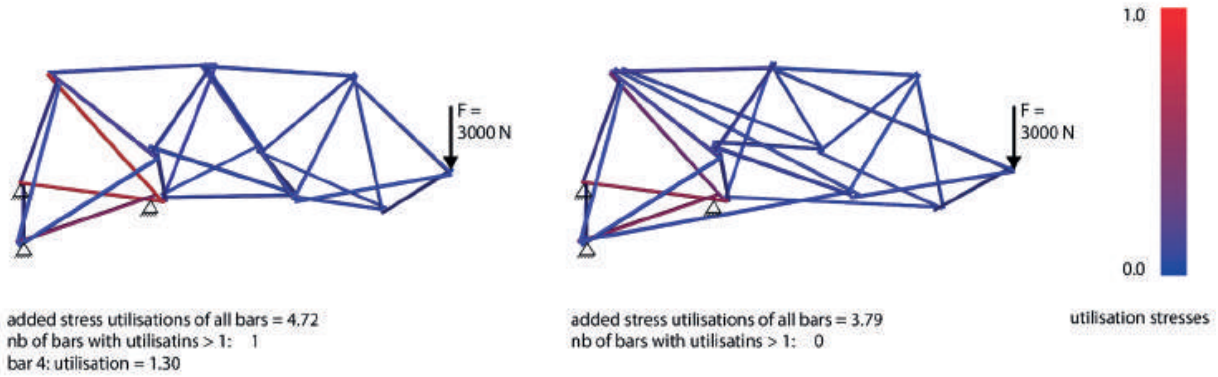


Figure 21: Example of topology generated by connecting to three closest vertices (left) and after the calculation process (right). Bars varying between 800 mm and 1800 mm in length, 25 mm of diameter and a thickness of 2 mm were used.

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.

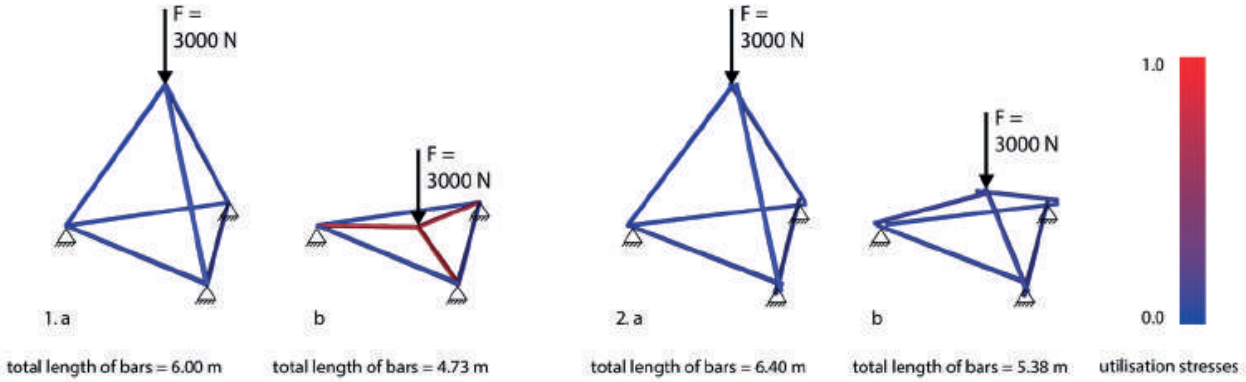


Figure 22: Optimisation results: 1. Simplified model: a) initial guess, b) optimisation result; 2. Model including connection: a) initial guess, b) optimisation result.

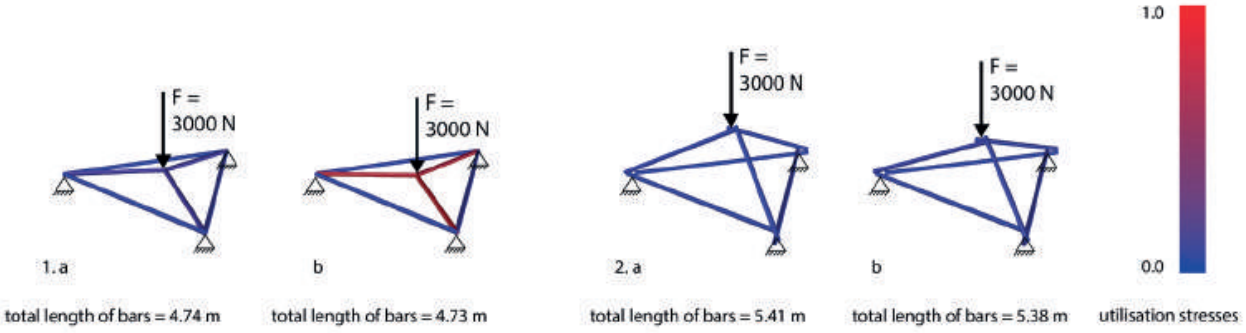


Figure 23: Optimisation results compared to brute force results: 1. Simplified model: a) brute force result, b) optimisation result; 2. Model including connection: a) brute force result, b) optimisation result.

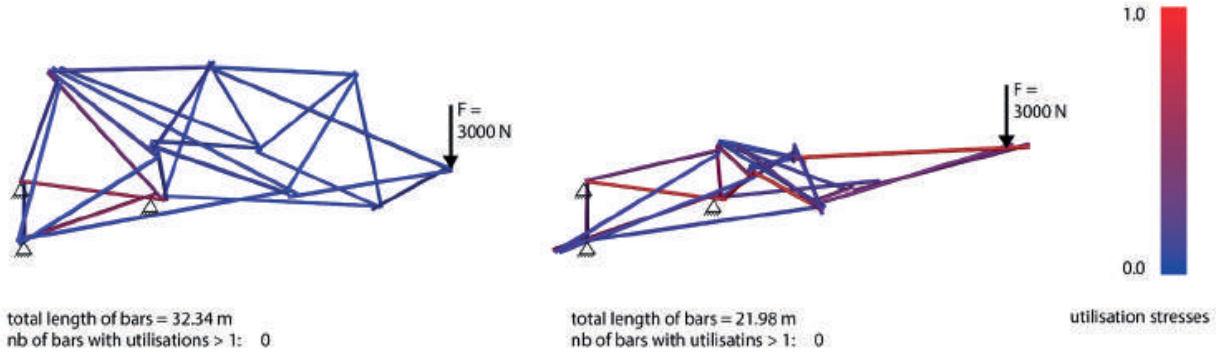


Figure 24: Example of structure before (left) and after (right) minimising material volume while constraining stress utilisations in bars.

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 preconditions. 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, Koln, 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 Joaqim 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).

Design and construction of curved support structures with repetitive parameters

Eike Schling, Martin Kilian, Hui Wang, Jonas Schikore, Helmut Pottmann

Eike Schling
eike.schling@tum.de
Technical University of Munich, Germany

Martin Kilian
kilian@geometrie.tuwien.ac.at
Vienna University of Technology, Austria

Hui Wang
hwang@geometrie.tuwien.ac.at
Vienna University of Technology, Austria

Jonas Schikore
jonas.schikore@tum.de
Technical University of Munich, Germany

Helmut Pottmann
pottmann@geometrie.tuwien.ac.at
Vienna University of Technology, Austria

Keywords:
Curved support structures, CMC-surfaces, elastic deformation, developable surfaces, strained gridshell, asymptotic curves, minimal surfaces, FEM

Abstract

The fabrication and construction of curved beams along freeform skins pose many challenges related to their individual and complex geometry. One strategy to simplify the fabrication process uses elastic deformation to construct curved beams from flat elements. Controlling the curvature of the design surface and beams has the additional potential to create repetitive building parts with beneficial beam orientation. We aim for strained gridshells built entirely from straight or circular lamellas of the same radius and with orthogonal nodes. The lamellas are aligned normal to a reference surface enabling an elastic assembly via their weak axis and a local transfer of loads via their strong axis. We show that the corresponding reference surfaces are of constant mean curvature and that the network of beams bisects principal curvature directions. We introduce a new discretization of these networks as quadrilateral meshes with spherical vertex stars and present a computational workflow for the design of such structures. The geometric advantages of these networks were key for the fabrication and assembly of a prototype structure, the Asymptotic Gridshell. We describe the complete process from design to construction, presenting further insights on the symbiosis of geometry, fabrication and load-bearing behavior.

1. Introduction

Gridshells are highly efficient structures because they carry loads through their curved shape with very little material. Their construction however, poses great challenges related to their complex geometry. In a freeform grid every node and every beam are likely to be different and have to be fabricated individually using computer aided, 3D manufacturing tools. Controlling the curvature parameters of design surfaces and beam networks, and using the elastic behavior of material to shape these grids opens up new strategies for fabrication-aware design.

We study structures that can be constructed with congruent nodes from lamella that are orientated normal to the underlying reference surface and have straight or circular development (**Fig. 1**). The slender lamellas allow for an elastic assembly via their weak axis and a local transfer of loads via their strong axis. The lamella network can be transformed elastically following a predetermined kinetic behavior. This

Figure 1: The Asymptotic Gridshell. The strained gridshell was assembled with orthogonal nodes from straight and flat lamellas, and erected elastically. (Image: Felix Noe)

enables a simple erection process without formwork. The final grid forms a doubly-curved network, enabling an efficient, spatial load transfer as a shell structure. We are interested in the possible shapes, their computational design and solutions for construction.

Related work. We follow up on recent work by Tang et al. (2016) on curved support structures from developable strips. A prominent example of this type is provided by the Eiffel Tower Pavilions (Schiftner et al. (2012), **Fig. 2**). However, this support structure follows principal curvature lines and does not lead to lamellas with straight or circular development. The design of strained grid structures with the use of developable strips has been investigated by Schling and Barthel (2017).

From the construction perspective our approach is inspired by the strained timber gridshells of Frei Otto (Burkhardt (1978)), namely, the Multihalle in Mannheim, see Figure 2.

Overview and contribution. We show that the requirements on lamellas and nodes lead to special curve networks on surfaces. Circular lamellas of constant radius and right node angles live on surfaces of constant-mean-curvature (CMC). Our computations use a novel discrete representation, namely quadrilateral meshes with *spherical vertex stars*. They generalize the well-known asymptotic nets with planar vertex stars (Bobenko and Suris (2008)).

We present a method for the computation of isothermic networks on CMC surfaces. The diagonals of such a network form curves of constant

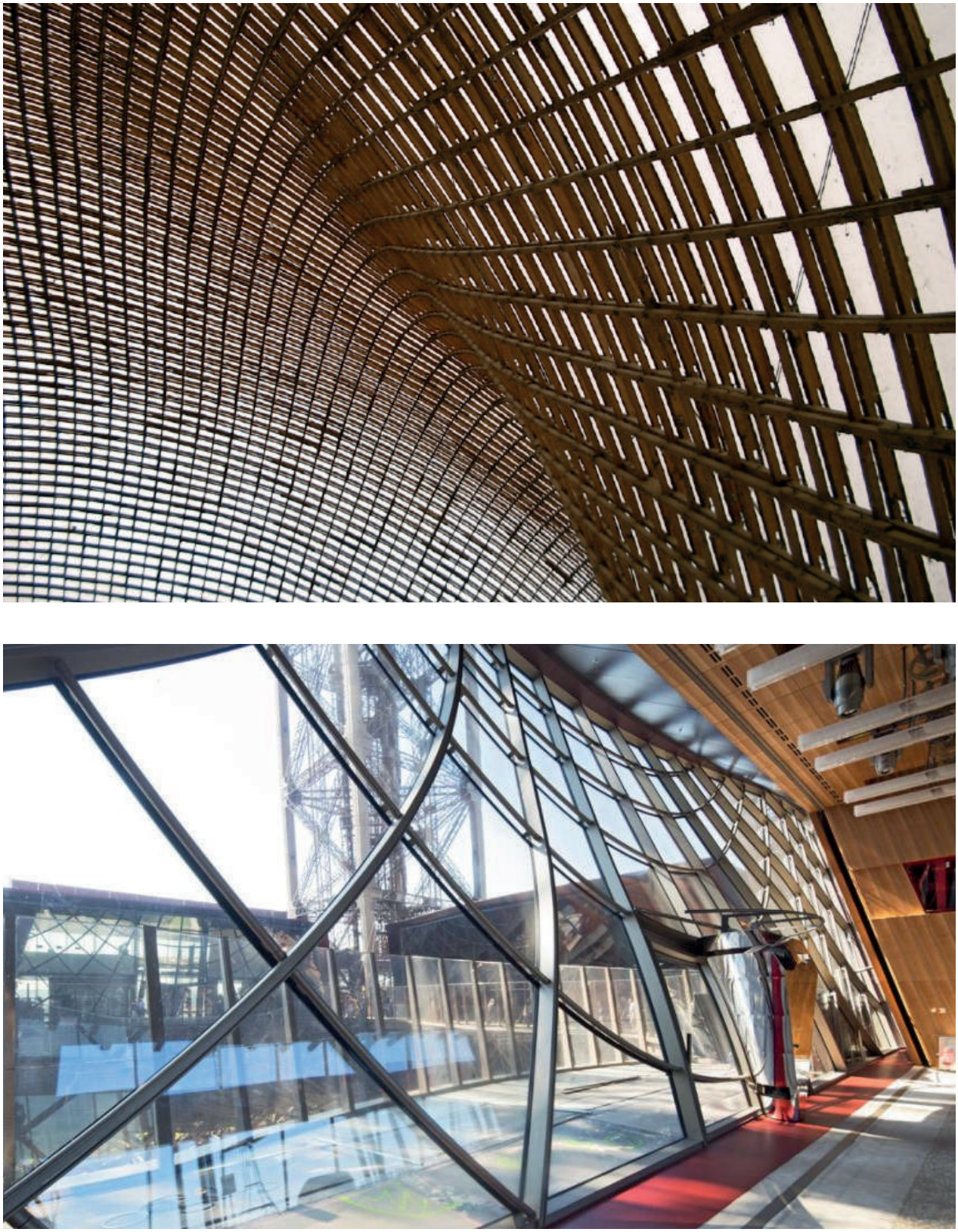


Figure 2: Curved grid structures: Left: Multihalle Mannheim by Frei Otto, 1975. The strained timber gridshell is formed from elastically-bent timber laths. Right: The Eiffel Tower Pavilions by Moatti Rivière Architects. The facade structure follows the principal curvature directions. The curved steel beams were fabricated from flat strips of steel. (Images: Rainer Barthel, Michel Denancé)

normal curvature κ_n and define the attachment points of lamellas of radius $r = 1/\kappa_n$ with 90 degree intersection angles. This includes the special case of straight lamellas with $r = \infty$.

The implications of planning and constructing such networks for a load-bearing gridshell are described in Section 3. Our case study, the Asymptotic Gridshell, was designed using asymptotic curves (vanishing normal curvature) on a minimal surface (zero mean curvature) and constructed from straight lamellas and orthogonal nodes. We discuss the architectural design process of surface and network, introduce a detail solution for a typical grid-joint, and present the fabrication and erection process. The load-bearing behavior is analyzed using a novel workflow to simulate residual stresses.

2. Theory and computation

Let us briefly recall a few facts from differential geometry. It will be helpful to know about the *Darboux frame* which is attached to a curve $\mathbf{c}$ on a surface S. Let $\mathbf{c}(s)$ be an arc length parametrization of that curve. At each point $\mathbf{c}(s)$, the Darboux frame consists of the unit tangent vector $\mathbf{t}(s)$, the unit vector $\mathbf{n}(s)$ orthogonal to the surface S, and the sideways vector $\mathbf{u}(s) = \mathbf{n}(s) \times \mathbf{t}(s)$, see Figure 3. As the frame moves along the curve, at any time s the angular velocity vector $\mathbf{d}$ is given by

$$\mathbf{d} = \tau_g \mathbf{t} - \kappa_n \mathbf{u} + \kappa_g \mathbf{n}.$$

Its coefficients are important quantities of the surface curve $\mathbf{c}$: *geodesic curvature* κ_g, *normal curvature* κ_n and *geodesic torsion* τ_g. The derivatives of the frame vectors with respect to s satisfy $\mathbf{t}' = \mathbf{d} \times \mathbf{t}$, $\mathbf{u}' = \mathbf{d} \times \mathbf{u}$, $\mathbf{n}' = \mathbf{d} \times \mathbf{n}$: Inserting the above expression for $\mathbf{d}$, one finds

$$\kappa_g = \mathbf{t}' \cdot \mathbf{u}, \quad \kappa_n = \mathbf{t}' \cdot \mathbf{n}, \quad \tau_g = \mathbf{u}' \cdot \mathbf{n}.$$

Thus, κ_g and κ_n are the tangential and normal components of the curvature vector $\mathbf{t}'$, and τ_g is the normal component of $\mathbf{u}'$.

The geometric model of a curved support structure is a network of

developable surface strips which are orthogonal to a reference surface S. Let us consider such a developable strip D, attached to S along the common curve $\mathbf{c}$. If we want to make a model from originally straight flat strips, the curve $\mathbf{c}$ must map to a straight line in the planar unfolding of D. This means that $\mathbf{c}$ has to have vanishing geodesic curvature with respect to D. At each point of $\mathbf{c}$, the tangent planes of D and S are orthogonal. Therefore, $\mathbf{c}$ has vanishing normal curvature with respect to S; it is a so-called *asymptotic curve* on S.

We will also study models from strips whose flat developments are circular. In order to achieve a circle of radius r as the image of $\mathbf{c}$ in the flat development of D, $\mathbf{c}$ must have *constant geodesic curvature* $1/r$ with respect to D and therefore *constant normal curvature* $1/\mathrm{r}$ with respect to the reference surface S.

Let us summarize these important facts: *A flat circular strip which is subject to bending and no stretching can be attached orthogonally to a given surface S only along a curve $\mathbf{c}$ of constant normal curvature. In particular, a straight strip can only be attached orthogonally along an asymptotic curve of S.*

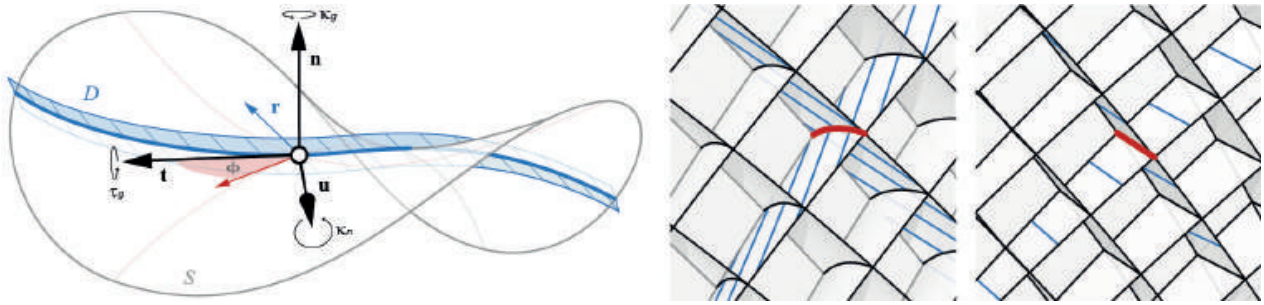


Figure 3: A developable strip attached orthogonally to a surface S along $\mathbf{c}$. Its rulings $\mathbf{r}$ are generally not parallel to the normal vector $\mathbf{n}$. This results in curved intersections of strips.

The developable surface D which is orthogonal to S along $\mathbf{c}$ is in general not formed by the surface normals along $\mathbf{c}$. The surface D is enveloped by planes orthogonal to S and tangent to $\mathbf{c}$, but its straight lines (rulings) are in general not orthogonal to S (**Fig. 3**). As discussed in detail by Tang et al. (2016), the ruling vectors are given by $\mathbf{r} = \kappa_g \mathbf{n} + \tau_g \mathbf{t}$ and thus agree with the surface normal $\mathbf{n}$ for $\tau_g = 0$, which in most of our examples is not the case. Related to this fact is the following one: The strips intersect at a node along a curve $\bar{\mathbf{n}}$ (**Fig. 3**, middle) and not in the surface normal (**Fig. 3**, right). However, this curve $\bar{\mathbf{n}}$ is usually nearly straight and

for practical purposes may be approximated by a straight line. We will also talk about a *node angle*, which refers to the one measured directly at the reference surface S. Theoretically, the angle between the two strips differs slightly along $\bar{\mathbf{n}}$.

To have more repetition in parameters, one may want to achieve the same node angle for all nodes. To discuss this, we need Euler's formula for the distribution of normal curvatures at a surface point:

$$\kappa_n = \kappa_1 \cos^2 \phi + \kappa_2 \sin^2 \phi .$$

Here, κ_1, κ_2 are the two principal curvatures and ϕ is the angle between the first principal curvature and the direction for which we compute the normal curvature κ_n. Directions with the same normal curvature are symmetric with respect to the principal directions as they are represented by ϕ and $-\phi$. If we want a constant right node angle and work with strips of the same radius in the development, the two directions meeting at a node are given by $\phi = \pm \pi/4$ and therefore $\kappa_n = (\kappa_1 + \kappa_2)/2 = \mathrm{H}$, H denoting *mean curvature*. This means that such a structure can only realize surfaces S for which the mean curvature equals $1/r$. These CMC surfaces are very well studied in differential geometry. A special case occurs when we use straight strips ($r = \infty$), where we obtain $H = 0$ and thus *minimal surfaces*.

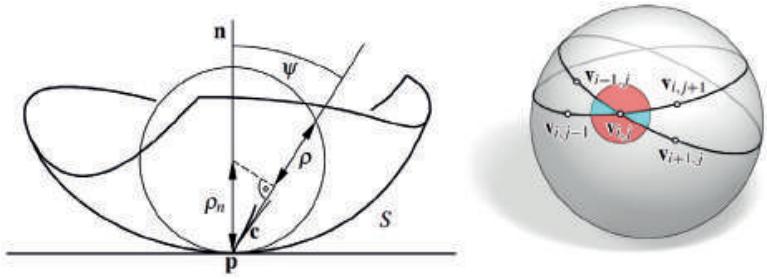


Figure 4: Left: Meusnier's theorem, relating the curvature of a curve $\mathbf{c}$ passing through $\mathbf{p}$ to normal curvature at $\mathbf{p}$ in direction of $\mathbf{c}'$. Right: Spherical vertex star.

CMC surfaces are a mathematical representation of inflated membranes, such as soap bubbles or pneus. Their curvature behavior corresponds to the equilibrium shape caused by a pressure difference and can form both synclastic and anticlastic surface regions. Minimal surfaces are a subset of CMC surfaces, in which the pressure difference is zero. They

can be found in nature in the form of soap films, creating the minimal area within given boundaries.

We have just derived another important fact: *Curved support structures from circular strips of the same radius r and with a right node angle model surfaces with constant mean curvature $H = 1/r$; in particular, straight strips yield models of minimal surfaces. The strips of the support structure are attached along those curves which bisect the principal directions.* These bisecting directions are those with extremal geodesic torsion.

CMC surfaces, and in particular minimal surfaces, are so-called *isothermic surfaces*. They possess a parameterization $\mathbf{s}(u,v)$ in which the isoparameter lines are principal curvature lines and which describes a conformal (angle preserving) mapping from the (u,v)-parameter plane to the surface. This parameterization maps the bisecting grid $u \pm v =$ const. onto those curves along which our strips can be attached. This fact is used later in our algorithm.

If we require a constant, but not necessarily right node angle 2ϕ, Euler's formula shows that the surface S possesses a linear relation between its principal curvatures, $A\kappa_1 + B\kappa_2 = 1/r$, with $A = \cos^2\phi$, $B = \sin^2\phi$. Structures from straight strips $(r = \infty)$ lead to surfaces with a constant negative ratio of principal curvatures $\kappa_1/\kappa_2 = -B/A$. Those have recently been studied by Jimenez et al. (2018).

2.1 Discretization

For digital design of the structures we have in mind, it is very useful to have discrete models of the network of curves along which the strips are attached. This means that we have to come up with quad meshes whose mesh polylines discretize the system of curves of constant normal curvature κ_n on a smooth surface.

It is useful to know about Meusnier's formula and its geometric interpretation (**Fig. 4**). The formula relates curvature κ of a curve $\mathbf{c}$ on a surface to its normal curvature κ_n via $\kappa_n = \kappa \cos\Psi$, where Ψ is the angle between the curve's osculating plane and the surface normal. Geometrically, this means that the osculating circle of $\mathbf{c}$ (which has radius $\rho = 1/\kappa$) lies on a sphere of radius $\rho_n = 1/\kappa_n$, which is tangent to the surface. Note that κ_n only depends on the tangent direction. Hence, all curves on a surface which pass through a given point $\mathbf{p}$ with a fixed tangent possess osculating circles at $\mathbf{p}$ which lie on the corresponding

Meusnier sphere. This knowledge allows us to prove the following fact:

A quad mesh of regular combinatorics for which each vertex and its four connected neighbors lie on a sphere of constant radius r, discretizes a network of curves of constant normal curvature $1/r$ on a smooth surface.

For a proof, we consider a vertex $\mathbf{v}_{i,j}$ and its four connected neighbors $\mathbf{v}_{i-1,j}$, $\mathbf{v}_{i+1,j}$, $\mathbf{v}_{i,j-1}$, and $\mathbf{v}_{i,j+1}$ (**Fig.4**). By our assumption, these 5 points lie on a sphere $S_{i,j}$ of radius r. The three points $\mathbf{v}_{i-1,j}$, $\mathbf{v}_{i,j}$, $\mathbf{v}_{i+1,j}$ are consecutive points on a discrete parameter line and lie on a circle, which is a discrete version of the osculating circle of that parameter line at $\mathbf{v}_{i,j}$. Of course, this circle lies on $S_{i,j}$. Likewise, the three points $\mathbf{v}_{i,j-1}$, $\mathbf{v}_{i,j}$, $\mathbf{v}_{i,j+1}$ define a discrete osculating circle for the other discrete parameter line, which also lies on $S_{i,j}$. These osculating circles can be seen as tangent to an underlying surface and thus we see that the sphere $S_{i,j}$ is tangent to that surface and contains the osculating circles. Hence, it is the common Meusnier sphere to the two parameter lines through $\mathbf{v}_{i,j}$ As all these vertex spheres have radius r, we have a discretization of the network of curves of constant normal curvature $1/\mathrm{r}$.

In order to achieve a right angle in the discrete sense we require that the sum of opposite angles around $\mathbf{v}_{i,j}$ is equal (**Fig.4**, right). This discrete orthogonality condition is also used for conical meshes (Liu et al. (2006)). If we apply the right angle condition in addition to the sphere condition, we obtain a new discretization of CMC surfaces. Only in the special case of $r = \infty$, where the spheres degenerate to planes, do we arrive at a known asymptotic discretization of minimal surfaces.

A careful study of meshes with vertex spheres (not necessarily of constant radius) and their special cases is left for future research; it is more a topic of discrete differential geometry rather than architectural geometry.

2.2 Implementation

A key step is the computation of isothermic networks on CMC surfaces. Networks are represented as quad dominant meshes. An isothermic mesh M on top of a reference surface S is characterized by (i) edges aligned to principal curvature directions of S and (ii) quadrilateral faces that are as square as possible. As mentioned above, such networks always exist on CMC surfaces. The main difference between two such networks on the same CMC surface is the size of the squares.

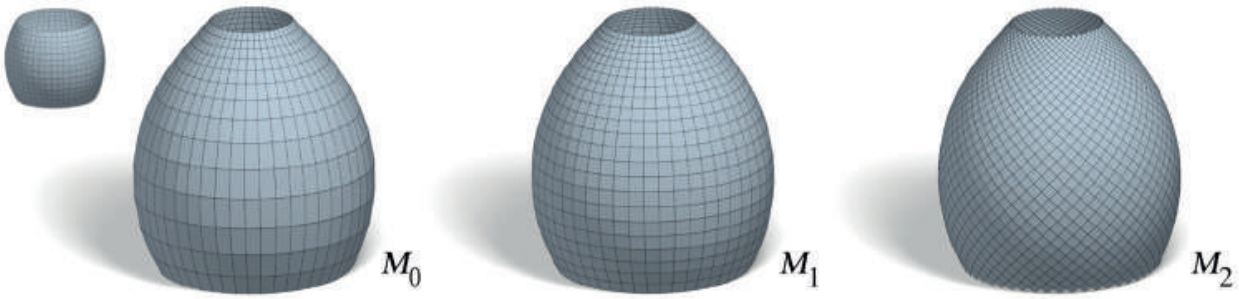


Figure 5: From left to right: Principal but not yet isothermic mesh on an unduloid surface. Straightening deforms the rectangles of M_0 into squares of M_1. Mid-edge subdivision yields a mesh M_2 with edge polylines aligned to directions of constant normal curvature on the unduloid.

Approach. We start from an initial quad mesh whose edges are aligned to principal curvature directions of a reference CMC surface S. If such a mesh M_0 cannot be created with the help of a known parametrization of S, we use T.MAP (Evolute GmbH (2018)) to initialize M_0.

We iteratively deform M_0 to an isothermic mesh by letting it slide along principal curvature directions of S until all faces are as square as possible; we refer to this process as *straightening*. Note that straightening does not provide a solution to the difficult problem of singularity resolution since it does not change mesh combinatorics.

The main tool used during straightening is so-called *guided projection* as introduced by Tang et al. (2014). Guided projection allows us to prescribe a set of constraints in terms of vertex coordinates, face normals, curvature directions, and other mesh/surface-related quantities. A solution to this set of constraints yields an isothermal mesh M_1 to which we apply mid-edge subdivision in order to obtain a mesh M_2 whose edges are aligned to directions of constant normal curvature on S (**Fig. 5**). If the density of curves is not chosen with care, the resulting isothermic mesh may only cover part of S as illustrated in the small inset. In this example we added additional horizontal curves and let the straightening process distribute them across the surface. As a rule of thumb, it is beneficial to have a reference surface S larger than the actual structure to give the straightening process enough room.

We lay out strips orthogonal to S along each polyline of constant normal curvature of M_2. To this end we use the corresponding normals of the reference surface S as initial rulings. Those strips are not yet developable and are subject to optimization via guided projection.

Guided projection. In a nutshell, guided projection takes a set of simultaneous equations and "solves" them by performing Gauss-Newton iterations. The important observation made by Tang et al. (2014) is that this simple idea performs especially well if the involved equations are, at most, quadratic in the unknowns. We will not go into detail of the Gauss-Newton algorithm and refer to Tang et al. (2014). In the remainder of this section we will talk about the actual equations that we use.

Given a polygonal mesh M, its vertex coordinates $\mathbf{v}_i$, $i = 1,\ldots,n$, define our main set of variables. All other quantities, such as face normals etc., are derived quantities that are tied to vertex coordinates via equations. To transform M into a mesh with planar faces f_j, $j = 1,\ldots,m$, we introduce the vertex normals $\mathbf{n}_j$ as additional variables and the equations

$$1 = \mathbf{n}_j^T \mathbf{n}_j$$
$$0 = \mathbf{n}_j^T (\mathbf{v}_{i_2} - \mathbf{v}_{i_1})$$

where j run over all faces f_j of M and the pair $(\mathbf{v}_{i_1}, \mathbf{v}_{i_2})$ runs over all edges of f_j. A mesh that satisfies these equations has planar faces f_j and corresponding normal vectors $\mathbf{n}_j$. We use these *planarity constraints* to make a quad strip developable.

Recall that we are not trying to solve a form finding problem: the vertices $\mathbf{v}_i$ are constrained to move on S. We can implement this restriction by introducing equations of the form

$$0 = \left(\mathbf{m}_i^T (\mathbf{v}_i - \mathbf{p}_i)\right)^2 + \varepsilon \|\mathbf{v}_i - \mathbf{p}_i\|^2$$

where $\mathbf{p}_i$ is the footpoint of $\mathbf{v}_i$ on S and $\mathbf{m}_i$ is the normal of S at $\mathbf{p}_i$. This is commonly referred to as *tangent-distance-minimization* and restricts the movement of $\mathbf{v}_i$ to the tangent plane of S at $\mathbf{p}_i$. Note that $\mathbf{p}_i$ and $\mathbf{m}_i$ are not treated as variables – their values are updated between Gauss-Newton iterations.

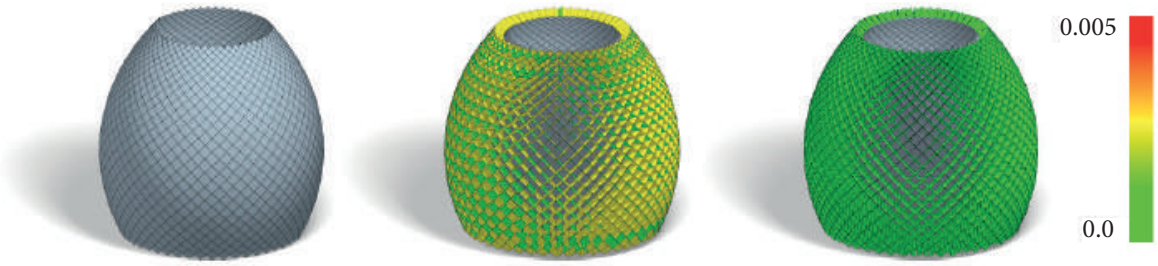


Figure 6: From left to right: Base mesh aligned to curves of constant normal curvature. Initial set of strips, color coded according to developability. Post optimized strips after one round of subdivision and 20 iterations of guided projection.

With the help of the projection operator we can also achieve alignment of mesh edges to prescribed directions – in our case directions of principal curvature which are precomputed on S. We project the midpoint of edge $(\mathbf{v}_i, \mathbf{v}_j)$ onto S to obtain the principal curvature directions $\mathbf{d}_1$ and $\mathbf{d}_2$. The edge $(\mathbf{v}_i, \mathbf{v}_j)$ should be aligned to one of those directions. The corresponding *alignment equation* reads

$$0 = \mathbf{d}_1^T(\mathbf{v}_i - \mathbf{v}_j)\mathbf{d}_2^T(\mathbf{v}_i - \mathbf{v}_j) = (\mathbf{v}_i - \mathbf{v}_j)^T\mathbf{d}_1\mathbf{d}_2^T(\mathbf{v}_i - \mathbf{v}_j).$$

To obtain a conformal parametrization we use an equation introduced in the context of circle packings (Schiftner et al. (2009)). For each vertex $\mathbf{v}_i$ we introduce a scalar variable $l_i > 0$ and the equations

$$0 = (\mathbf{v}_i - \mathbf{v}_j)^T(\mathbf{v}_i - \mathbf{v}_j) - (l_i + l_j)^2$$

where j runs over all neighbors of vertex $\mathbf{v}_i$.

Optimizing discrete structures typically requires a *fairing term* to ensure overall mesh quality. When dealing with quadrilateral meshes it is sufficient to require that a generic vertex $\mathbf{v}_{i,j}$ (cf. **Fig. 4**) and its four neighbors satisfy

$$2\mathbf{v}_{i,j} = \mathbf{v}_{i-1,j} + \mathbf{v}_{i+1,j}$$
$$2\mathbf{v}_{i,j} = \mathbf{v}_{i,j-1} + \mathbf{v}_{i,j+1}.$$

Finally, we may want to optimize a given mesh towards a mesh with *spherical vertex stars* as explained above. For each vertex we introduce

a radius r_i, sphere center $\mathbf{c}_i$, and equation

$$0 = (\mathbf{v}_i - \mathbf{c}_i)^T(\mathbf{v}_i - \mathbf{c}_i) - r_i^2.$$

A neighboring vertex $\mathbf{v}_j$ has to satisfy $0 = (\mathbf{v}_j - \mathbf{c}_i)^T(\mathbf{v}_j - \mathbf{c}_i) - r_i$.

2.3 Results

We start with a remark on the color coding of strips in this section. To judge the developability of a quad strip we measure the planarity of individual quads $(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4)$ as the distance of its diagonals. To factor out scaling this number needs to be normalized. To do this we divide by the mean length of diagonals and arrive at the following planarity score:

$$\mathrm{ps}(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4) = \frac{2d(\overline{\mathbf{v}_1\mathbf{v}_3}, \overline{\mathbf{v}_2\mathbf{v}_4})}{d(\mathbf{v}_1, \mathbf{v}_3) + d(\mathbf{v}_2, \mathbf{v}_4)}.$$

If we imagine a 1×1 m square, a diagonal distance of 1 cm maps to a planarity score of about 0.007. When applying color, pure red maps to a planarity score of 0.005 or higher.

Starting from an initial quad mesh M aligned to principal curvature directions of a reference surface S, we used the fairness, alignment, and conformality constraints while restricting movement to S via the closeness term to turn M into an isothermic mesh. The diagonals of M define the contact curves along which strips are attached. Strips are optimized for developability using the planarity constraint while constraining their lower boundary curves to S and their upper boundaries to a parallel surface at prescribed distance h.

Unduloid. The unduloid is obtained as a surface of revolution of an elliptic catenary. Figure 6 shows a network of curves with constant normal curvature, an initial set of strips using surface normals of a triangle mesh representation as node axis, and a set of optimized strips.

We use the spherical vertex star property to explore the deformation behavior of the curve network. The network of diagonals extracted from the isothermic network M_I (**Fig. 5**) is a very good starting point to compute a discrete structure that satisfies this condition. Figure 7 shows the effect of reducing the radius of the red circle shown in the small inset while preserving edge lengths and the spherical vertex star condition

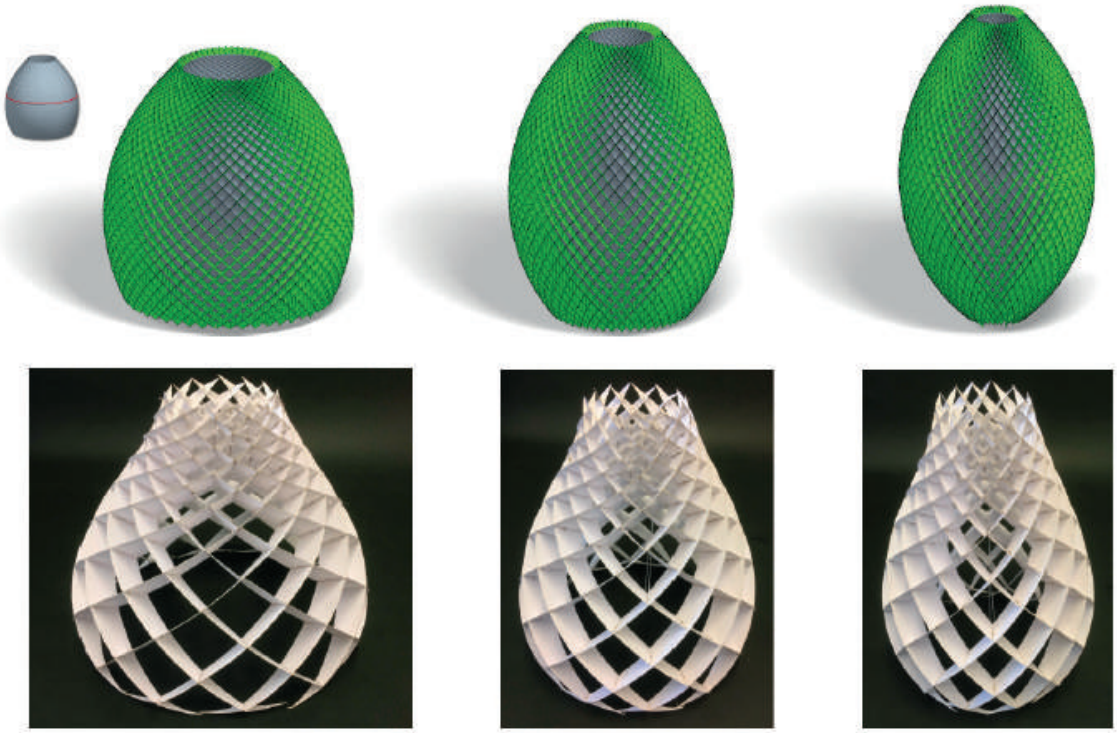


Figure 7: Deformation of the unduloid using the spherical vertex star property. The radius of the red curve was reduced by 10, respectively 20% to drive the deformation. The bottom row shows a corresponding paper model employing a coarser curve network on the upper half of the unduloid.

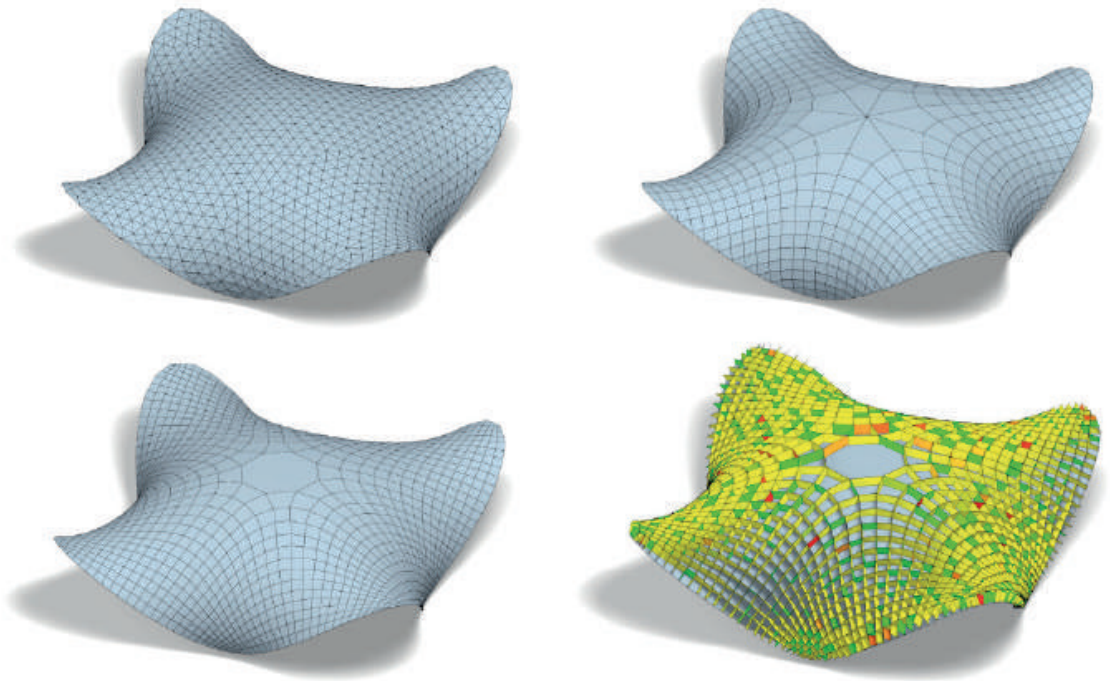


Figure 8: From left to right and top to bottom: Ocean reference surface, isothermic mesh, network of constant normal curvature curves, and the initial set of strips following the curve network.

with a fixed radius for all vertex spheres equal to the inverse of mean curvature H of the undoloid. The value of $H = 1.25$ was estimated on a triangle mesh representation of the unduloid reference surface.

Ocean. Reference surface ($H = 0.68$), isothermic mesh, curve network, and initial strips are shown in Figure 8. The set of optimized strips is shown in Figure 9. Strip quality deteriorates when approaching the boundary. This cannot be fixed by optimization of strips since rulings are determined by geodesic torsion τ_g and normal curvature κ_n of the guiding curves (which, in our case are uniquely defined by S and hence cannot be changed individually). As a remedy one needs to explore nearby reference shapes with a more favorable ratio of τ_g and κ_g along the curve network, or consider twisting lamellas during construction to allow deviation from a developable strip.

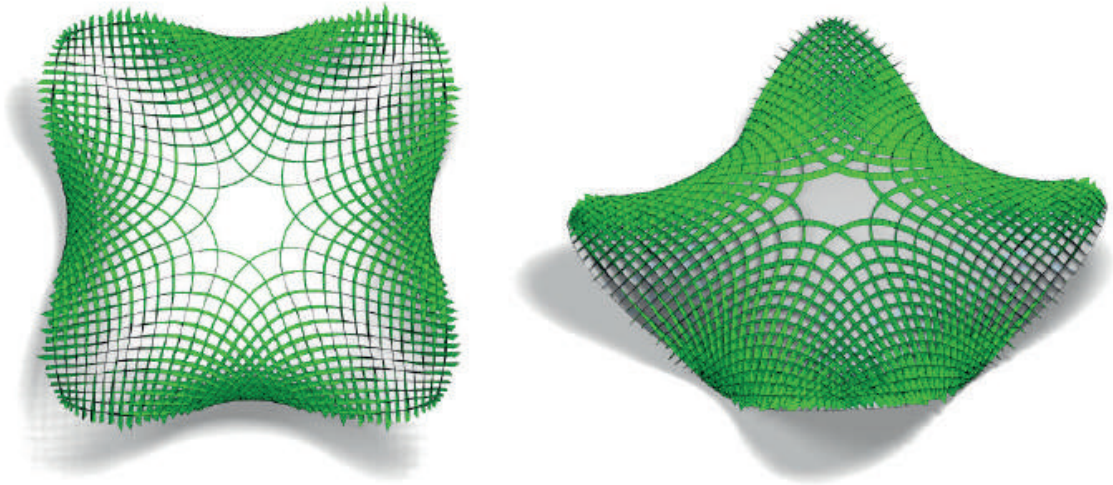


Figure 9: Strips (of radius 1.47) after 2 rounds of subdivision, each followed by 20 iterations of guided projection.

3. The Asymptotic Gridshell

The design and construction of the Asymptotic Gridshell simultaneously serve as motivation and case study for this paper. The structure illustrates the transfer from a purely geometrical concept to an architectural project, and presents the benefits and challenges of designing with rigorous geometrical constraints, fabricating and assembling a strained lamella grid, and analysing its load-bearing behavior.

3.1 Design process

Surface. The initial surface was designed using a fast digital routine for minimal surface approximation. While the algorithm implements the geometric requirements of a CMC surface, the designer is responsible for all other requirements like site, safety and functionality. A key challenge was to find a shape that would benefit an efficient shell-like load-transfer by approximating qualities of a funicular form. Manipulating the position and shape of two boundary curves, we created an intricate, mussel-shaped design with high double curvature and arch-shaped edges. Three curved horizontal supports nestle well along the complex site boundaries. The surface creates a circular oculus around an existing tree and opens two archways that allow circulation throughout the courtyard (**Fig. 11**, left). Once the boundary curves were defined, the minimal surface was modeled more accurately as NURBS surface using the Rhino plugin TeDa (Philipp et al. (2016)).

Figure 10: The Asymptotic Gridshell was completed in October 2017. The structure is 5 m high and spans 9 × 12 m. It is built entirely from 1.5 mm-thick and 100 mm-wide steel lamellas. (Image: Felix Noe)

Network and lamellas. The network is designed along the paths of constant normal curvature (asymptotic curves) bisecting an isothermic principal curvature network (Section 2.2). This produces an almost square cell layout which is beneficial both structurally and graphically. Furthermore, the diagonal alignment with the principal curvature directions creates advantages for future facade solutions with single-curved or planar quadrilateral panels (Liu et al. (2006)).

The lamella geometry was simply defined by the normal vectors $\mathbf{n}$ (**Fig. 3**, right). This creates a well-defined ruled surface strip with straight intersections deviating from a truly developable surface. As a consequence, the structural strips are twisted during assembly and experience elastic strain.

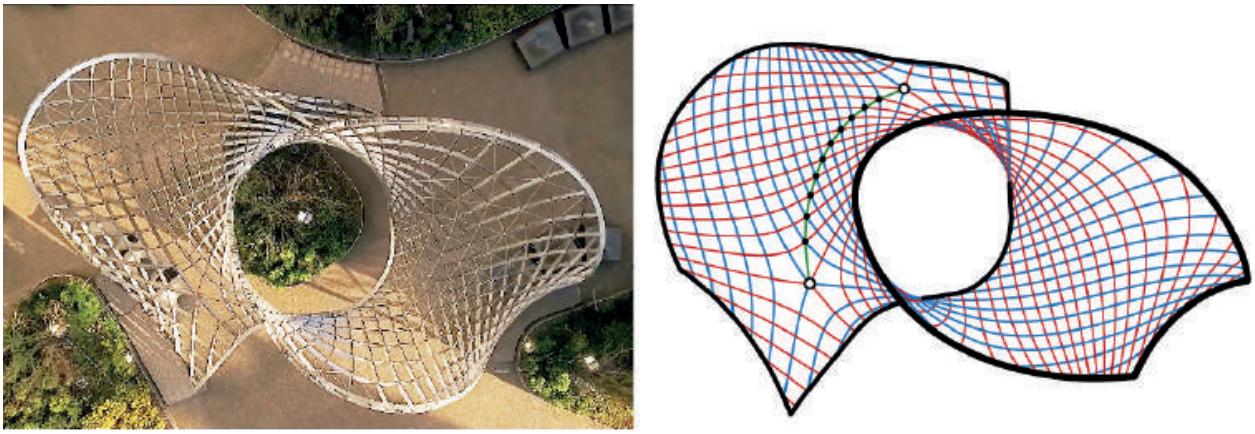


Figure 11: Left: The Asymptotic Gridshell was designed to fit an existing green courtyard. The arch-shaped design fosters the load-bearing behavior of a gridshell. Right: Two planar surface points create singularities which were iteratively adjusted during the design process to align in one principal curvature line. (Image: Felix Noe)

The geometry of network and lamellas is dependent on the curvature of the surface. A high Gaussian curvature causes a high torsion of the lamellas which is limited by the elastic capabilities of the material. Planar surface points, on the other hand, create singularities within the network, and thus have a large impact on the layout and stability of the grid structure. Both factors were carefully adjusted by controlling the progression of boundary curves, re-computing the surface and testing the new network layout. In the case of the Asymptotic Gridshell, there are two singularities on opposite sides, east and west of the central oculus. Both singularities are designed as congruent, regular, hexagonal nodes.

They were arranged on the same principal curvature line.

The grid density is determined by the subdivision of this connection axis. (**Fig. 11**, right)

3.2 Construction development

Implications of curvature. The three curvatures (normal curvature, geodesic curvature and geodesic torsion) within the structure are created either during fabrication (of circular lamellas) or during assembly (through elastic bending and twisting). The $\mathbf{t}$, $\mathbf{u}$ and $\mathbf{n}$-vector resemble the x, y and z-axis of our lamellas profiles. The stresses due to elastic deformation are directly related to the curvature values.

Let us first look at the gridshells of Frei Otto which mark the starting point of our construction development. (**Fig. 2**, top) Otto's design network is subject to all three curvatures. The timber lattice had to be constructed from slender, doubly symmetrical profiles in order to be bent and twisted around all axes. Any shape within the permissible bending radii can be built from such a grid.

Our curve networks, on the other hand, follow the paths of constant normal curvature. The grid can be constructed from straight or circular lamellas orientated perpendicular to the surface. As a consequence, no bending around the local y-axis (sideways vector) of the profiles is necessary during assembly. The geodesic curvature results in bending around the z-axis (normal vector), and the geodesic torsion creates twisting of the lamellas around their x-axis (tangent vector). When choosing the profile thickness, the stiffness has to be adjusted to accommodate the maximum twist and minimal bending radii and keep deformation elastic.

In contrast to the timber gridshells of Frei Otto, the lamella grid is restricted to the family of shapes described in Section 2. This is due to the restricted deformation (i.e., high stiffness) in respect to the y-axis (i.e., constant normal curvature).

Post-stiffening strategy. If the elastic deformation of a material is used to construct a curved geometry, this inevitably poses the question of deflection and stability under self-weight and external loads. Increasing the bending stiffness is not an option if all elements are to be bent elastically into a curved geometry. Lienhard calls this discrepancy a "paradoxon that underlies all bending-active structures" (Lienhard (2014), p. 141). These opposing requirements are solved by introducing two parallel lay-

ers of lamellas. Each layer is sufficiently slender to be bent and twisted elastically into its target geometry. Once the final geometry is installed, the two layers are coupled with shear blocks in regular intervals to increase the overall stiffness.

This construction technique was tested with two prototypes, one in timber and one in steel, each with an approx. 4×4 m span (**Fig. 12**). The timber lamellas were bent individually and connected to a rigid edge-beam. The lamellas are arranged on two levels to allow for uninterrupted timber profiles. The steel prototype, on the other hand, was assembled flat (a benefit of straight lamellas) and subsequently transformed into the spatial geometry. Here the lamellas are slotted and interlocked at one level.

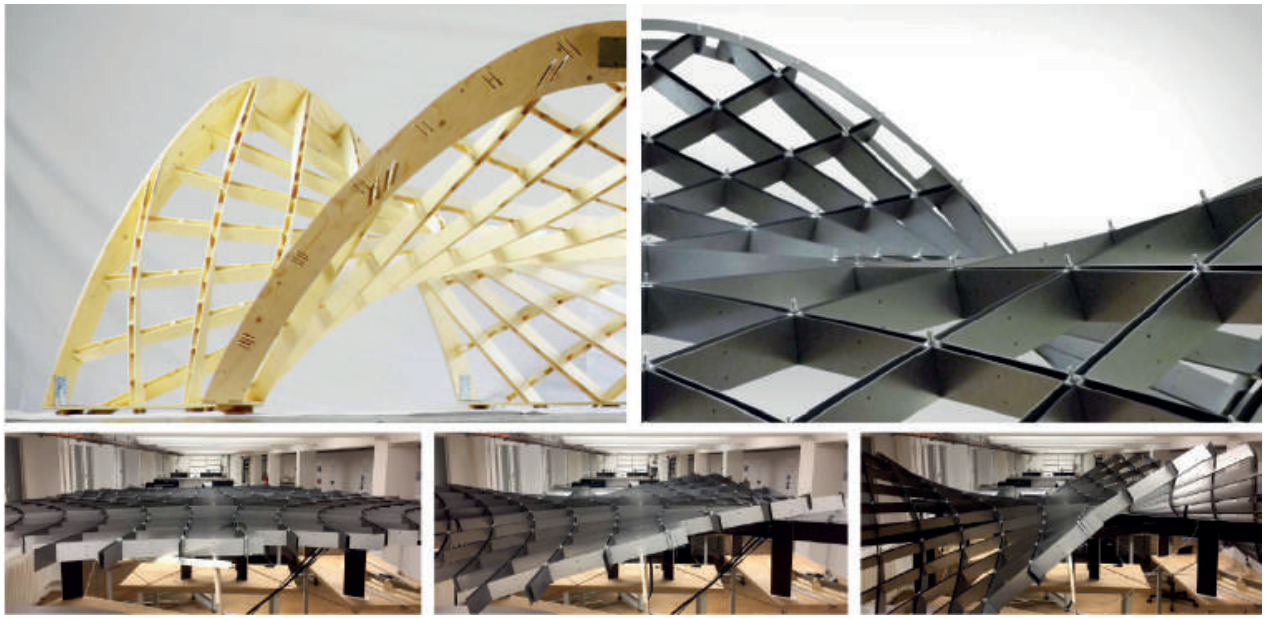


Figure 12: The structure was tested with two prototypes, one in timber and one in steel, each with an approx. 4×4 m span. Left: The lamellas of the timber prototype are arranged on separate levels to allow the use of uninterrupted profiles. Right: The lamellas of the steel prototype are interlocked in one level. They were first assembled flat and then transformed into the curved geometry (bottom row). (Images: Eike Schling)

Grid joint. All nodes are congruent with an intersection angle of 90 degrees. They can thus be constructed with repetitive, orthogonal joints (**Fig. 13**). At each intersection, two pairs of parallel lamellas are interlaced through perpendicular slots. The slots are twice as wide as the material thickness to allow a rotation of up to 60 degrees during assembly. The lamellas are locked by two star-shaped washers on top and bottom.

A single carriage bolt and nut is used to fix each joint after they are transformed into the target geometry. An additional cross-shaped clamp fixes the diagonal cables. The Asymptotic Gridshell was constructed from 100 mm high and 1.5 mm thick, straight, stainless steel lamellas at parallel offset of 25 mm following the detailing and construction strategy of the steel prototype.

Figure 13: The typical grid joint is assembled with two parallel lamellas in each direction. Two standardized star-shaped washers fix the 90 degree intersection angle and create a central axis for the carriage bolt. The steel cables are also constructed in pairs and are fixed by a cross-shaped clamp. (Images: Felix Noe)

3.3 Construction process

Fabrication. Designing networks along constant normal curvature lines greatly simplifies fabrication: All lamellas are fabricated flat as either straight strips (on minimal surfaces) or circular strips (on any CMC-surface). The edge lengths from node to node, are the only variable information needed to produce fabrication drawings. The distances are simply marked along the standardized strips.

The lamellas of the Asymptotic Gridshell were laser-cut straight, which allowed for minimal offcuts and easy transport. The fabrication of washers and clamps was incorporated in the same laser-cutting procedure offering a cost-efficient production of all parts.

Erection process. The lamellas are slotted together to form a flat (for minimal surfaces) or spherical (for general CMC surfaces) girder (**Fig. 14**). In this state, the lamellas display no geodesic torsion. The intersection angles are not yet constant. The joints are flexible and allow for a scissor movement. This lamella grillage can be deformed within a predefined family of shapes, one of which is the designed reference surface. It is found by enforcing a constant node angle of 90 degrees. The

Figure 14: The straight lamellas are interlocked by hand into flat segments. The segments are then transformed elastically into their designed shape by fixing each node to 90 degrees. Nine of these segments were prefabricated off site. (Images: Eike Schling)

Figure 15: Installation on site. The prefabricated segments of up to 400 kg, where positioned with a crane, temporarily supported, and bolted together by hand. To activate the structural behavior of a gridshell, the completed grid is braced diagonally and fixed at supports in vertical and horizontal direction. (Images: Andrea Schmidt)

deformation behavior follows the same principles as described in Section 2.3 (**Fig. 7**). This kinetic behavior is called a *compliant mechanism* (Howell (2002)). It enables an elastic erection process without formwork. Of course, this mechanism is subject to gravity and other external loads and needs to be verified by selective measurements. Its further study is part of future research.

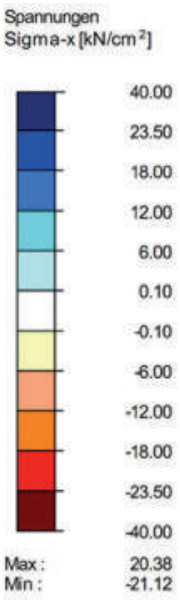
The Asymptotic Gridshell was prefabricated in nine individual segments (**Fig. 15**). Each grillage was first assembled flat, then placed on a simple, cross-shaped stand and elastically deformed into its designated anticlastic curvature. Locking each node at 90 degrees and adding edge supports created rigid segments, which were then combined on site, like a large 3D puzzle. By fixing the supports and adding diagonal steel cables, this structure becomes an efficient, load-bearing gridshell.

3.4 The completed pavilion

The Asymptotic Gridshell is the first architectural structure that utilizes the geometric potentials of a constant normal curvature network on a constant mean curvature surface (**Fig. 16**). The gridshell spans 9×12 m and covers an area of approx. 90 m². Its surface weight is approximately 18 kg/m², a total of 1.6 tons. A decisive factor for the aesthetical quality of both the shape and the lamella grid are owed to their formation process, following the curvature constraints of this design method. The slender lamellas create a gradient graphical effect with virtually full transparency at a straight view, and an almost opaque appearance at an inclined view (**Fig. 1**).

3.5 Load-bearing behavior

FEM analysis. The network geometry was modeled in Rhino/Grasshopper and exported as a discrete model to RFEM (Dlubal Software GmbH (2018)), where all necessary structural information was added. The geometric values of geodesic curvature and geodesic torsion were measured individually for each discrete element along the smooth curves and translated into strain loads in RFEM. This strategy enabled us to induce the residual stresses without modeling the actual assembly process (**Fig. 16**). Due to intense twisting of the lamellas, additional normal stresses according to the effects of helix torsion are to be expected (Lumpe and Gensichen (2014) p. 118–128). These effects are not considered in the FE analysis which uses beam elements.



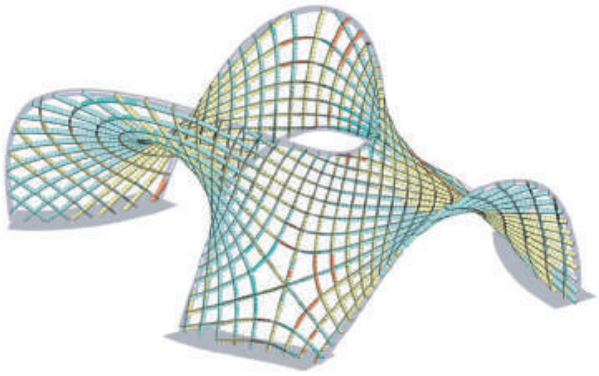


Figure 16: The load-bearing structure of the Asymptotic Gridshell. The lamellas are bent and twisted to form an anticlastic network with two singularities. The diagonal bracing is arranged at every second node. The diagram shows the surface stresses of the lamella grid resulting from both the elastic erection process and self-weight. All stresses stay within the elastic range.

Global and local behavior. We observed the hybrid load-bearing behavior of two competing mechanisms; a grillage and a gridshell. Due to the bending stiffness in their strong axis, the lamellas are able to act as a beam grillage. This is needed to account for the local planarity of the asymptotic curves (due to their vanishing normal curvature) and to stabilize the open edges. At the same time, the lamellas form a doubly-curved structure. Bracing this quadrilateral network with diagonal cables and creating fixed supports (in vertical and horizontal direction) activates behavior of a gridshell. Which of the two mechanism dominates is highly dependent on the design shape.

The arch-shaped boundaries of the Asymptotic Gridshell promote a shell-like behavior. Expanding the design spectrum to all CMC surfaces enables us to create synclastic shapes and further adapt to a funicular form. dependent on the design shape. The arch-shaped boundaries of the Asymptotic Gridshell promote a shell-like behavior. Expanding the design spectrum to all CMC surfaces enables us to create synclastic shapes and further adapt to a funicular form.

The elastic erection process results in restraint (residual) stresses within the lamellas. Due to the low profile thickness, the initial bending moments stay low and have minor effects on the global behavior. However, compression of these curved elements increases the bending moment in their weak axis. The strategy of doubling and coupling lamellas is therefore essential to control local buckling.

The optimal orientations for compression and tension elements of a gridshell run along the principal stress trajectories. However, in our method, we choose to follow a geometrically optimized orientation along the directions of constant normal curvature, taking into account an increase of stresses.

4. Conclusion

Combining repetitive curvature parameters with an elastic construction holds great potentials for the fabrication, assembly and load-bearing behaviour of strained gridshells.

The technical requirements (straight or circular lamellas, congruent nodes) translate nicely into differential geometric characterizations of the curve networks and reference surfaces realizable with this approach.

They even motivated the development of novel discrete structures (quad meshes with spherical vertex stars) which deserve interest from a purely geometric perspective.

The geometric properties greatly simplify the construction process: The lamellas have a beneficial orientation orthogonal to the design surface. They can be fabricated flat and straight or with a constant radius. All joints are identical and orthogonal. The elastic erection process takes advantage of a compliant mechanism, determining the design shape without formwork.

The elastic behavior, however, poses the challenge to avoid deflections and instability under self-weight and external loads. This paradoxon of bending-active structures was addressed within the design and construction process. Even though our structures can only assume CMC surfaces, a substantial freedom in the design process remains with the potential to adjust to architectural and structural requirements.

Future Research. Meshes with spherical vertex stars are a novel surface discretization which opens up new avenues of research in discrete differential geometry. Our study also opens up two promising research fields that combine the disciplines of mathematics, architecture and engineering: (i) the investigation of the kinetic behaviour of elastic grids (compliant mechanisms) and the dependency of geometry and mechanics therein, (ii) the optimization of surfaces for both geometric requirements (like constant mean curvature) and structural performance (for shell structures).

Finally, we aim to develop further construction techniques and facade solutions for strained gridshells built from straight and circular lamellas.

Acknowledgements.

The Asymptotic Gridshell was designed and planned in collaboration with Denis Hitrec from the University of Ljubljana. We would like to thank Matthias Müller, locksmith at the Technisches Zentrum, TUM, as well as the Brandl Metallbau GmbH & Co. KG in Eitensheim for their extensive support in steel fabrication. We further thank PFEIFER Seil- und Hebetechnik GmbH, in Memmingen for supporting the tailoring of the steel cables. The plugin TeDa was provided and supported by Anna Bauer and Benedikt Phillipp from the Chair of Structural Analysis, Prof. Dr.-Ing. Bletzinger, TUM. Further support in MESH modelling and curvature line extraction was granted by Alexander Schiftner and Evolute GmbH

in Perchtoldsdorf, Austria. The Asymptotic Gridshell was assembled with the help of five dedicated students: Beatrix Huff, Andrea Schmidt, Viktor Späth, Miquel Lloret Garcia and Maximilian Gemsjäger. The Project was funded by the Faculty of Architecture of TUM, the Dr. Marschall Foundation, the Leonhard Lorenz Foundation and the Architectural Research Incubator (ARI). We thank Christian Müller for providing the paper model of the unduloid (**Fig. 7**) and Hao Pan for sharing a triangle mesh of the CMC surface used in Figure 8. This work was supported by SFB-Transregio program Geometry and Discretization (FWF grant no. I 2978) and by the project "Geometry and Computational Design for Architecture and Fabrication" at Vienna University of Technology.

References

Bobenko, A. I. and Y. B. Suris (2008). *Discrete Differential Geometry: Integrable Structure*. American Mathematical Society.

Burkhardt, B. (Ed.) (1978). *IL13: Multihalle Mannheim*. Stuttgart: Institut für leichte Flächentragwerke (IL)/Karl Krämer Verlag.

Dlubal Software GmbH (2018). RFEM 5. www.dlubal.com.

Evolute GmbH (2008–2018). EvoluteTools T.MAP. www.evolute.at.

Howell, L. L. (2002). *Compliant Mechanisms*. New York: Wiley & Sons.

Jimenez, M. R., C. Müller, and H. Pottmann (2018). Discretizations of surfaces with constant ratio of principal curvatures. *Discrete & Computational Geometry*. to appear.

Lienhard, J. (2014, April). Bending-active structures: form-finding strategies using elastic deformation in static and kinetic systems and the structural potentials therein.

Liu, Y., H. Pottmann, J. Wallner, Y.-L. Yang, and W. Wang (2006, July). Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graph.* 25(3), 681–689.

Lumpe, G. and V. Gensichen (2014). *Evaluierung der linearen und nichtlinearen Stabstatik in Theorie und Software: Prüfbeispiele, Fehlerursachen, genaue Theorie*. Ernst & Sohn.

Philipp, B., M. Breitenberger, I. D'Auria, R. Wüchner, and K.-U. Bletzinger (2016). Integrated design and analysis of structural membranes using the isogeometric B-rep analysis. *Computer Methods in Applied Mechanics and Engineering 303*, 312–340.

Schiftner, A., M. Höbinger, J. Wallner, and H. Pottmann (2009). Packing circles and spheres on surfaces. *ACM Trans. Graphics 28*(5), #139,1–8. Proc. SIGGRAPH Asia.

Schiftner, A., N. Leduc, P. Bompas, N. Baldassini, and M. Eigensatz (2012). Architectural geometry from research to practice: The eiffel tower pavilions. In *Advances in Architectural Geometry*.

Schling, E. and R. Barthel (2017). Experimental studies on the construction of doubly curved structures. *Detail structure* (01), 52–56.

Tang, C., M. Kilian, P. Bo, J. Wallner, and H. Pottmann (2016). Analysis and design of curved support structures. In S. Adriaenssens, F. Gramazio, M. Kohler, A. Menges, and M. Pauly (Eds.), *Advances in Architectural Geometry 2016*, pp. 8–23. VDF Hochschulverlag, ETH Zürich.

Tang, C., X. Sun, A. Gomes, J. Wallner, and H. Pottmann (2014). Form-finding with polyhedral meshes made simple. *ACM Trans. Graphics 33*(4). Proc. SIGGRAPPH.

Discrete CMC surfaces for doubly-curved building envelopes

Xavier Tellier, Laurent Hauswrith, Cyril Douthe, Olivier Baverel

Xavier Tellier
xavier.tellier@enpc.fr
Laboratoire Navier, Ecole des Ponts, IFSTTAR, CNRS, France

Laurent Hauswirth
laurent.hauswirth@u-pem.fr
Université Paris-Est, Laboratoire d'Analyse et de Mathématiques Appliquées, France

Cyril Douthe
cyril.douthe@enpc.fr
Laboratoire Navier, Ecole des Ponts, IFSTTAR, CNRS, France

Olivier Baverel
olivier.baverel@enpc.fr
Laboratoire Navier, Ecole des Ponts, IFSTTAR, CNRS, France

Keywords:
Free form building envelopes, rationalization, constant mean curvature surfaces

Abstract

Constant mean curvature surfaces (CMCs) have many interesting properties for use as a form for doubly curved structural envelopes. The discretization of these surfaces has been a focus of research amongst the discrete differential geometry community. Many of the proposed discretizations have remarkable properties for envelope rationalization purposes. However, little attention has been paid to generation methods intended for designers.

This paper proposes an extension to CMCs of the method developed by Bobenko, Hoffmann and Springborn (2006) to generate minimal S-isothermic nets. The method takes as input a CMC (smooth or finely triangulated), remeshes its Gauss map with quadrangular faces, and rebuilds a CMC mesh via a parallel transformation. The resulting mesh is S-CMC, a geometric structure discovered by Hoffmann (2010). This type of mesh have planar quads and offset properties, which are of particular interest in the fabrication of gridshells.

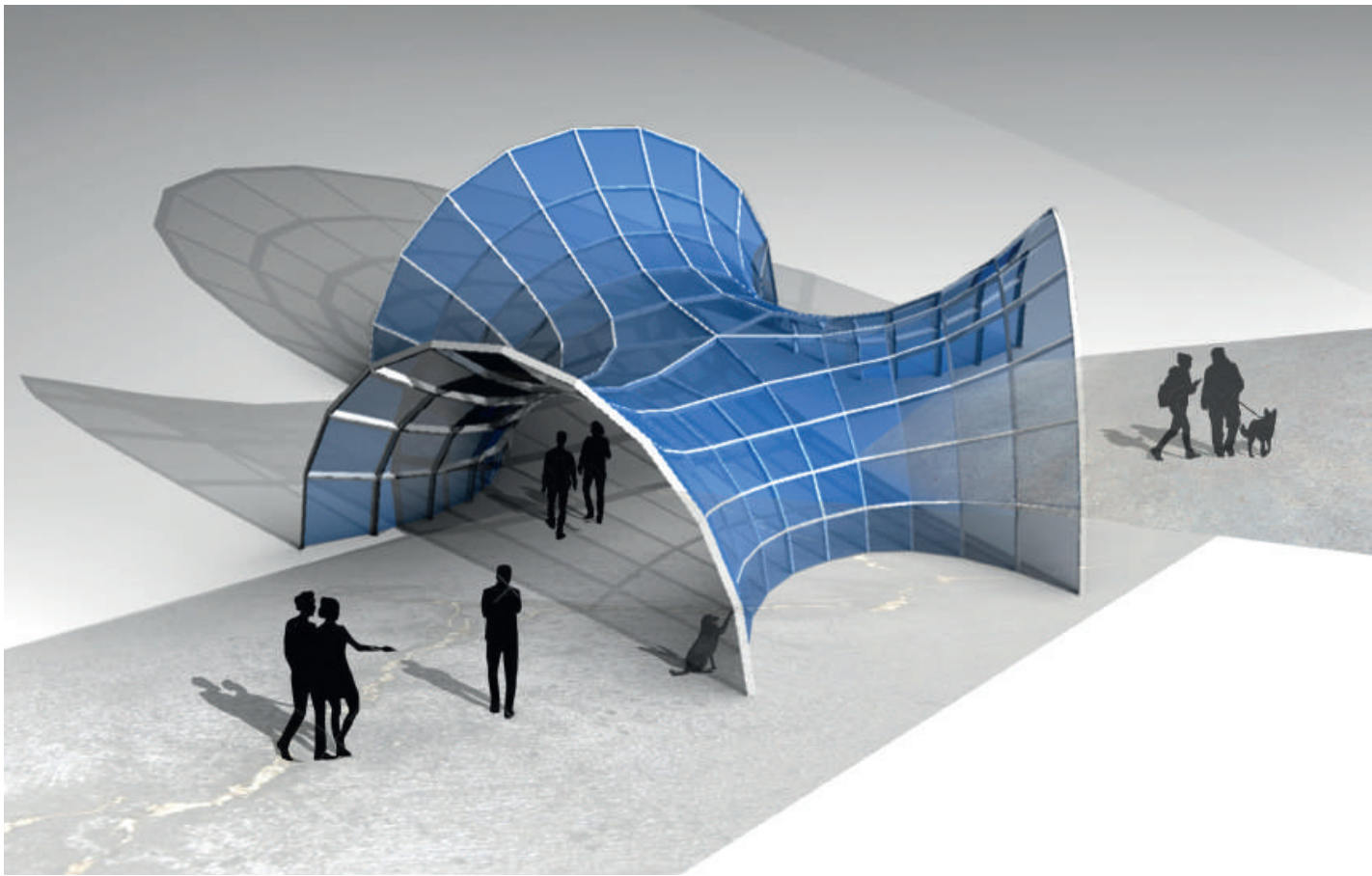


Figure 1: A steel-glass gridshell with geometry based on an S-CMC trinoid.

1. Introduction

1.1 Constant mean curvature surfaces for architecture

CMCs are defined mathematically as surfaces whose mean curvature is constant. The mean curvature of a surface at a given point is the average of the maximum and the minimum principal curvatures. Some CMCs can be easily created: any soap film or bubble in static equilibrium takes the shape of a CMC. However, the family of CMCs also contains surfaces that could theoretically take the form of a bubble, but that are too unstable to exist. CMCs have other unique properties, including the fact that they solve the Plateau problem: CMCs are the surfaces with minimal area fitting a given boundary and englobing a given volume.

CMCs are particularly interesting for the design of building envelopes for the following reasons:

» They can be fitted on any boundary. This property is interesting for applications such as covering courtyards.

» They are aesthetically pleasing, as they take the harmonious shape of an inflated soap bubble.

» Rogers and Schief (2003) showed that under normal pressure, principal stress directions in CMC membranes are aligned with directions of curvature. Curvature directions are preferred directions to lay beams in a gridshell: they minimize panel curvature and node torsion, and also have offset properties. Therefore, on CMCs, curvature lines combine mechanical performance with fabrication advantages.

Minimal surfaces are the most well-known CMCs. They are a special subclass of CMC surfaces for which the mean curvature is null. They can be easily generated with a physical model (e.g. a soap film), or a numerical model (the input then being a boundary curve). However, because of their null mean curvature and due to the estimate of curvature for a stable minimal disk (Schoen 1983), they tend to be flat at their center. They thus require a boundary with a high variation of height in order to be interesting aesthetically, mechanically, and functionally. Allowing the mean curvature to be different from zero significantly broadens the spectrum of possible shapes: minimal surfaces can be “inflated” – as can be seen in Figure 2.

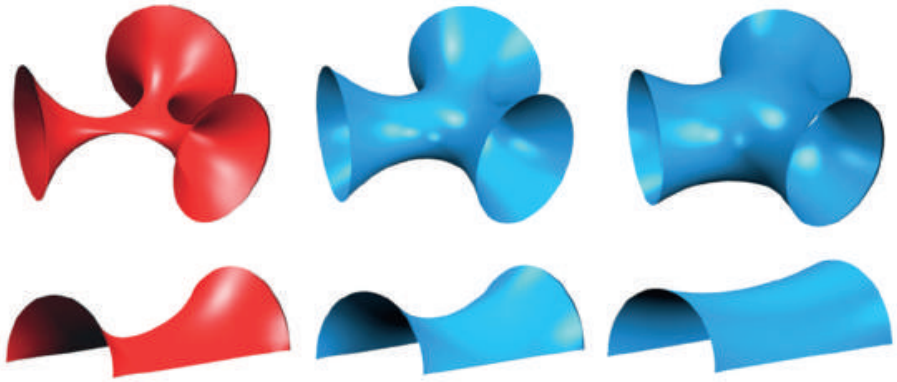


Figure 2: Comparison of minimal surfaces (left) and non-minimal CMCs (middle and right) with the same boundary. Pictures generated with Kangaroo2.

In architecture, CMCs have been used frequently in the work of Frei Otto. The most famous example is the Munich Olympic stadium, whose cable net describe a minimal surface. Other examples include membrane envelopes and inflatable structures, such as the Unite Pneu or the Airhall of Expo64. Despite the interest for smooth CMCs, the potential of discrete CMCs for building envelopes has not yet been exploited.

1.2 Related work

We will first briefly review previous work on the physical form exploration of CMCs. We will then review literature on discrete CMC surfaces relevant for the current paper. Amongst this literature, two approaches are of interest for this study: methods enabling generation of a CMC meshes on a given boundary, and one discretization of the notion of CMC, called S-CMC, which offers interesting properties for gridshell fabrication.

Form potential of CMCs

The shape of a soap film in static equilibrium is a CMC surface. This is due to the fact that a soap film has no bending stiffness and its membrane tension is uniform and isotropic. The mean curvature of a film is directly proportional to the difference of pressure between the two sides of the film.

Bach, Burkhard and Otto (1988) performed a vast exploration program of the shape potential of soap films at the IL in Stuttgart. They tested several types of film support: frames, ropes, friction-free surfaces, and even other soap films. Each type of support has a different flow of forces and yield different forms. They also explored the effect of

difference of pressure between the two sides of a film. Their work revealed the ability of CMCs to fit boundaries with holes and thus assume complex topologies. Trying to fit the same boundaries with traditional methods such as NURBS surfaces would be highly tedious. Inspired by their work, Figure 3 shows a soap bubble whose boundary is a model of the British Museum atrium.

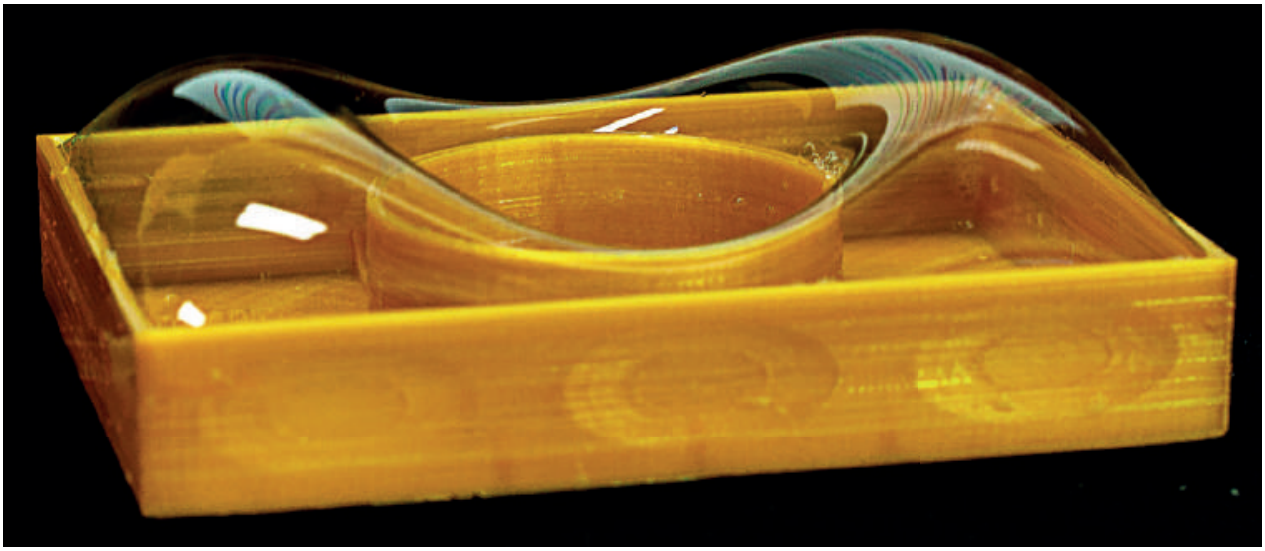


Figure 3: A soap bubble on a boundary similar to the one of the atrium of the British Museum.

Generation of triangular CMC mesh by searching critical points of a functional

Many methods have been developed to generate a triangular mesh with minimal area under a volume constraint. One well-known software implementing such a method is Surface Evolver, developed by Brakke (1992). Oberknappl and Polhier (1999) generate minimal surfaces in $\mathbf{S}^3$ by minimizing an area functional. They then transform them into CMCs in $\mathbf{R}^3$ using the Lawson correspondence, which has been recently generalized in the discrete case by Bobenko and Romon (2017). In order to improve the robustness of CMC mesh generation, Pan et al (2012) propose to look for critical points of an energy based on a Centroidal Voronoi Tessellation rather than minimizing the area. For designers, one of the most accessible tools to generate CMCs is the plugin Kangaroo2 for Grasshopper, which is based on the algorithm developed by Bouaziz *et al* (2012) to handle various geometric constraints.

S-CMC meshes

Smooth CMCs have the property of being parametrized along curvature lines by isothermic coordinates. Bobenko and Hoffmann (2016) propose a discretization of this property with S-isothermic meshes. A subclass of this family (referred to as type 1), have the particularity of having an inscribed circle in each face, and sphere associated with each summit – two spheres being tangent if the corresponding nodes share an edge. Bobenko, Hoffmann and Springborn (2006) developed a theory of minimal S-isothermic meshes based on this structure. Numerous discrete minimal surfaces were then constructed by Bücking (2007).

Hertrich-Jeromin and Pedit (1996) show that smooth CMCs are characterized by the fact that their Christoffel dual is also a Darboux transform of the surface. Hoffmann (2010) proposes a discretization of this property for S-isothermic meshes of type 1. Meshes fulfilling this property are called S-CMCs.

S-CMC surfaces have geometric properties which are of particular interest for fabrication purposes. Firstly, they are quad meshes with planar faces and torsion-free nodes. This property significantly eases the fabrication of a structure such as a gridshell. Secondly, they admit an offset in which some edges are located at constant distance h1 from the mesh, and the other edges are located at a distance h2. This property enables a perfect alignment of the beams at the node while using only two different beam cross sections, as illustrated in Figure 4. We will use the term orthotropic edge offsets to refer to this kind of offset. Thirdly, each face has an inscribed circle. As a result, faces are “roughly square”, which provides aesthetic value to the mesh, and also minimizes material loss if panels are cut out of a larger sheet. Finally, S-CMC meshes have interesting mechanical properties. They are close to a smooth CMC, which is funicular under a uniform pressure loading. Furthermore, since the mesh approximates the curvature lines of the smooth CMC, the orientation of the edges is optimized for beams to resist such a load (Rogers and Schief, 2003).

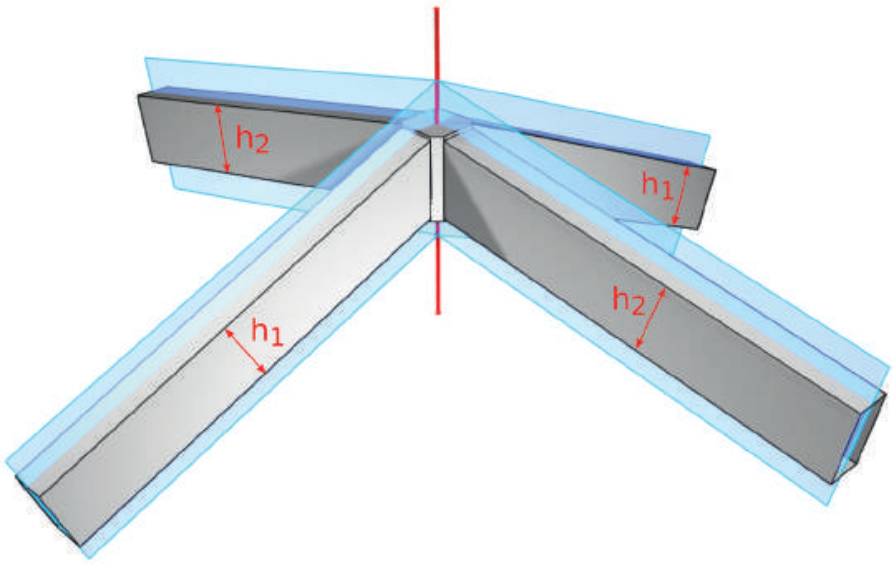


Figure 4: A torsion free node in an orthotropic edge offset mesh.

1.3 Contribution and overview

In this paper, we propose a method to generate quadrangular S-CMC meshes and a structure that allows a change of curvature sign. In Section 2, we present an overview of the method. Section 3 describes how smooth or finely triangulated CMCs can be generated. Section 4 explains how the Gauss map of the smooth CMC can be discretized. The construction of a discrete S-CMC surface from this Gauss map is detailed in Section 5. In Section 6, we explain how the work presented in Sections 3 and 4 must be modified in areas with a change of curvature sign. Finally, in Section 7, we give some examples of S-CMC surfaces and discuss the use of the method in practice.

2. Overview of the method

The workflow is similar to the one used by Bobenko, Hoffmann and Springborn (2006) to generate minimal meshes. The process consists of four steps, which are shown in Figure 5.

In the first step, a CMC surface – smooth or triangulated – is generated. An isothermic network of curvature lines is generated. In the second step, the Gauss map of the surface is calculated. The boundary

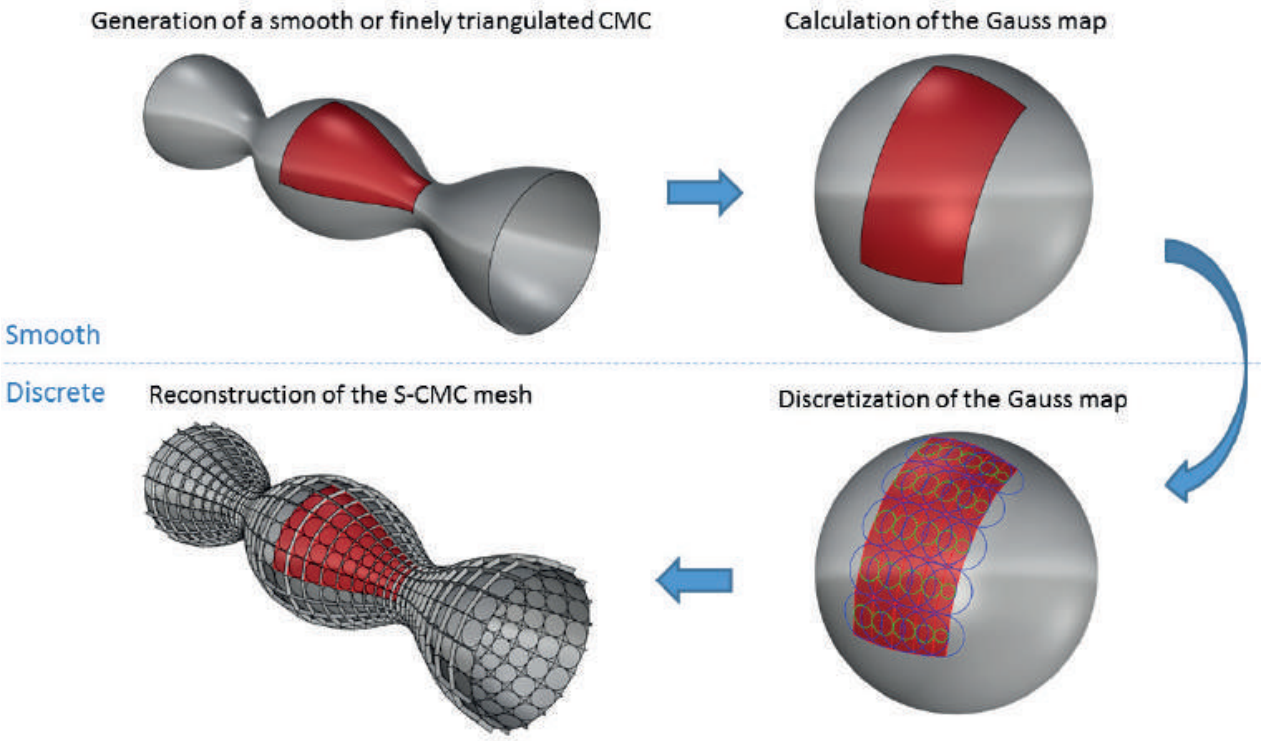


Figure 5: Overview of the discretization method.

of the Gauss map and the topology of the curvature lines are used to generate a discrete Gauss map in the third step. Finally, in the fourth step, the Gauss map is transformed into an S-CMC mesh by a parallel transformation.

3. Generation of input smooth CMCs

In this section, we shall present how we generate smooth CMC surfaces for use as an input in our algorithm.

CMC generation

For the first step of our process, smooth or finely triangulated CMCs are generated. The former option is used when an analytical equations is known for the surface. An example is the unduloid, shown in Figure 5. When no analytical equation is available, a CMC triangular mesh is generated by using the functions “SoapFilm” and “Volume” of the software Kangaroo2. CMC surfaces shown in Figure 2 are generated by this method.

Isothermic orthogonal net

An isothermic network of curvature lines is then drawn on the surface. The isothermic property means that each face is "square", this is necessary for the net to be approximated by an S-CMC mesh. This part is performed in the CAD software Rhino. For smooth surfaces, a code was developed for this purpose using the geometry functions of RhinoScript-Syntax. For triangulated surfaces obtained by Kangaroo2, a network of curvature lines is drawn using the software EvoluteTools T.MAP. Singularities of the network shall be located on the umbilical points of the surface. The order of these umbilical points is a multiple of ½ (Gutierrez and Sotomayor 1986), so the singularities have an even valence: singularities of valence 3, 5 and 7 are not possible since they correspond to umbilics of order ¼, -¼, and -¾ respectively. There are exceptions of course if a singularity is located on the mesh boundary.

Gauss map

The Gauss map of the surface is then computed. For analytical surfaces, the exact normal is computed. For triangulated surfaces, the direction of the normal at a given vertex is computed as the gradient of the area of the adjacent faces.

4. Discretization of the Gauss map

The discretization of the Gauss map is done by generating an orthogonal double circle packing (ODCP) on the unit sphere with a boundary close to the one of the smooth Gauss map. The geometric structure of ODCP is explained in Section 4.1 and the generation method in Section 4.2. The transformation of the ODCP into a discrete Gauss map is described in Section 4.3. The rich structure of this discrete CMC Gauss map – which allows generation by an ODCP – was developed by Hoffmann (2010).

4.1 Orthogonal double circle packings

An *orthogonal double-circle packing* (ODCP) in the plane consists of pairs of circles, where two circles of a given pair are concentric. Such a structure is shown in Figure 6. The packing can be decomposed into two families, represented by red and blue. Having in mind the construction of the Gauss map, one family will be called the *node-centere*d circles (in

red), and the other one the *inscribed* circles (in blue). For each family, the smaller circles are tangent in one direction, and the larger circles in the other one.

When a pair of circles from each family intersect, they fulfill the following rule: the smaller circle of one pair intersects orthogonally the larger circle of the other pair. This property is shown in Figure 7. Thanks to this rule, a quad mesh can be drawn between the node-centered circles, and the inscribed circles (in blue) are then tangent to the edges of this mesh.

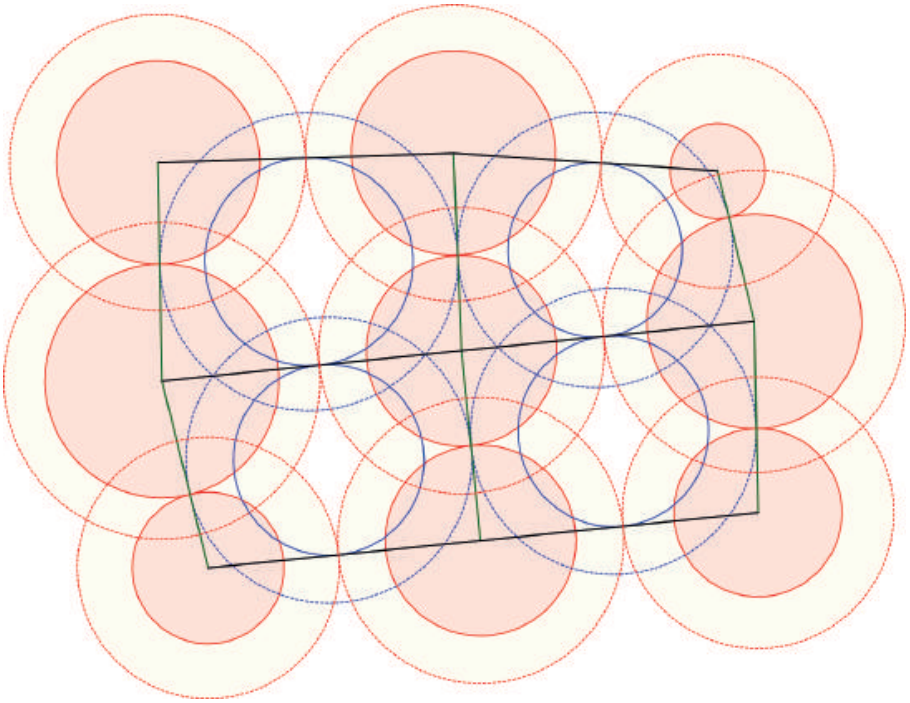


Figure 6: Orthogonal double-circle packing (ODCP).

In order to generate the Gauss map, ODCP will be generated on S^2, the unit sphere. The rules described above are applied in the same way as in the plane, except that straight lines are replaced by arcs of great circles.

4.2 Generation of an ODCP

In this section, we show how an ODCP can be generated with given combinatorics and boundary angles. In a first step, radii of circles compatible with the ODCP structure and the boundary conditions are found using a Newton algorithm. The compatibility of the circles can be expressed by two sets of constraints. In a second step, the ODCP is constructed using the radii and the orthogonal properties.

First Constraint on radii: orthogonal intersection

The orthogonality condition between two secant pairs of circles yields one constraint per pair of circles. As shown in Figure 7, let r_0 and R_0 be the spherical radii of one pair of circles, and r_1 and R_1 the radii of the second one. The geodesic distance d between the centers of the two pairs can be calculated by the spherical cosine rule:

$$\cos(d) = \cos(r_0)\cos(R_1)$$
$$\cos(d) = \cos(R_0)\cos(r_1)$$

Assuming that all circles have a radius lower than $\pi/2$, and thus a non-null cosine, we obtain the following relation:

$$\frac{\cos(r_0)}{\cos(R_0)} = \frac{\cos(r_1)}{\cos(R_1)}$$

Since this relation must hold for all intersecting pairs of circles, the cosine ratio must be identical for all pairs of circles:

$$\frac{\cos(r)}{\cos(R)} = t = cte \geq 1 \qquad (1)$$

The constant t will play an important role in the structure of the offset, as will be shown in Section 4.3.

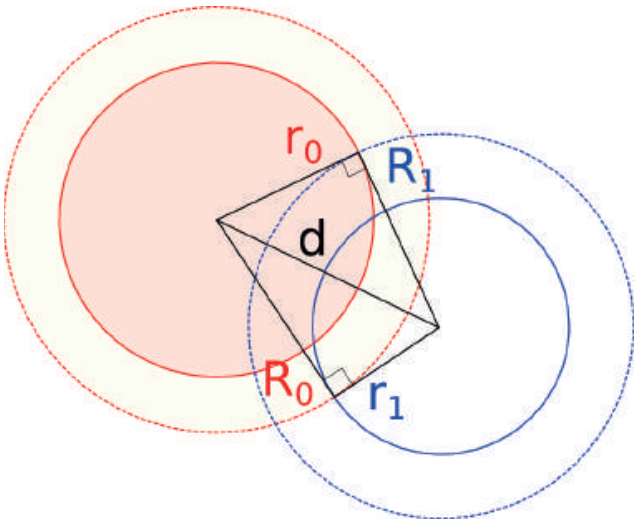


Figure 7: Two secant pairs of circle of an ODCP.

Second constraint on radii: closure of mesh faces

The second set of constraints concerns how all the neighboring circles of a given circle close around it.

Bobenko et al (2006) showed that the Napier formula for a right spherical triangle can be expressed as follows:

$$\varphi = \arctan(e^{\gamma_2-\gamma_1}) + \arctan(e^{\gamma_2+\gamma_1})$$

in which $\gamma_i = \ln(\tan\frac{r_i}{2})$ and r_1, r_2 and φ are shown in Figure 8 :

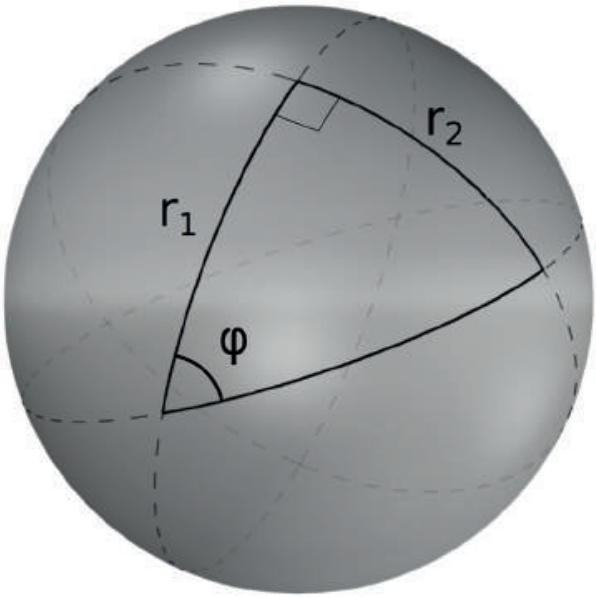


Figure 8: Napier rule for a right spherical triangle.

Since circles intersect orthogonally, the Napier formula can be used to compute all the angles centered at a point M, as shown in Figure 9:

$$\varphi_k = \arctan(e^{\Gamma_k-\gamma_i}) + \arctan(e^{\Gamma_k+\gamma_i})$$
$$\psi_k = \arctan(e^{\gamma_k-\Gamma_i}) + \arctan(e^{\gamma_k+\Gamma_i})$$

Where

$$\gamma_i = \ln\left(tan\tfrac{r_i}{2}\right) \;;\; \Gamma_i = \ln\left(tan\tfrac{R_i}{2}\right) \;;\; \varphi_k = \widehat{A_kMP_k} \text{ and } \psi_k = \widehat{P_kMB_k}$$

For a pair of circles not located on the boundary, the angles must add up to 2π:

$$\sum_{k=1}^{n}(\varphi_k + \psi_k) = 2\pi \qquad (2a)$$

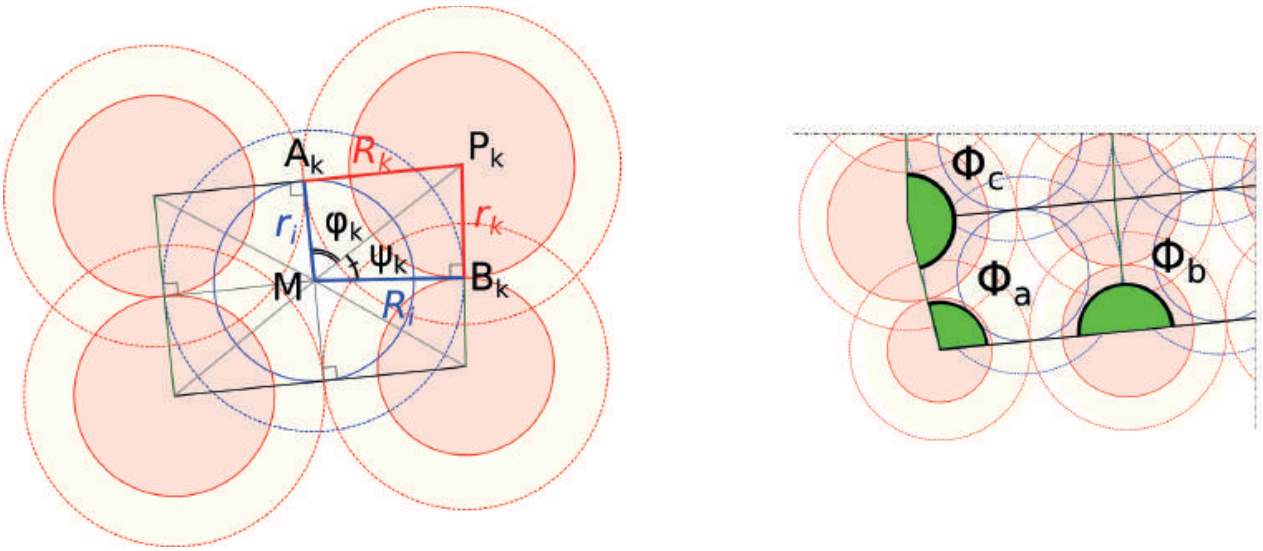


Figure 9: Angles around the center of a circle (left) and boundary angles Φ (right).

For pairs located on the boundary, the sum of the angles around a point is a boundary angle Φ that needs to be calculated from the smooth Gauss map, as shown in Figure 9:

$$\sum_{k=1}^{n} (\varphi_k + \psi_k) = \Phi \qquad (2b)$$

Calculation of the radii

The system of nonlinear equations determined in the two previous sub-sections is square: the number of equations is the same as the number of unknowns. Since the equations are analytical, the Jacobian matrix of the system can be calculated exactly. Radii fulfilling all the constraints are searched using the Newton-Raphson method. Note that the value of the radii need to be higher than 0 and lower than ϖ. This constraint is automatically fulfilled using the logarithmic radii as variables. The following initial spherical radii were used for the pictures shown in this paper: *0.24 rad* for the larger circles of each pair, and *0.15 rad* for the smaller. This algorithm converges fairly quickly. Eight iterations are sufficient to generate the trinoid shown in Figure 1.

Construction of the ODCP

These two sets of constraints are sufficient for radii to be compatible with a simply connected ODCP structure. The ODCP is built from the circles as follow:

» First, pairs of circles are placed on two edges of the boundaries of the packing. Only the circle radii and the boundary angles are needed for this purpose.

» The remaining circles are added by propagation from the edges using the orthogonality property and the radii.

4.3 Construction of the discrete Gauss map

The construction of the Gauss map starts with the construction of the circular cones which are tangent to S^2 along the larger node-centered circles of the ODCP. Such cones are shown on the right side of Figure 10.

Prop 1

The apexes of these cones are the vertices of a polyhedral mesh with planar faces and orthotropic edge offset property, i.e. each edge is tangent to either S^2 or tS^2 (a sphere or radius t concentric with S^2).

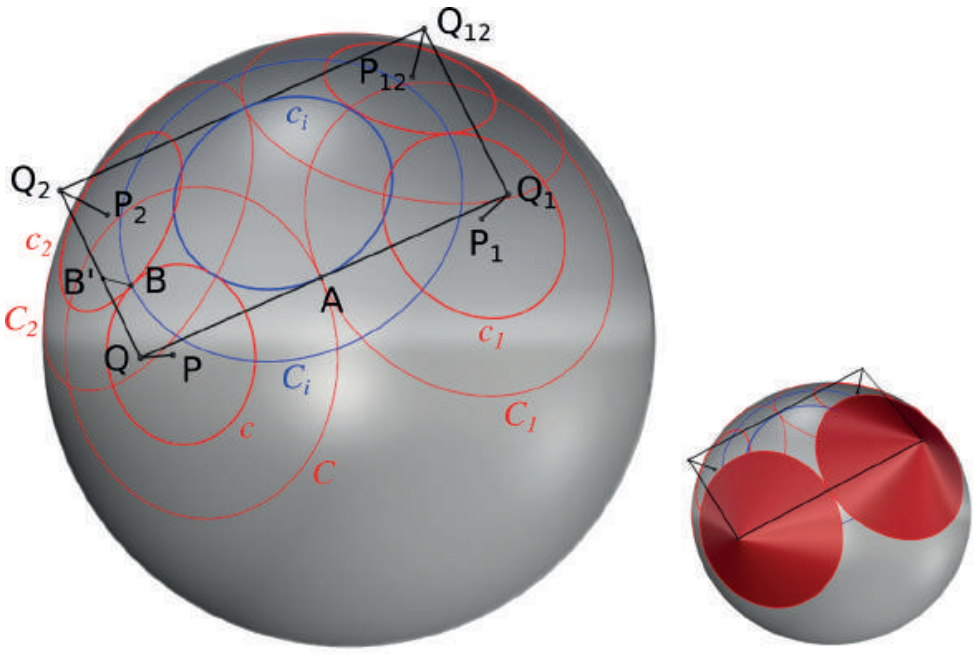


Figure 10: Construction of the Gauss map from the ODCP. Cones used to build the mesh are shown on the right.

Proof:

Starting from an object X of the ODCP, we will call X_1 (resp. X_2) the next object in the direction of higher (resp. lower) curvature – i.e. the direction in which the larger (resp. smaller) circles are tangent.

Let us call (see Figure 10):

» P the center of a node-centered circle of the ODCP ($P \in S^2$) ;

» Q the cone apex corresponding to P ;

» O the center of S^2 ;

» C and c the node-centered circles centered at P, whose spherical radii are respectively R and r (spherical radii are angles in S^2, see Figure 11);

» C_i and c_i the inscribed circles of the spherical face $PP_1P_{12}P_2$.

Q, Q_1 and A are aligned, because the cones centered on P and P_1 are tangent to S^2 at A, and A belongs to the plane OPP_1. Since (QQ_1) and c_i are incident (at A) and since (QQ_1) is tangent to S^2 at A, (QQ_1) and c_i are necessarily coplanar. The same argument can be used to show that (Q_2Q_{12}) and c_i are coplanar. Therefore the quad $QQ_1Q_{12}Q_2$ is planar.

Let us now build the circle c', which is the projection of c onto tS^2, and then build the cones tangent to tS^2 along c'. Q', the apex of this cone belongs to (OP), and its distance to O is (see Figure 11), using Equation (1):

$$OQ' = OB' / \cos r = t / \cos r = 1 / \cos R = OQ$$

Therefore $Q' = Q$, and we conclude that (QQ_2) and (Q_1Q_{12}) are tangent to tS^2.

5. Reconstruction of the surface from the Gauss map

In this section, we will show how to construct an S-CMC surface from the Gauss map built in Section 4. We start by constructing a double-sphere packing thanks to the underlying ODCP. To each node-centered pair of circles, we associate a pair of spheres centered on the node of the Gauss mesh. Figure 12 shows on the left (resp. right) a section of the double-sphere-packing along the edge of the mesh where the larger (resp. smaller) spheres touch each other:

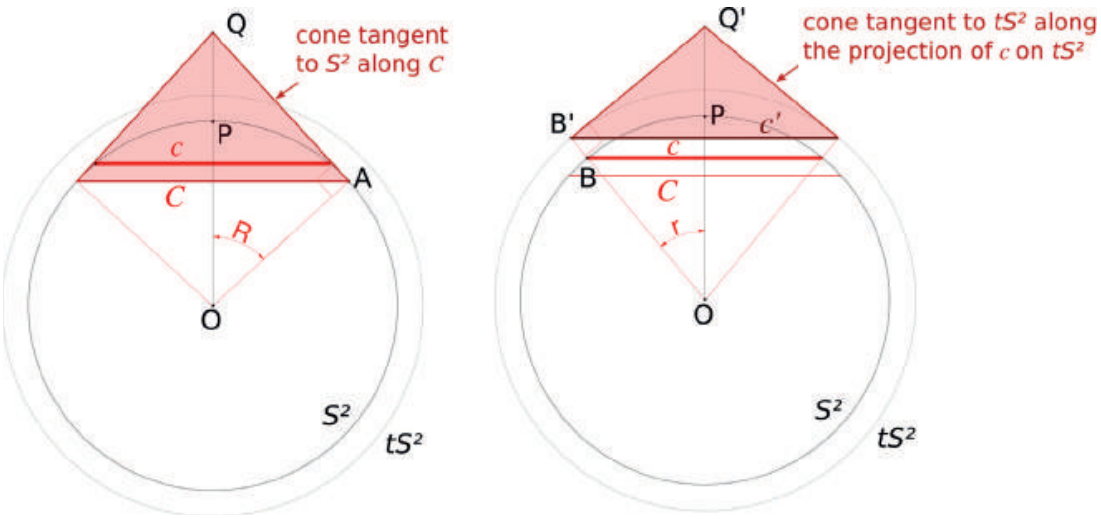


Figure 11: Tangency of edges with tS².

The radii of the larger and the small spheres are given respectively by:

$$R_S = AQ = \tan(R)$$
$$r_s = B'Q = t * \tan(r)$$

Prop 2

Let G be a Gauss map constructed in Section 4. Let *R* and *r* be the radii of the associated double sphere packing. There exist two S-isothermic meshes, M^+ and M^-, which are edgewise parallel to *G*. The radii of the associated spheres are $(R+r)/2$ for M^+, and $(R-r)/2$ for M^-.

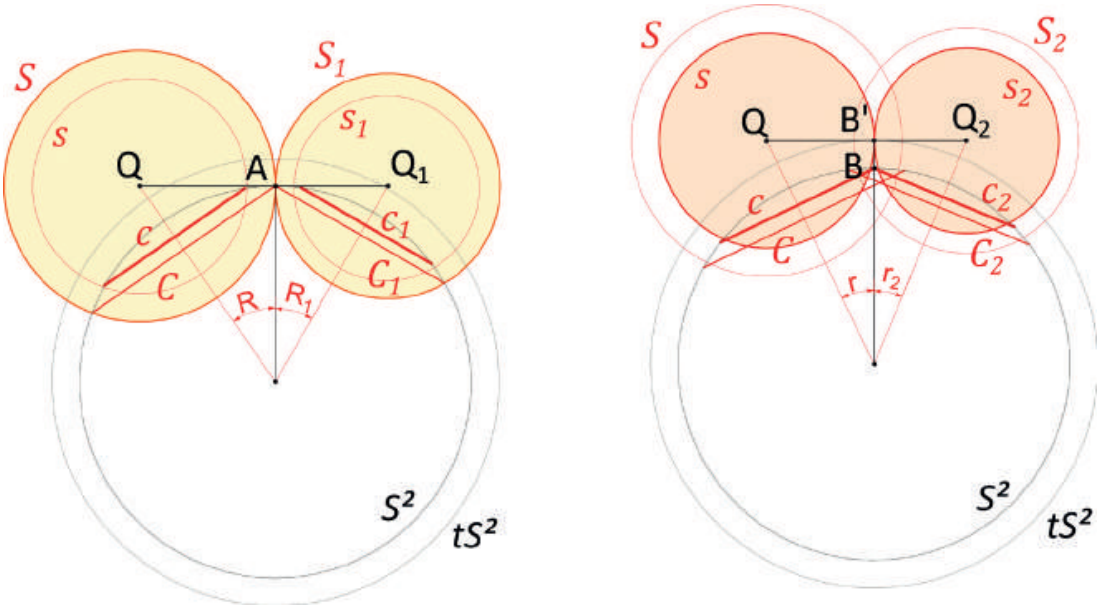


Figure 12: Double-sphere packing associated with the Gauss map.

Proof:

Figure 13 shows a top view of a face of the Gauss map, with the associated double-spheres. Since the face is closed, we have:

$$(R + R_1)u + (r_1 + r_{12})v_1 - (R_{12} + R_2)u_2 - (r_2 + r)v = 0$$

In which $u = \frac{\overrightarrow{QQ_1}}{QQ_1}$ and $v = \frac{\overrightarrow{QQ_2}}{QQ_2}$

As shown in Figure 13, we can obtain a second sphere packing by switching the direction of tangency of the smaller spheres with that of the larger spheres. This switch can be executed by applying a reflection to each colored quad.

Thanks to the fact that each colored quad has two right angles, the flipped Gauss mesh is parallel to the original one. Therefore, we obtain the following equations, which corresponds to the closure of the quad $\widetilde{Q}\widetilde{Q_1}\widetilde{Q_{12}}\widetilde{Q_2}$:

$$(r + r_1)u + (R_1 + R_{12})v_1 - (r_{12} + r_2)u_2 - (R_2 + R)v = 0$$

As a result, spheres of radius $(R+r)/2$ can be packed in directions parallel to the Gauss mesh:

$$\left(\frac{R+r}{2} + \frac{R_1+r_1}{2}\right)u + \left(\frac{R_1+r_1}{2} + \frac{R_{12}+r_{12}}{2}\right)v_1 - \left(\frac{R_{12}+r_{12}}{2} + \frac{R_2+r_2}{2}\right)u_2 - \left(\frac{R_2+r_2}{2} + \frac{R+r}{2}\right)v = 0$$

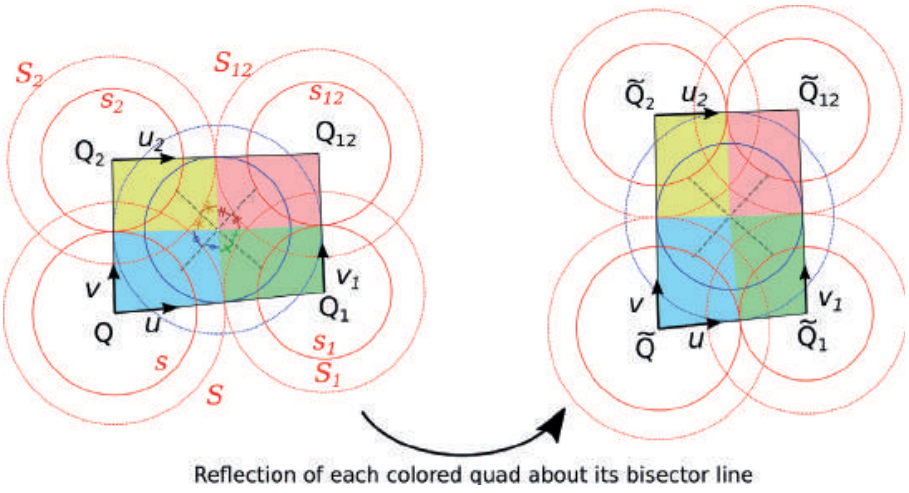


Figure 13: Change of the direction of the packing of double spheres.

This compatibility equation insures that the whole Gauss map can be deformed into an S-isothermic mesh by a Combescure transformation. The edge length modification ratios of this transformation are simply obtained from the sphere radii.

The same result holds for a packing of spheres of radii *(R-r)/2*.

Prop 3

The S-isothermic meshes M^+ and M^- mentioned in Prop 2 are also S-CMC.

Proof:

Note: For sake of conciseness many of the mathematical concepts used in this proof (such as the Christoffel dual) are not introduced. The reader is advised to browse the paper by Hoffmann (2010) beforehand.

S-CMC meshes are defined as S-isothermic meshes for which the Christoffel dual mesh is also a Darboux transform of the mesh. We start by constructing the mesh $M^* = M^+ + n$ where n is the Gauss map and "+" is the sum on vertices. We call C_i the vertices of M^+, C_i^* those of M^*, and A_i the points of tangency of the spheres of M^+.

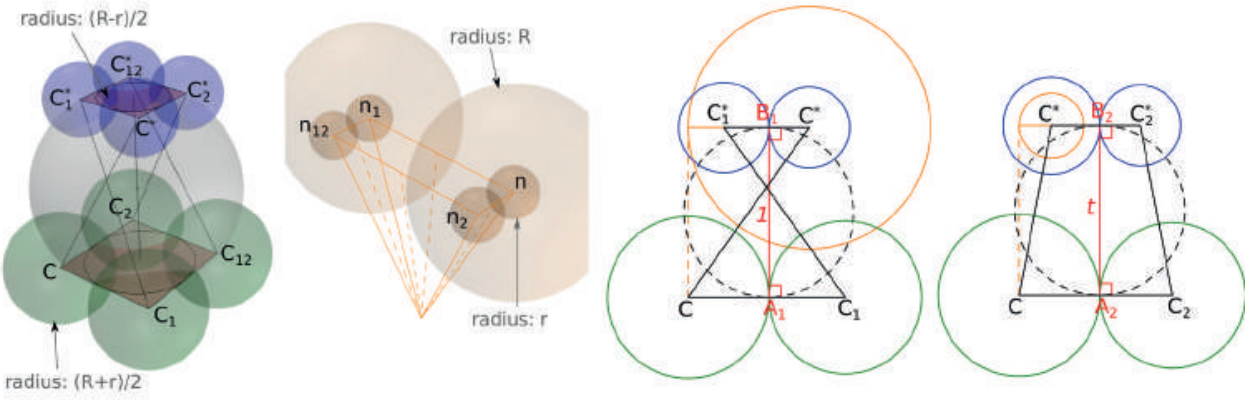


Figure 14: Construction of the Christoffel dual. From left to right: 3D view of meshes, 3D view of Gauss map (larger spheres centered on n_2 and n_{12} are hidden for clarity), section in higher curvature direction, section in lower curvature direction.

Figure 14 shows the construction in the planes (CC_1C^*) and (CC^2C^*). In each of these planes, we draw a line perpendicular to (CC_i) going through A_i. We call B_i the intersection of this line with $(C^*C_i^*)$. Since edges of the Gauss map n are tangent to S^2 and tS^2, $A_1B_1 = 1$ and $A_2B_2 = t$.

We note that:

$$C^*B_1 = R - \frac{R+r}{2} = \frac{R-r}{2}$$

$$C^*B_2 = \frac{R+r}{2} - r = \frac{R-r}{2} = C^*B_1$$

Therefore, we can construct a packing of tangent spheres of radii $\frac{R-r}{2}$ centered on vertices of M^*. Since M^* is parallel to M^+, it is also parallel to M^-. As a result, M^* corresponds to the mesh M^-.
The product of the radii of corresponding spheres of M^+ and M^* is:

$$\frac{R+r}{2}\frac{R-r}{2} = \frac{R^2-r^2}{4} = \frac{t^2-1^2}{4} = cte \qquad (3)$$

where we use the fact that:

$$CC^{*2} = R^2 + 1 = r^2 + t^2$$

Therefore M^* is the Christoffel dual of M^+.
The circles inscribed in the quads $CC_1C_{12}C_2$ et $C^*C_1{}^*C_{12}{}^*C_2{}^*$ are coaxial. The sphere containing these two circles is orthogonal with the eight spheres centered on each vertex. Therefore, M^* is a Darboux transform of M^+. We can then conclude that M^+ is S-CMC.

6. Change of curvature

The junctions between zones of positive and negative curvature require a specific treatment. At such a location, the Gauss map of the surface is "folded". This section describes how the discrete Gauss map can be folded while keeping the geometric properties described in the previous sections.

6.1 Structure of the Gauss map on a fold

In the model presented in this paper, the curvature is defined on the nodes of the mesh: if a node has a positive (resp. negative) curvature, the associated sphere in the S-CMC mesh has a radius of *(R+r)/2* (resp. *(R-r)/2*). In the cases treated in the previous sections, each face had four nodes with the same curvature sign. As a result, all the circles

of the ODCP (and consequently all the spheres of the sphere packing) were tangent on the outside. When a change of curvature occurs, two adjacent smaller circles touch each other on the inside, as shown in Figure 15 :

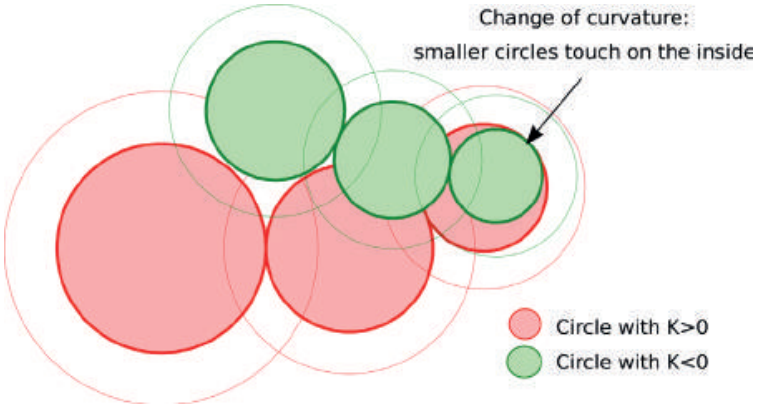


Figure 15: Change of curvature sign in a line of double-circles of an ODCP of a Gauss mesh.

Quads of the Gauss map with nodes of different curvature signs can be classified in the following types, as represented in Figure 16:

» Faces of type A: two nodes have positive curvature, and the two others have negative curvature. The change of curvature occurs when traveling in the direction of low curvature (the direction in which smaller circles are tangent).

» Faces of type B: same as type A, except that the change of curvature occurs when traveling in the direction of higher curvature (the direction in which larger circles are tangent). In that particular case, the inside tangency shown in Figure 15 does not apply.

» Faces of type C: this type is only encountered in highly coarse meshes and will not be treated here.

» Faces of type D: one node has a curvature sign different from the other three.

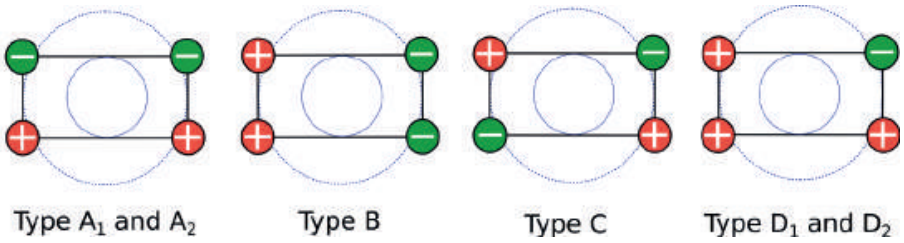


Figure 16: Types of quads with non-uniform node curvature signs.

The full tangency pattern for each type of face is shown in Figure 17. For faces of type A and D, it can be noted that, depending on relative size of the adjacent circles, the quad can auto-intersect. Faces of type B always auto-intersect, in the way of a candy wrapping paper. For faces A and D, the tangency of quad edges with tS^2 happens outside of the quad. The types of fold of a quad are analogous to how a rectangle of fabric can be folded, as shown in Figure 18.

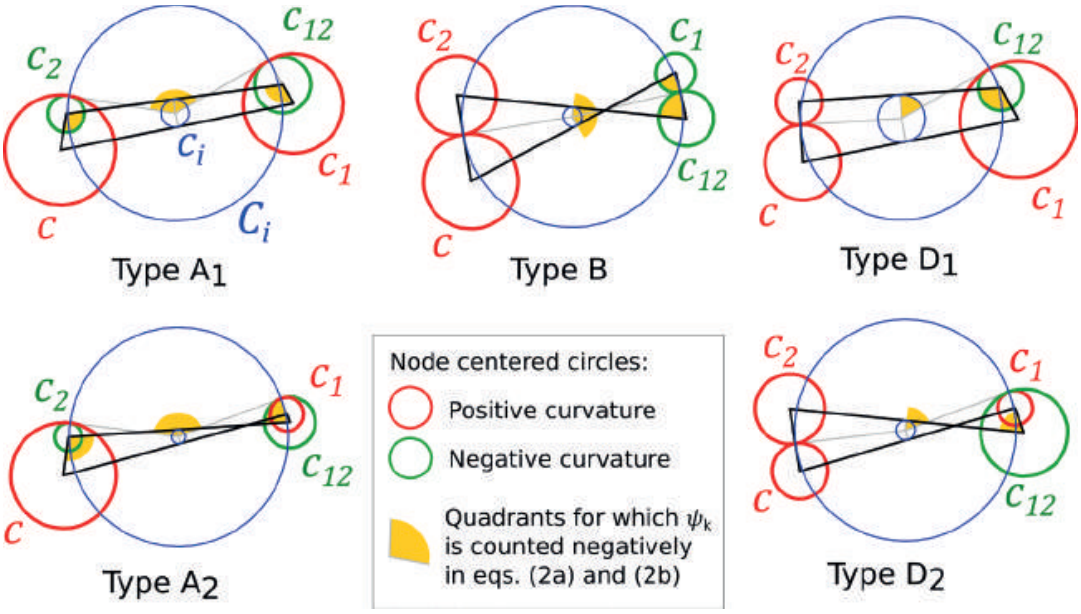


Figure 17: Tangency of circles for the five types of face with a change of curvature sign (larger node centered circles not shown for clarity).

Figure 18: An illustration of the five types of fold with a piece of fabric.

6.2 Reconstruction of the surface

Prop 4

Each of the five proposed Gauss map folds can yield a transition part between synclastic and anticlastic portions of a mesh that conserves the S-CMC property.

Proof:

For sake of conciseness, we will only prove the result for faces of type A1. Looking at one face $QQ_1Q_{12}Q_2$ on Figure 19, we notice that we can pack spheres of radius $(R_i+r_i)/2$ at Q and Q_1 and $(R_i-r_i)/2$ at Q_2 and Q_{12} to form a quad with an inscribed circle:

$$(R+R_1)u+(r_1-r_{12})v_1-(R_{12}+R_2)u_2-(r-r_2)v=0$$
$$(r+r_1)u+(R_1+R_{12})v_1+(r_{12}+r_2)u_2-(R+R_2)v=0$$
$$\Rightarrow \left(\frac{R+r}{2}+\frac{R_1+r_1}{2}\right)u+\left(\frac{R_1+r_1}{2}+\frac{R_{12}-r_{12}}{2}\right)v_1-\left(\frac{R_{12}-r_{12}}{2}+\frac{R_2-r_2}{2}\right)u_2$$
$$-\left(\frac{R_2-r_2}{2}+\frac{R+r}{2}\right)v=0$$

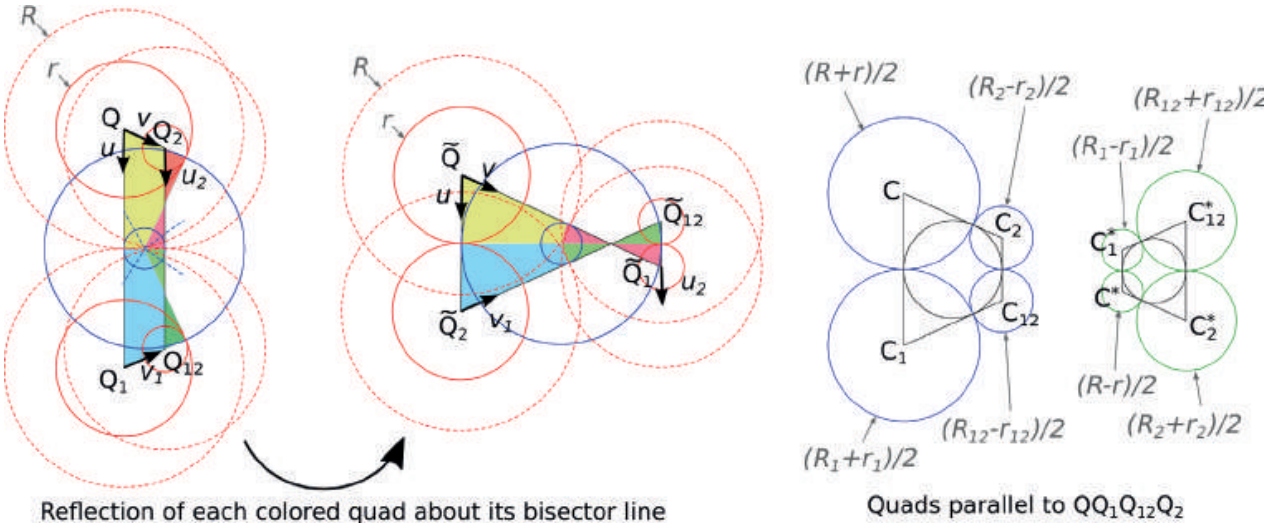


Figure 19: Construction of faces with inscribed circle from a Gauss face of type A1 (circle radii are indicated in grey with an arrow).

Note that the same result can be achieved with spheres of radii $(R_i-r_i)/2$ at Q and Q_1 and $(R_i+r_i)/2$ at Q_2 and Q_{12}:

$$-\left(\frac{R-r}{2}+\frac{R_1-r_1}{2}\right)u+\left(\frac{R_1-r_1}{2}+\frac{R_{12}+r_{12}}{2}\right)v_1+\left(\frac{R_{12}+r_{12}}{2}+\frac{R_2+r_2}{2}\right)u_2$$
$$-\left(\frac{R_2+r_2}{2}+\frac{R-r}{2}\right)v=0$$

If we look at a strip of quads (i.e. a mesh with only one row) of type A_1, we can now obtain a strip of S-isothermic mesh Str^+. Vertices can be assigned a sphere or radius $(R_i+r_i)/2$ on side of the strip and $(R_i-r_i)/2$ on the other side.

If we now look at $Str^* = Str^+ + n$, the same reasoning as in the proof of prop 3 shows that Str^* is the dual and a Darboux transform of Str^+. Therefore, Str^+ is S-CMC. It can thus connect an S-CMC mesh with spheres of radii $(R+r)/2$ (synclastic) to an S-CMC mesh with radii $(R-r)/2$ (anticlastic). Figure 20 shows the connection of the face A_1 with adjacent faces.

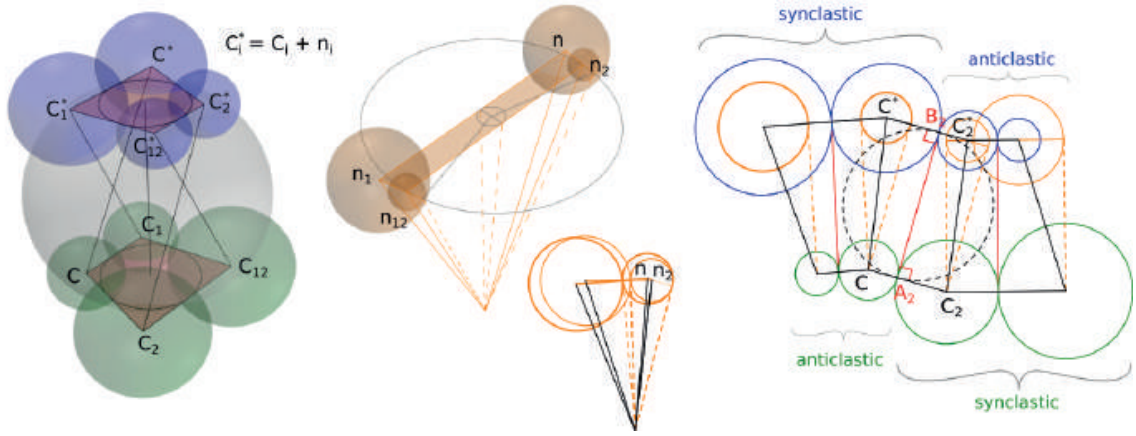


Figure 20: Arrangement of spheres at a face of type A^1. Left: The face and its dual form a Darboux pair; Middle: Corresponding Gauss map in 3D and in side view including adjacent faces; Right: Side view of M and M*.

It can be noted that equations (2a) and (2b) need to be modified on the fold of the Gauss map: the angles ψ_k shall be counted negatively at locations shown on Figure 17. The type of quad is thus a necessary input of the algorithm. The convergence is much less robust when there is a change of curvature.

7. Applications and discussion

Examples

The unduloid is a periodic cylindrical CMC. Although a discrete S-CMC unduloid can be generated rather simply by a so-called elliptic billiard, as explained in Hoffmann (2010), the unduloid shown in Figure 5 was generated with our framework, using as input the analytical equations of the smooth unduloid.

Figure 1 shows an S-CMC version of the trinoid, another well-known CMC surface. Singularities, such as the valence-6 node at the center, can be efficiently handled by the method. The constant t for this mesh is 1.004. Therefore, the edge offset in the higher curvature direction is only 0.4% lower than in the lower curvature direction. This fact is particularly interesting considering one major limitation of the edge offset meshes: at locations of a surface where there is a significant difference between the higher and the lower principal curvature, faces are highly elongated. This effect can be observed in some of the work of Pottman *et al* (2007). In the case of this trinoid, we observe that by allowing a slight change between the edge offsets in the two curvature directions, we can obtain faces with an aspect ratio close to one. Furthermore, the difference between the two offsets is low enough to be considered as a regular edge offset for fabrication purposes. Finally, it is important to note that this S-CMC mesh can fulfill the properties (planarity, offset, etc.) with arbitrary precision.

Figure 21 shows an S-CMC mesh with changing curvature sign. The associated sphere packing is shown on the right. The mesh is generated from a portion of 4-noid, and successive reflections yield the full mesh. The eight-valent nodes could be replaced by planar octagons for improved uniformity of panel sizes.

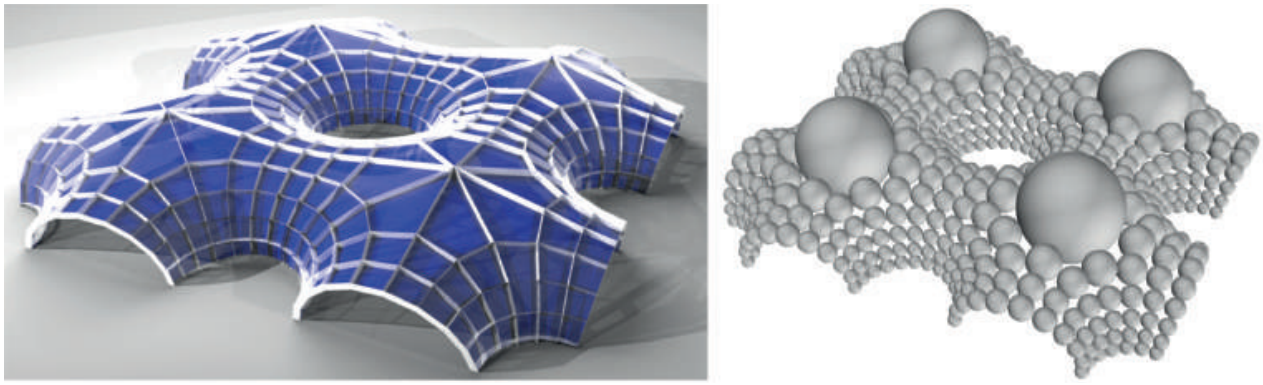


Figure 21: An S-CMC mesh with changing curvature sign.

Figure 22 shows multiple morphologies that can be obtained with a given trinoid combinatorics. The boundaries of the meshes are planar, this simplifies the fabrication of the edge beams. The various shapes are obtained by varying the position and orientation of the boundary planes.

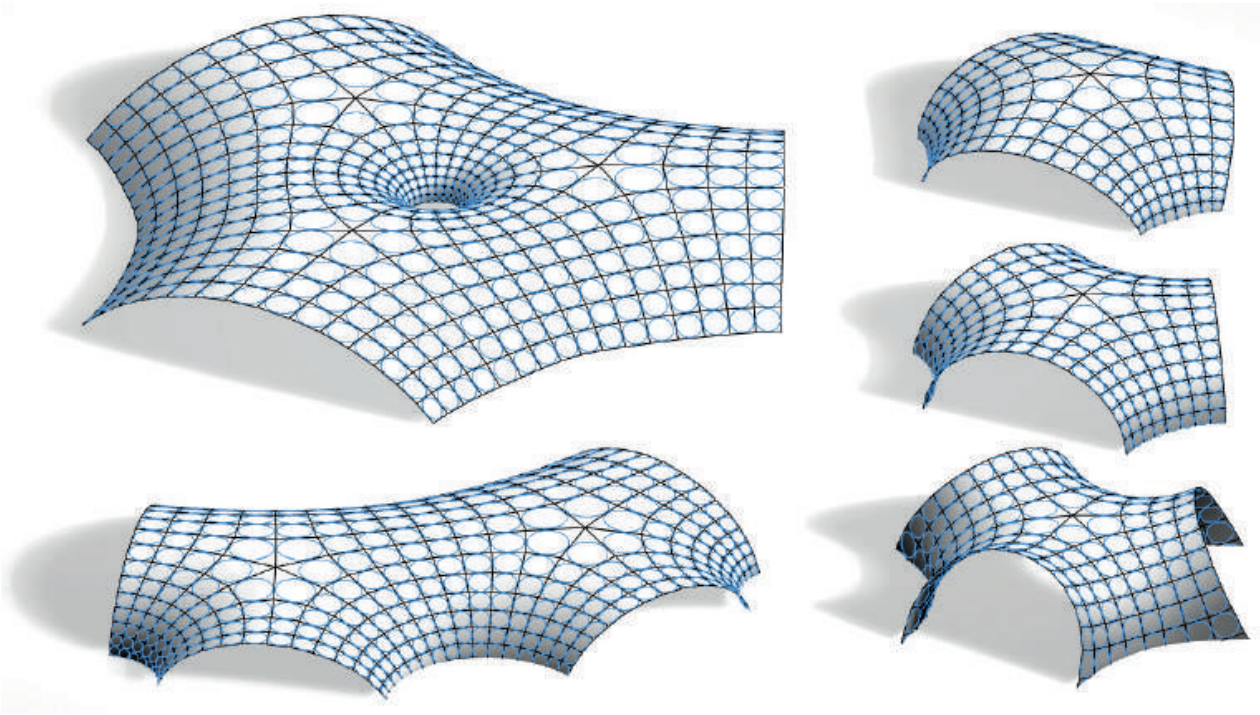


Figure 22: Several S-CMC trinoids. Left: combining three and two trinoids; Right: different ways to inflate a trinoid.

Limitations

The following limitations apply:

» As with other types of meshes with torsion-free nodes, S-CMC meshes can be interpreted as a curvature line network. As such, one cannot choose the orientation of the mesh.

» The final geometry is highly dependent on the combinatorics of the curvature line network. The isothermic condition and the positioning of singularities on umbilics can be difficult to obtain with commercially available software. Furthermore, the network (and therefore the S-CMC mesh) might need significant refinement when some umbilics are located close to each other.

» In the meshes shown in this paper, boundaries are planar curvature lines. For other types of boundaries, the computation of the boundary angles often requires an optimization loop to approximate the desired smooth surface. This aspect is under development and will be detailed in further publications.

» CMC surfaces that are not simply connected (e.g. surfaces with holes) need periodicity constraints on the top of the ones given in Section 4.2 to ensure proper closing.

Comparison with other generation methods

As a final remark, S-CMCs could also be generated by optimizing directly a mesh. Both vertex positions and vertex normals would then need to be optimized simultaneously. This would make the optimization quite more complex than for circular and conical meshes, for which vertex positions are the only variables. An advantage of such a method would be a stronger control of the boundary, allowed by the ability to "relax" the S-CMC property. Comparatively, our method uses less degrees of freedom, fulfills the S-CMC property exactly and fit boundaries in an approximate manner.

8. Conclusion

In this paper, we identified the potential of S-CMC meshes for construction-aware design of free-form architectural envelopes. We proposed a method to generate these meshes by discretizing smooth CMCs. We developed a geometric structure that allows the construction of S-CMCs with changing curvature sign. Finally, we demonstrated the morphological potential of S-CMCs on several examples

Acknowledgment

This work is supported by Labex MMCD (http://mmcd.univ-paris-est.fr/), Labex Bezout (http://bezout.univ-paris-est.fr/) and I-SITE GAMES Impulsion project. We warmly thank Tim Hoffmann for inspiring discussions about S-CMC meshes. We would also like to thank Laurent Monasse and Pierre Margerit for their support on numerical and computational aspects and Siavash Ghabezloo for his help with bubble photography.

References

Bach, K. , Burkhard, B. and Otto, F. 1988. *"IL 18. Forming Bubbles"*. Institute for Lightweight structures, University of Stuttgart.

Bouaziz, S. , Deuss, M. , Schwartzburg, Y. , Weise, T. and Pauly, M. 2012. "Shape-Up : Shaping Discrete Geometry with Projections." *Eurographics Symposium on Geometry Processing* 31. doi:10.1111/j.1467-8659.2012.03171.x.

Brakke, K. (1992). The Surface Evolver. *Experimental mathematics Volume 1*,1992 – Issue2 p141–165.

Bobenko, A. , Hoffmann T. , and Springborn, B. 2006. "Minimal Surfaces from Circle Patterns : Geometry from Combinatorics." *Annals of Mathematics* 164: 231–64.

Bobenko, A. , and Hoffmann, H. 2016. "S-Conical CMC Surfaces. Towards a Unified Theory of Discrete Surfaces with Constant Mean Curvature." In *Advances in Discrete Differential Geometry*, 287–308. doi:10.1007/978-3-662-50447-5.

Bobenko, A. and Romon, P. 2017. "Discrete CMC Surfaces in R 3 and Discrete Minimal Surfaces in S 3 : A Discrete Lawson Correspondence." *Journal of Integrable Systems* 2 (April): 1–18.

Bücking, U. 2007. "Approximation of Conformal Mappings by Circle Patterns and Discrete Minimal Surfaces." *PhD Diss., Technischen Universität Berlin.*

Gutierrez, C, and Sotomayor, J. 1986. "Principal Lines on Surfaces Immersed with Constant Mean Curvature." *Transactions of the American Mathematical Society* 293 (February). doi:10.1090/S0002-9947-1986-0816323-5.

Hertrich-Jeromin, U. and Pedit, F. 1997. "Remarks on the Darboux Transform of Isothermic Surfaces." In *Documenta Mathematica*, 313–333.

Hoffmann, T. 2010. "A Darboux Transformation for Discrete S-Isothermic Surfaces." J*ournal of Math-for-Industry* 2 ((2010B-6)): 157–69.

Oberknappl, B., and Polthier, K. 1997. "An Algorithm for Discrete Constant Mean Curvature Surfaces." *Visualization and Mathematics,* 141–61.

Pan, H. , Choi Y.-K. , Liu, Y. , Hu, W. , Du, Q. , Polthier, K. , Zhang, C. , and Wang, W. 2012. "Robust Modeling of Constant Mean Curvature Surfaces." *ACM Transactions on Graphics (TOG)* 31 (4): 11–85. doi:10.1145/2185520.2185581.

Pottmann, H. , Liu, Y. , Wallner, J. , Bobenko, A., and Wang, W. 2007. "Geometry of Multi-Layer Freeform Structures for Architecture." *ACM Transactions on Graphics* 26 (3): 65. doi:10.1145/1276377.1276458.

Rogers, C., and Schief, W.-K. 2003. "On the Equilibrium of Shell Membranes under Normal Loading. Hidden Integrability." *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 459 (2038): 2449–62. doi:10.1098/rspa.2003.1135.

Schoen, R. 1983. "Estimates for Stable Minimal Surfaces in Three Dimensional Manifolds." In *Seminar on Minimal Submanifolds,* Ann. of Math. Stud. 103, 111–126. Princeton Univ. Press.

Environmental-performance morphology generation: Combining physical wind tunnel and dynamic building model

Yuqiong Lin, Jiawei Yao, Jingyun Zheng, Philip F. Yuan

Yuqiong Lin
1630237@tongji.edu.cn
College of Architecture and Urban Planning, Tongji University, China

Jiawei Yao
jiawei.yao@tongji.edu.cn
College of Architecture and Urban Planning, Tongji University, China

Jingyun Zheng
1530298@tongji.edu.cn
College of Architecture and Urban Planning, Tongji University, China

Philip F. Yuan
philipyuan007@tongji.edu.cn
College of Architecture and Urban Planning, Tongji University, China

Keywords:
Wind tunnel, wind environment visualization, environmental performance, building morphology generation, dynamic mechanical model

Abstract

Performance-responsive buildings are establishing a sustainable architectural future. The proliferation of urban spatial density has led to a series of environmental problems including pedestrian thermal stress and air pollution. To deal with these problems, architects are suggested to understand and employ the natural ventilation. This paper introduces the latest achievements of environmental-performance architectural morphology generation method based on the customized physical wind tunnels visualization, which integrated the dynamic mechanical models. Moreover, “twisting” and “retreat” architectural morphologies are chosen as the movement mode of the mechanical models to conduct some experiments. A set of systematic method of building morphology generation is presented, which is suitable for the early stage of architectural design and guided by wind environment performance.

1. Introduction

With the advent of the digital era, sustainable and environmental design concept has become the important factor of considering built environment performance. The building morphology is changed by novel design method and construction technology. Andrew Marsh has combined Generative Design and Performative Design, while the building’s performance based design steps to the digital age (Marsh 2008). Integration of Digital simulation and morphology generation asks architects to pay attention to the design logic, as performance responsive environment based architectural generation design will promote the development of sustainable buildings.

The dramatic increase in urban spatial density has brought increasingly number of serious problems in terms of building ventilation, air pollution, and pedestrian comfort. To deal with these problems, architects can apply natural ventilation as the main consideration factor in the early stage of building morphology design. With the continuous development of performance-based simulation tools, the “trial and error” design method has gradually been replaced by logically generated design method. At present, the simulation of the wind environment is mainly through computational fluid dynamics (CFD) software and physical wind tunnel. However, the former has the disadvantages of large computatio-

nal burden, long time-consuming and high-level barriers to expertise, as well as the high cost of the latter, and the difficulty of establishing initial models, which both may not provide timely feedback for the optimization of the scheme in the early stages of design. The post-evaluation paradigm no longer satisfies the architect's pursuit of performance-based design for the built environment (Lin et al.2018).

This study is based on the customized physical wind tunnel, dynamic models will be applied in the research of optimized morphology generation system. Therefore, with the application of Arduino open source platform, the connection between virtual and real-world contributes to efficient wind environmental performance building design. Indeed, simulated environment information and dynamic building models create a feedback loop together.

2. Morphology design strategy

The plane form of high-rise buildings has a great influence on the outdoor microclimate and pedestrian thermal comfort. The wind-driven plane form design methods are usually divided into three categories: fillet, chamfering, and indentation; while to reduce the wind load on the top volume of building, the changes in facade form are mainly divided into two types: top reduction and corner cutting. On the other hand, twisting and retreat are also applied for morphology optimization. The last two strategies do not change in volume, but rather the relative plane orientation and position.

Moreover, U.S. High-rise Buildings and Urban Housing Committee (CTBUH) revealed that there have been currently 28 twisted tall towers, which were either built or under construction (CTBUH 2016). These twisted buildings are generating a new landmark building tendency respectively, that is, to pursue more freedom in the vertical dimension. These representative buildings include the Shanghai Tower, Cayan Tower in Dubai and Absolute World Towers in Canada. However, its specific form may mainly consider aesthetics and structural engineering, their twisting angle and mode might not take environmental performance into account. The outdoor wind environment around tall building with the retreat design strategy is closely related to the windward angle of the stair-like facade. The Seoul Comprehensive Trade Center in South Korea adopted

a single-sided step-back approach, which weakened the impact of the subsidence on the pedestrians, while reduced the feeling of depression caused by the large number of buildings (Xie and Yang 2013).

In this paper, twisted form and retreat form have been selected as main mode of motion applied in the dynamic models. This morphology generation method that is beneficial to architects, according to wind environmental performance during the early stages of architectural design. Based on the principle of dynamic form-finding, this method transforms the relationship between complex architecture and wind environments into computer language and a programmable, adjustable model. It prompts the design to focus on the relationship between the building and the surrounding wind environment, and is not limited to the building form itself.

3. Experiment platform

3.1 Customized physical wind tunnel

In order to obtain rapid simulation feedback on the air flow around the building, a customized mini-wind tunnel was generated in 2016 (Yuan et al. 2016). This 3 meters length wind tunnel was made by wood planks and organic glass panels. Due to the open source nature of the wind tunnel module, any module can be rebuilt or additional functional modules can be added according to different experimental requirements. According to the order of air flow, it was divided into: stable section, contraction section, experimental section, diffusion section and fan section. In 2017, a smoke section was installed according to visualization demand (Zheng et al. 2016). The module is located upstream of the experimental section and is connected to the contraction section. It can generate fine smoke particles. The wind field simulated by the experimental section is seen by the naked eye and provides real-time observation air flow. In order to avoid the influence of the sensor on the fluid, the experimental section of the wind tunnel was modified in this study. The baseboard of the test section was composed of a detachable composite wood and multiple sensor's position was reserved opening according to the experimental requirements for pedestrian height wind environment measurement. The single opening must meet the pass of the sensor's hot wire

probe, and the size is set to 3 mm × 20 mm. The side of the opening is sealed with a sealed brush to maintain the airtightness of the experimental space. The number and location of openings must be tailored to specific experimental needs. The transformed wind tunnel is shown in Figure 1.

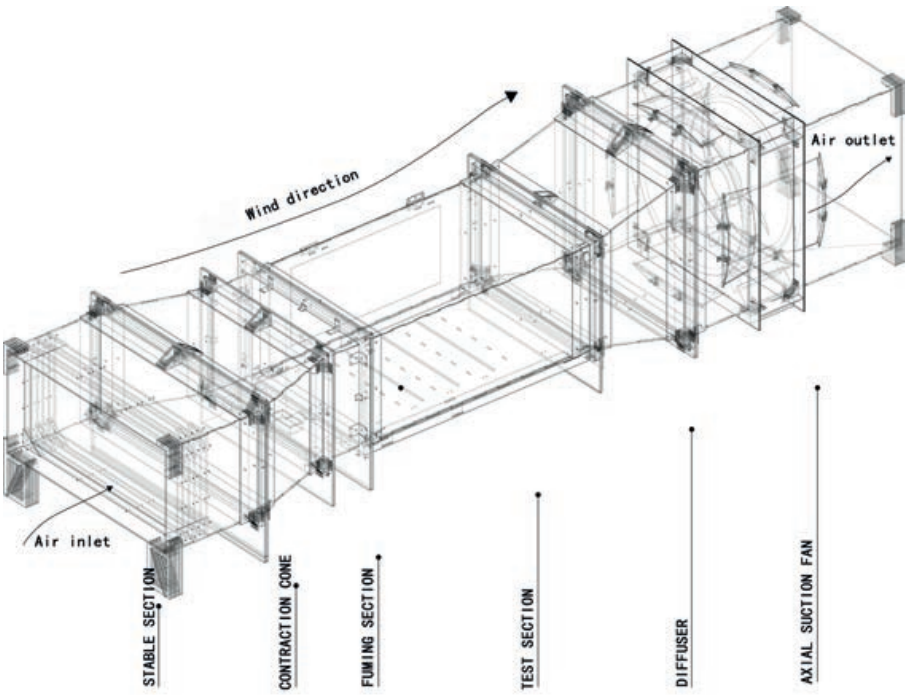


Figure 1: Customized physical wind tunnel.

3.2 Sensing system

The experiment chooses Rev. P as the main sensor, which can sense the wind speed and temperature in the environment in real time and transmit the electrical signal of the data to the Arduino board and convert it through the matching formula. The more accurate wind speed and temperature values can be obtained, which is more continuous than the wind speed value converted from pressure (Moya 2015). There are fixing holes on both sides of the top of the sensor electronic component. The insulated wire with a certain hardness can be used to clamp in the wind tunnel experimental section. The lower part of the sensor is located outside the wind tunnel, and the circuit and signal receiver can be connected (**Fig. 2**).

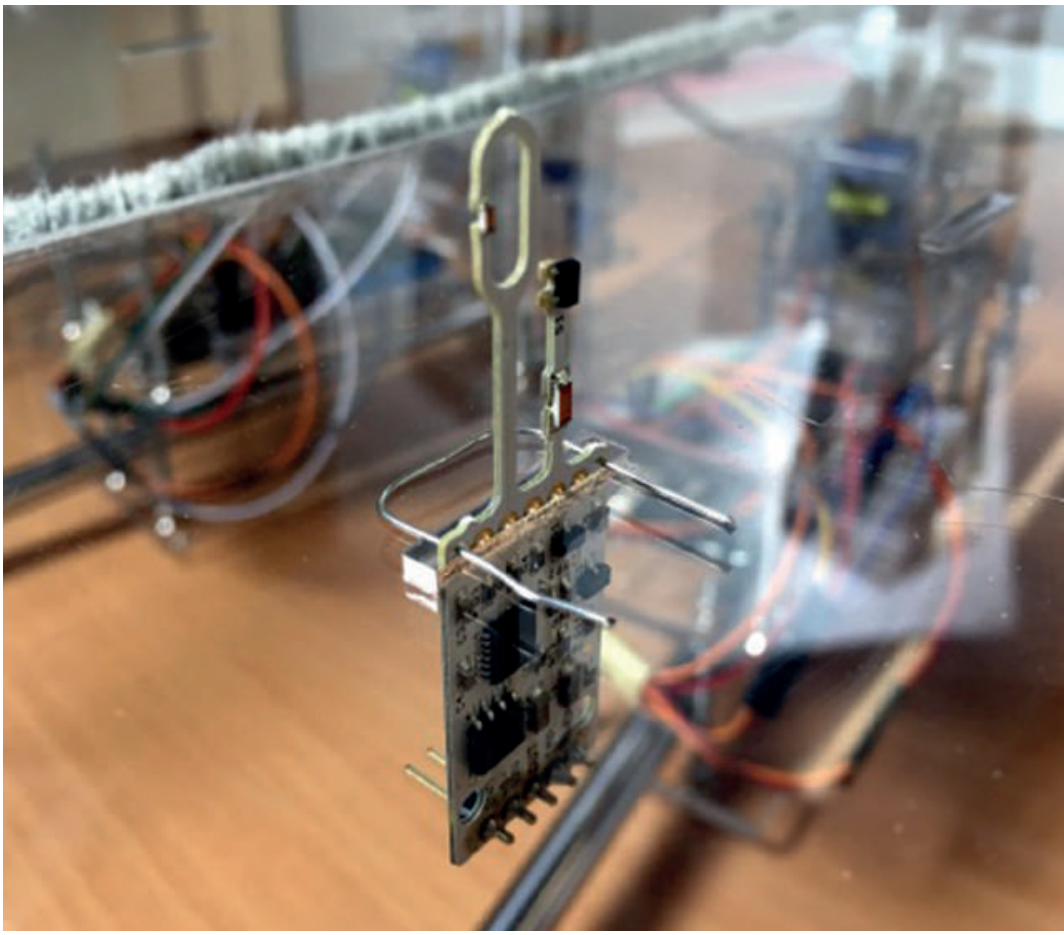


Figure 2: Wind sensor fixed on the baseboard of wind tunnel.

3.3 Arduino platform

A performance-optimized architectural design process often requires a feedback-capable control system that can integrated sensors and actuators into physical materials. Therefore, this study selected Arduino, a low-cost open-source electronic prototype platform for data transmission

control (**Fig. 3**). It includes an I/O board and C language based development environment (IDE) for software. Usually, simple programming in the IDE is programmed into the micro controller's circuit board to control the operation of sensors, servos and other components connected to the hardware.

At the same time, in order to make it easier for architects to control morphology generation algorithms, this study uses the Firefly plug-in in Grasshopper, which compensates for the disadvantages of Grasshopper. Although Grasshopper has a graphical programming environment interface, it is difficult to communicate with hardware devices. Firefly has replaced the Arduino IDE as a control terminal for programming and logic generation. On the one hand, the sensor data connected to the Arduino board can be obtained, and on the other hand, the designer is allowed to send control information from the Grasshopper to the hardware device to stimulate actuators (such as servos) in the real world (Payne and Johnson 2013).

Before the sensor operates, upload the matching code on the Arduino board. The Arduino serial port can display the wind speed in real time. When multiple sensors work together, the original code can be modified. This process cannot be completed in Grasshopper through Firefly. Therefore, programming in the Arduino IDE and customizing the number of sensors can better suit different experimental requirements.

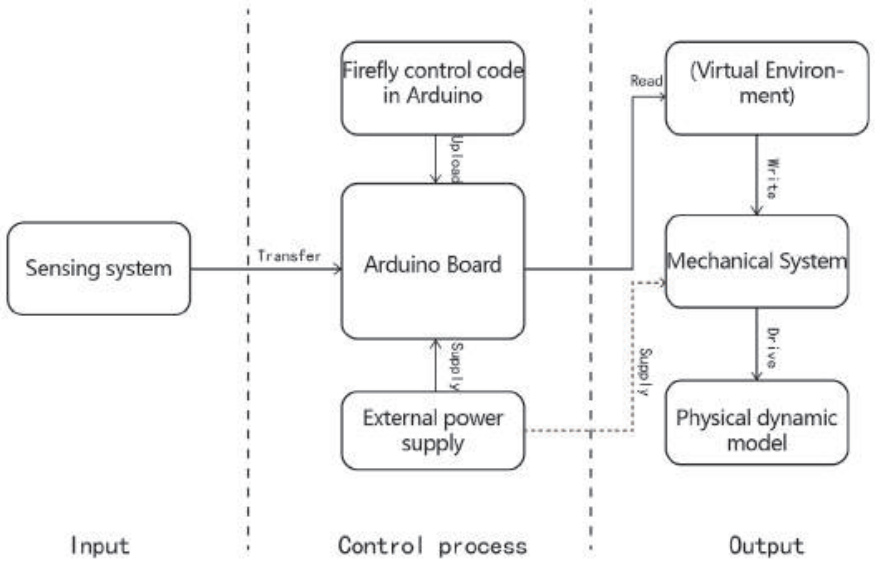


Figure 3: The workflow based on Arduino Platform.

4. Dynamic mechanical building model

The dynamic model design needs to consider issues such as wind penetration and mass integrity. Therefore, the architectural model of this study selected organic glass plates to be formed by stacking and using elastic threads to penetrate each layer of organic glass plates for integral connection. The superposition of materials provides the possibility of changing the relative position of each layer, while the elastic thread has well ductility and toughness, and can be connected inside the building model; at the same time, the elastic thread is pulled by applying horizontal mechanical force. Each layer of material is displaced so as to obtain a change of building morphology, and the change has a tightening force throughout the elastic thread, which can maintain the airtightness of the building model.

In order to convert the direction and motion trajectory of the horizontal force of the main model, the main principle of the mechanical drive design of the main model in this study is to apply the servo as the active part. The direction and speed, etc., are converted into the expected pattern of body changes. The number of steering gears plays a role in increasing the diversity of movement throughout the structure, resulting in more abundant physical changes. The mechanical structure of the two dynamic models is shown in Figure 4.

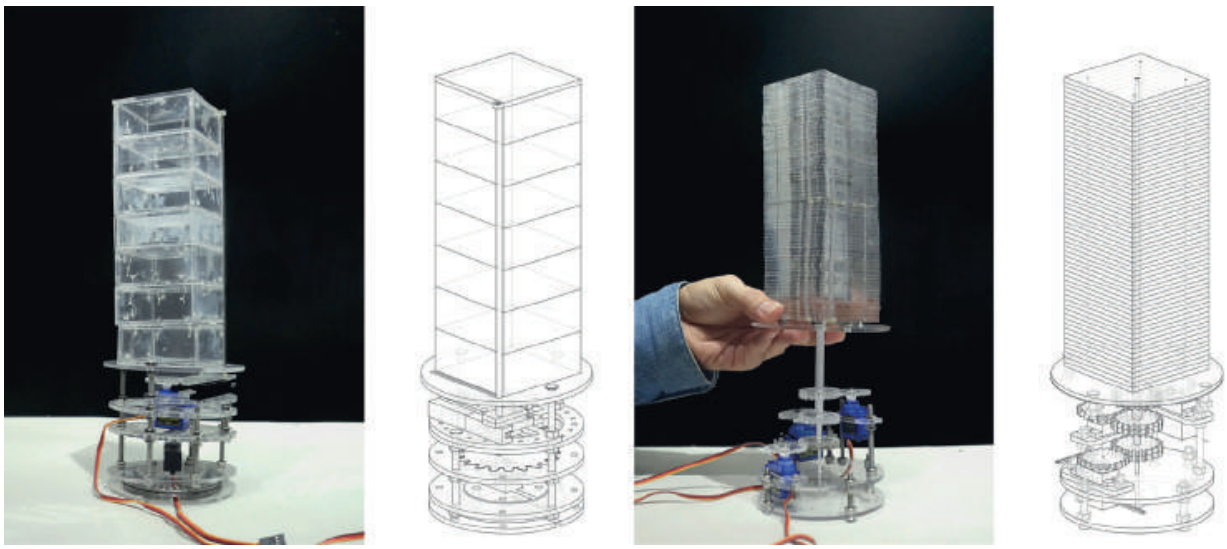


Figure 4: Twisting (up) and Retreating (down) building model.

4.1 Twisting mode based mechanical model

The “twist” building morphology changing strategy is to make the materials of each layer of the main building model rotate in the horizontal direction at different speeds with the center as the axis. The experimental model consists of a laser-cut 4 mm plexiglass sheet, which has a maximum frame size of 65 × 65 × 200 mm and uses a 65 × 65 mm rectangle as the standard layer prototype for a total of 50 layers. Among them, three plexiglass plates connect the steering gear as the active rotating plate, and the rest of the volume plates drive their changes through four elastic wires running through the inside. On the one hand, the elastic wire can provide a horizontal force for rotation of the driven rotating plate and is moderate in tension relative to a latex film, etc., and does not affect the rotation of the steering gear. Meanwhile, each passive rotating plate can be pulled for rotation. On the other hand, the elastic wire can be rotated. The continuity on the driven plate body can be maintained: the relative angle between adjacent driven plates is the angular difference between the two nearest active rotating plates in the vertical direction divided by the number of driven plates. As shown in Figure 5, the three active rotating plates of the main building model part are respectively controlled by three different servos. The steering gear drives the three rotating shafts that are nested together to rotate through the gear transmission. Each of the three rotating shafts consists of a hollow ABS tube, which respectively transmits the rotation of the corresponding steering gear to the corresponding active rotating plate.

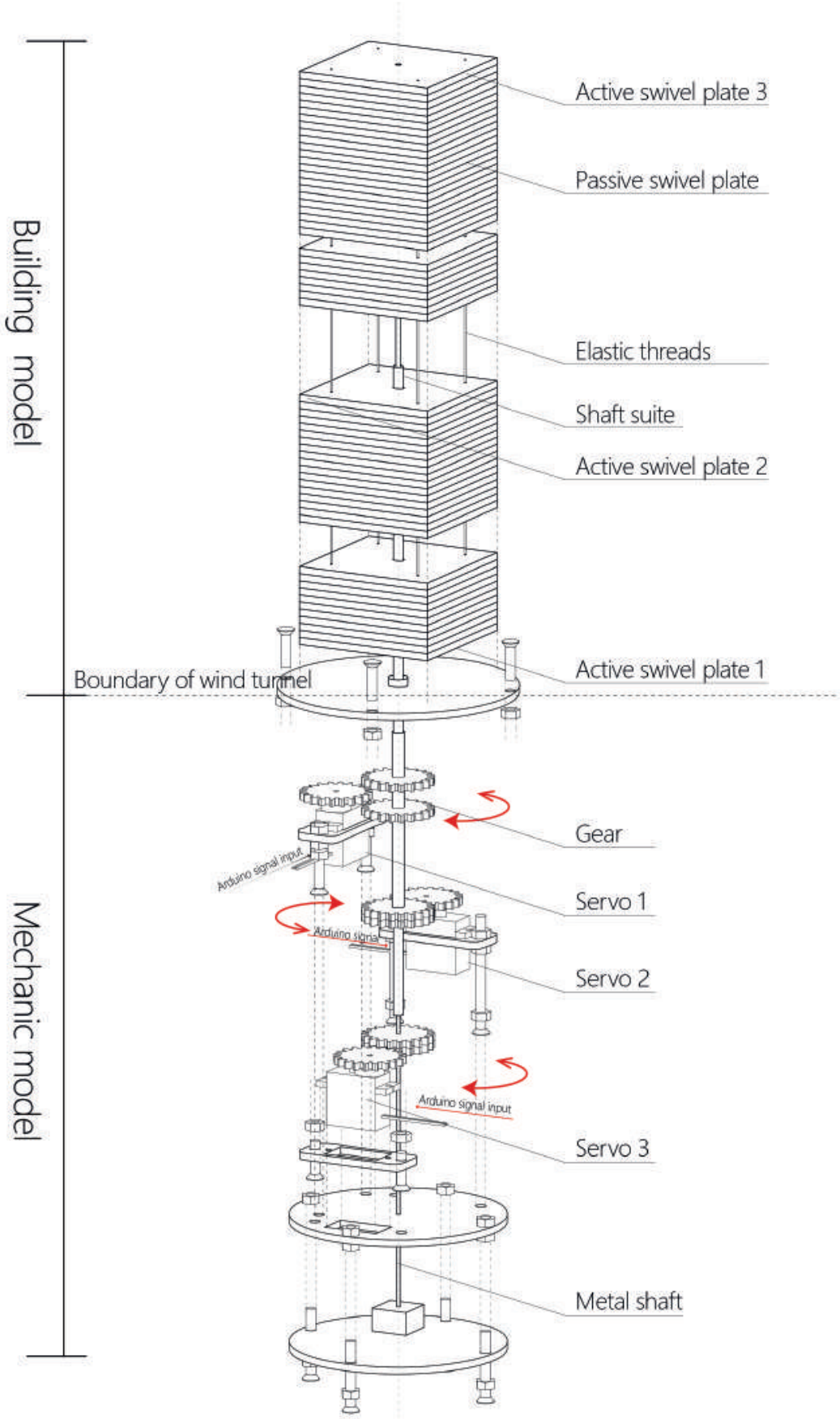


Figure 5: Explosive view of twisting dynamic model.

4.2 Retreat mode based mechanical model

The strategy of retreating is to make the material of each layer of the main model move at different distances. The volume of the main model is mainly composed of seven boxes with a height of 26 mm. The top layer is the active displacement layer and is moved by the force of the metal shaft driving translation. The bottom layer of the main model is fixed with the bottom plate, and each layer is connected by four elastic wires. The displacement of the top layer can drive the rest of the driven layers to be displaced. The elastic wire mainly plays the role of driving the driven layer to perform displacement in the model. As shown in Figure 6, the rotation of the steering gear 5 drives the gear meshing with the linear rack to convert the rotation into translational motion. The displacement distance is related to the angle of rotation of the steering gear and the number of gears. In this experiment, the steering gear is rotated by 45 degrees per tooth equals to 10mm movement. Due to the complexity of the site environment, the unidirectional translation of the evacuation body lacks the diversification of the experimental objects. Therefore, an overall rotation device is added to the mechanical structure of the evacuation model, and the wind environment can be applied to different degrees of translation of the body in different directions. The gear connected by the steering gear 4 drives the entire building body model to rotate, and at the same time includes the rotational movement of the steering gear 5 and the translational sliding block. As a result, the physical parameters of the retreating model mainly consist of the model displacement distance represented by the degree of rotation of the steering.gear 5 and the overall rotation angle of the steering gear 4.

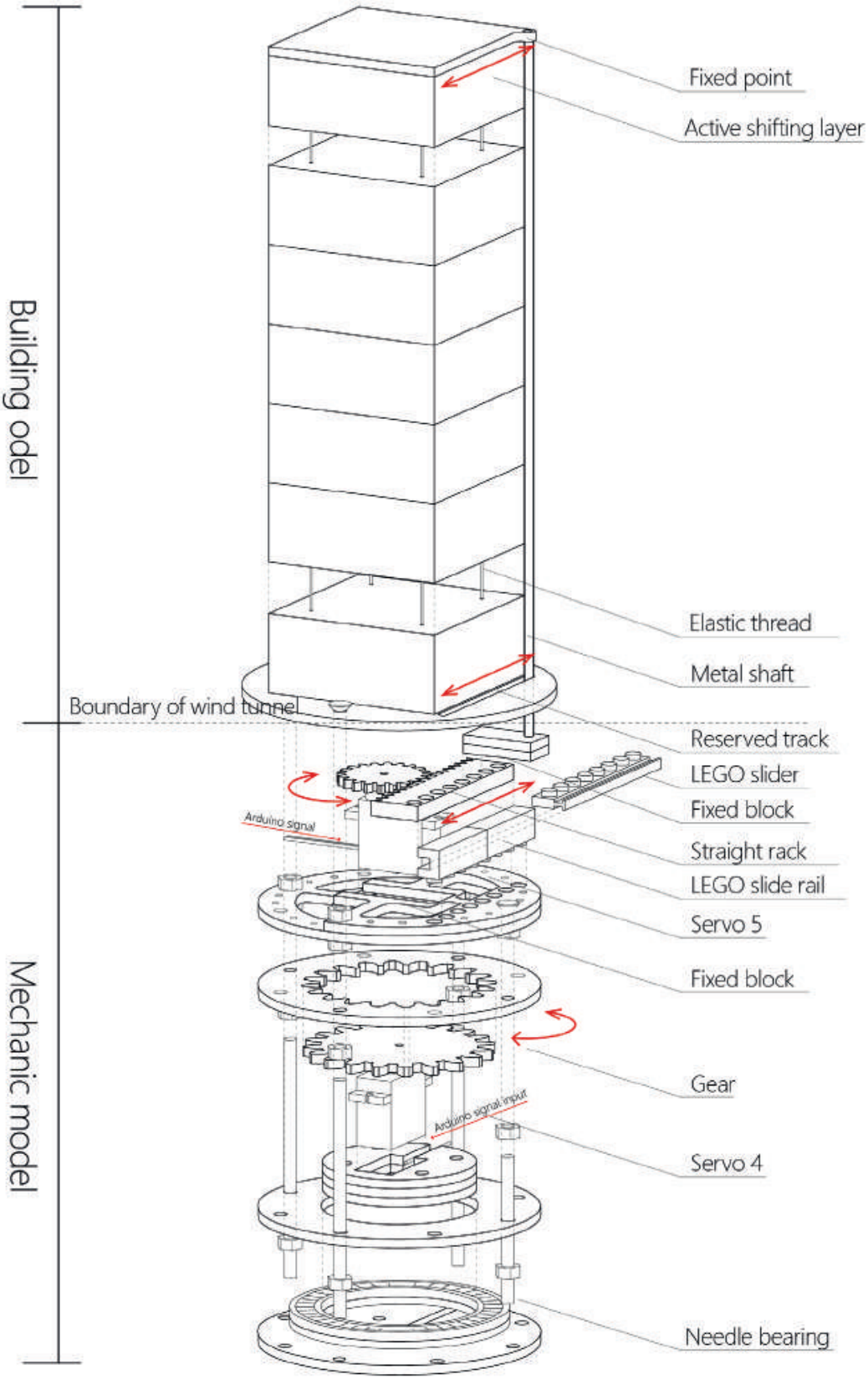


Figure 6: Explosive view of retreat dynamic model.

5. Morphology generation approach and workflow

The flow of the generative design driven by the environmental performance is composed of three elements: performance simulation data, morphology change rules and evaluation criteria. Architectural morphology design is no longer just the feeling of space and the pursuit of aesthetics. Mass, components and their changes are described by parameters in the design process and are constrained by the digital environment elements. The final morphology of the building is generated in the evaluation standard with environmental data screening and control.
In Galapagos' architectural vocabulary logic using “genetic algorithm”, the change of “independent variables” promoted the generation of diversification of digital forms, and thus prompted the generation of “dependent variables” according to the set “logical correlation”, and “dependent variables”. The evaluation determines the optimal solution of the building.
The physics logic of this study (Fig.7) is similar to that of Galapagos, but the control subject of the “independent variables” is transformed from the virtual electronic model into the mechanical model of the servo drive; the simulation tools involved in “logical correlation” are changed from computational fluid dynamics (CFD) software to physics. Wind tunnel experiment, through the digital measurement system to obtain “dependent variables”. After the establishment relationship of the data, a certain “evaluation standard” can be used to screen the “dependent variable” and the building morphology under the control of the optimal “independent variable” can be obtained.

5.1 Logic programming for servo movement

The “independent variables” in the logic of morphology change consist of servo movement data. First, set the time interval of the steering gear rotation and the step angle in the Grasshopper, that is, the mechanical angle rotated by the rotor for a pulse signal without a reduction gear. At the beginning of the form-finding experiment, a series of servo position parameters are obtained as time increments. The parameters of the servos generated at the same time are combined into an “argument” parameter group. These parameter sets correspond to different physics models. Each pattern will obtain a series of wind speed values during wind tunnel simulation. The wind speed values are read into Grasshopper through the Arduino platform and Firefly. The pre-processed effecti-

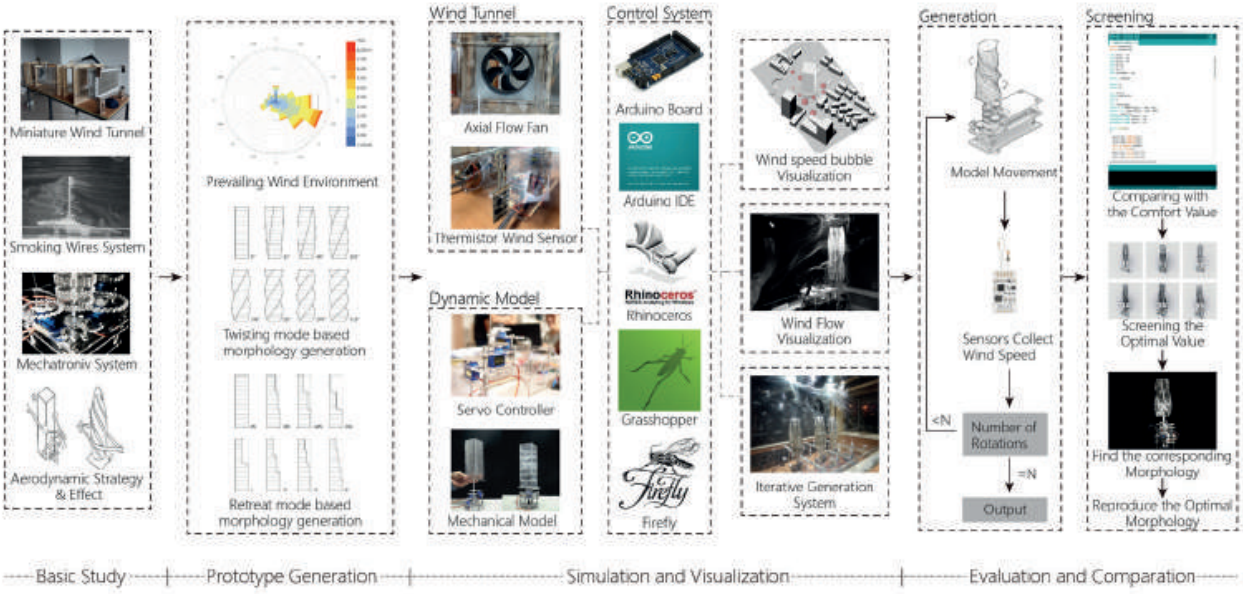


Figure 7: The workflow of architectural generation method combining physical wind tunnel and dynamic model.

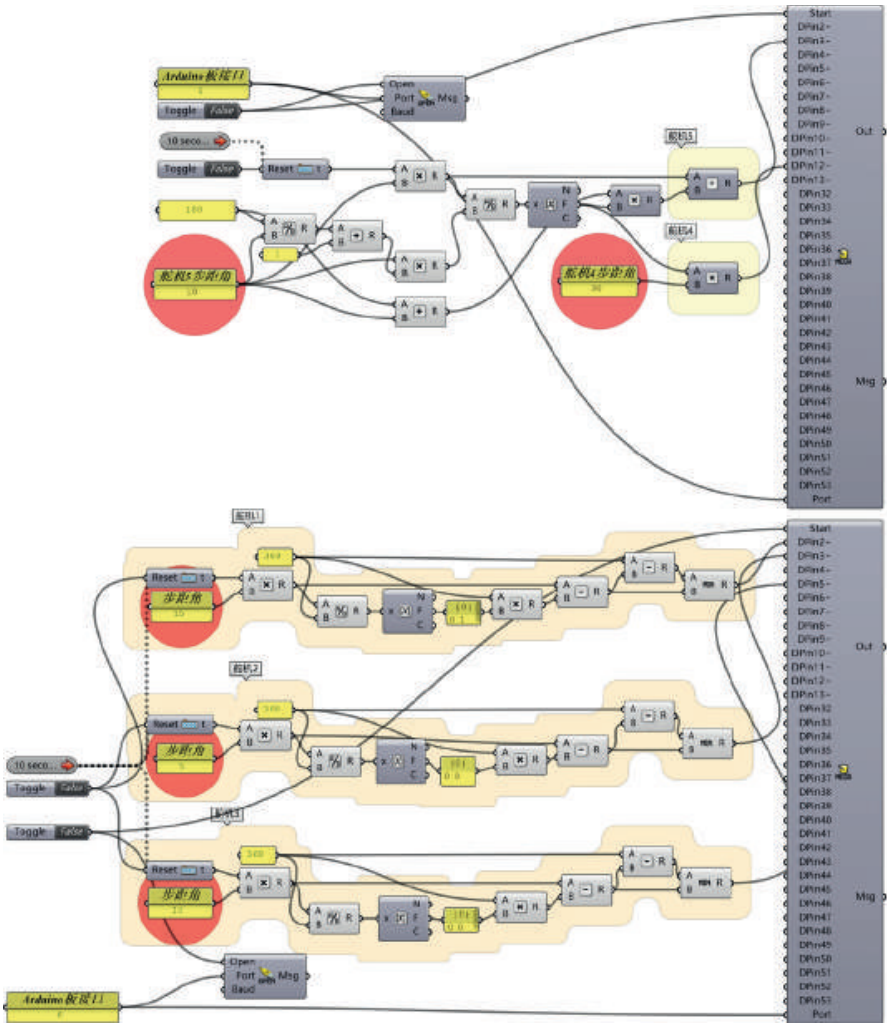


Figure 8: Grasshopper diagram of steering gear data controlling of retreating (up) and twisting (down) dynamic model.

ve and stable wind speed is again converted into a form of a parameter that can be evaluated as a “dependent variable”, and the computer is guided through a certain evaluation criteria to screen it (**Fig. 8**).

5.2 Data collecting of the wind tunnel experiment

After the parameter logic is set up, in the physical wind tunnel, the site model will be installed, initial wind speed settings and sensor location should also be determined. The position of the sensor can be determined based on the pseudo-color map obtained by CFD software simulation, and the representative points are selected. During the experiment, Grasshopper will record servo parameters corresponding to all morphology and data from sensors around these models (**Fig. 9**). When all data are collected to the expected value, Grasshopper will process and filter the data through the preset evaluation criteria, and finally show the three groups of optimal physical parameters.

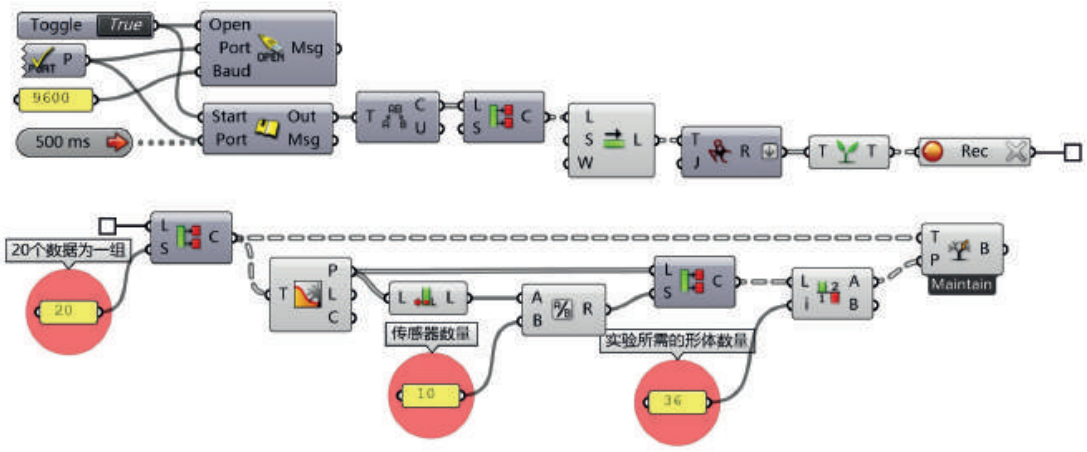


Figure 9: Sensor data processing.

5.3 Evaluation rules translation

In this experiment, three criteria commonly used in wind environment evaluation were selected to assess the effective data generated in the experiment: the deviation of comfortable temperature, the probability of uncomfortable wind speed and the discrete value of wind speed at the measurement point. Among them, the comfort of the wind environment takes the comfortable wind speed of 1m/s at the height of pedestrians in Shanghai at a summer temperature (around 27). The evaluation is based on the minimum wind speed value and comfort wind speed

deviation value as the optimal solution; the uncomfortable critical wind speed value is set to 3.9 m/s according to the pedestrian height and the street wind speed and frequency evaluation standard, and the pedestrian is standing. The probability of occurrence of the critical wind speed value is less than 80 % as the comfort range, while the generation of the minimum frequency larger than the critical wind speed value is the best; the evaluation of the wind speed dispersion value is to prevent the wind speed mutation from adversely affecting pedestrians. The higher the wind speed, the greater the sudden change in wind speed and the harsher wind environment.

5.4 Comparison and screening out the optimal morphology

After inputting the valid data measured by the sensor, the three evaluation criteria all take the minimum value of the battery output as the optimal option. After the effective data collected by the experiment is converted by the evaluation logic, weights are assigned to the data under different evaluation criteria. The weighting factor can be set according to different functional requirements around the main building. The processed data is added and sorted, among which the smallest three data are the optimal solution environmental data group. As shown in Figure 10, the operator inputs all effective wind speed values of 10 sensors from the leftmost end, and calculates and estimates the comfortable wind speed deviation data of a single body sample, the probability of uncomfortable wind speed values generated by the morphology body and the discreteness of wind speed, respectively. The right side can output the body number that produces these three optimal environment data. Finally, according to the body number, the corresponding main building morphology for the next qualitative test can be found.

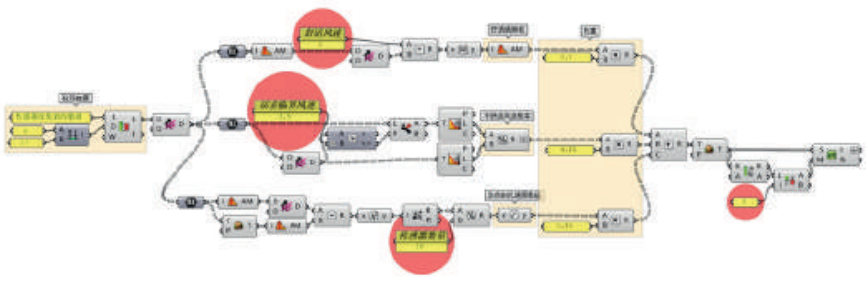


Figure 10: Comprehensive assessment of environmental data.

6. Experiment results

This study describes the design method through a series of form-finding experiments. This paper adopts “twisting” and “retreating” mode to carry out dynamic form finding experiments in rectangular standard plane shapes with the same site environment. Based on the same change rule, the weights of the environmental data evaluation in the warping strategy experiment are: 70% for comfort, 15% for wind speed dispersion, and 15% for uncomfortable wind speed; There is a stagnation area with a low wind speed. Therefore, in order to ensure the homogeneity of the airflow around the body, the evaluation weights for the regression model in this experiment are: 60% for the comfort deviation, 30% for the wind speed dispersion, and uncomfortable wind speed, while the probability is

No.		Servo parameters	Average wind speed(m/s)		Comfort deviation	Wind speed dispersion	Uncomfortable wind speed probability
Rectangular plane & Twisting strategy	RA-8	Servo 1 : 105°	S1 : 1.51	S6 : 2.01	0.578	0.579	0
			S2 : 0.54	S7 : 1.85			
		Servo 2 : 35°	S3 : 0.43	S8 : 2.02			
			S4 : 1.96	S9 : 1.58			
		Servo 3 : 70°	S5 : 2.15	S10 : 1.73			
			Total average : 1.578				
	RA-21	Servo 1 : 160°	S1 : 1.73	S6 : 1.55	0.602	0.452	0
			S2 : 0.82	S7 : 2.08			
		Servo 2 : 100°	S3 : 1.02	S8 : 1.22			
			S4 : 1.54	S9 : 2.31			
		Servo 3 : 60°	S5 : 2.03	S10 : 1.72			
			Total average : 1.602				
	RA-31	Servo 1 : 60°	S1 : 1.43	S6 : 1.56	0.555	0.738	0
			S2 : 1.00	S7 : 1.32			
		Servo 2 : 150°	S3 : 0.32	S8 : 0.78			
			S4 : 1.99	S9 : 2.57			
		Servo 3 : 90°	S5 : 2.86	S10 : 1.72			
			Total average : 1.555				
Rectangular plane & Retreat strategy	MA-15	Servo 4 : 90°	S1 : 1.56	S6 : 1.53	0.356	0.584	·0
			S2 : 0.41	S7 : 1.65			
			S3 : 0.55	S8 : 0.50			
		Servo 5 : 180°	S4 : 1.81	S9 : 1.82			
			S5 : 2.00	S10 : 1.73			
			Total average : 1.356				
	MA-19	Servo 4 : 135°	S1 : 1.05	S6 : 1.46	0.31	0.679	0·
			S2 : 0.35	S7 : 1.81			
			S3 : 0.21	S8 : 0.61			
		Servo 5 : 135°	S4 : 1.87	S9 : 1.73			
			S5 : 2.31	S10 : 1.70			
			Total average : 1.31				
	MA-20	Servo 4 : 135°	S1 : 1.00	S6 : 1.44	0.302	0.686	0
			S2 : 0.32	S7 : 1.59			
			S3 : 0.28	S8 : 0.52			
		Servo 5 : 180°	S4 : 1.98	S9 : 1.95			
			S5 : 2.22	S10 : 1.72			
			Total average : 1.302				

Figure 11: Servos and corresponding morphology results.

10%. The three optimal solutions produced by each body under quantitative evaluation will be analyzed by qualitative smoke experiments, and the form of the wind field will be judged again so as to obtain better physical results, as shown in Figure 11. And the other building models with different morphology design strategies in the smoke visualization can be seen in Figure 12 and Figure 13.

Figure 12: Morphology generation results.

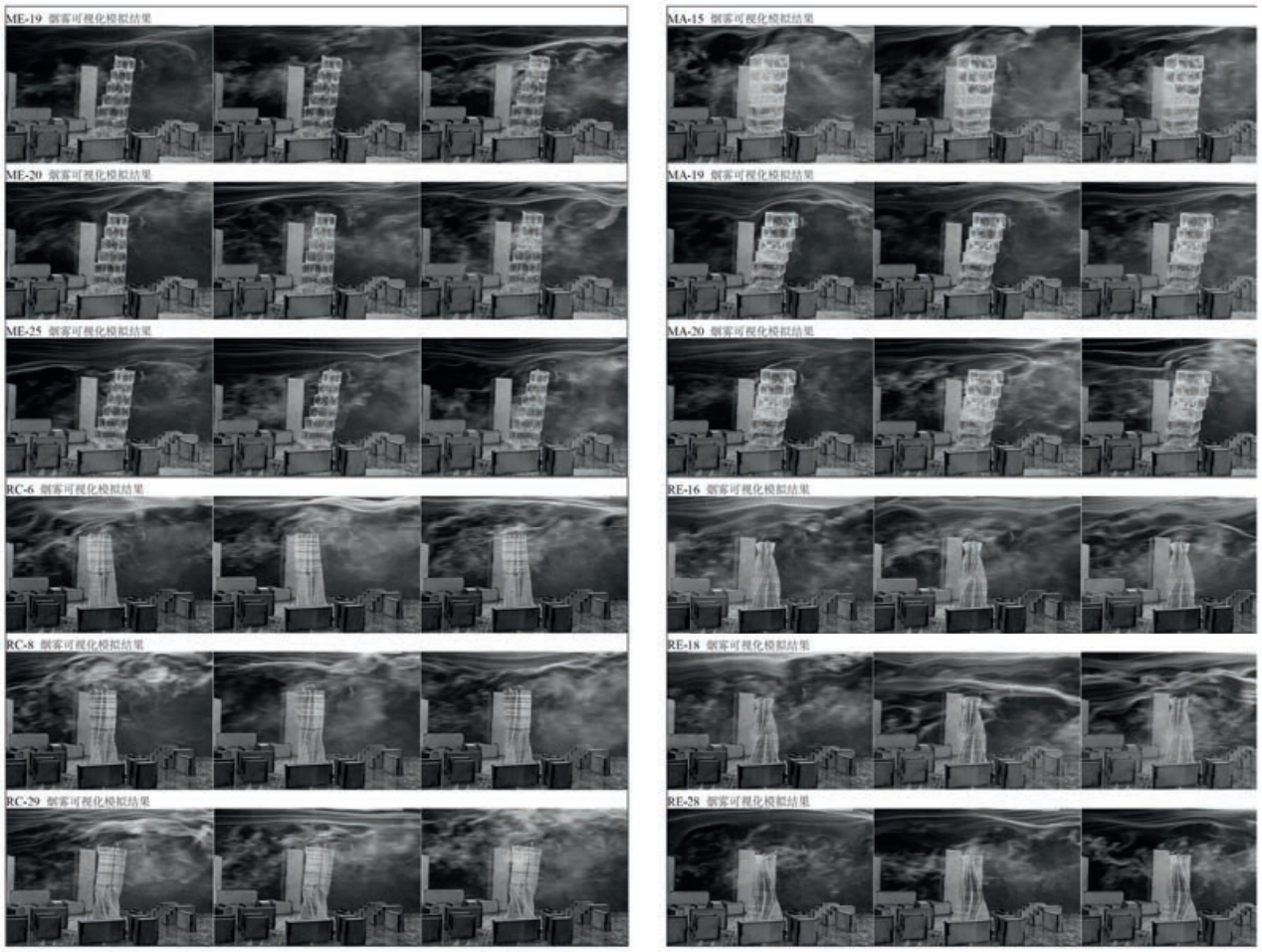


Figure 13: Dynamic models with wind tunnel visualization.

7. Summary and future work

This study introduced a new design method, built the feedback logic between physical wind tunnel and the architects common software Grasshopper. A generated feedback dynamic model of the Arduino connected sensors and actuators, making the simulation data in the wind tunnel efficiently applied. This logical system can be easily controlled by digital software, thus constructing a set of environmental performance design methods suitable for architects in the early design stages. This method provides a set of implementable logic flow frameworks. Based on user-defined site environment and evaluation criteria, the overall building morphology can be digital controlled. The wind environment simulation is involved in the design process at the early stage of design. On the one hand, it can not only create better natural ventilation conditions for the microclimate around the building and improve the comfort of the users in the site; on the other hand, the performance control of

the building form at the beginning of the design can avoid the loss of economy and resources brought by the optimization and adjustment stage in the later stage of design.

The experiments that can be performed using this design method are not limited to this. Under this open design framework, designers can change the physical strategy, site environment, sensor locations, and data evaluation rules according to specific experimental objectives. In addition, sensors such as wind pressure, air temperature and humidity can be installed in the wind tunnel to evaluate the comprehensive built environment.

References

CTBUH, Council on Tall Buildings and Urban Habitat. 2016. "Tall Buildings in Numbers Twisting Tall Buildings." CTBUH Journal, 3, 46–47.

Lin, Y. Q., Zheng, J. Y., Yao, J. W. and Yuan, F.: 2018, Research on Physical Wind Tunnel and Dynamic Model based Building Morphology Generation Method, Proceedings of CAADRIA 2018.

Moya, R.: Empirical evaluation of three wind analysis tools for concept design of an urban wind shelter[J]. Proceedings of Computer-Aided Architectural Design Research in Asia CAADRIA, 2015: 313–322.

Payne, A. O., Johnson, J. K. Firefly: Interactive Prototypes for Architectural Design [J]. Architectural Design, 2013, 83(2): 144–7.

Xie, Z. Y. and Yang, N.:2013, Optimization Design Tactics for High-rise Building Shape on Improvement of Outdoor Wind Environment, Journal of Architecture, 2013(2):76–81.

Yuan, F., Huang, S. Y. and Xiao, T.: 2016, Physical and numerical simulation as a generative design tool, Proceedings of CAADRIA 2016, 353–362.

Zheng, J. Y., Yao, J. W. and Yuan, F.: 2017, Architectural generation approach with wind tunnel and simulation: Environmental Performance-driven design approach for morphology analysis in the early design stage, Proceedings of CAADRIA Short Papers 2017, 13–18.

Form finding and design of a timber shell-nexorade hybrid

Romain Mesnil, Douthe Cyril, Tristan Gobin, Olivier Baverel

Romain Mesnil
romain.mesnil@enpc.fr
Laboratoire Navier, Ecole des Ponts/IFSTTAR/CNRS, France

Douthe Cyril
cyril.douthe@ifsttar.fr
Laboratoire Navier, Ecole des Ponts/IFSTTAR/CNRS, France

Tristan Gobin
t.gobin@hal-robotics.com
HAL Robotics/GSA, France

Olivier Baverel
baverel@enpc.fr
Laboratoire Navier, Ecole des Ponts/IFSTTAR/CNRS, France

Keywords:
Nexorade, shell-nexorade hybrid, timber structure, PQ-mesh, form finding, marionette technique

Abstract

The aim of this paper is to discuss the form-finding of an innovative structural system through the design and construction of a full-scale timber pavilion. Nexorades, or multi-reciprocal grids, are structures where members support each other along their spans. This structural principle allows simple assembly and connection details, but leads in counterpart to poor structural performance. Introducing planar plates as bracing components solves this issue, but result in a complex and intricate geometry of the envelope and supporting structure. This paper discusses the different challenges for the designers of shell-nexorade hybrids and algorithmic framework to efficiently handle them in a project workflow.

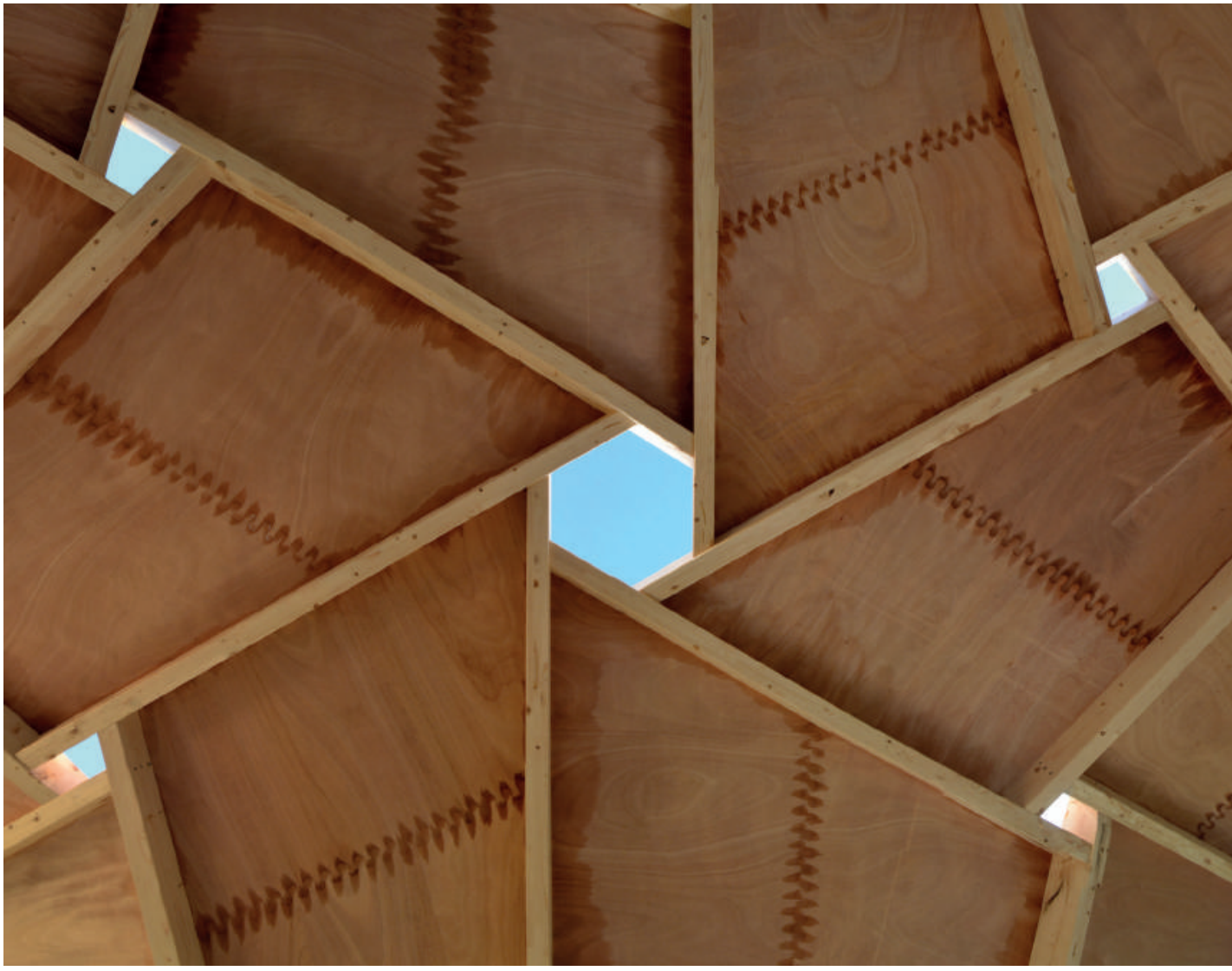


Figure 1: The central fan of the pavilion.

1. Introduction

Nexorades, also known as reciprocal frames or multi-reciprocal grids, are structures constituted of short members which supports on their ends. They are simple to assemble because of they avoid the construction of complex connection details and can be built with short members. For that matter, they have been used since medieval architecture, for example by Villard de Honnecourt, or in sketches of Leonardo da Vinci in the Codex Atlanticus (Bowie 1960). Recent examples of nexorades include the "Plate Pavilion" in Malta, the "KREOD Pavilion", designed by architect Chun Qing Li, engineers Ramboll and geometry consultants Evolute, and timber nexorades in ETH Zürich (Kohlhammer, et al. 2017).

Despite their ease of assembly, the structural behavior of nexorade is far from optimal, because of the low valence, which implies a bending-dominated behavior, even for funicular shapes (Brocato 2011). This has been observed in the material science community, where this property of nexorades is used to create auxetic materials. In structural engineering, this behaviour limits the range of spans where those structure are economical and efficient. The structural behavior of nexorades can however be improved by bracing them with planar quadrilateral panels. The authors call "shell-nexorade hybrids" the new resulting structural system.

The handling of both facet planarity and of the geometry of nexorades at the same time is unexplored up to now, and requires tailor-made geometrical algorithm for construction-aware structural form finding. This paper discusses thus a dialectic approach between constructive geometry and structural mechanics and its application to the form-finding and construction of a timber pavilion, shown in Figure 1.

2. Structural and fabrication requirements

In engineering practice, optimization or rationalization have multiple competing objectives and constraints. When aiming at high slenderness, structural detailing and envelope detailing merge, so that the fabrication process and structural design process intertwine. The design of the pavilion is driven by the structural and fabrication constraints, which are described in this section.

2.1 Project description

The project aims at illustrating the potential of shell-nexorade hybrids as an efficient and easy-to assemble structural system. First, we have to introduce some vocabulary specific to nexorades. Nexorades are constituted of load-bearing members, which support each other along their span and not their extremities. The generation of their geometry is based on the displacement of edges of a watertight mesh. The displacements create engagement windows, shown in thick blue lines in Figure 2. Two values characterize the engagement windows: their lengths, called engagement lengths, and the eccentricity, which is the distance between the neutral axis of two concurrent members.

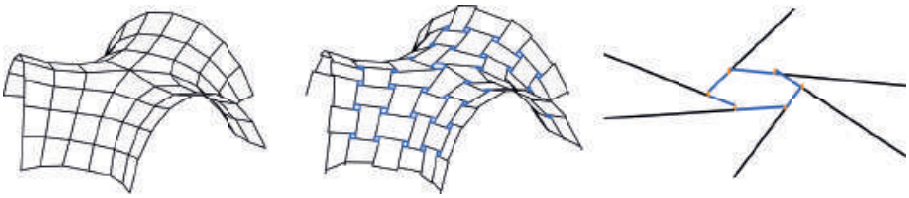


Figure 2: The reference geometry (left), a nexorade resulting from a transformation of the mesh with engagement windows in blue (middle), a close-up on a fan: the engagement length is the length of a blue line, while the eccentricity is the distance between the lines, highlighted in orange (right).

The pavilion is constructed with cross-laminated timber beams and 10mm thick plywood panels. The pavilion has a span of approximately 7 meters and covers an area of 50 m^2. Its shape is inspired by the CNIT, a thin shell supported on three punctual supports: the geometry and pattern topology allow to build by cantilevering from a central tripod. The beams weigh approximately 5 kilograms and can be assembled in-situ by two people. Mechanical attachment, as opposed to chemical attachments, are used to guarantee on-site assembly. With a thickness of 14 cm, the slenderness ratio is of 50, a rather high value for timber structures.

2.2 Fabrication and construction requirements

Fabrication constraints are induced by the robotic fabrication process, the materiality of timber and by structural considerations. The panels covering the structure must be as planar as possible, in order to guarantee

their fabrication from plywood and to avoid coupling between bending and axial forces, as they are used as a bracing system.

The detailing chosen for the connection of beams and panels is shown in Figure 3: beams are connected with end-grain screws, while beam/panel attachment is made by screws. Tenons and mortises are milled in the beams for the assembly process. Grooves are milled in the timber beams as alignment failsafe between panels and beams. The top surface of the beams are milled in order to avoid timber exposure created by the eccentricities. The higher the eccentricity, the less material after milling, and the lower the lever arm and resistance of the connection. The detailing implies thus to minimize the eccentricities, a rather unusual optimization target in nexorades.

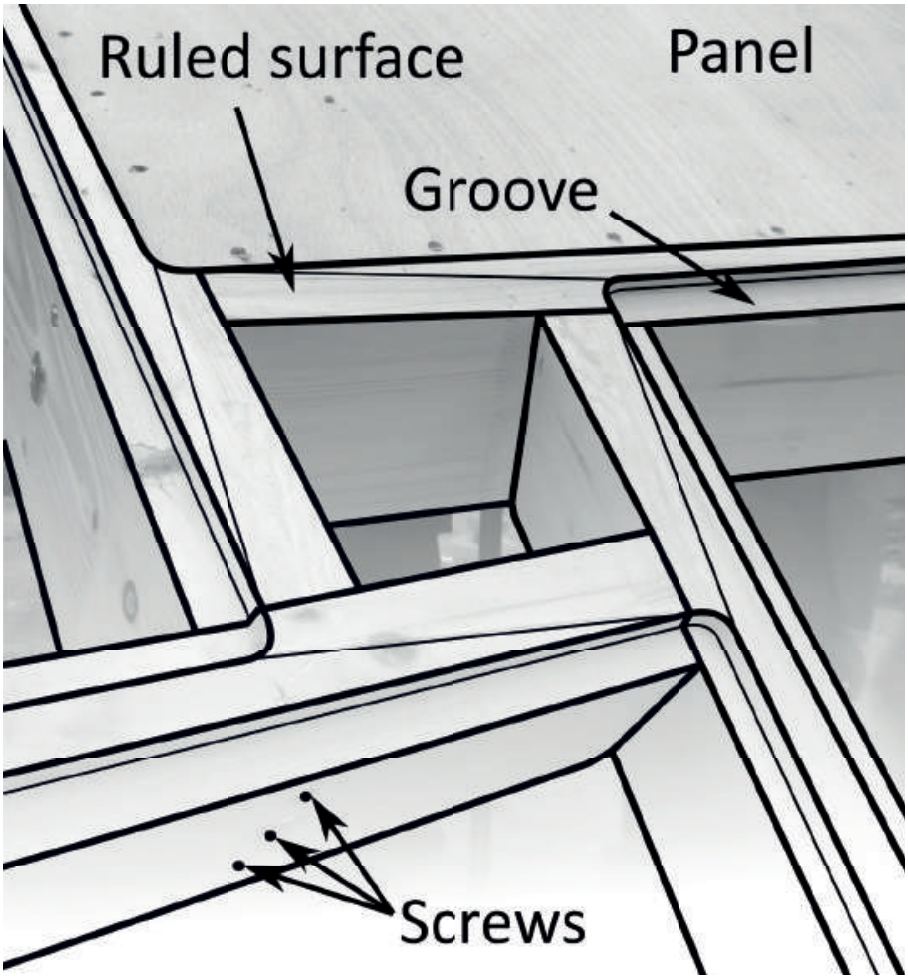


Figure 3: A fan during construction showing the detailing: beams are connected by end-grain screws which create a moment-rigid connection, grooves are made in the beam to fit planar panels, and ruled surfaces are milled on top and bottom of the members to avoid timber exposure.

The beams are fabricated with the aid of 6 axes robots shown in Figure 4: one robot on a track with a gripper that is used as a mobile frame for the beam and the other to perform the milling operations. Since the milling of the beam is performed with 13 (6+6+1) axes, the complexity of the attachment between beams and panels can be treated with the milling of the beam. The plates are thus cut with simple 2.5 axes CNC machines. Their width of 1030 mm sets constraints on the bounding box of the panels. Other geometrical constraints are imposed by the robotic fabrication process: the size of the gripper imposes a minimal length of 750 mm between two mortises, while vibrations restricted the beam length to 2 000 mm. The speed of cut was adjusted to avoid vibration and to minimize cutting forces. Angles between members also had to be minimized in order to ease the approach of the robotic arms.

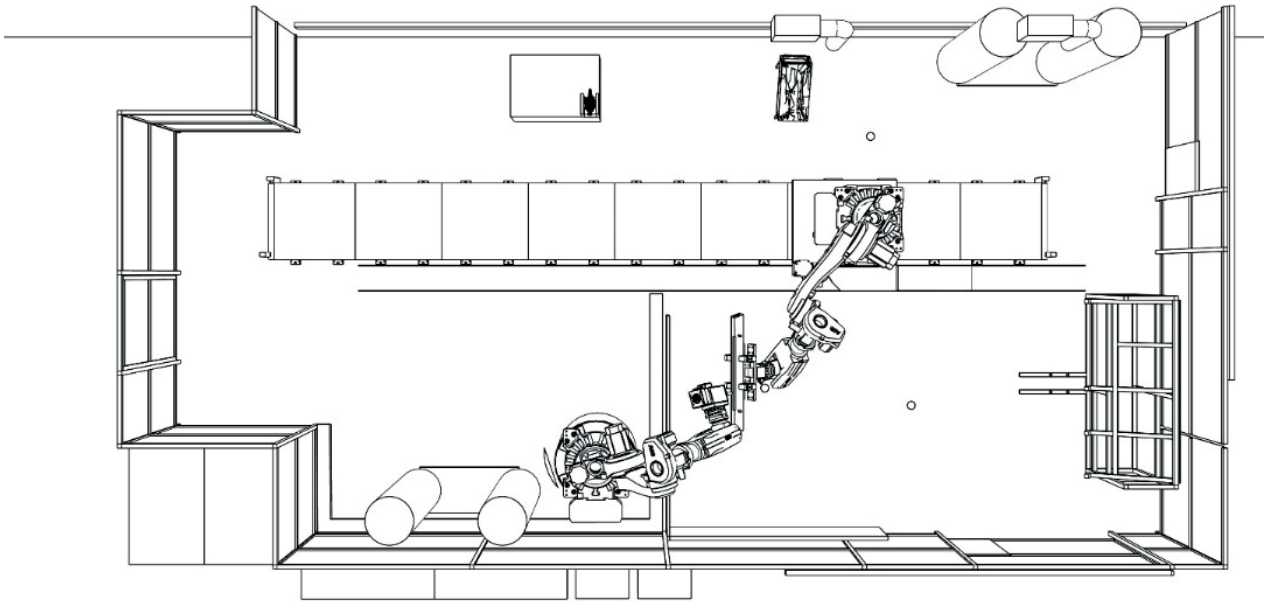


Figure 4: Top view of the robotic cell with gripping robot on track, milling robot (bottom) and fixed tools (top).

2.3 Structural requirements

The structure presented in this paper is a temporary building with a lifespan of one year. As such, it has to withstand climatic and acciden-

tal loads. The envisioned accidental load is a non-symmetrical load of 700 kg (approximately 10 people climbing on the structure). The loads and design capacity of beams have been derived from the Eurocode 1 and 5 respectively. The connection details, which use end-grain screws, are not designed within Eurocode, and a European Technical Approval (European Technical Approval ETA-11/0190) must be used. The serviceability and ultimate strength requirements are defined as follows:

$$\begin{cases} \delta_{\max} < \frac{L}{300} \\ f(N, V_y, V_z, M_x, M_z) < 1 \end{cases} \quad (1)$$

Where L is the span (6.5 m), $\delta_{\max}$ is the maximal deflection under SLS load combination, f is a convex function defined by the technical agreement of the screws describing the utilization of the connection details. Buckling was also checked, but, due to the relatively small span, it is not the governing phenomenon. The ULS design is conservative because not a single nodal failure is allowed.

The final structure weighs approximately 15 kg/m^2, so that self-weight is far from being the governing load case for a temporary building, where creep can be neglected. Therefore, the geometry does not have to follow a funicular shape, and CAD tools can be used to generate a structural shape. Real-time feedback from a finite element analysis is therefore necessary to optimize the structural behaviour (Bletzinger, Kimmich et Ramm 1991). The reference geometry was thus generated as a collection of NURBS.

2.4 Computational workflow

The fabrication and structural requirements are integrated in a computational workflow is presented in Figure 5. The workflow is based on several optimization algorithms that solve construction problems and allow to iterate on the different design parameters to improve the performance of the design. We focus here on the geometrical aspects of the computational workflow, treating the robotic setup as a design constraint. In reality, iterations between the design and the organization of the fabrication platform have been made to guarantee the constructability of the pavilion. It is very likely that an industrial with different machines would have another set of design constraints.

Two levels of geometrical complexity are handled through the design. In early design stages, the architect and engineer deal at a coarse level,

called “design geometry” in the flowchart: the members and plates are represented by lines and surfaces respectively, and a priori cross-section are used. This allows to discard bad designs and to quickly iterate and “optimize” the design, although some modelling assumptions are up to the knowledge of the designer. Then, the designer needs to work at a finer level of detail and thus to generate the “fabrication geometry”. At this stage, beams are generated as BREP, the proper cross-sections are assigned according to the as-built geometry. The feasibility of the fabrication is also assessed, in our case with the aid of HAL (Schwartz 2012). This step is much more resource demanding, as a considerable amount of fabrication data has to be generated.

The main geometrical operations performed for the form finding of the pavilion aim at complying with the main fabrication constraints. First, the designer sets an input geometry, it is then fitted by a mesh with planar facets. This mesh is then transformed with a custom algorithm, so as to create a nexorade with planar panels. The different fabrication data can then be generated and the structural response is evaluated. The next section discusses in detail these design steps.

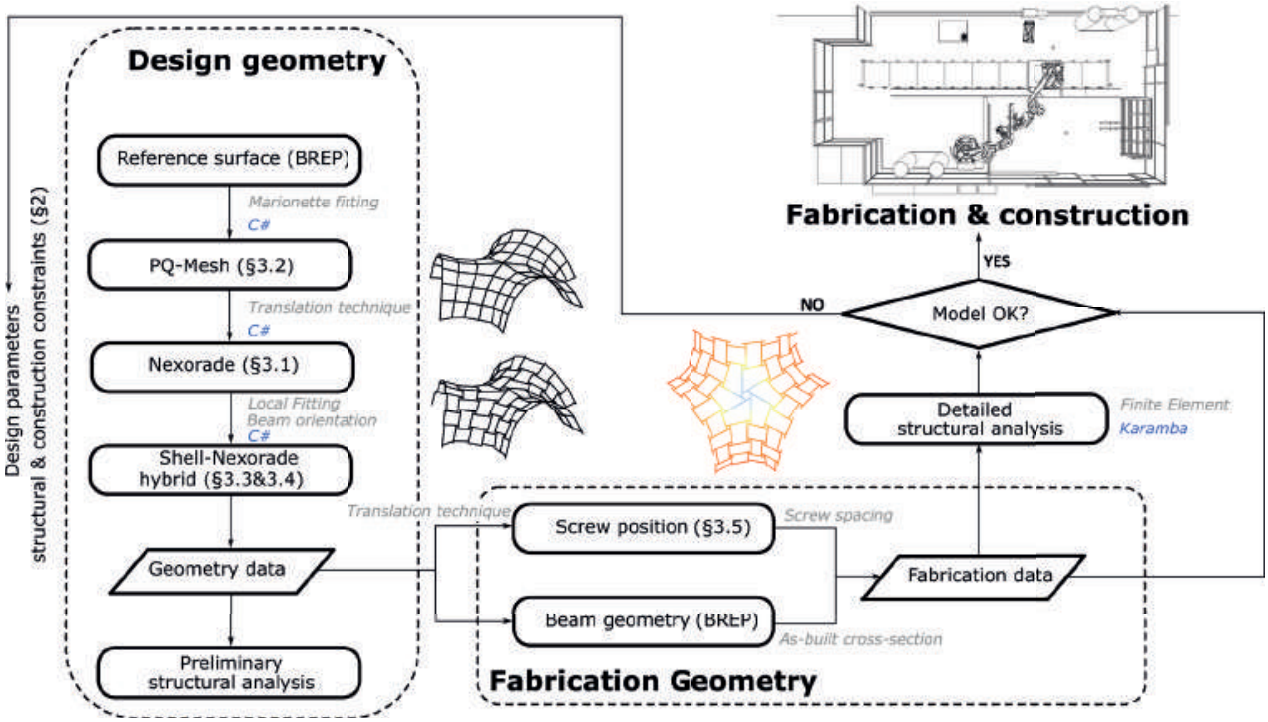


Figure 5: Computational workflow.

3. Construction-aware form finding

3.1 Form finding of nexorades with the translation method

The form-finding of shell-nexorade hybrid is based on translations of the members, as initially proposed by Baverel (Baverel 2000) and as illustrated in Figure 6. The method takes a mesh as input, the edges correspond either to the neutral axis of the member or the member apex. The geometrical object resulting from the translation of the mesh edges is a collection of lines, which are non-concurrent in general. The translation technique is based on the fact the eccentricity between two lines varies linearly with the translation components (Mesnil, Douthe, et al., Form-finding of nexorades with translation technique 2018). As such, nexorades can be form-found by solving linear least square problems, in the manner of what has been done for polyhedral meshes (Poranne, Chen et Gotsman 2015).

The simplicity of the problem is not surprising, although it has not been noticed before: indeed, edge translations appear in polyhedral mesh modelling. Transformations that preserve edge orientation also preserve facet planarity, and create a linear subspace for shape modelling (Pottmann, et al. 2007) (Poranne, Chen et Gotsman 2015). As a consequence, if the input mesh has planar facets, the nexorade created with the translation technique can be covered with nearly planar panels, although the designer has to deal with eccentricities (distance between the non-concurrent neutral axes of the beams).

The translation technique allows to cover nexorades with planar panels, which can be used as a bracing system. The authors call shell-nexorade hybrid the resulting structural system. The practical generation of the structural layout requires first to generate a mesh with planar facets, and then to optimally fit a panel in order to accommodate eccentricities.

3.2 Shape-fitting problem

The input geometry is fitted with the marionette technique, which considers a projection of the mesh as an input, in our case, the plane view, leaving only the altitudes of the mesh vertices as design variables. The technique allows to express the planarity constraint with a linear equation. Additional positional constraints are imposed to some nodes for a better control of the shape, and are also linear.

$$\begin{cases} \mathbf{A}.\boldsymbol{X} = \mathbf{0} \\ \mathbf{B}\boldsymbol{X} = \boldsymbol{C} \end{cases} \tag{2}$$

The matrix $\mathbf{A}$ encodes the planarity constraint for the facets, as described in (Mesnil, Douthe, Baverel, & Léger, Marionette Mesh: from descriptive geometry to fabrication-aware design, 2016). The matrix $\mathbf{B}$ is a sparse matrix, the only non-zero coefficients $\mathbf{B}_{ij}$ are so that the $\mathbf{i}^{\text{th}}$ node has the altitude $\mathbf{C}_j$. The two constraints can be assembled by stacking the matrices $\mathbf{A}$ and $\mathbf{B}$ in columns. The optimization becomes a linearly constrained linear least square problem.

$$\min_{\begin{bmatrix}\mathbf{A}\\\mathbf{B}\end{bmatrix}\mathbf{X}=\begin{bmatrix}\mathbf{0}\\\boldsymbol{C}\end{bmatrix}} (\boldsymbol{X}-\boldsymbol{X_t})^T.(\boldsymbol{X}-\boldsymbol{X_t}) \tag{3}$$

The solution of the problem is classical, but recalled here for the sake of completeness. The constrained problem is easily solved with the aid of Lagrange multipliers $\boldsymbol{\lambda}$: optimal values $\boldsymbol{X}^*$ and $\boldsymbol{\lambda}^*$ satisfy following linear equation.

$$\begin{bmatrix} 2.\boldsymbol{I_n} & \begin{bmatrix}\mathbf{A}\\\mathbf{B}\end{bmatrix}^T \\ \begin{bmatrix}\mathbf{A}\\\mathbf{B}\end{bmatrix} & 0 \end{bmatrix} . \begin{bmatrix}\mathbf{X}^*\\\boldsymbol{\lambda}^*\end{bmatrix} = \begin{bmatrix} 2.\boldsymbol{X_t} \\ \begin{bmatrix}\mathbf{0}\\\boldsymbol{C}\end{bmatrix} \end{bmatrix} \tag{4}$$

In practice, the system is solved by performing Cholesky decomposition of the symmetrical matrix on the left-hand side. Few position constraints are chosen so that the problem is not over-constrained, and the matrix of the left-hand side remains invertible. This guarantees the feasibility of the solution using Cholesky decomposition. The solution is fast, even for large number of facets (Sorkine et Cohen-Or 2004).

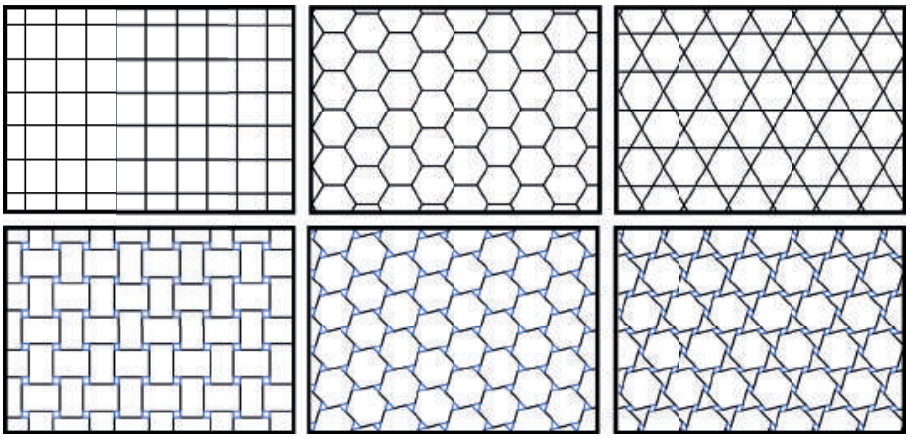


Figure 6: Some planar tiling (top) and associated nexorade patterns created by edge translation (bottom).

3.3 Mesh planarization

The transformation of a mesh into a nexorade introduces eccentricities, as illustrated in Figure 7, where the end the members are highlighted with black dots (notation $\mathbf{P}^{+}\mathbf{i}$) and the corresponding closest point on the attached member is highlighted in white (notation: $\mathbf{P}^{-}\mathbf{i}$). We write $\mathbf{t}\mathbf{i}=\mathbf{P}^{-}\mathbf{i}-\mathbf{P}^{+}\mathbf{i}$.

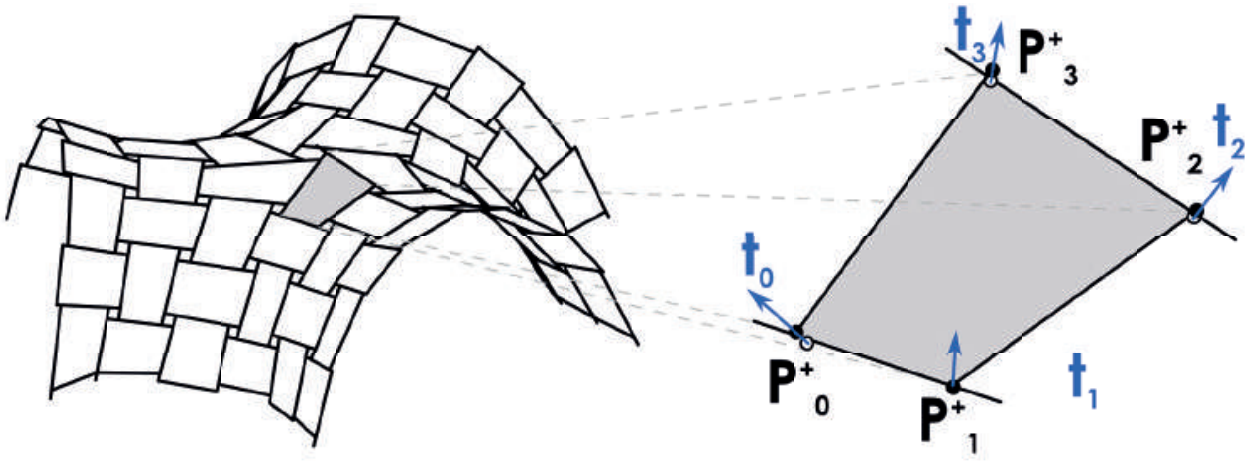


Figure 7: The form-found nexorade as a collection of lines and the local planarization problem

With the geometry of Figure 7, the designer does not deal with a watertight mesh anymore and must fit envelope panels to the beams and wants thus to minimize following quantity:

$$\sum_{i=0}^{N} \|P_i - P_i^+\|^2 + \|P_i - P_i^-\|^2 \qquad (5)$$

With a planarity constraint and the additional design restriction:

$$P_i = P_i^+ + a_i . t_i \qquad (6)$$

This optimization is a specific example of the marionette technique, with non-parallel lines and is also expressed as a linearly constrained least square problem (Mesnil, Douthe, Baverel, & Léger, Marionette Mesh: from descriptive geometry to fabrication-aware design, 2016). Each equation can be solved independently for each facet. This makes the computation extremely fast and reliable.

In order to better understand why eccentricities arise from the transformation into a nexorade pattern when constructing with planar facets,

one can count the degrees of freedom imposed by the planarity and member straightness. We write n_e and n_v the number of edges and vertices in the nexorade, as shown on the bottom of Figure 6. Each edge of a nexorade pattern contains four nodes, except at the boundaries (see bottom of Figure 6), so that $n_v \sim 2n_e$, in addition there are $4n_e$ alignment constraints in the whole nexorade patterns. For two-dimensional nexorade patterns (for example in the XY plane), there are initially $2n_v$ degrees of freedom and $2n_e$ alignment constraints. The estimation of the number of degrees of freedom for nexorade patterns without eccentricities is thus given by equation (7).

$$\begin{cases} d_{3D} \sim 6n_e - 4n_e = 2n_e \\ d_{2D} \sim 4n_e - 2n_e = 2n_e \end{cases} \tag{7}$$

The dimensions of the design spaces are similar, and differ slightly in practice essentially because of "free" borders (where some members have less than four nodes). When adding planarity constraints on the facets (the number of constraints is proportional to the number of facets), one over-constrains the design space of eccentricity-free nexorade patterns and is left only with nexorade patterns inscribed in a plane. The complexity of milling operations mentioned in this article is thus not a limitation of the proposed form finding technique, but rather an intrinsic limitation of nexorades.

3.4 Beam orientation

In timber structures, rectangular cross-sections are commonly used. It is therefore preferable to build torsion-free beam layouts, i.e. to find beam orientation where the beam central plane meet along a common axis. The solution for this problem is not obvious for quadrilateral meshes, but three valent meshes always admit constant face offsets (Pottmann, et al. 2007).

For nexorades, the offsetting problem can easily be solved, because there are only three-valent connections (from a combinatorial point of view), but two-valent connections from a technological point of view: any choice of beam discrete normal yields a torsion-free beam layout. Figure 5 illustrates this statement and the notations for orientation of the beams and panels. The letters Y and Z describe the local material frame corresponding to the strong and weak axis respectively.

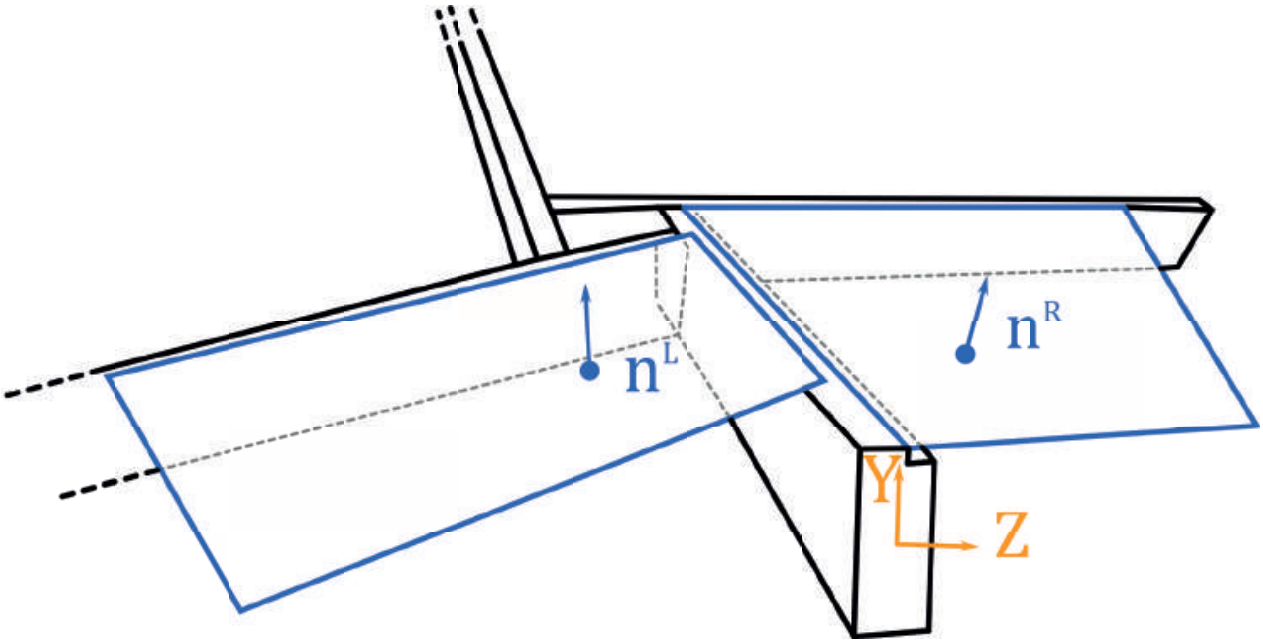


Figure 8: Notations for the orientations of beams and panels.

We can define the normal as the bisector vector between the adjacent faces normal, as shown in equation (8).

$$Y_i = (n_i^L + n_i^R) - (n_i^L + n_i^R).t_i \qquad (8)$$

This choice minimizes the maximal angle between a panel and its supporting beams, which is a constraint in the chosen fabrication set-up. Indeed, as shown in Figure 9, a robot mills the groove for the assembly between beam and panel: large angles can lead to collisions between the tools or robots.

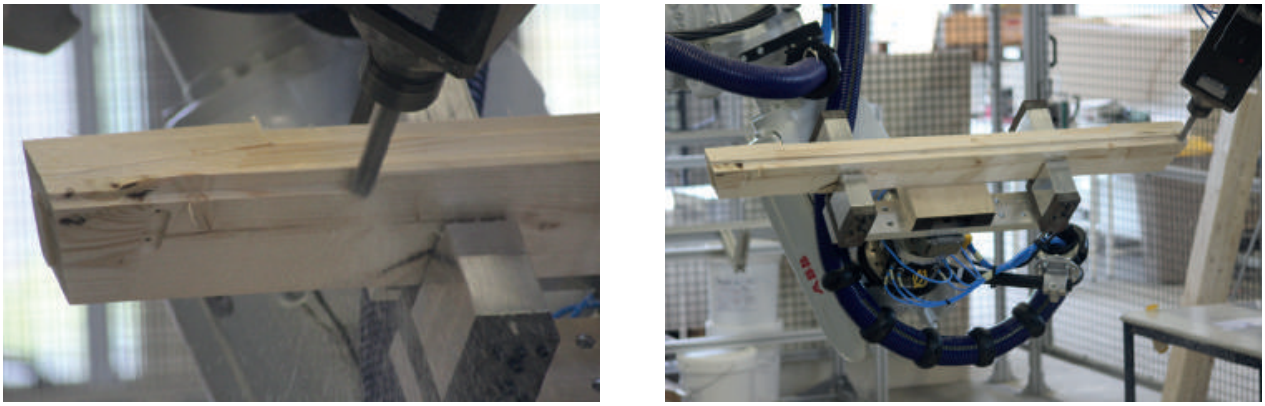


Figure 9: Groove milling for panel/beam attachment.

3.5 Optimization of connection details

The choice of structural connection details depends on mechanics and assembly. End grain screws are used in the pavilion because they are simple to assemble and do not constrain the assembly kinematics. Details using connecting plates in the timber could not be implemented due to the small width of the members (60 mm), but could be used if high structural performance is required. Their main limitation is however the restriction of assembly kinematics, which add an new challenge for the construction sequence planning and execution. Fire safety can also be an issue if the plates are exposed to fire. Glued connections are another alternative with good mechanical performance, but are not suited for in-situ assembly, and do not fit the design requirements of the project. End-grain connections are thus a good alternative to more conventional connections. Moreover, the forces in the connectors are limited in shell-nexorade hybrid, so that yield of connections is a lesser design issue than in classical nexorades, and their lower mechanical performance is not as critical as in other timber structures. Nonetheless, they yield some difficulties for the detailed planning, which are discussed in this section.

The end-grain screwed connections are subject to practical limitations, illustrated in Figure 10: the distance between the screw axis and the beam boundary should remain inferior to 24 mm, while the distance between screws should be superior to 24 mm. For some fans, the engagement length is inferior to the screw length (200 mm), meaning that some collisions between screws might occur: as a result, the position of the screw must be adjusted.

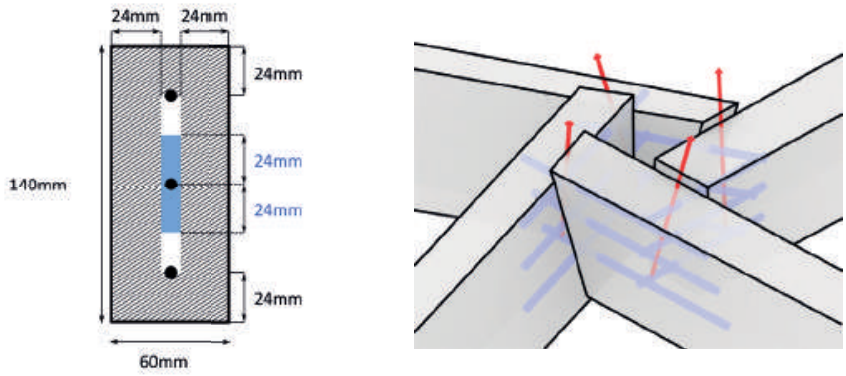


Figure 10: Geometrical constraints for the screws layout. Left: Admissible position for screws (white area), and minimal spacing between screws (blue area). Right: a nexor where the screws are longer than the engagement length and potential intersections between them.

The designer aims at maximizing the distance between the top and bottom screws, as it increases the lever-arm, and thus resistance of the connection detail. This must be done without collisions between the screws. This is therefore a constrained optimization problem: the screw layout must be collision-free and respect the bounding box shown on the left of Figure 10. The parameters of the problem are the position of the neutral axes in the local plane of the in-coming beam: the screws can be moved along the red arrows drawn in Figure 10. The screws have a determined orientation: they are aligned with the beam neutral axis. Just like in the form-finding of nexorades by translations presented in Section 3.2, the distance between two screws depends linearly on the amplitude of the translation. The optimization problem of equation is thus a linear programming problem. Fortunately, this problem is not highly coupled: the optimization problem can be solved for each fan separately with the simplex method.

4. Structural behavior of shell-nexorade hybrids

4.1 Modelling assumptions and design iterations

The structure is modelled with the finite element technique, in order to assess the structural response under non-symmetrical loads. In the preliminary structural design, the cross-section was set to 120 mm × 60 mm, assuming that 20 mm of static height at most would be milled because of the eccentricities. This conservative assumption allowed to quickly iterate over the geometry without calculating the beam cross-section after milling. The plywood plates can be modelled with an isotropic material law with a Young's Modulus of 8 GPa. A linear elastic model is computed with Karamba, a finite element software integrated in Rhino/Grasshopper (Preisinger 2013), and it was checked that every connection detail was safe. This ensures that the pavilion satisfies the ultimate limit state, but a better approximation of the collapse load could be given by yield design theory or by a geometrically and materially nonlinear analysis (GMNA).

Design iterations were performed at the level of the design geometry, without generating the final geometry of the members. It allowed to cre-

ate a shape with a strong curvature that provide geometrical stiffness. It also appeared unsurprisingly that decreasing the engagement length improved the structural response of the structure. A design approach purely driven by the optimization of the structural response would result in reducing the engagement lengths. The engagement lengths of individual fans became thus design parameters once a satisfying geometry was found. The minimization of engagement lengths is limited in practice by two constraints: it widens the bounding box of the panels, and small engagement lengths might result in unreachable areas to insert the end-grain screws. The design iterations had thus to take structural response, fabrication and assembly constraints into account.

4.2 Benefits of shell-nexorade hybrids

We propose to assess the benefits of introducing plates as bracing elements by comparing the performance of the as-built geometry and cross-sections for the shell-nexorade hybrid and a nexorade without panels. We assume that the load apply in the same way to both structures. The displacements are significantly lower in the shell-nexorade hybrid, especially for non-symmetrical wind and accidental loads, as seen on the left of Figure 11. Forces are also significantly decreases: the right of Figure 11 shows the utilization factor of timber under wind load alone, as prescribed by the Eurocode 5 (the material class is GL24h). It can be seen that, even without combination factor, some members of the unbraced nexorade are over-stressed. Under ULS combination *1.35G+1.5W*, the utilization factor can go up to 200%, even without considering reduction factor for long term load ($k_{mod}=0$), which is absolutely not conservative. The utilization factor of the beams in the shell nexorade hybrid is approximately ten times lower. The introduction of plates as a bracing system is thus highly beneficial, since forces and displacements are divided by ten, with an additional mass of 30%.

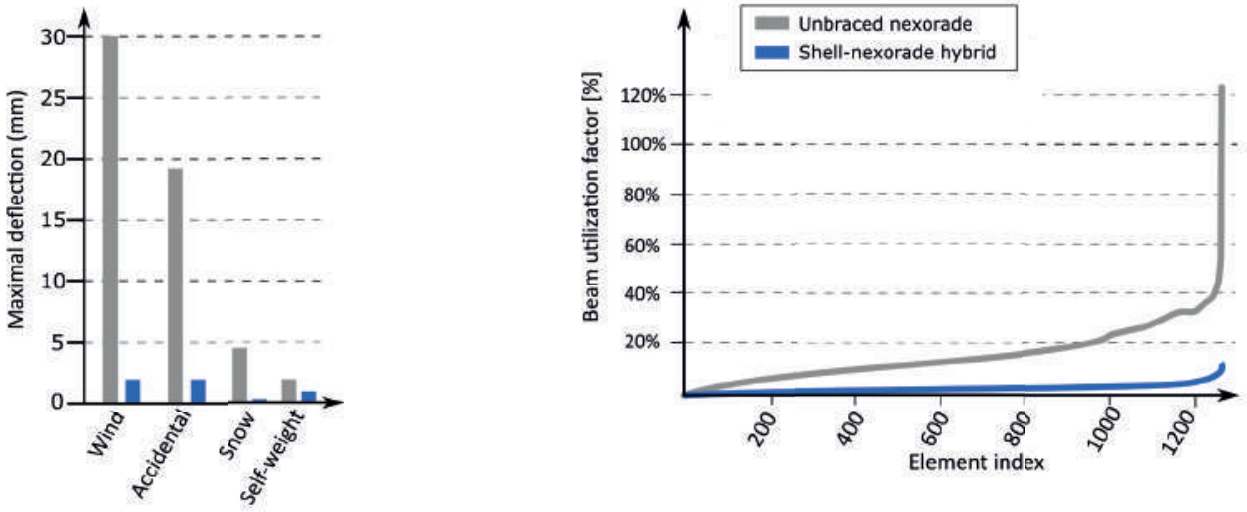


Figure 11: Displacement under different loads (left) and utilization ratio of beams under wind load alone.

4.3 Scalability of nexorades and shell-nexorade hybrid

Different structural systems are available to cover areas with doubly curved structures. This paper focuses on nexorades, or reciprocal systems, and shell-nexorade hybrids. They are easy to assemble and do not require complex connection details. Gridshells are also a classical solution: they are highly efficient, but at the cost of more complex connection details. The relative performances of the different structural systems with respect to the change of scale are assessed in the followings with a simplified comparative study. The geometry of the built pavilion is used as a reference but re-scaled to span larger areas. Then, a sizing optimization is performed under the constraint that the structure satisfies serviceability criterion and ultimate strength criterion, set here to 30 % of the characteristic yield strength. The height-over-width ratio of the beams is set arbitrarily to 2.5, and the plates have a thickness-over-span ratio of 100. The only parameter in the sizing optimization is thus the beam width b.

$$\begin{cases} \min\limits_{\delta_{SLS}<\delta_d} m \\ \sigma_{ULS}<0.3\sigma_k \end{cases} \tag{9}$$

The results of the sizing optimization for different spans are shown in Figure 12. The connections between beams are assumed to be extremely rigid (more than 10 000 kNm/rad), in order to sensitivity to nodal stiffness, an important issue for gridshells. The shell-nexorade hybrid

and gridshell follow the same trend: the weight increases linearly with the span for spans superior to 25 meters, while the nexorades follow a power law and are clearly outperformed by the two other structural systems. Notice that the gridshell is lighter than the shell-nexorade hybrids: the plates represent a significant part of the total weight in shell nexorade hybrids with large spans here. This could be fixed by working on hollow plates, or by using a finer mesh pattern for the shell nexorade hybrid. A precise comparison of gridsells and nexorades should be the topic of a more precise study.

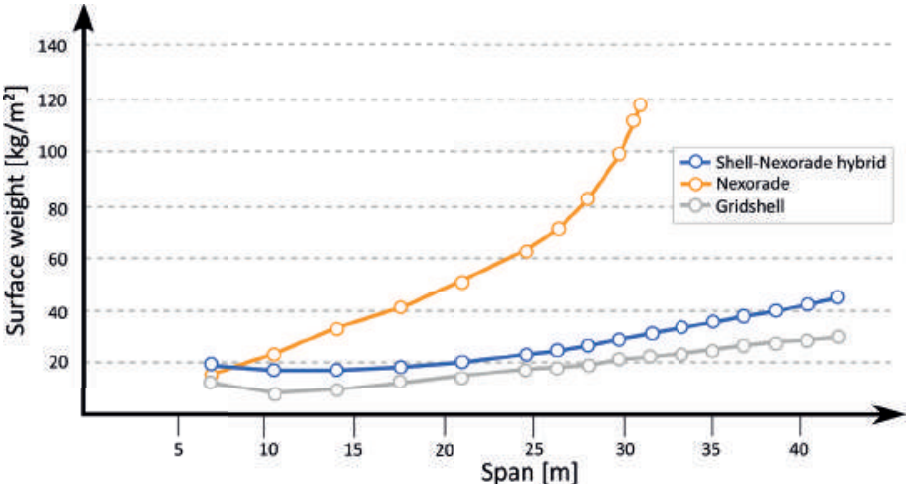


Figure 12: Influence of the span on the weight of different structural systems.

The trends seen in Figure 12 can be explained with simple arguments. First, it should be noticed that for large spans, the governing load case for the nexorade is self-weight, although the shape is close to a funicular shape.

We explain this trend by considering a cylindrical vault of radius R under uniform load p, proportional to the self-weight. We write b,h the width and height of the beams, E the Young's modulus and ρ the volumic mass of timber. An equivalent in-plane membrane stiffness $\mathcal{A}$ can be computed with homogenization techniques, as already done in (Mesnil, Douthe, Baverel, & Léger, linear buckling of quadrangular and *kagome* gridshells: a comparative assessment, 2017). It has already been observed that the membrane stiffness is proportional to the flexural rigidities of the members, which depends on the I_2 and their length L, defined as $L * N = R$, where N is the number of subdivisions.

$$\mathcal{A} \propto \frac{EI_2}{L^3} = \frac{Eb^3hN^3}{12R^3} \tag{10}$$

The applied load is proportional to the mass:

$$p \propto \rho bh \tag{11}$$

The meridian force T is given by the classical formula $T = pR$, so that the strain ε is simply given by:

$$\varepsilon = \frac{\delta}{R} = \frac{T}{\mathcal{A}} \propto \frac{\rho R^4}{Eb^2N^3} \tag{12}$$

The value $\frac{\delta}{R}$ should be limited by a SLS criterion, for example $\frac{\delta}{R} < \frac{1}{300}$. This imposes a condition on the beam width. The proportionality laws derived previously allow to affirm that there exist a factor k so that:

$$b > k\sqrt{\frac{\rho}{EN^3}}.R^2 \tag{13}$$

The height h being proportional to b by hypothesis, one can thus conclude on the trend of the optimal weight for a nexorade that satisfies serviceability under self-weight.

$$p_{opt} \propto R^4$$

The power law observed in Figure 12 does therefore find a simple analytical explanation. However, we observe a strong dependency of the result with respect to the number of subdivision in equation (13). The study presented in this section should thus be further extended to evaluate the influence of subdivision.

The membrane stiffness of gridshells and shell-nexorade hybrids is proportional to the cross-sectional area, which itself is proportional to the surface weight. Thus the sizing optimization under self-weight is scale-invariant, and gridshells and shell-nexorade hybrids are sized with respect to out-of-plane loads.

5. Conclusion

Nexorades are structures based on an elegant assembly principle that generally suffer from poor structural behavior. Introducing planar plates as a bracing system opens a new potential of application for nexorades. This paper illustrates the possibilities offered by shell-nexorade hybrids and practical implications of using this new structural principle for fabrication and geometrical modelling. Several optimization problems must be solved to guarantee facet planarity, structural reliance and constructability. The understanding of geometrical properties of nexorades is fundamental in the design workflow. The numerous design iterations are made possible by the flexibility, robustness and speed of the proposed framework and by the handling of geometrical representations of increasing complexity.

A full-scale timber pavilion, shown in Figure 13, was built to validate the methodology proposed in this paper. The structural calculations show that the plates multiply the stiffness by ten with a mass increase by 30 %. The pavilion is checked as a temporary building with building codes and technical agreement, so that despite innovations on form finding and fabrication, it has the potential to be proof-checked by an independent engineer. The robotic fabrication within tolerances allowed the manual assembly of the structure with minimal difficulties. Shell-nexorade hybrids combine thus the ease of assembly of nexorades with the stiffness of ribbed shell structures.

Figure 13: A view of the completed pavilion.

References

Baverel, O. (2000). *Nexorades: a family of interwoven space structures*. University of Surrey.

Bletzinger, K.-U., Kimmich, S., & Ramm, E. (1991). Efficient modeling in shape optimal design. *Computing Systems in Engineering*, 2(5–6), 483–495.

Bowie, T. (1960). *The sketchbook of Villard de Honnecourt.*

Brocato, M. (2011). Reciprocal frames: *Kinematical determinacy. International Journal of Space Structures,* 26(4), 343–358.

(n.d.). *European Technical Approval ETA-11/0190.*

Flöry, S. (2017). Goat. Rechenraum e.U., Vienna.

Kohlhammer, T., Apolinarska, A., Gramazio, F., & Kohler, M. (2017). Design and structural analysis of complex timber structures with glued T-joint connections for robotic assembly. *International Journal of Space Structures,* 32(3–4), 199–215.

Mesnil, R., Douthe, C., Baverel, O., & Gobin, T. (2018). Form-finding of nexorades with the translation method. *Automation in Construction*, in press.

Mesnil, R., Douthe, C., Baverel, O., & Léger, B. (2016). Marionette Mesh: from descriptive geometry to fabrication-aware design. *Advances in Architectural Geometry* (pp. 62–81). Springer.

Mesnil, R., Douthe, C., Baverel, O., & Léger, B. (2017). linear buckling of quadrangular and *kagome* gridshells: a comparative assessment. *Engineering Structures*, 132, 337–348.

Poranne, R., Chen, R., & Gotsman, C. (2015). On linear spaces of polyhedral meshes. *IEEE transactions on visualization and computer graphics, 21*(5), 652–662.

Pottmann, H., Liu, Y., Wallner, J., Bobenko, A., & Wang, W. (2007, August). Geometry of multi-layer freeform structures for architecture. *ACM Transactions on Graphics, 26*(3), 6.

Powell, M. (2007). A view of algorithms for optimization without derivatives. *Mathematics Today-Bulletin of the Institute of Mathematics and its Applications, 43*(5), 170–174.

Preisinger, C. (2013). Linking structure and parametric geometry. *Architectural Design, 83*(2), 110–113.

Schwartz, T. (2012). *Rob| Arch* (pp. 92–101). Vienna: Springer.

Sorkine, O., & Cohen-Or, D. (2004). Least-squares meshes. *Shape Modeling Applications* (pp. 191–199). IEEE.

Liquid rock – Agent based modeling for concrete printing

Eftihis Efthimiou, Georg Grasser

Eftihis Efthimiou
eftihis@decodefablab.com
Faculty of Architecture, University of Innsbruck, Austria, Decode Fab Lab, Athens, Greece

Georg Grasser
georg.grasser@uibk.ac.at
Faculty of Architecture, University of Innsbruck, Austria

Keywords:
Concrete printing, computational design, agent-based modeling, morphogenetic design, optimization

Abstract

This paper focuses on the computational design and optimization aspects of a large scale, realized robotic concrete printing project, as well as material behaviour and robotic fabrication layout. The project has been fully realized within Grasshopper3d, from design to fabrication control, with the use of several plug-ins, such as Anemone, Boids library and Taco, and a series of custom components created specifically for this purpose, through light scripting. The core of it is an agent based variable growth algorithm, optimised for concrete tolerances, fall-off angle and weight management, as well as the robot's information management threshold.

The paper presents the design methodology as well as the optimization techniques embedded within the morphogenetic process of the objects. In addition to that there will be given justification for the choice of an agent based modeling approach in contrast to a physics simulation process that was initially used, with a comparative assessment of the two methods. Finally, a series of other issues will be discussed, from material tolerances and fabrication optimization, to real-time visualization methods for the geometry.

1. Introduction

Over the last few years additive manufacturing with concrete has continuously increased in size and applicability (Lim S. et al, 2012) but still remains a highly complex fabrication process. Among a variety of potential materials, concrete has emerged as one of the most promising materials for automated layered depositing fabrication at the building scale. Due to its weight there is a major impact on the stability (Khoshnevis et al, 2006) of the object-in-print but also the issue of printing cantilevers should be addressed. The necessity and opportunity for design related aspects is demonstrated in the establishment of several new companies (Dini, 2018, CyBe-Consturuction, 2018, XtreeE, 2018, incremental3d, 2018). However the presented project does take full advantage of the articulation abilities of six-axis industrial robots and implies the possibilities of differentiated material organizations that allow for performance-oriented construction methodologies (Oxman et al, 2011). Apart from that, it is an exploration of the potential for formal expression that concrete extrusion may unlock, through the use of self organizational design strategies.

Figure 0: Exhibition: LIQUID ROCK (by Marjan Colletti, Georg Grasser, Eftihis Efthimiou, Alexander Karaivanov, Javier Ruiz,Institute for experimental Architecture, REX/LAB, University of Innsbruck, The Bartlett School of Architecture, UCL, BAUMIT GMBH) January 24–March 3, 2018, GALERIE GÖTTLICHER, Krems, Austria. Photo: (Philip Super).

2. Design strategy/growth algorithm

With the democratization of robotic fabrication, various approaches have been implemented for the layout, but also for the creation of paths for the robots (eg. Bhooshan et al., 2018; Rust et al., 2016; Vasey et al., 2015). Each approach describes a different level of awareness of the overall design, on the level of the algorithm, that signifies an equivalent methodology for optimization.

The design paradigm followed for this project is Agent Based Modeling. Agent-Based Modeling is the modeling of phenomena as dynamical systems of interacting agents (Castiglione, 2006). Every calculation of the system is being performed on an agent level, and the agent is not aware of the system. Instead, the gnosis of the agent lies on a lower level of interaction. In this type of design systems, any type of complex behaviour emerges from the interactions of the agents on a local scale. This approach has been integrated in architectural design (e.g. Stuart-Smith, 2016), for both the morphogenetic and optimization potential it provides.

More precisely, in the center of the project lies a variable growth algorithm, developed for Grasshopper, with the use of Anemone (Zwierzycki, 2015) and Boid Library (Pernecky, 2015).

2.1 Morphogenetic strategy

The process starts with the design of a base curve. The curve then gets divided in a number of points, such that they maintain a set distance between them. Every single one of these points is trying to avoid its neighboring points, by aiming to maintain a distance that is incrementally larger than the current one. To achieve this goal, for every single point, the neighbours within a specific range (the aforementioned distance) are retrieved and their average point is calculated. The point then moves away from it, relative to the distance that it is trying to maintain. After this movement, a new curve is interpolated over the updated points and the process is iterated, with the initial distance value. The growth, at this point, occurs from the imbalance that occurs between the distance between the points and the repulsion radius. At each iteration the total length of the curve increases, resulting in an ever increasing number of division points.

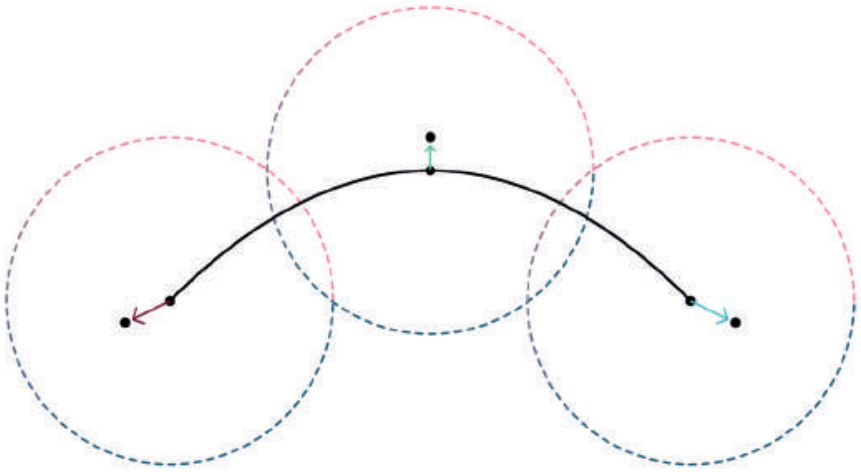


Figure 1: Agent repulsion, relative to distance.

Since the growth is generated on an agent level (agents being the points from the subdivision), the same logic can be applied to multiple curves. In order for that to work, we need to be calculating the repulsion amongst all points, following the curve division, regardless of which initial curve they stemmed from. After the vector operation is completed, the

agents are placed back in lists as per their parent curves, in order for the new curves to be interpolated. This allows for different curves to be interweaving along their growth, so that visual continuity is maintained.

Although initial distance between the points is fixed, the search radius for the repulsion may differ from point to point. This leads to a variable growth rate along the curve. The variation in search radius is controlled by a scalar field, created either by an attractor system or by a noise function. Whenever the curve enters the “stronger” areas of the scalar field, it grows more wildly, whereas inside the “weaker” areas it grows less vividly (or even not at all, should the repulsion radius reach values below the division distance). This allows for control over the directionality of the growth and the overall volume placement of the structure.

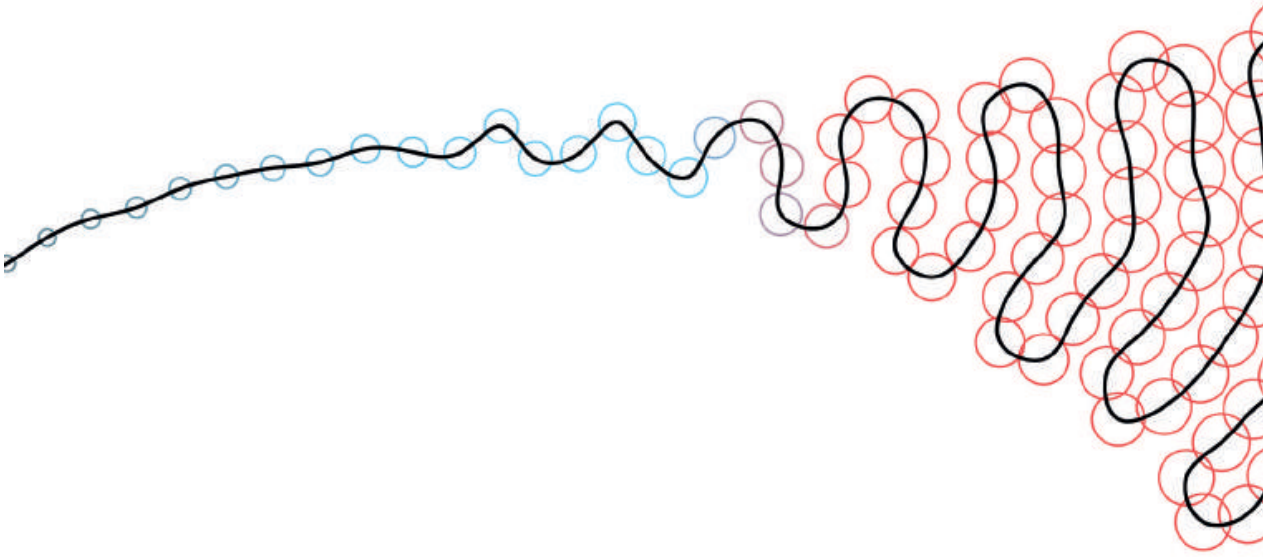


Figure 2: Variable repulsion distance.

Along with the growth movement, each point is also moved vertically by an increment. Each new curve that is formed, lies on top of the previous one. The aggregation of these growing curves creates the general structure and also describes the morphogenetic process of the object. The increment by which the agents are moved vertically amounts to the layer height of each printed layer. By using a scalar field, that controls elevation, variable layer height can be achieved.

2.2 Print optimization and stability

Each iteration of the curve growth is being used as a print layer path. To achieve a continuous printing path, the final 10 orientation planes from each curve are tweened between the current curve and the next, so as to gradually cover the ground between the two curves. In order to achieve that, it is crucial that the seam of each curve is placed at the closest point along said curve to the seam of the previous one. However a seam is still visible after this process (**Fig. 3**), mainly due to the extrusion speed, which needs to be specifically adjusted for the seam. This problem can be also solved by using an alltogether spiral print path, by tweening the elevation value of every point along a curve. However, taking into account the variable layer height and the diverging print plane per layer, the tween solution appears more straightforward.

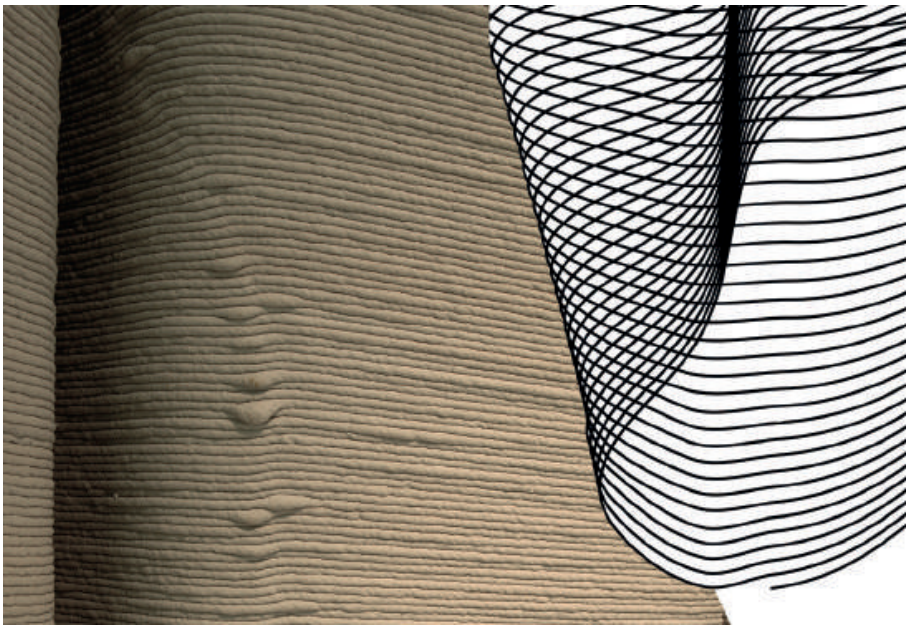


Figure 3: Seam management and resulting imperfections.

Given that the fabrication method is Fused Deposition Manufacturing, we needed to maintain control over the overhang angle per print layer, in order to ensure stability during printing and to avoid drips. In this domain we need to take into consideration that the use of supports was not an option. This situation called for an optimization step that was integrated within the morphogenetic scheme of the object, so that the object never exceeds a maximum overhang angle value at any single point. In order to solve that, we used a trigonometric function, that multiplies the elevation

value (ie. the distance between the curves at any given point) with the desired overhang angle tangent. This yielded the maximum length that the movement vector of the agents may attain, without counting in the elevation. From that point, it was quite easy to have each movement vector cap out at this length, with the use of a simple conditional. For the object fabricated, the overhang value was set to 45o. However, we were able to achieve much greater overhang angles, during our trials.

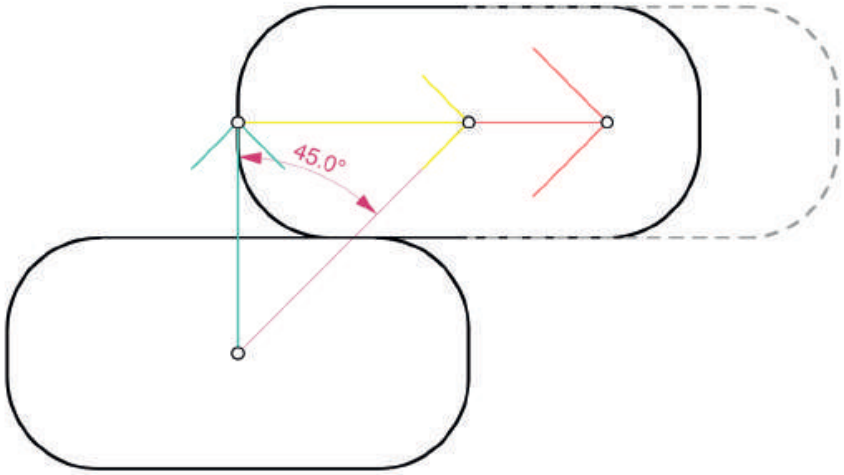


Figure 4: Overhang angle thresshold.

Another issue that had to be addressed was overall stability of each geometry, both as a final object, as well as during every stage of the printing procedure. In order for the objects to stand, the center of mass needs to be within the base curve, ie the initial curve. To ensure that, the average point of each layer was calculated, as well as the average point of every point in each geometry, up to the current iteration. Both points where then projected on the base curve's plane, along the gravity vector. Should any of the projected centers of mass prove to be outside of the confines of the base, the code would exit the loop and a warning would be given. Taking into consideration that the base curves need not necessarily be convex, during the test period the convex hull of the base curve would be used. However, for the final objects the actual base curves where used.

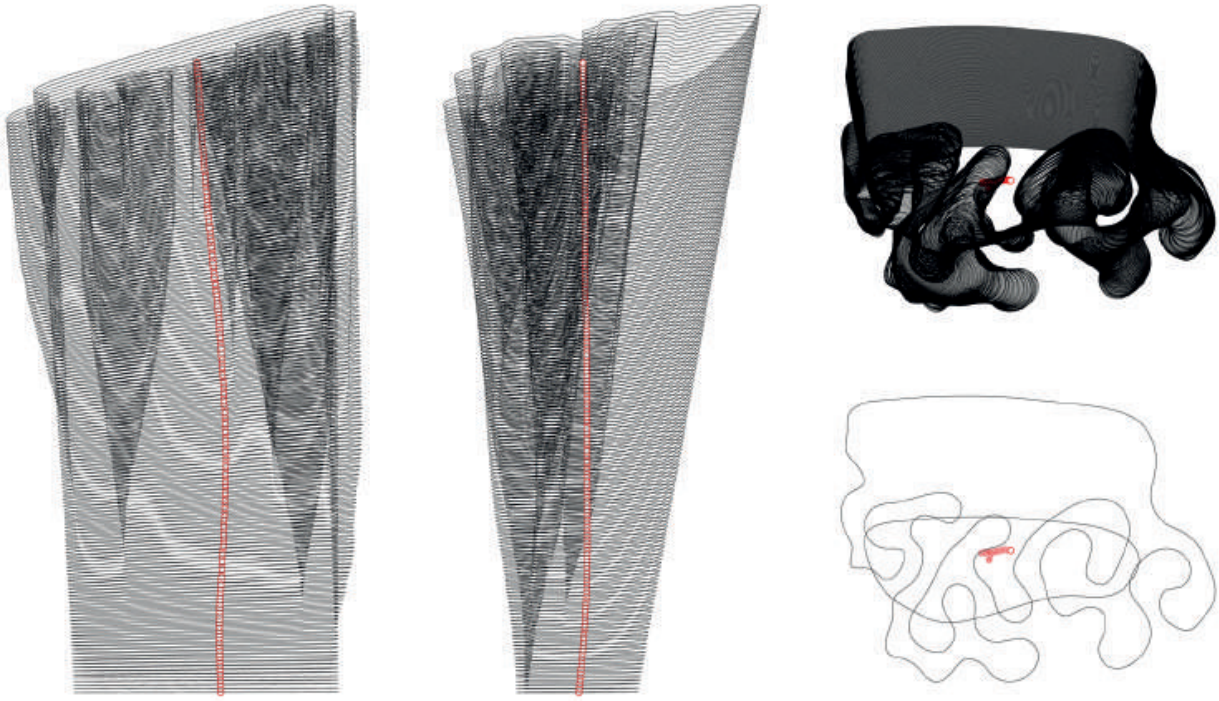


Figure 5: Center of mass calculation, per layer and as a whole.

Since in the final objects growth was oriented solely towards the center of the installation, it is only logical that this would lead to an uneven distribution in weight, that would work towards tipping the objects over. To counteract this unilateral growth that was driving the center of mass away from the center of balance, a second force was implemented, that was pulling the geometries away from the center of the installation. This force, although it remained unnoticed and was counteracted by the growth in the more volatile area of the geometry, created an inclination at the backside of the objects and provided a comprehensive method for any center of mass correction necessary.

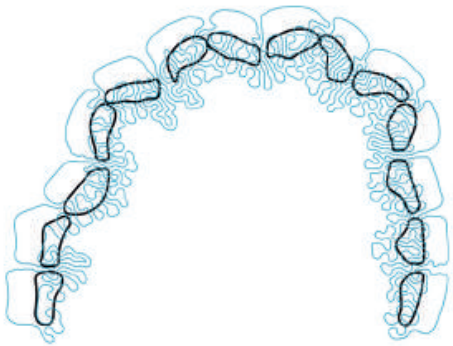


Figure 6: The initial curves are in black, the final in blue. The amount of backwards inclination needed for center of mass correction is visible, here.

2.3 Processing data

A major hurdle we had to overcome was the data processing capability of the robots. It appears that the robots have a cap on their processing capabilities, due to a limited memory, for loading and running printing paths. This translates as an upper threshold on the amount of orientation planes, that is to say the amount of points, that the robot can be programmed to pass through during a print. This attribute indicated towards the resolution of the final objects and, alongside printing time, weight management and material issues, has been crucial to the design's formation. Our early designs would amount to upwards of 250 000 orientation planes, which would translate to a significant amount of path curves per object, that had to be loaded individually. We strove to have each of the geometries be comprised of no more than 70 000 orientation planes, which translates to a more manageable amount of paths per object, that had to be individually loaded. We aimed for the same amount of general volume covered as in the initial tests. Towards that goal, we had to work in two directions. First, we managed to optimize the design itself by increasing the distance between the agents and by adjusting the repulsion radius accordingly, something that, generally, increased the size of the folds. Also, we exaggerated the variability of the growth rates, resulting in areas that would almost not be affected at all by the growth.

Secondly, we followed a post-processing optimization protocol, where we removed the less significant agents for the curves, in terms of curvature. This was made possible by merging points that had an angular deviation from their neighbors under a specific threshold. The aim for that implication was mainly to drop the point count on the smoother areas on the curves, where there was minimum to no growth. For this reason we used a custom function that worked on the level of the agents and thus does not affect the curvature degree of the curve. Even though the "Curve to Polyline" command that is built-in in Grasshopper, and could also be used, is very effective in decimating the smoother areas of the curves, it tends to increase the point count at the areas of the folds, where we can find bigger curvature values. That is to be expected, since curvature cannot be represented in curvature degree 1 and has to be approximated, instead, with many short polyline segments. This process led to a decrease at the scale of 8–12 % in the total amount of the points, without any major impact on the appearance of the curves. The mass of the objects is directly linked to the length of the paths used.

2.4 Visualization

An intricate issue that had to be addressed during design was how to visualize the outcome of the growth algorithm in real time, in order to assess the results both in terms of aesthetics and optimization. Given that the curves generated tended to stack up extremely densely, the display of the curves alone could not give an adequate overview of the general geometry. In terms of appearance, piping the curves works quite nicely towards depicting the final outcome, since, apart from wrapping geometry to allow for shading, it generates the stratified look of the extrusion layers. Since the number of control points per curve was rather large, a simple NURBS piping was not an option, due to performance issues. For a curve of 1 000 control points, the amount of time required for a pipe measures upwards of 4 seconds on a performance pc, and the definition would have to work for 14 curves in parallel, that would grow for a total of 168 iterations. An alternative to that was using a Mesh pipe. This was a faster procedure by the tenfold, but although it would not crash, it would fail to generate a valid mesh pipe at some curves and would sometimes show inconsistent normals at the seam of the curves, when the point count was too high. However much faster this method may have been, in order to visualize the procession of the growth in real time and to avoid weighing down on our definition, in the end we put the piping approach to the side and opted for a custom lineweight & color approach. We used a color value, taken from a color gradient, relative to each curve's print layer index, and an absolute thickness value, relative to median layer thickness. Although this did not generate any shading, it created an over-stylized, comprehensive display of our geometry with the minimum amount of computational strain. Since the actual curves that are being generated are used and no extra geometry is being created, this is the lightest approach. It has to be stated that this option was only used for the real time assessment of our geometry, whilst the code was running.

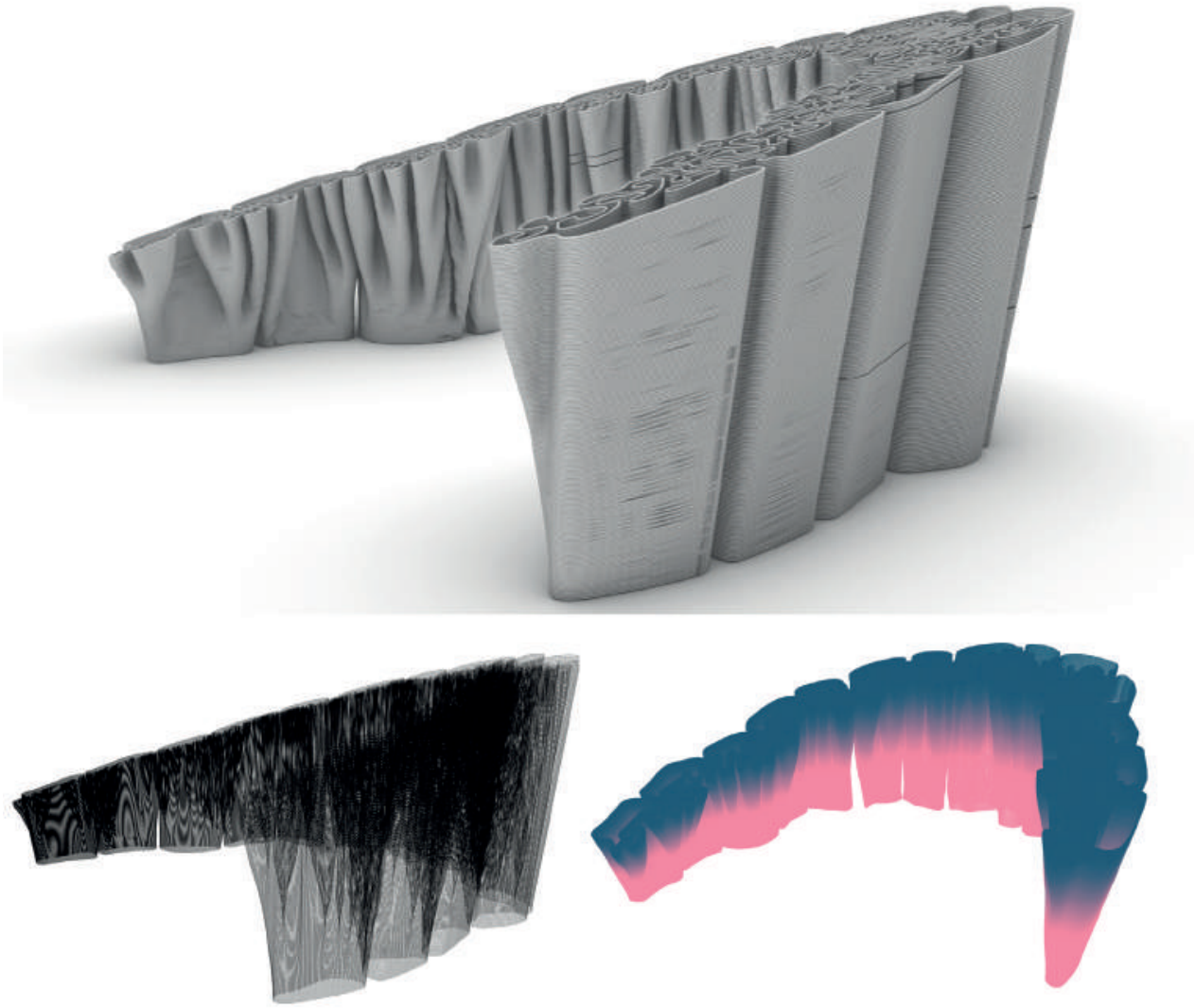


Figure 7: Comparison of the three visualization strategies. Top: Mesh piped & rendered. Bottom left: vanilla curves. Bottom right: Color gradient and thickness.

3. Agent based vs physics approach

During the early stages of design, we utilized a physics-based pipeline to simulate growth, with the use of "Kangaroo Physics 2.42" (Piker, 2017). This would entail subdividing a base curve into a series of segments of a set length, which were set as springs trying to attain a length incrementally larger than their current one, and by using sphere colliders, with a radius incrementally larger than half the subdivision length, so as to avoid overlap and drive the points away from their neighbours. This process was then iterated, by further subdividing the line segments, when they would reach more than twice the set length, resulting in growth.

3.1 Comparison

In general, the physics approach proved to be more computationally extensive, resulting in an all around heavier definition both to run, and to tweak. After each iteration, the process would jitter the list of lines, demanding reordering for seam continuity, in order to achieve a spiraling, continuous extrusion path. A means of re-ordering the lines is joining them and then exploding the resulting curves. Since the whole logic of the definition works on a line basis, the resulting curves were in fact polylines, or of curvature degree 1, in contrast to the smooth, degree 3 curves of the agent based pipeline. Taking into consideration the angle and length management goals that Kangaroo is providing us with, it was easier to optimize the definition for a better "length/ area covered" ratio, by normalizing the length and angle values for each line segment. However, it was way more difficult to achieve variable layer height in this manner, as well as setting an overhang angle threshold, and next to impossible to achieve variable growth, since "Sphere Colliders" only calculates collisions between equally sized spheres. Another issue that remains, is some gaps that appear in the growth, since most of the curves would increase in size in every generation, but would cross the subdivision threshold every "n" generations. This would lead to a periodic increase of the growth rate, that would create areas that would easily missprint.

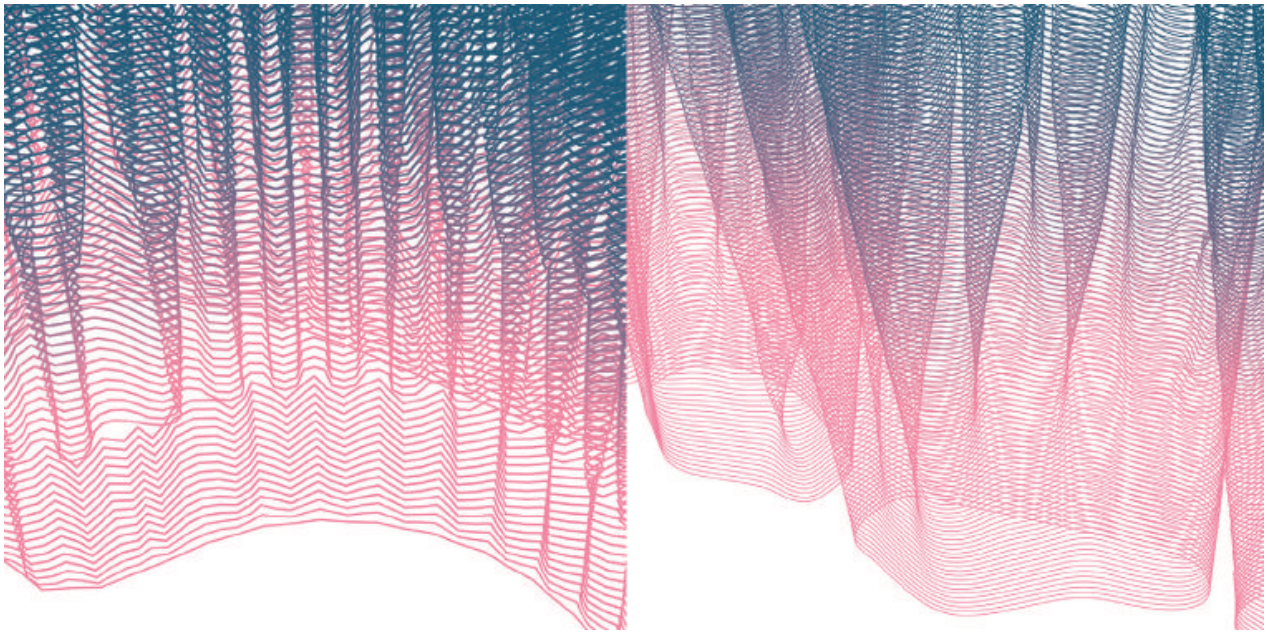


Figure 8: Close view of the inconsistencies, in the physics approach, in contrast to the smoother agent based approach curves.

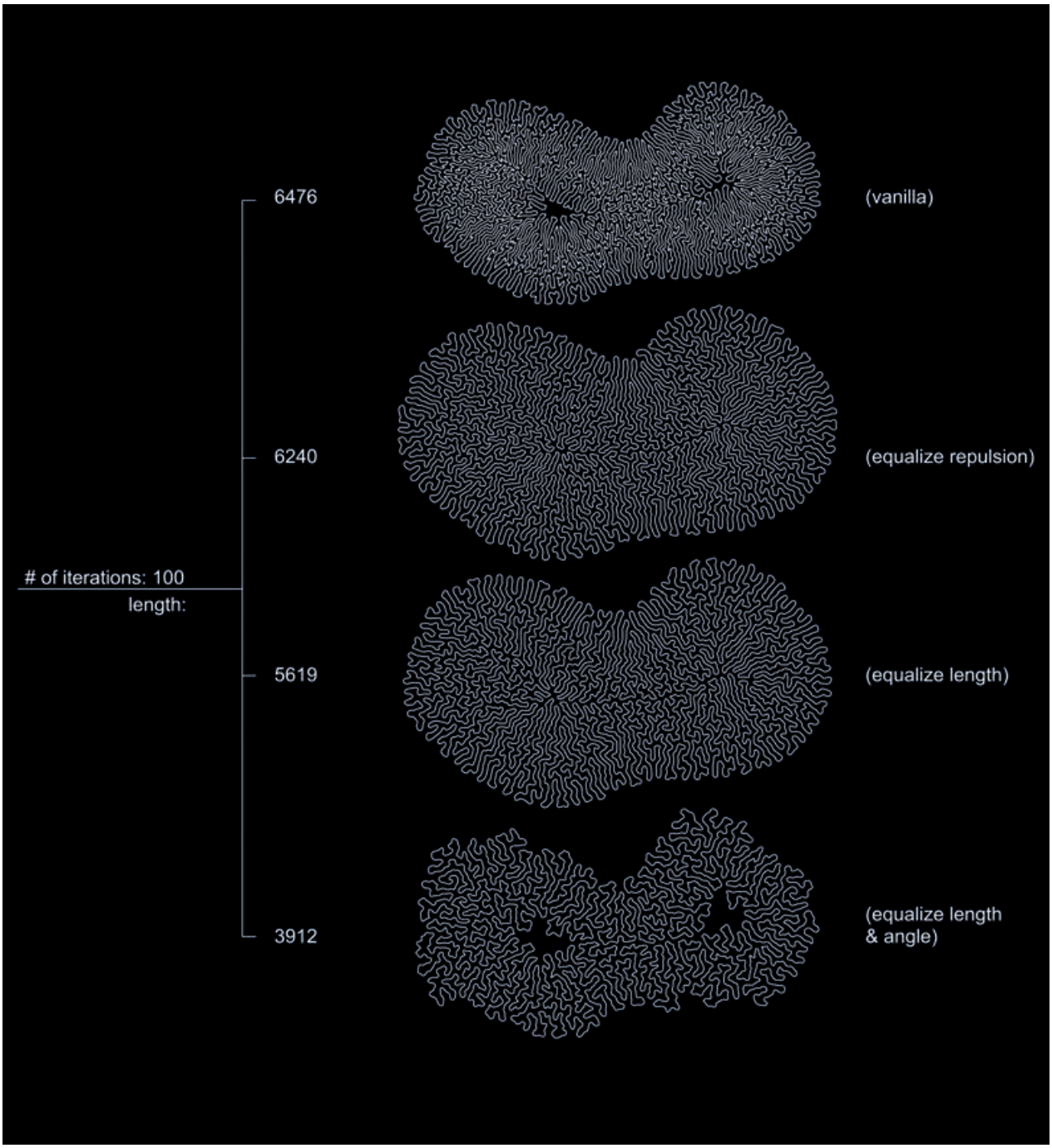


Figure 9: Path length optimization analysis in the physics based approach. The length is shown in arbitrary units.

An agent based approach, as was described, proved to be by far and large lighter computationally, easier to write and optimize, more flexible to support variability and locality and, all in all, more intuitive. The latter is a very important factor, taking into account the fact that the project was part of a university class and, hitherto, had to be assessed also didactically.

4. Additive layered deposition layout

Our material extrusion process directly positions material that is mixed and ready to cure. Currently we are able to choose between 3 different 3d printing materials that slightly vary in performance and color:

Labeling	Max. Grain Size (mm)	Spez. Weight (kg/m³)	Compressive Strength (MPa)	Bending Tensile Strength (MPa)
PrintCrete 230	2,0	1.850	25	8
PrintCrete 260	2,0	1.950	55	8,5
PrintCrete 230 white	2,0	1.850	25	8

Although many cement products do already include catalysts that speed up the process of curing, the necessity to add high impact catalysts allows curing reaction within several seconds which demands higher timing precision and challenges stability during printing (Lloret E. et al. 2017).

Our extrusion tool is mounted on a 6-axis industrial robot system with a maximum reach of 2.55 m but using different robot models in size and speed doesn't influence the general settings of printing.

In context of concrete extrusion, the system has to meet high demands in mechanical stability as high pressures within the extrusion system require solid tooling properties. Initial toolsets made with plastic rapid prototyping immediately indicated the inevitable change to metals. The end-effector consists of three essential elements, the extrusion pipe, a mixer and an injection nozzle for the catalyst. Working with rapid hardening concrete mixed concrete requires the end-effector to be able to easily clean, quickly assemble and disassemble in order to guarantee longevity and avoid that concrete cures within the system.

The printing process requires a concrete pump as well as a specific pump for the catalyst that both continuously deliver material to the extrusion tool itself. The amount of material per time period determines the flow rate which has a crucial impact to the whole process itself. Low rates of flow are required in order to exactly deposit material and allow initial curing of previous printed morphologies. Ideal settings extrude lengths of approximately 15 m/min and 2.5 kg/min of concrete. The generated paths are highly connected to the material properties and the

designed morphology and challenge curing time, the flow-rate as well as its own weight.

Given that layer height spans from 4 mm to 12 mm, we had to sub-divide geometries into four groups that amount to an equal number of consecutive domains of layer height. We then assigned a different feed rate to each layer height domain, in order to achieve the appropriate variable layer thickness for every layer height. We were able to move from the continuous nature of the variance in layer height to a discrete variation in feed rate due to the forgiving nature of the material, with only a fluctuation in layer width. A base speed was set, that was only slightly tweaked manually during the print, to ensure smooth deposition of the material. Based on the above, we could measure the projected weight of each object rather accurately and incorporate that information into the selection process of the final objects.

Name of Part	Length of Path (m)	Weight (kg)	Time of Printing (min)
P1	535	124	29
P2	554	129	30
P3	707	221	39
P4	639	199	36
P5	583	237	32
P6	793	354	44
P7	627	305	33
P8	665	335	36
P9	686	355	38
P10	640	309	34
P11	687	336	38
P12	696	323	37
P13	697	308	40
P14	571	236	31
TOTAL	9080	3771	497

Figure 10: Fabrication layout.

5. Results

The agent based approach proved to be efficient in the local optimization of the geometry, in a manner such that translates and generalizes to the global. In reality, the structural optimization of the form is embedded in the morphogenetic scheme of the object. The user retains control through the ruleset for local interaction, but also through the use of intricate scalar fields that allow for targeted growth. The design strategy resulted in an intricate, highly complex form that satisfied construction needs. This justifies agent based modeling as a fabrication informed modeling strategy for concrete extrusion.

During the early stages of construction we opted for a total of 7 objects. This amounted to a longer printing time, more printing paths per object that had to be streamed individually and a heavier final object. Taking into consideration that concrete extrusion is not yet a fool proof procedure, the longer printing time both translated to a higher margin in which some printing error could occur (mostly mechanical reasons), but also, in the case of an error, more printing hours and actual material would be wasted. The number of successive print paths per object is also relevant.

At the small time interval that occurs between two paths, both pumps, the concrete and the catalyst, stop flowing. At this point there is an increased risk for a clog, if the remaining reagents in the extruder do not get properly removed. Given that a clog can not just stop the extrusion, but can also drastically alter the viscosity of the mixture, this may be destructive for the entire print. For this reason, we opted for a solution with the minimum amount of print paths used per object.

Finally, the weight of the resulting object proved to be crucial for handling, transportation and logistics in general. For all of the above, we decided to split the initial curves in half and construct a larger amount of smaller, more manageable objects.

The installation consisted of a total of 14 objects, with 168 print layers each. The resulting height of the objects spanned from 67 cm, at the lowest top point of the shortest object, to 198 cm for the highest point of the tallest object. The print paths, as was mentioned, incorporate variable layer thickness, with the variability occurring even along the same layer. Each printing layer is generally planar, however the printing plane is not always horizontal; it instead diverges from the z axis, to up to 17 degrees. The outcome, as can be seen in figure 10 is a smooth, continuous print,

with the continuity of the overall form extending to the striation of the print layers in each separate object. This occurs because all of the curves were grown concurrently. This may be a problem if we want to add an extra object at a later time, which evidently will not be able to grow together with the initial objects but will at best grow around them. More importantly, though, for the aforementioned reason, the process cannot be segmented, to cater for an indefinite amount of initial curves or an immense level of resolution.

6. Outlook

Concrete printing offers a new way to think of concrete elements without molding even with complex geometries, and therefore huge savings in resources and costs. While many new forming paradigms, may not fit within existing production lines for now, there is a lot of opportunity in the further development of processes, the engagement within a larger system as well as architectural design possibilities. Robotic additive manufacturing allows generating differentiated local qualities without additional effort and material waste, it can specifically be designed towards structural, energetic or even visual qualities (Dillenburger & Hansmeyer, 2014).

Through the creative act of encoding behavior, capacities, affordances and material constraints, resulting forms are often organic in appearance and resemble the results of a physical topology optimization process suggesting that there may be structural implications of the material behavior and therefore enable performance-based material optimization.

Emerging from a custom workflow between the physical and the digital, our approach widens the general design spectrum and offers enormous architectural potential through specific design methodologies. These digital tools require analyzing methods, generative design approaches and the implementation of robotic as well as fabrication simulation tools to make a serious push towards applicability of concrete based 3D printing for the construction industry in order to benefit the discipline of architecture.

Figure 11: View of the installation. Photo: Nikolaus Korab.

References

[1] Bhooshan S. et al. 2018. "Equilibrium-Aware Shape Design for Concrete Printing". In: De Rycke K. et al. (eds) Humanizing Digital Reality. Springer, Singapore

[2] Castiglione F., (2006) "Agent based modeling". Scholarpedia, doi:1(10):1562.

[3] CyBe-Construction. 2018 "CyBe Construction—Redefining Construction." Accessed July 11. https://cybe.eu/

[4] Dillenburger B. and Hansmeyer H. 2014. "Printing Architecture: Castles made of Sand" In Fabricate: Negotiating Design and Making, 92–97.

[5] Dini, E. 2018 "D-Shape". Monolite UK Ltd. Accessed July 11. http://www.d-shape.com

[6] Fritschi, E. et al. 2017. "Smart Dynamic Casting: Slipforming with Flexible Formwork – Inline Measurement and Control." Second Concrete Innovation Conference (2nd CIC), Paper no. 27. Troms , Norway.

[7] Fischer T. and Herr C. 2016. "Parametric customisation of a 3D concrete printed pavilion". In:

Chien, S. et al. (eds) Proceedings of the 21st International Conference of the Association for Computer-Aided Architectural Design Research in Asia CAADRIA 2016, pp. 549–558. The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), Hong Kong.

[8] Incremental3d. 2018. “Opening the world for innovative 3d concrete printing.” Accessed July 11. https://www.incremental3d.eu

[9] Khoshnevis B. et al. 2006 “Mega-Scale Fabrication by Contour Crafting.” In International journal of Industrial and System Engineering, Vol 1, no. 3.

[10] Lim S. et al. ”Developments in construction-scale additive manufacturing processes.” In Automation in Construction, 21 (1), 262-268.

[11] Oxman N., Keating S. and Tsai E. 2011 “Functionally Graded Rapid Prototyping” In Proceedings of Innovative Developments in Virtual and Physical Prototyping, The 5th International Conference on Advanced Research in Virtual and Rapid Prototyping, Leiria, Portugal

[12] Piker, D. 2017 “Kangaroo Physics” ver. 2.42. www.food4rhino.com/app/kangaroo-physics

[13] Pernecky, J. 2015 “Boid Library” www.food4rhino.com/app/boid-library

[14] Rust et al. (2016) Force Adaptive Hot-Wire Cutting Integrated Design, Simulation, and Fabrication of Double-Curved Surface Geometries.

[15] Stuart-Smith R., 2016 “Behavioural Production: Autonomous Swarm-Constructed Architecture”. AD Volume 86, Issue 2 Mar. /Apr. 2016. Sp. Issue: Parametricism 2.0: Rethinking Architecture's Agenda for the 21st Century, John Wiley & Sons, 54–59.

[16] Vasey et al. (2015) Behavioral Design and Adaptive Robotic Fabrication of a Fiber Composite Compression Shell With Pneumatic Formwork. In: Proceedings of ACADIA 2015 April 2016 (2015), pp. 297309.

[17] XtreeE. 2018. “XtreeE| The large-scale 3D” Accessed July 11. http://www.xtreee.eu

[18] Zwierzycki, M. 2015 “Anemone 0.4”. www.food4rhino.com/app/anemone

[19] Zwierzycki, M. 2013 “Mesh Tools.” www.grasshopper3d.com/group/milkbox/forum/topics/mesh-pipe-and-sweep-mesh

Nuclear thresholds: Geometric strategies for flexible cord assemblies

Luke Ogrydziak, Zoë Prillinger

Luke Ogrydziak
luke@oparch.net
OPA, United States

Zoë Prillinger
zoe@oparch.net
OPA, United States

Keywords:
Cords, nuclear, flexible

Abstract

Much current research in architectural geometry focuses on developing formal complexity through robotic fabrication. Most robotic fabrication processes take place in prefabrication, not on site. In prefabrication, robots fabricate sub-assemblies in a factory, and these sub-assemblies are later manually assembled to a larger aggregate structure on site. An alternative approach is proposed which uses a flexible material, for which specific geometric strategies are developed to exploit its unique material characteristics. The advantages of this approach are the possibility of achieving formal complexity through relatively simple means, along with the zero material-waste typical of additive fabrication processes.

This approach is implemented at the Nuclear Thresholds installation at Chicago, Illinois, which applies three geometric strategies to an assembly of flexible cords. These include close packing, recursive branching, and looping and knotting. Digital models were used to study the formal possibilities of each of these strategies, focusing in particular on alternative types of spatial controls, both deterministic and stochastic. The final installation demonstrates an alternative approach to material form in which complex geometries are developed by exploiting the specific characteristics of the material itself.

1. Introduction

This project explores the representational possibilities inherent in large diameter flexible cords. These cords are left full length and manipulated through operations such as linking and stacking. An obvious advantage of this approach is zero material waste, along with the potential to be reconfigured. This focus on waste reduction follows exemplary recent work such as the Wood Chip Barn made of naturally grown tree forks (Mollica & Self, 2016).

We digitally simulated different geometric possibilities for these cord assemblies, developing two primary systems of spatial control, one entirely deterministic the other more stochastic. Computationally, the position of a flexible cord in space can be fully defined by a 3D spline curve, a data type used in architecture extensively since the late 90s (Lynn, 1999). In reality, flexible cords have thickness, mass, and flex-

ibility. Consequently, digitally simulating the behavior of flexible cord assemblies requires computationally intensive physics simulations. The application of such physically accurate simulations to architecture is an open frontier for architectural geometry.

Nuclear Thresholds combines advanced digital simulation techniques with hand-assembly. Along its length, the project transforms from a single large bundle, to a series of progressively finer recursive branches, until each individual cord is free. This recursive structure embeds a material tendency within the material itself, provoking it to self-assemble into specific complex geometric systems through relatively simple means. We believe this is an important paradigm for advanced architectural geometry, combining computationally intensive studio work with low-tech site assembly.

Conceptually, our approach was largely inspired by the research work of Frei Otto. In particular, we find the idea of taking the materiality of a design model seriously and embracing its vagaries remains a liberating idea today (Spuybroek, 2014; Keller, 2017). Recent work on tensile structures has combined structural simulations with robotic fabrication (Mirjan et al., 2016). Such projects use particle spring system for their strand simulations. The success of this approach inspired our own material simulations. Our project similarly develops digital models simulating the behavior of material strands. Although in our case the primary forces are gravity and self-avoidance rather than tension.

2. Project description

Nuclear Thresholds is a commemorative installation for 75th anniversary of Enrico Fermi's "Chicago Pile -1" (CP-1), the first controlled, self-sustaining nuclear chain reaction. The project brief invited us to reflect on the nature of Fermi's experiment, as well as the tension between control and the loss of control engendered by the birth of the Nuclear Age. The project site is a 17.5-meter square plinth on the University of Chicago campus with a 3.6-meter-high Henry Moore sculpture in the center.

The installation uses an assembly of flexible cords to represent the moment of "going critical", when a chain reaction becomes (barely) self-sustaining – such as was the case with Fermi's experiment. The installation also represents supercriticality, the turning point when the rate

of fission increases, to the point of being out of control. In addition, the installation echoes the complex materiality embodied in the original CP-1 experiment such as the tightly-packed pile of graphite blocks (Murray, 2015). Finally, the installation suggests matter as something not solid but actually composed largely of space and energetic particles.

The Nuclear Thresholds installation is an assembly of 241 50 mm diameter x 23-meter-long EPDM rubber cords. The installation begins with a 90° arc of tightly packed cords. Then suddenly the assembly hits a virtual threshold and splits into two bundles. This recursive splitting continues seven more times at regular intervals until each cord is free from bundling. The development of the project combined material research and the creation of both physical and computational models. The installation contrasts a coherent, organized "solid form" with a breakdown into atomized independent parts. In so doing, the installation challenges the model of solidity upon which architectural geometry has historically relied, instead suggesting the structures of matter on a subatomic scale using complex full-scale geometries.

Figure 1: Nuclear Thresholds installation, Chicago, Illinois.

3. Close packing

The 1942 CP-1 experiment took place in a disused squash court under the University of Chicago football stadium. So, throughout our research on nuclear processes there was a ghostly presence of squash balls, echoed by the perfect spheres of physics diagrams. Initially we actually considered making the installation of squash bulls, but it would require over 15 000 40 mm diameter balls to fill a one-meter cube. Could the project start with close packed region, which was then dispersed with ever greater porosity across the site? As built, Nuclear Thresholds incorporates many elements of this initial approach.

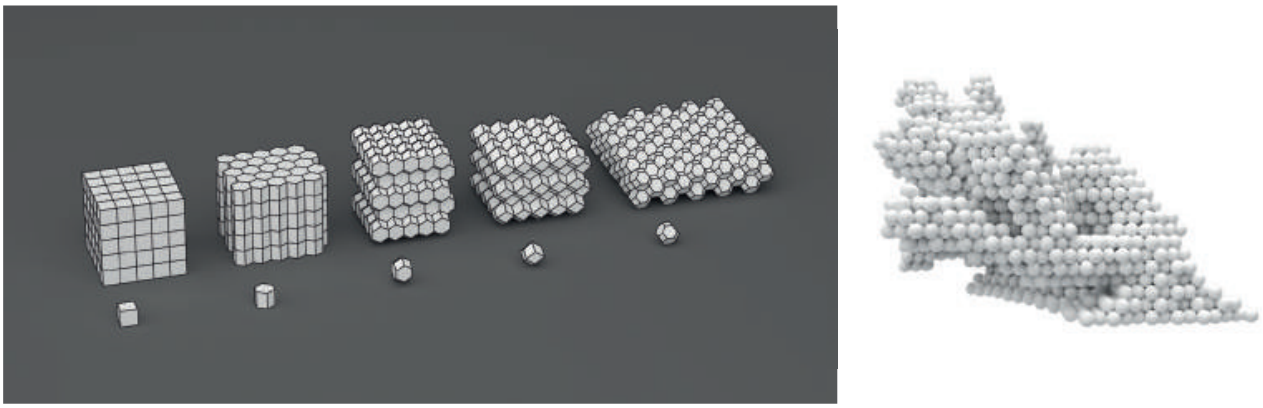


Figure 2: Initial design studies of 3D close packing : (a) Simple planar faced tiles which pack space without gaps; array #4 is the one we used for our spherical packing. (b) Spheres in a 3D array filling an arbitrary boundary envelope.

We began our research studying close packed spheres. To keep the data lightweight, we used a 3D hexagonal array (**Fig. 2a**). This allowed us to store the data in extremely terse text files with whole number coordinate position (i, j, k). Our first study was a rhino C++ plugin which filled arbitrary boundary envelopes with spheres (**Fig. 2b**). This approach successfully evoked piling, but we needed a method to add porosity. In the original CP-1 experiment, the particles moved through the graphite block in random walks. To evoke this aspect of the process, we created a diffusion-limited aggregation (DLA) simulation using the same 3D hexagonal array (**Fig. 3a**). We also tried translating this same data into a planar mesh shell, which we then relaxed to create a more organic morphology (**Fig. 3b**). This latter approach yielded porosity, but ultimately felt too controlled. We were searching for a system that had a sharper internal break, one that was more evocative about the threshold of

criticality which the CP-1 experiment had passed. Our project should feel explosive. But unlike an explosion, our installation would necessarily be a static form. In other words, the development of the system over time would need to be implied rather than enacted – a common architectural conundrum in our digital age (Carpo, 2011).

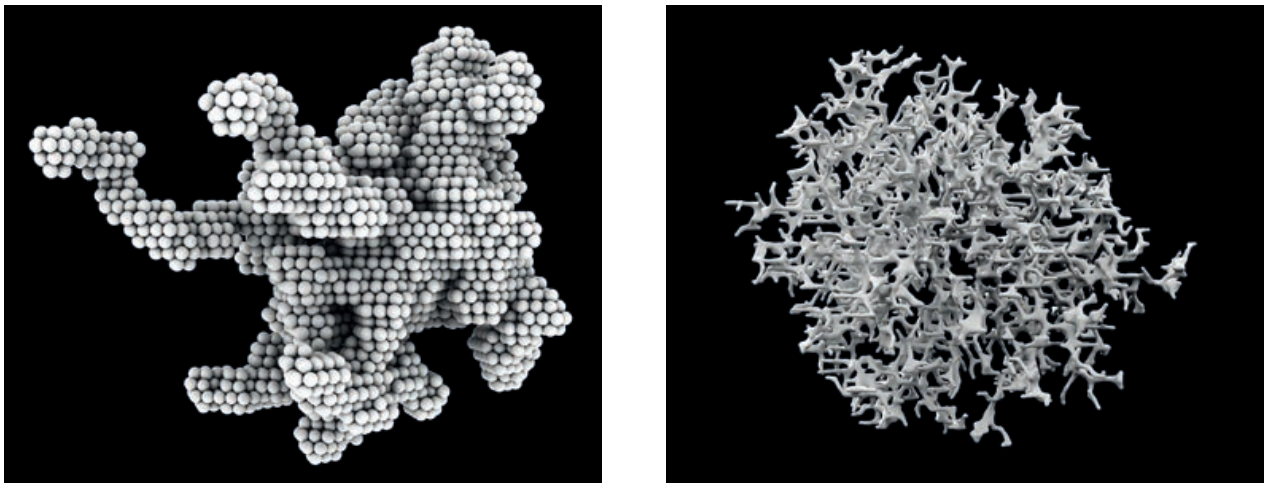


Figure 3: DLA simulations using a 3D hexagonal array: (a) Spheres with boundary inflation. (b) Planar mesh with relexation.

Ultimately, we decided to use cords rather than spheres. We could still represent close packing, essentially extruding a 2D hexagonal array along a line, like the second tile in figure 2b. This retained some memory of the squash court, as if the original squash balls were extruded into strands. To develop a typical bench section, we developed a 2D hexagonal array then drew a section of bench using these points. In deference to simplicity and gravity we limited ourselves to simple piling operations. This piling generally follows the 60° slope of the array, other than at the front of the seat where the cords are arranged vertically to make room for comfortable seating (**Fig. 4a**). Because of friction between the ground and bottom course, only minimal attachments were required at this area: loops of UV resistant zip ties at 0.5 m intervals in a running bond arrangement. As long as the bottom course is secure, this configuration allows the material to slump into the desired configuration (Murphy, 2017).

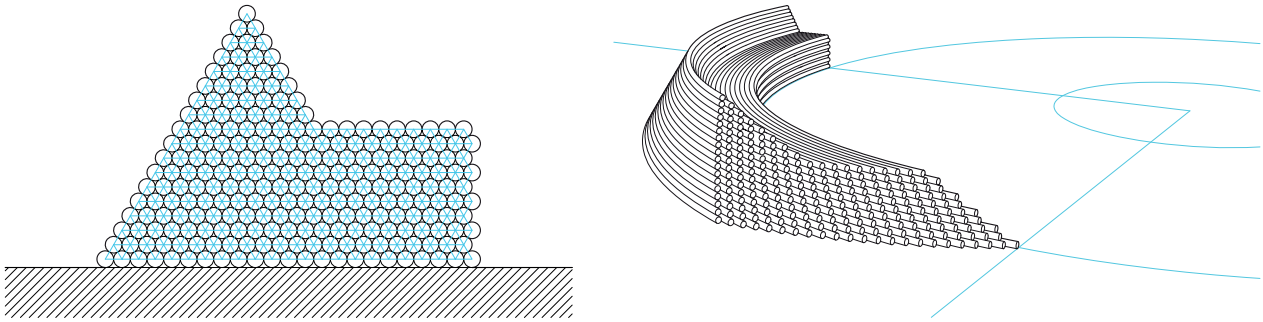


Figure 4: Nuclear Thresholds (a) Typical 241 cord section demonstrating hexagonal array inside constraint envelope of "bench" shape. (b) This section follows a 4.5 m radius 90° arc until "criticality" is reached.

To begin the layout, we established a 4.8-meter radius guide curve on the ground. The first strand follows this curve exactly, starting tangent to the street edge of the plinth and running clockwise. Then all the subsequent cords start rotated 1° in plan relative to the last cord placed. The result is a complex sheared and curved surface (**Fig. 4b**). The start of each vertical course shares a similar 1° plan rotation. Our piling rule is extremely simple, but because of the embedded complexity of combining concentric curves and the hexagonal grid section, the implied sheared figure which emerges has a complex, non-uniform curvature. This specific result emerges from a bottom up construction process which proceeds strand by strand based on a simple rule. The start of the arc creates a narrative where a solid form is progressively produced by the organized accretion of smaller elements (**Fig. 5**).

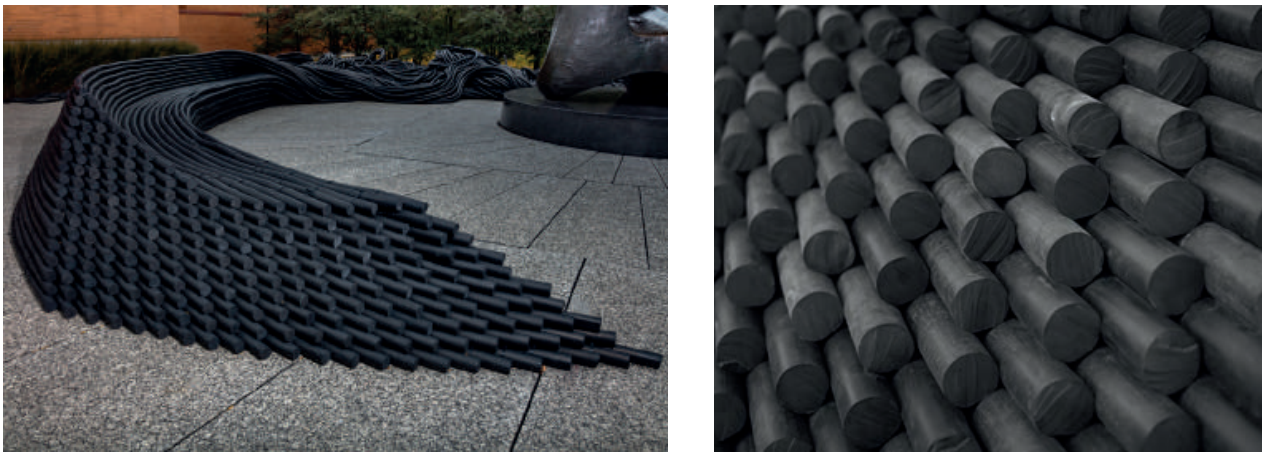


Figure 5: Nuclear Thresholds initial 90° arc: (a) Piling cords with 1° plan offset results in complex shearing at edge. (b) Detail of sheared tip.

4. Recursive branching

After the initial quarter circle, the pile of cords suddenly "goes critical". At this virtual threshold, the organization of the material changes entirely, from a tightly packed, geometrically controlled pile to a series of independent bundles of progressively fewer cords. In our research, we reviewed many chain reaction diagrams (**Fig. 6a**). Such systems always involve exponential growth. In controlled experiments like CP-1 the chain reaction is slowed by the graphite pile. In a bomb, a similar chain reaction happens at incredibly fast speeds and is intended to go supercritical. Representing such exponential growth presents an intriguing challenge for architectural geometry.

Our approach was to consider the installation as a pile composed of roughly 256 individual elements. At the bench arc, these elements are bound together into a single, simple form. After 90°, a "critical point" is reached and this bundle is split in half at regular intervals, resulting in a recursive branching structure (Ball, 2009). These bundles manifest an inverse relationship between the size of the bundles and the number of branches as follows.

RECURSIVE BRANCHING STRUCTURE

Branches at Interval	Cords in Branch	Sectional Area of Branch
1 (2^0)	256 (2^8)	5.02 m^2
2 (2^1)	128 (2^7)	2.51 m^2
4 (2^2)	64 (2^6)	1.26 m^2
8 (2^3)	32 (2^5)	0.63 m^2
16 (2^4)	16 (2^4)	0.31 m^2
32 (2^5)	8 (2^3)	0.16 m^2
64 (2^6)	4 (2^2)	0.079 m^2
128 (2^7)	2 (2^1)	0.039 m^2
256 (2^8)	1 (2^0)	0.019 m^2

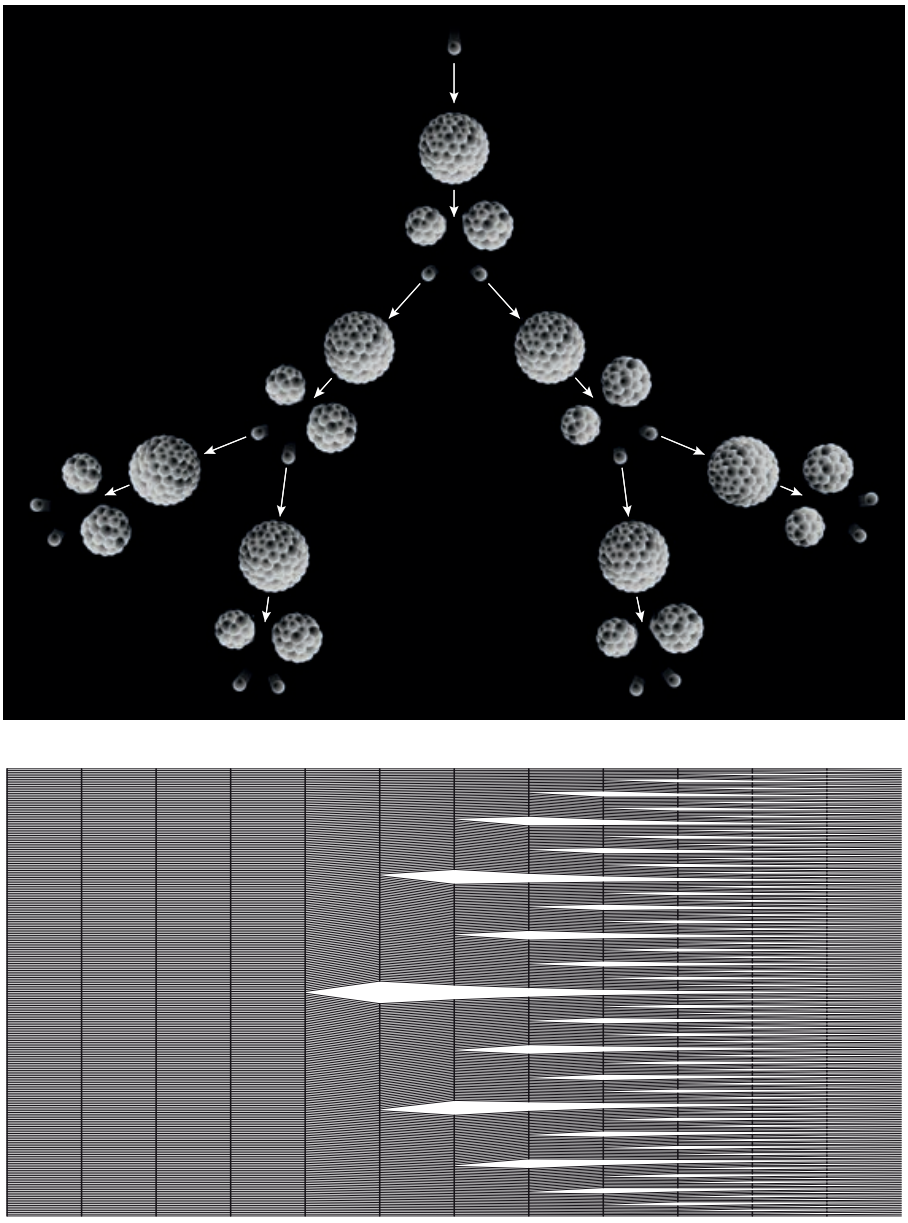


Figure 6: Chain reactions : (a) Literal chain reaction diagram starting with Uranium 235. (b) Nuclear Thresholds recursive splitting diagram showing progressive breakdown of form from single 256 cord bundle to individual cords.

This recursive branching system provides a physical analog to the exponential logic of a chain reaction. The relatively simple arc bench progressively dissolves into exponential complexity (**Fig. 6b**). We wanted the first split to be extremely dramatic. This first split is horizontal and the top flops over itself, creating a wavelike form. This involved two very heavy bundles. Each 23-meter-long cord weighs 54.4 kg or 2.4 kg/m. Since the project is entirely hand constructed by a team of 5-7 people, this move had to be planned out before finishing the bench arc (**Fig. 7**). To work around the weight of the strands further downstream, we deve-

loped a temporary fastening system based on the recursive branching logic. All the cords were temporarily linked together as powers of two. E.g., bundles of 2, then these bundles linked together to make 4, etc. This temporary system allowed for us to embed the recursive structure onto the assembly early on, while still allowing for flexibility in terms of the layout of the looping and knotting structures. One tension throughout the process was keeping the branching legible, without the overall gestalt becoming too tree-like. We wanted a slightly malevolent feeling in keeping with our own ambivalence about the nuclear threshold that was momentously breached 75 years ago by Fermi and his team.

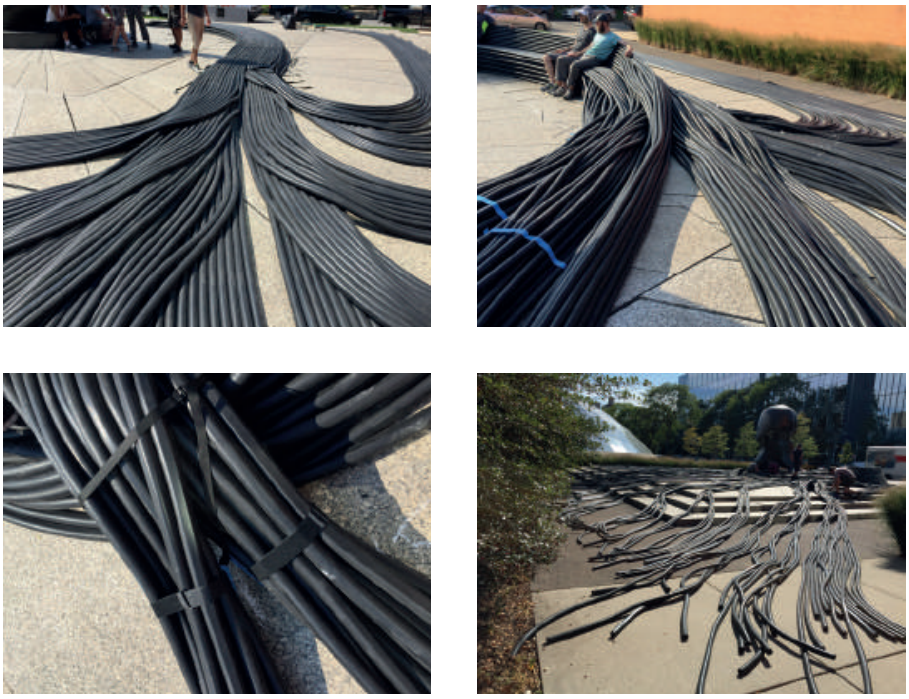


Figure 7: Construction process for recursive branching: (a) Initial setup sorting by bundles. (c) Typical branch using ziplock temporary ties. (d) Final setup for looping.

5. Loops, tendrils, and knots

The installation has a perceptual ambiguity, enacting a conflict between two distinct formal systems. From certain vantage points, the curving, form is simple and Platonic – essentially a truncated cone. This region identifies with a classical, centered space, the realm of dutiful reflection (**Fig. 8a**). But as you traverse the plinth, the form becomes restless and

incoherent. As the solid form breaks down, the successive branches erupt into loops, tendrils, and knots (**Fig. 8b**). The swirling induces a loss of orientation and an immersion that undermines modern notions of clarity and order. Architecturally speaking, we are excited about both models, but believe that swirling presents an opportunity for critical architecture to challenge many of the normative conditions and received ideas embedded in the world as typically constructed. Swirling space, however disturbing, may ultimately be a more accurate representation of our current world.

Figure 8: Two views of the "critical point" : (a) The truncated cone of the bench directs focus to Henry Moores "Nuclear Energy". (b) The rupture of the initial split, after which the coherent form of the arcing bench collapses into an amorphous swirl. (c) The swirl.

To study the formal possibilities for this swirling space, we developed game engine physics models at various resolutions. All of these simulations are based on the same basic approach. A flexible cord is approximated as a series of linked convex meshes. The fidelity of the model directly relates to the granularity of these approximations. To balance speed and accuracy, the circular section of the cord is approximated as a hexagon (**Fig. 9a**). Our next step was to establish the length of each link. The higher the link count the more accurate the simulation, but at a certain point the simulation becomes too slow to become useful. Our solution was to create two simulations at different resolutions.

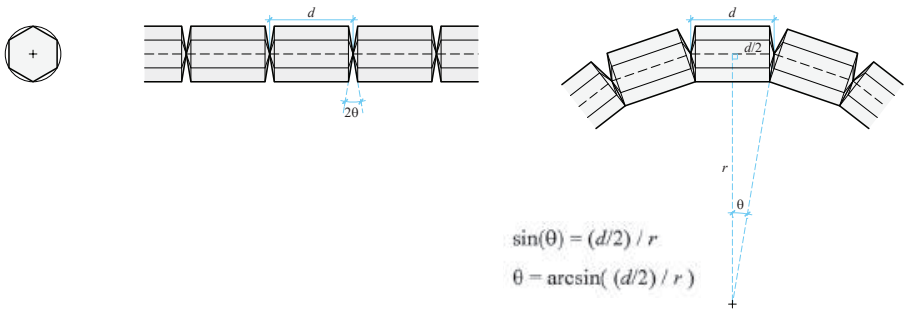


Figure 9: Basic mechanism for physics simulation : (a) Hexagonal section approximates circular cord (b) solid prisms with point connections limit bending angle with collision detection (c) The centers of the prisms are connected with a spline curve, which is piped to create a single cord.

First, we developed a game engine physics simulation focused on a detail of four recursion levels. This model has a 3:1 ratio of length to cord diameter (**Fig. 9b**). The bending radius of the EPDM cords used is approximately 0.46 meters, which defines the maximum twist at each joint (**Fig. 9c**). We approximated the cord bundling as a simple doubling of the sectional area of the guide cord. This approach reduces the number of at a relatively low fidelity cost, resulting in a model with 375 solid elements rather than 800 (**Fig. 10a**). This detail model is extremely accurate in terms of replicating the physical behavior of the cord assemblies. Such accuracy allowed us to test a variety of looping configurations, as well as study typical morphologies for how the assemblies come to rest after various manipulations such as dropping, pushing, and knotting. Second, we developed a keyframed physics simulation of the entire 241 cord system. This model extends the ratio of length to cord diameter to 12:1. The resulting model has 12,050 solids and is extremely 'heavy' compared to the first study model (**Fig. 10b**). This loss of resolution results

in behaviors that are slightly less realistic, but adequate for studying the overall project gestalt. We tested morphologies resulting from dropping the assembly and solid object collisions (**Fig. 10c**, **10d**).

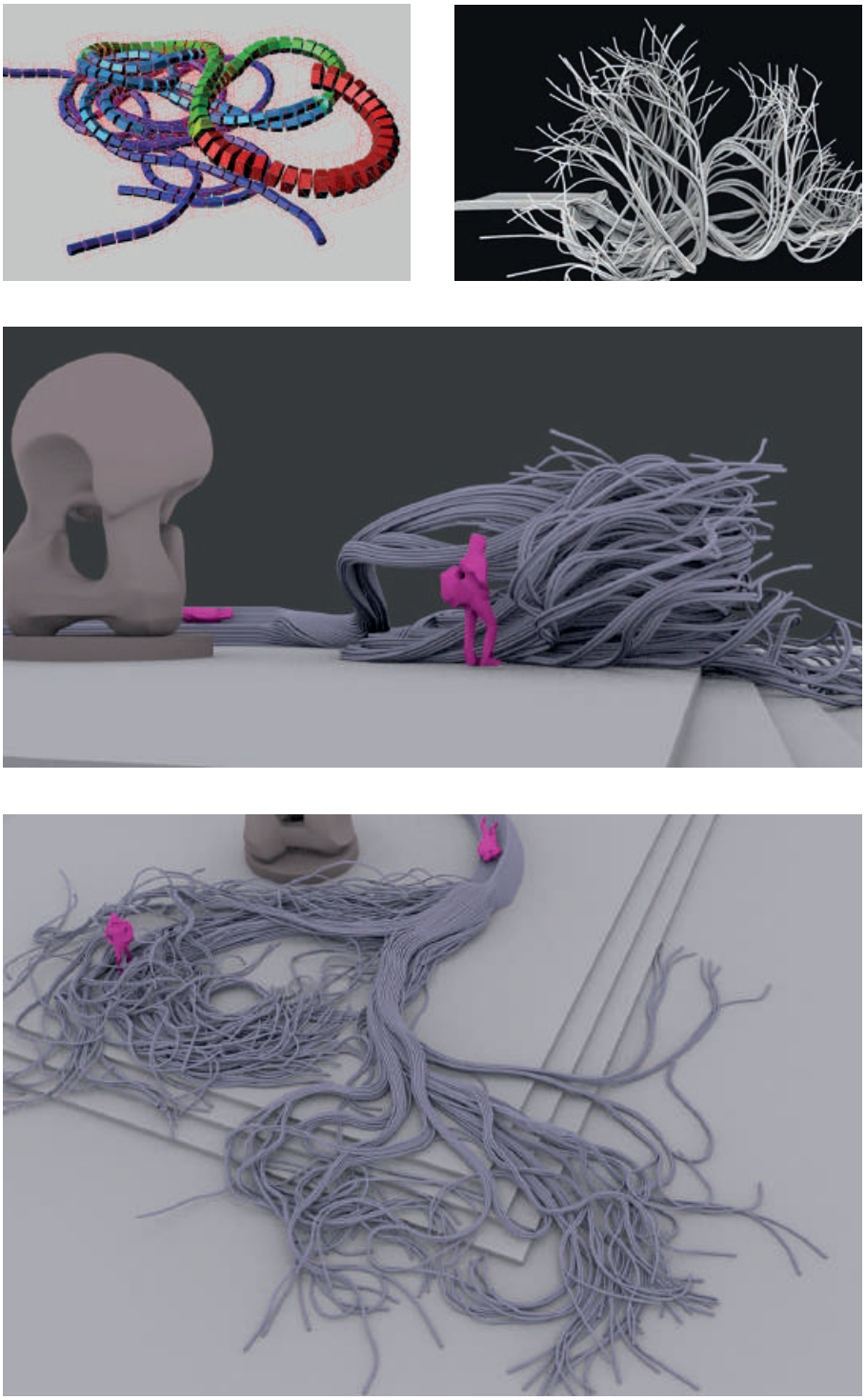


Figure 10: Physics simulations at various resolutions and states: (a) Detail model at focusing on 4 recursion levels. (b) Low-resolution full size (241 strand) model mid-simulation. (c) Full-size model near the end of a simulation, before gravitational settling. (d) Full-size model used to study interaction with site.

The branching area represents the loss of control implied by the original CP-1 experiment. Ideally, we would have set up the recursive branching connections, then throw the piece high in the air and let it expand and fall; but this was not a realistic option. Instead the installation was built by hand at ground level, with the thicker bundles required the most extensive advanced planning. Areas involving looping or knotting of the thicker bundles, were built up gradually out of 4-strand sub-assemblies. At the free ends we followed the examples of our detail simulation, which generated extensive looping near the bending limit of the cords. The overall gestalt of the branching region with its loops, tendrils, and knots provides a strong formal contrast to the highly controlled beginning arc (**Fig. 11**).

Figure 11: Aerial photo of Nuclear Thresholds installation.

6. Gaps

We began this project with a simple plan: produce a fairly accurate digital simulation of our installation based on the material properties of our flexible cords. This simulation would allow us to explore different formal possibilities both in terms of bundling and looping. As it turned out, the complexity of this simulation at a "high resolution" quickly overwhelmed our computational resources. Consequently, most of our time was spent negotiating the representational gaps between our digital models and the actual material behavior. Several such gaps have already been described, in terms of granular trade-offs in the resolution of various simulations.

Reviewing this process from a broader perspective, the most significant representational gap was between the methods for manipulating our simulation and the actual fabrication techniques. In real life, the process was entirely limited by the moves a group of 5–7 people could achieve. Accurately simulating this was conceptually possible using game engine physics. But in practice, at any kind of realistic level of detail these simulations were too computationally intensive to be useful. The only way to simulate the behavior of the entire bundle was to use a physics simulation, and keyframe various impact forces. Setting up these types of scenes took a long time (12+ hours), but at least ran cleanly. However, being limited to this type of pre-set simulation meant that precisely the kinds of behaviors we were most interested in were not possible at higher resolution levels. For instance, we could only tie knots effectively using interactive physics simulations, but these had to be run at much lower detail levels.

In the end, we decided to embrace this lack of material control as part of the beauty of the project. The branching was rigorously set but the looping portions were more contingent, working from the types of shapes we liked in the simulations but only following a strictly choreographed setup for the initial branchings. Given the specific representational goals for this piece, this approach made sense. But in a different situation a project that focused exclusively on no more than 20 strands could have used the same tools at a finer level of detail. This would allow for a significant reduction in the gap between the digital simulations and construction technique.

7. Conclusion

Nuclear Thresholds challenges the model of solidity upon which architectural geometry has historically relied. The project demonstrates an alternative approach to material form in which complex geometries are developed by exploiting the specific characteristics of the material itself, extensively using digital simulations to predict this behavior combined with low tech hand fabrication techniques.

One possible trajectory for this type of research would be to introduce robotic construction techniques. This would allow for greater complexity in terms of both the patterns established and in the manipulation of the materials themselves. A major constraint of this installation was working around what a team of average strength people could lift. Changing this constraint would modify the field of formal possibility, introducing an entirely new set of opportunities and constraints.

A second, alternative trajectory would be opportunity for development would be to focus primarily on self-assembly. Working exclusively with manipulation by dynamic physical forces would yield an entirely different set of formal possibilities. In our digital simulations the most dynamic results occurred by throwing the model high in the air and letting the cord assemblies expand stochastically. In this context the focus might be the specific binding patterns rather than the specific placement cord by cord. The site setup would simply involve adding energy to the system. In both cases, the accurate digital simulation of computationally intensive physical behaviors remains a rewarding frontier for research in architectural geometry.

References

Aste, Tomaso and Denis Weaire. 2008. *The Pursuit of Perfect Packing*. London: Taylor & Francis.

Ball, Philip. 2009. *Branches.* London: Oxford University Press.

Carpo, Mario. 2011. *The Alphabet and the Algorithm*. Cambridge, Massachusetts: MIT Press.

Comaroff, Joshua and Ong Ker-Shing. 2013. *Horror in Architecture*. China: Oro Editions.

Keller, Sean. 2017. *Automatic Architecture: Motivating Form After Modernism*. Chicago: The University of Chicago Press.

Lynn, Greg. 1999. *Animate Form*. New York: Princeton Architectural Press.

Mirjan, Ammar, Federico Augugliaro, Raffaello D'Andrea, Fabio Gramazio, and Matthias Kohler. 2016. "Building a Bridge with Flying Robots." In *Robotic Fabrication in Architecture, Art and Design 2016* edited by Dagmar Reinhardt, Rob Saunders, and Jane Burry. Heidelberg: Springer.

Mollica, Zachary and Martin Self. 2016. "Tree Fork Truss: Geometric Strategies for Exploiting Inherent Material Form." In *Advances in Architectural Geometry 2016* edited by Sigrid Adrianenssens, Fabio Gramazio, Matthias Kohler, Achim Menges, and Mark Pauly. 138–153. Zürich: vdf.

Murphy, Kieran, Leah Roth, and Heinrich Jaeger. 2017. "Adaptive Granular Matter." In *Active Matter* edited by Skylar Tibbets. 287–290. South Korea: MIT Press.

Murray, Raymond and Keith Holbert. 2015. *Nuclear Energy: An Introduction to the Concepts, Systems, and Applications of Nuclear Processes*. London: Elsevier.

Spuybroek, Lars. 2014. "The Structure of Vagueness." In *Theories of the Digital in Architecture* edited by Rivka Oxman and Robert Oxman. 263–276. New York: Routeledge.

Taylor, William. 2005. "Protein Folds, Knots and Tangles." In *Physical and Numerical Models in Knot Theory* edited by Jorge Calvo, Kenneth Millett, Eric Rawdon, and Andrzej Stasiak. 171–200. Singapore: World Scientific Publishing Company.

Tuniz, Claudio. 2012. *Radioactivity: A Very Short Introduction*. London: Oxford University Press.

Oasis of light – Manufacturing the cladding of the Louvre Abu Dhabi

Goswin Rothenthal, René Ziegler, Djordje Spasic

Goswin Rothenthal
goswin.rothenthal@waagner-biro.com
Waagner-Biro Stahlbau AG, Austria

René Ziegler
rene.ziegler@waagner-biro.com
Waagner-Biro Stahlbau AG, Austria

Djordje Spasic
djordje.spasic@waagner-biro.com
Waagner-Biro Stahlbau AG, Austria

Keywords:
Panelization of sphere, nodeless mesh design, data flow collaboration

Abstract

The dome structure of the Louvre Abu Dhabi and its cladding is a milestone in architectural design and construction. The design development and the fabrication of a large scale light test model of the dome spanning over the museum galleries has been presented in the AAGs 2010 and 2012. We want to pick up the thread and give an insight on how we applied geometry to manufacture, assemble and install the cladding of the Louvre Abu Dhabi.

1. Introduction

The starting point for the museum was a sketch on a sheet of paper and the effort of the city of Abu Dhabi to set up a museum in the spirit of openness and dialogue. The dialogue began with the annual Abu Dhabi Art. Both Foster + Partners and Shigeru Ban have already presented so-called "Architecture Statements". The current "Statement" is the dome of the Louvre Abu Dhabi. Jean Nouvel has designed a museum that fits into the traditional landscape of the United Arab Emirates. The museum is located at the transition from land to sea. The buildings are islands in the water connected by bridges and pontoons. The flat 180 m wide dome creates the cohesion of the parts. The initially very clear geometric shape of the dome shows in detail an extremely complex structuring. The seemingly random interweaving creates shadows. At the same time, the woven structure allows openings for the enter of bundles of sunrays. During the day, the dome shimmers in the sunlight and at night, the buildings are part of an urban promenade under the star dome. The ensemble of buildings, together with the dome, become a sanctuary and shelter for the valuable works of art and visitors.

Architect Jean Nouvel summarized the design idea as follows [1]: "This micro-city requires a micro-climate that would give the visitor a feeling of entering a different world. The building is covered with a large dome, a form common to all civilizations. This one is made of a web of different patterns interlaced into a translucent ceiling which lets a diffuse, magical light come through in the best tradition of great Arabian architecture."

With its start into the project, Waagner-Biro's design team was able to pick up a well-defined project in general and well-defined geometric set-up in specific developed by the teams in and around Ateliers Jean

Nouvel, Buro Happold, Gehry Technologies, Transsolar and further schools and consultants. To understand the design approach and steps of development we recommend reading the article presented and published at the AAG 2012 in Paris [1]. The authors had taken the project as a case study for the number of stakeholders involved in the development of a project of this scale. The stakeholders contribute to the project by defining requirements. The concurrent models of the various stakeholders are the expression of the requirements. When we refer to RIBA stages [3], the Developed Design will merge the coordinated architectural, structural and services models into one digital model. The digital model will always serve for multiple purposes. Benjamin Koren showed how the developed design may be advanced form the digital model into manufacturing data for a 1/33 visual mock-up. He presented this at the AAG 2010 in Vienna [2].

Within the standard workflow, the coordinated developed design model is the source for a robust framework of quantities. This is one of the primary outcome of the developed design. This framework is basis for a reliable cost estimation for the specialist contractors.

In this workflow, Waagner-Biro was the chosen specialist contractor for the dome structure. The scope of works of this package included the design and delivery of the architecturally designed structural steel and the multiple layers of aluminium cladding enveloping the load-bearing steel. In our paper, we will focus on the geometrical aspects of the technical design of the cladding.

It was Waagner-Biro's engineers and designers task to pick up the *developed design* and work it forward to the RIBA stage Technical Design. The offsite manufacturing of the components, the assembly of the components to deliverable parts and the development of the onsite installation methodology are all the outcome of the technical design. The deliverables to site can be summarized to the groups of structural steel, aluminium cladding and the supporting temporary works. The permanent structural steel consists of 11 000 members connected by 2 700 nodes. The design of the bespoke steel towers positioned between the museum buildings during construction phase were the key elements of the temporary works. The aluminium cladding enveloping the dome is the composition of more than 8 000 individual stars.

The authors of the paper want to illustrate the influence of geometry on the workflow of the preparation of production information by putting

the focus on the cladding design. We will show how the work of the "development team" in the developed design stage was the basis for "technical team" in the technical design stage.

2. Geometric detailing of the cladding

The geometric pattern originates by arranging isosceles triangle repeated and rotated to form a system of squares and octagons [1]. The pattern may also be seen as four pointed stars meeting at their tips. The pattern is mapped to the "great circle" grid on the dome surface. The tessellation is true at the apex whilst it distorts towards the perimeter of the dome.

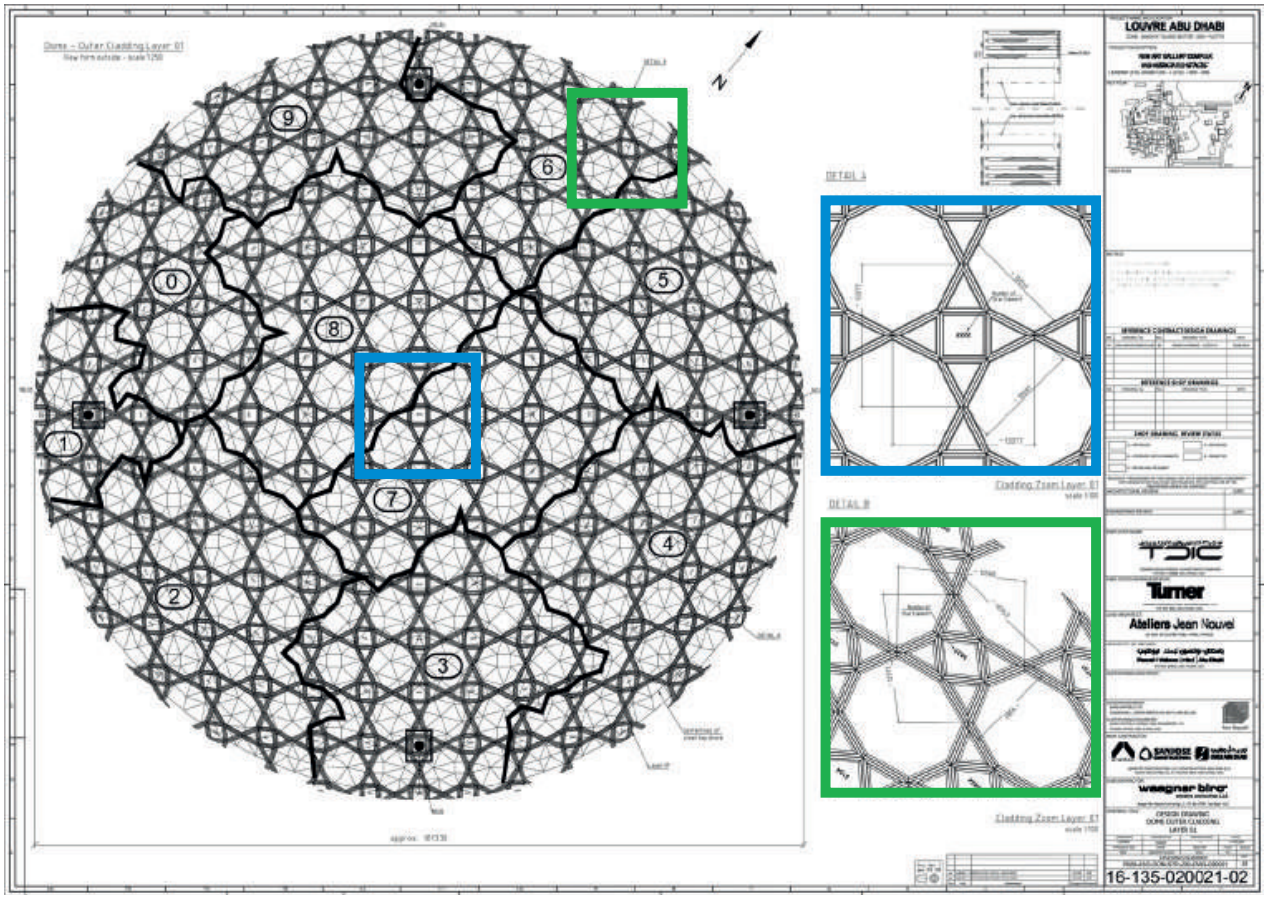


Figure 1: Arrangement of stars and distortion at perimeter.

The stars are scaled in size and mapped to the further layers of the cladding of the dome. The resulting pattern is rotated around the central axis

of the dome. These sequenced geometric operation has the consequence of that the layers themselves were symmetric, but the stacking of the layers was a fully non-symmetric structure.

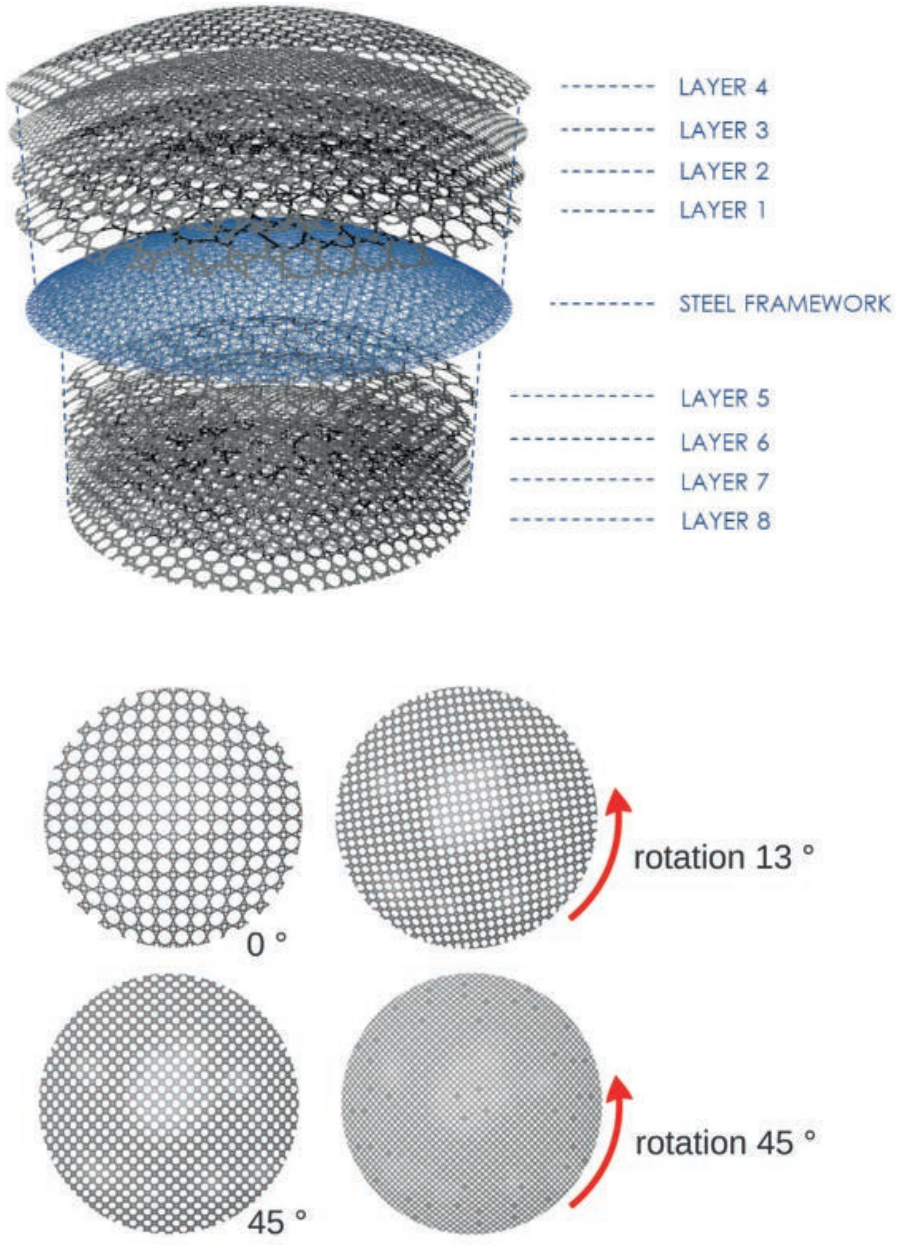


Figure 2: Arrangement of layer of stars enveloping the steel framework.

The decision to keep the faces of the triangles, quads and octagons planar had the consequence that bars connecting the nodes of the stars are folded in order to align with the adjacent faces. The bars were further required to have varying width to control the passing of light through the dome according to the translucency map of the museum.

The fold angles were analysed for the varying stars within the mapping and over the layer of the cladding. The result of the analysis was the basis of the decision on how to design the technical details for generating the kink. The analysis and clustering of the width of the profiles were the basis for the design of the aluminium extrusion profiles. To meet the required width, the bars are assemblies of several extruded profiles.

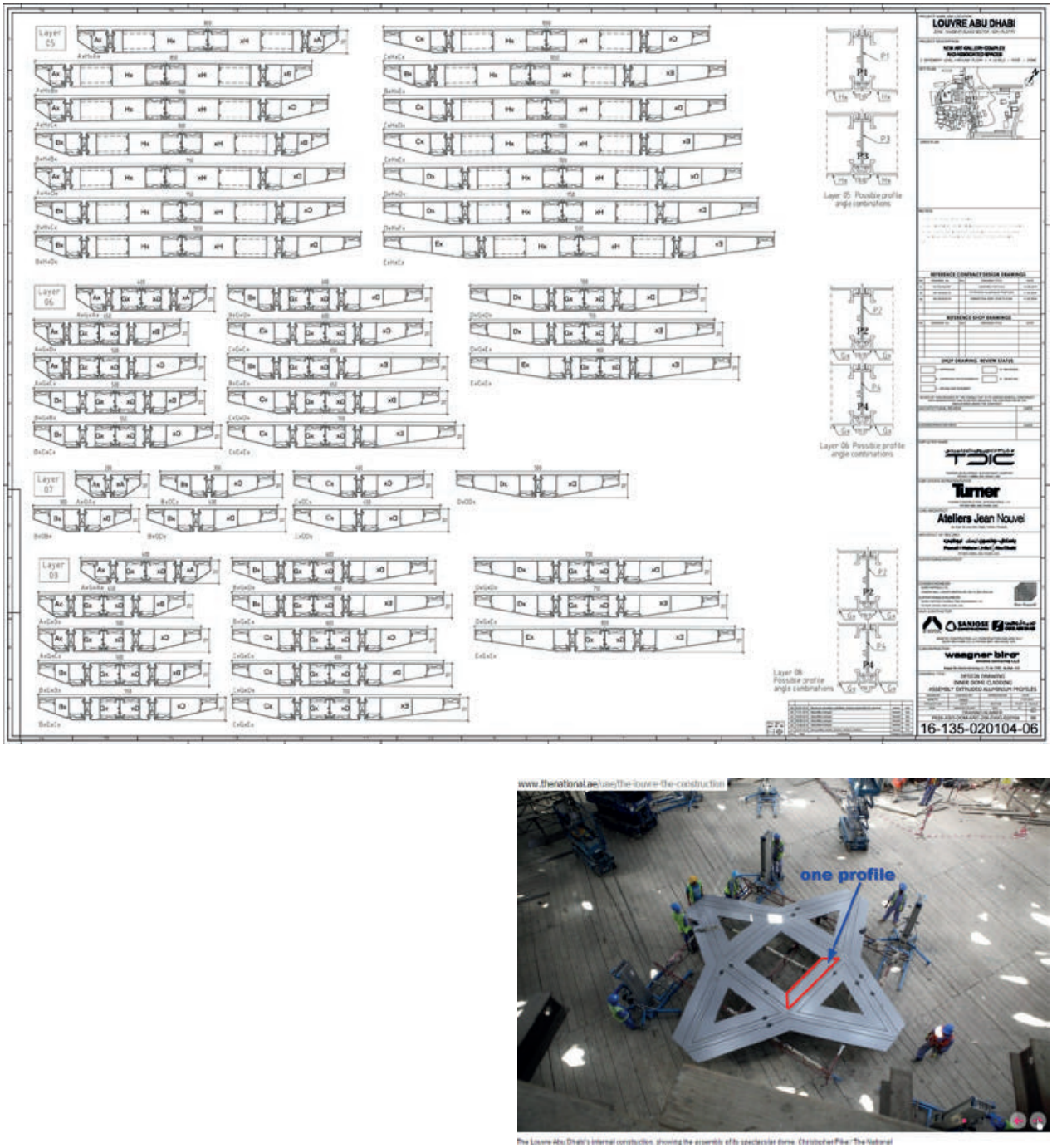


Figure 3: Bars as assemblies of extruded aluminium profiles with folds at intersection of faces.

The bars as assemblies of extruded profiles were connected at the four corners of the inner quads to stars. The mapping of the stars to the “great circle” grid lead to varying in-plane angles at the connections within the faces and non-congruent normals of the four faces meeting at single point. The corner cleats are the elements connecting the bars at the node of the stars. They remain within the plane of a face. Therefore, the design of the corner cleat must only consider the in-plane angle. The varying angles were analysed to define the range angles that the corner cleat needs to cover.

Having each face build fully planar allows significant simplification in the joint of aluminium extrusion:

1. The cutting plane is always perpendicular to the top plane of the profile. One angle was sufficient to describe the cutting plane. No 3D-incline-cutting planes were required.

2. The corner cleats connecting two profiles only have to accommodate rotation in a plane. No spatial ration is needed. The corner cleats can be designed and produced following the principal of a single axis hinge.

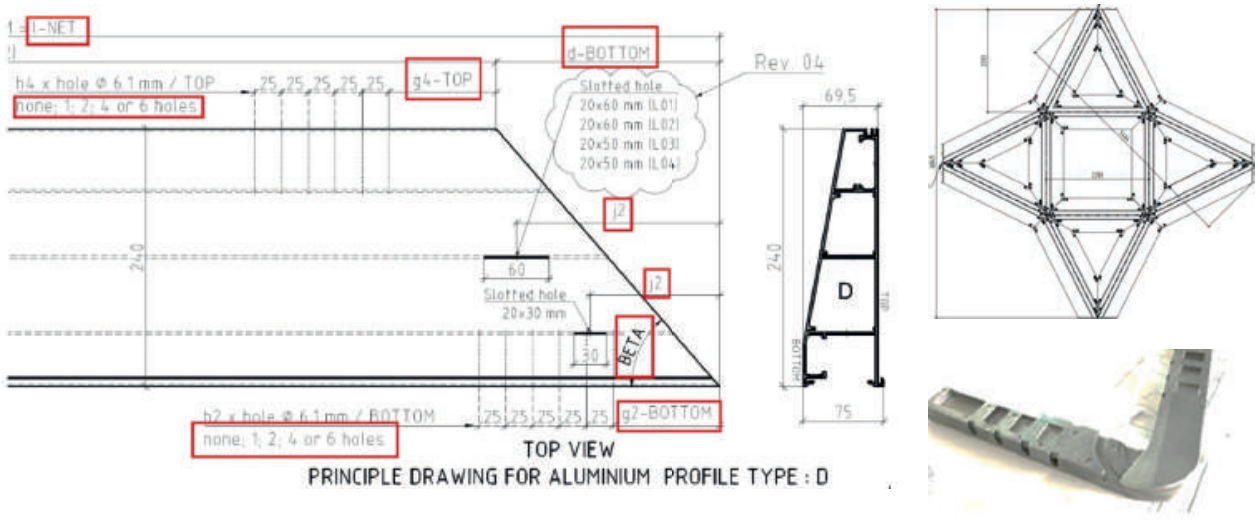


Figure 4: Geometric description of the end of a bar, the position of the corner cleats and a typical corner cleat.

All the connections within a star are fully rigid connections. The tips of the stars connect to their neighbours to form a continuous structure. In structural terms, the star-to-star connections behave as hinged and sliding connections. The connections transfer out-of-plane shear forces from one star to the next. The geometric requirement is to depict the kink necessary to clad the sphere of the dome.

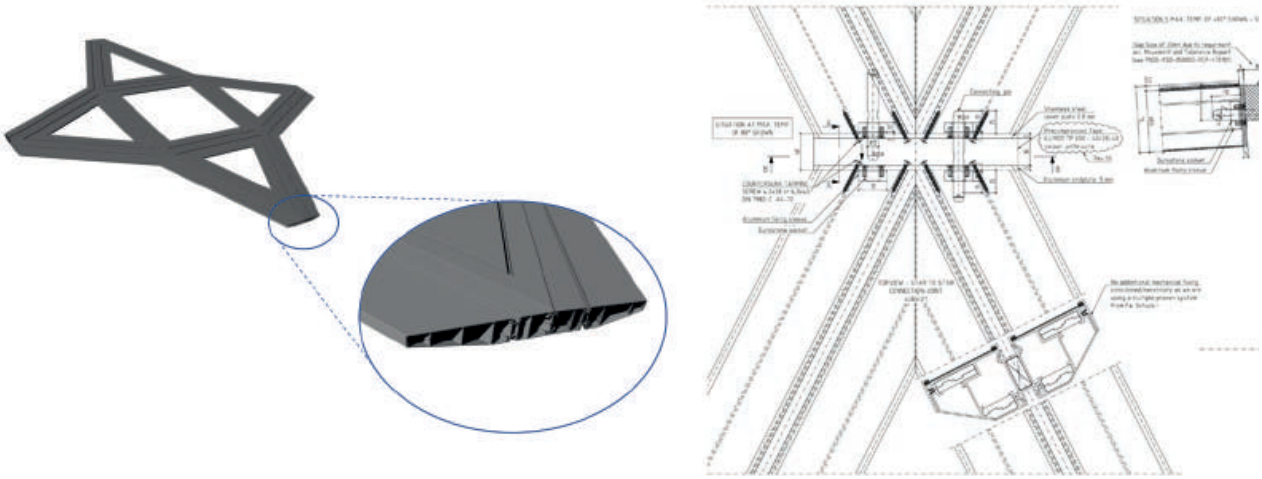


Figure 5: Isometric view of the tip of a star and star-to-star connection.

Each star-to-star connection is unique. The design of the connection required the consideration of the kink angles, the types of profiles and the inner angle of the bars connecting at the tips as well as the forces transferred between the stars. With the sum of this information defined the diameter of the connecting bolts, the thickness of the endplate and the number and position fastening screws.

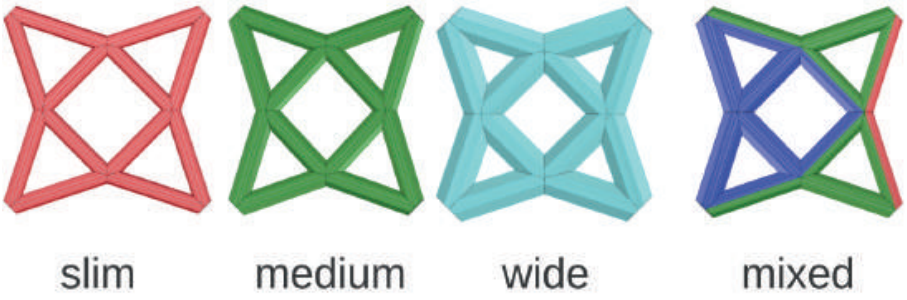


Figure 6: Basic star assemblies.

In simple term, the build-up of a layer is as follows. Extruded profiles are assembled to bars, corner cleats join bars to stars and stars are hinged at their tips to a continuous layer. The layers then require structural elements to connect the layers and transfer the forces to the load bearing steel structure.

3. Connecting the cladding to the supporting structure

The structural steel supports all the layers of cladding. *Layer-to-steel* connectors transfer the load form cladding to steel. Ideally, the cladding connects directly to the steel. The stacking of layers does not always permit the direct connection to steel; therefore, *layer-to-layer* connectors were introduced.

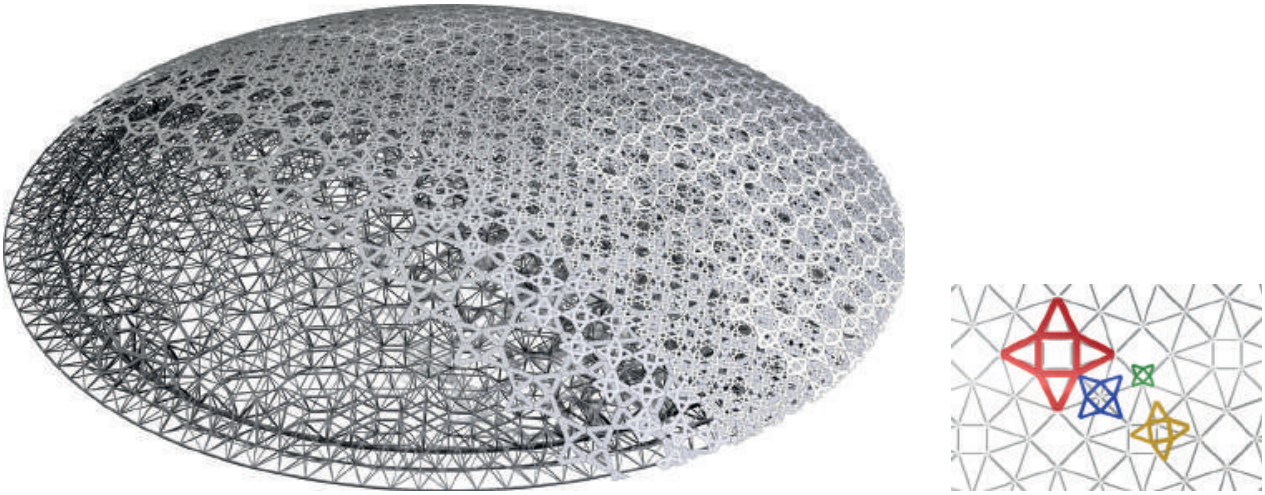


Figure 7: Layers of aluminium cladding enveloping structural steel and possible relative positions.

The general principle for the positioning layer-to-steel and layer-to-layer connectors are as follows:

1. Find all possible intersections between the channels of the aluminium bars and the structural steel.

2. For non-direct connectors (e.g. steel to layer 2 or layer 1 to layer 3) check for collision with in-between layers.

3. If a bar has more than two supporting points: Remove the inner supports.

4. Generate structural engineering model, calculate and evaluate.

5. Adapt and modify the components of cladding according the results of the structural calculation.

6. Regenerate FE model, calculate and evaluate results.

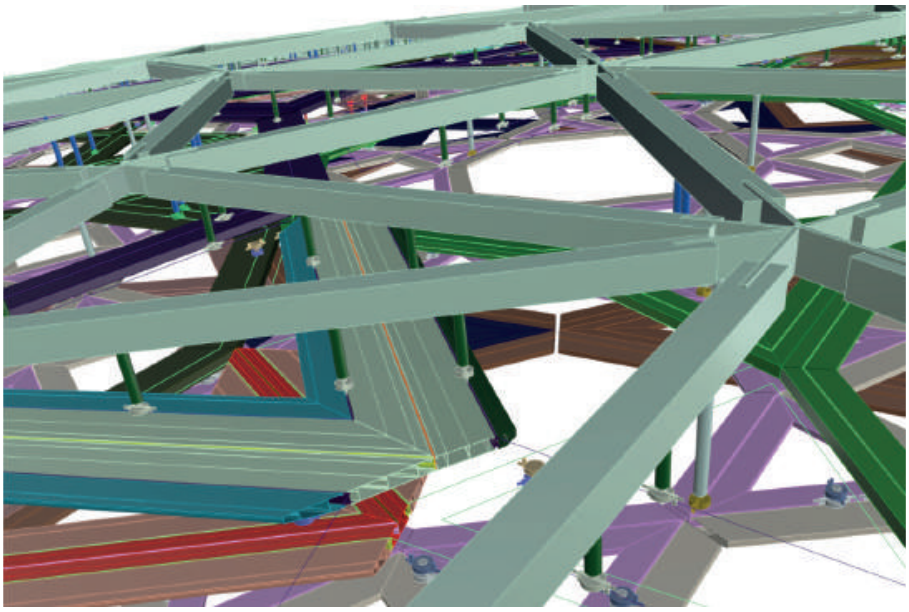


Figure 8: Digital model of layers of aluminium cladding connecting to structural steel.

Geometric data was necessary for the off-site fabrication. On site, the installation of the layers of stars had to follow the varying installation phase geometry of the steel dome. The steel dome was built in a pre-cambered geometry to compensate its dead load deflection. The final design geometry was reached after the de-prop of the structure from the scaffold and the full application of the cladding load. This made it necessary to constantly update the installation coordinates for the stars.

4. Conclusion

The Louvre Abu Dhabi Dome as proven to be a case study for the close collaboration of the teams within a stage of project execution as well as in successive stages of project execution. When taking reference to the RIBA stages it is obvious that it is necessary to share and coordinate geometric information throughout the workflow of project execution. To improve the performance of the workflow at the interfaces it proves to be helpful to share and understand the principles necessary to recreate the geometric information.

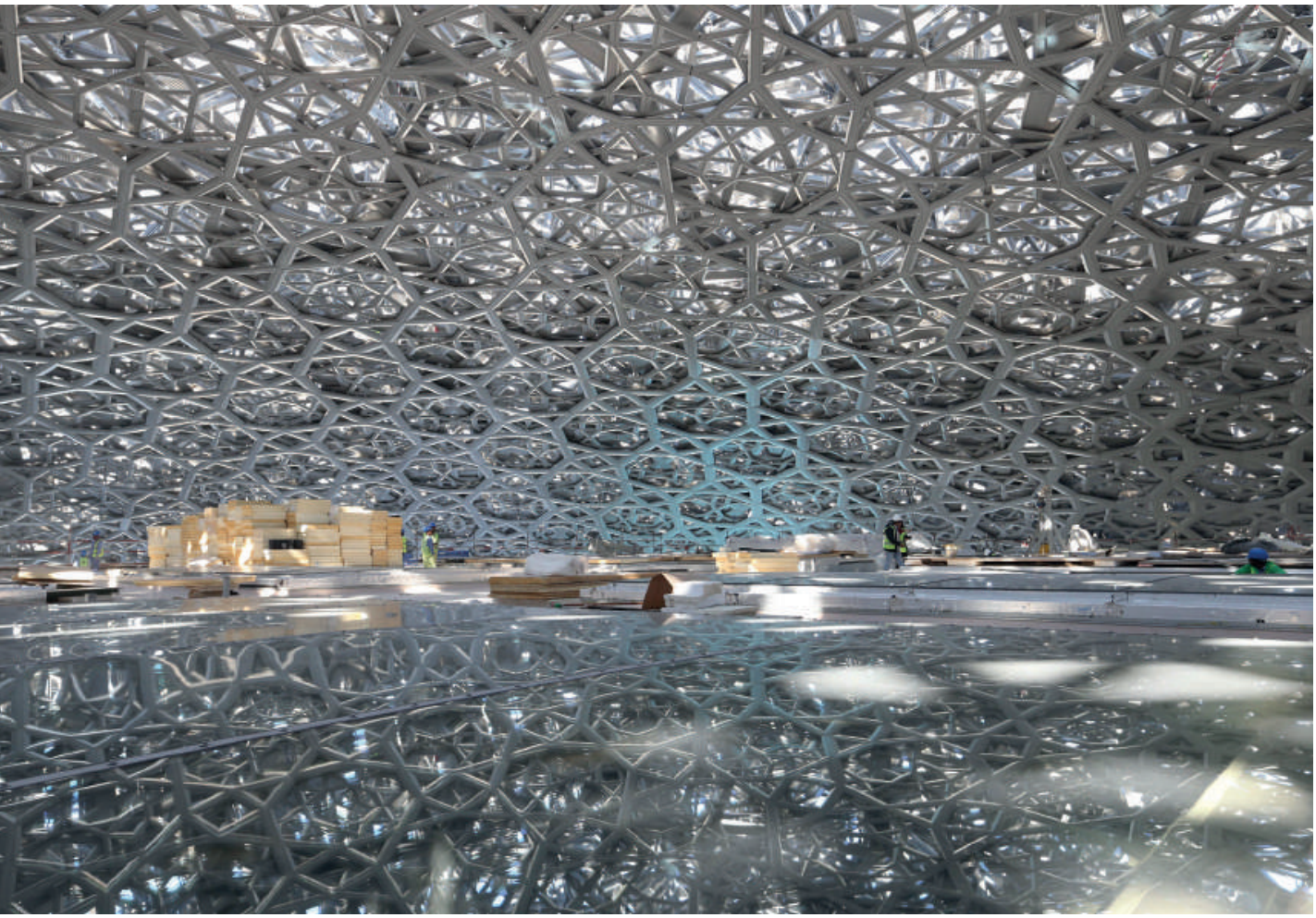


Figure 9: View of the inner layers of cladding of the dome (image courtesy of TDIC).

References

[1] Imbert, Stutts Frost, Fisher, Witt, Tourre, Koren. “Concurrent Geometric, Structural and Environmental Design: Louvre Abu Dhabi”, in AAG 2012, (pp. 77–90)

[2] Koren, Benjamin. “Louvre Abu Dhabi 1/33 – Fabrication of a Large-Scale Physical Light-Test Model”, in AAG 2010, (pp. 163–74)

[3] RIBA Plan of Work 2013, www.ribaplanofwork.com

The use of virtual work for the formfinding of fabric, shell and gridshell structures

Emil Adiels, Mats Ander, Erica Henrysson, Jens Olsson, Karl-Gunnar Olsson, Alexander Sehlström, Paul Shepherd, Chris Williams

Emil Adiels
emil.adiels@chalmers.se

Mats Ander
mats.ander@chalmers.se

Erica Hörteborn
erica.horteborn@chalmers.se

Jens Olsson
jens.olsson@chalmers.se

Karl-Gunnar Olsson
karl-gunnar.olsson@chalmers.se

Alexander Sehlström
alexander.sehlstrom@chalmers.se
Chalmers University of Technology, Sweden

Paul Shepherd
p.shepherd@bath.ac.uk
University of Bath, United Kingdom

Christopher Williams
christopher.williams@chalmers.se
Chalmers University of Technology, Sweden, University of Bath, United Kingdom

Keywords:
Virtual work, fabric, shell and gridshell structures, calculus of variations, conjugate directions

Abstract

The use of the virtual work theorem enables one to derive the equations of static equilibrium of fabric, shell and gridshell structures from the compatibility equations linking the rate of deformation of a surface to variations in its velocity. If the structure is treated as a continuum there is no need to consider its micro-structure provided that the grid is fine compared to the overall geometry. Thus we can include fabrics, ribbed shells, corrugated shells and gridshells with a fine grid, such as the Mannheim Multihalle. The equilibrium equations are almost identical to those obtained by assuming that a shell is thin and of uniform thickness, but are more general in their application. Our formulation introduces the concept of geodesic bending moments which are relevant to gridshell structures with continuous laths.

The virtual work theorem is more general than the energy theorems, which it in- cludes as a special case. Hence it can be applied to surfaces which admit some form of potential, including minimal surfaces and hanging fabrics. We can then use the calculus of variations for the minimization of a surface integral to define the form of a structure.

Many existing formfinding techniques can be rewritten in this way, but we concen- trate on surfaces which minimize the surface integral of the mean curvature subject to a constraint on the enclosed volume, producing a surface of constant Gaussian curvature. This naturally leads to the more general study of conjugate stress and curvature directions, and hence to quadrilateral mesh gridshells with flat cladding panels and no bending moments in the structural members under own weight.

1. Introduction

The principal of virtual work was formalized by Johann Bernoulli and Joseph-Louis Lagrange in the 18th century (Capecchi, 2012) and today is taught to all civil and mechanical engineering students. It is the basis of the application of the finite element method to structural mechanics, although the formulation it produces is often identical to that obtained using the Galerkin method. Virtual work is closely related to the calculus of variations and the concept of strain energy, but it is more general in that it can be applied to non-elastic materials and to loads which do not admit a potential, such as wind loads.

Virtual work considers a virtual or imaginary infinitesimal increment of displacement and deformation of a structure which may be undergoing a very large deformation. It then calculates the increment of work done by the loads on the structure and uses the divergence theorem to relate this to internal stresses and strains. It is perhaps better to instead imagine a virtual velocity, in which case the increment of work is replaced by the rate of work, or power, and in French they use the term puissances virtuelles, literally virtual power.

Virtual work requires the geometric compatibility equations relating increment of displacement to increment of deformation or strain and uses the virtual nature of the increment of deformation to derive the equations of static equilibrium. Thus the method is purely geometric, with no concept of resolution of forces or moments, which should hopefully appeal to those with a background in geometry. The advantage of using velocity instead of increment of displacement is that the velocity is the derivative of position with respect to time, and we can therefore use all the properties of differentiation, rather than the more unwieldy process of letting the magnitude of the displacement tend to zero.

It it should be emphasized that we have essentially 3 types of equation,

- the compatibility equations relating variations in velocity to rate of deformation, both stretching and bending,
- the virtual work equation and
- the equilibrium equations relating loads to internal forces and moments.

Any 2 of these imply the 3rd and we will use the compatibility equations and virtual work to obtain the equilibrium equations. One could use the compatibility equations and the equilibrium equations to prove the virtual work equation, and engineers often use the equilibrium equations and virtual work to solve a geometric problem involving the deformation of truss structures.

We shall assume that the structure is either a continuous shell or fabric structure, or has a fine grid so that it can be treated as a continuum, both from the geometrical and structural points of view.

In general we will follow the notation in Green and Zerna (1968) for both geometric quantities and quantities such as force and stress.

There are many books and papers on shell theory, for example Timoshenko and Woinowsk-Krieger (1959), Flügge (1973), Calladine (1982) and Axelrad and Emmerling (1988), but in our view the notation in Green and Zerna is to be preferred because it has the added advantage that it is essentially as that used by Einstein for the general theory of relativity (Dirac, 1975), where, of course, the stress-energy tensor causes the curvature of space-time so that stress is essentially a geometric concept with principal values equal to the density and the 3 principal stresses.

The references cited in the previous paragraph could be loosely described as "engi- neering" texts. But there are numerous other relevant references in mathematics, architectural geometry and computer graphics including Kupferman et al. (2017), Vouga et al. (2012), Yang et al. (2011), Jiang et al. (2015) and Diamanti et al. (2014).

2. The application of virtual work to pin jointed space structures

Before considering shell structures let us examine a simpler case, that of pin jointed space structures. This enables us to understand the fundamental idea behind the application of virtual work. One could imagine a shell, or even a 3 dimensional continuum, as being made up of a very fine structure of pin ended members, and indeed this is done in the numerical methods, peridynamics (Silling and Lehoucq, 2010) and smoothed particle hydrodynamics (Monaghan, 2012). At the molecular level the assumptions of continuum mechanics break down, so that a pin jointed framework is theoretically just as realistic as a continuum.

The length L_{i} of the member with ends at nodes numbered a_i and b_i is given by Pythagoras' theorem,

$$L_i^2 = (\mathbf{x}_{a_i} - \mathbf{x}_{b_i}) \cdot (\mathbf{x}_{a_i} - \mathbf{x}_{b_i}) \tag{1}$$

where $\mathbf{x}_j$ is the position vector of node j.

Differentiating with respect to time we obtain

$$2L_i\dot{L}_i = 2\,(\mathbf{x}_{a_i} - \mathbf{x}_{b_i}) \cdot (\dot{\mathbf{x}}_{a_i} - \dot{\mathbf{x}}_{b_i})$$

so that

$$\dot{L}_i = \frac{(\mathbf{x}_{a_i} - \mathbf{x}_{b_i}) \cdot (\dot{\mathbf{x}}_{a_i} - \dot{\mathbf{x}}_{b_i})}{L_i} \,. \tag{2}$$

The virtual work equation is

$$\sum_{j=0}^{n} (\mathbf{p}_j \cdot \dot{\mathbf{x}}_j) = \sum_{i=0}^{m} (T_i \dot{L}_i) \tag{3}$$

where the members are numbered from 0 to m and the nodes are numbered from 0 to n. p j is the load applied to node j and T_i is the tension in member i.

Therefore using (2),

$$\sum_{j=0}^{n} (\mathbf{p}_j \cdot \dot{\mathbf{x}}_j) = \sum_{i=0}^{m} \left(\frac{T_i}{L_i} (\mathbf{x}_{a_i} - \mathbf{x}_{b_i}) \cdot (\dot{\mathbf{x}}_{a_i} - \dot{\mathbf{x}}_{b_i}) \right) .$$

But this applies for *any virtual set of nodal velocities*. So if we imagine that all the nodes are stationary except for node j

$$\mathbf{p}_j \cdot \dot{\mathbf{x}}_j = \sum_{i=0}^{m} \left(\frac{T_i}{L_i} (\mathbf{x}_{a_i} - \mathbf{x}_{b_i}) \cdot (\delta_{ja_i} \dot{\mathbf{x}}_j - \delta_{jb_i} \dot{\mathbf{x}}_j) \right)$$

in which

$$\begin{aligned} \delta_{ja_i} &= 0 \quad \text{if} \quad j \neq a_i \\ &= 1 \quad \text{if} \quad j = a_i \,. \end{aligned} \tag{4}$$

Therefore since the direction of $\dot{\mathbf{x}}_j$ is arbitrary,

$$\mathbf{p}_j = \sum_{i=0}^{m} \left(\frac{T_i}{L_i} (\delta_{ja_i} - \delta_{jb_i}) (\mathbf{x}_{a_i} - \mathbf{x}_{b_i}) \right) \tag{5}$$

which are the equilibrium equations that we could have obtained by resolving forces at the nodes. We do not actually need the δ_{jai} and δ_{jai} in a numerical implementation since we sum over all the members adding forces to the nodes as appropriate.

Thus any 2 of equations (2), (3) or (5) imply the third.

Equation (3) looks like a statement of conservation of energy, rate of work being done by loads equal rate of work being absorbed by members, but there is no suggestion the the deformation or the forces are "real", provided that $\dot{\mathbf{x}}_j$ and $\dot{L}_i$ are geometrically compatible and $\mathbf{p}_j$ and T_i are in equilibrium with each other.

Note there is absolutely no assumption that the displacements are small, although we do have to consider velocities or increments

of displacement. L_i is the current length of a member, which might be stretched to many times its original length. Similarly $\mathbf{x}_j$ are the current positions and the initial member lengths and positions do not appear at all. Indeed there may be no meaningful concept of initial lengths and positions since we may have to move nodes and stretch or compress members to fit the structure together, possibly deforming the members permanently.

There is also no assumption about material properties, the members do not have to be elastic and we have made no assumption about the relationship between T_i and L_i.

2.1 The stiffness matrix

We are not concerned with stiffness in this paper. However, since there is a great deal of confusion attached to the subject, let us differentiate (5) with respect to time,

$$\dot{\mathbf{p}}_j = \sum_{i=0}^{m} \left(\frac{d}{dt}\left(\frac{T_i}{L_i}\right)\left(\delta_{ja_i} - \delta_{jb_i}\right)\left(\mathbf{x}_{a_i} - \mathbf{x}_{b_i}\right)\right) + \sum_{i=0}^{m} \left(\frac{T_i}{L_i}\left(\delta_{ja_i} - \delta_{jb_i}\right)\left(\dot{\mathbf{x}}_{a_i} - \dot{\mathbf{x}}_{b_i}\right)\right). \tag{6}$$

If the members are elastic, or if they are plastic and we have sufficient knowledge of past deformation, we can write

$$\frac{d}{dt}\left(\frac{T_i}{L_i}\right) = s_i \frac{\dot{L}_i}{L_i} = s_i \frac{d}{dt}\left(\log L_i\right) \tag{7}$$

where the member stiffness s_i may itself be a function of the current L_i.

Then

$$\begin{aligned} \dot{\mathbf{p}}_j &= \sum_{i=0}^{m} \left(s_i \frac{\dot{L}_i}{L_i}\left(\delta_{ja_i} - \delta_{jb_i}\right)\left(\mathbf{x}_{a_i} - \mathbf{x}_{b_i}\right)\right) + \sum_{i=0}^{m} \left(\frac{T_i}{L_i}\left(\delta_{ja_i} - \delta_{jb_i}\right)\left(\dot{\mathbf{x}}_{a_i} - \dot{\mathbf{x}}_{b_i}\right)\right) \\ &= \sum_{i=0}^{m} \left(\left(\delta_{ja_i} - \delta_{jb_i}\right)\left(s_i \frac{\left(\mathbf{x}_{a_i} - \mathbf{x}_{b_i}\right)\left(\mathbf{x}_{a_i} - \mathbf{x}_{b_i}\right)}{L_i^2} + \frac{T_i}{L_i}\mathbf{I}\right)\cdot\left(\dot{\mathbf{x}}_{a_i} - \dot{\mathbf{x}}_{b_i}\right)\right) \end{aligned} \tag{8}$$

in which $\mathbf{I}$ is the unit tensor in 3 dimensional space.

The term containing s_i is the elastic stiffness and the term containing $\frac{T_i}{L_i}$ is the geometric stiffness. The geometric stiffness is so called because it only depends upon the geometry and the state of stress, not the elastic properties, except in so much as they may influence the state of stress.

3. The equations of static equilibrium of shell structures

The concept of virtual work is much more difficult to grasp for shell structures than for the pin jointed structures described in section 2 because the associated mathematics involving the differential geometry of a deforming surface is complicated. Therefore, rather than leave the main results until after they have been proved, we will quote them now as an incentive to follow their derivation.

The equation of equilibrium of forces is

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{p} = 0 \,. \tag{9}$$

This is a vector equation corresponding to equilibrium in 3 directions, that is the 2 directions tangential to the surface and the normal direction. The vector $\mathbf{p}$ is the load per unit area on the structure and $\boldsymbol{\sigma}$ is a second order tensor containing the membrane stresses and the normal shear forces associated with bending. The $\nabla\cdot$ is the divergence described in Section 4.3. In the case of a structure which is not static we can include inertia forces using D'Alembert's principle.

The form of eq. (9) is essentially the same as eq. (5) and indeed much of the theory of peridynamics and smoothed particle hydrodynamics is to establish link tensions which will have the same effect as stresses in a solid or a fluid.

Equation (9) is identical to that for a 3 dimensional continuum and in relativity theory $\nabla \cdot \boldsymbol{\sigma} = 0$ is the equation for the conservation of momentum and mass-energy.

The equation of equilibrium of moments is

$$(\boldsymbol{\sigma} + \nabla \cdot \mathbf{m} + \mathbf{c}) = (\boldsymbol{\sigma} + \nabla \cdot \mathbf{m} + \mathbf{c})^{\mathrm{T}} \tag{10}$$

where the superscript T means the transpose. Thus we are saying that the second order tensor $(\boldsymbol{\sigma} + \nabla \cdot \mathbf{m} + \mathbf{c})$ is symmetric, which is equivalent to the resultant moment about 3 directions being zero, again 2 directions tangent to the surface and the surface normal. $\mathbf{m}$ is the third order tensor containing the moments in the surface, both "ordinary" bending and twisting moments about axes in the plane of the surface and "geodesic" bending moments about the normal which are relevant to gridshell structures with continuous laths, like the Mannheim Multihalle

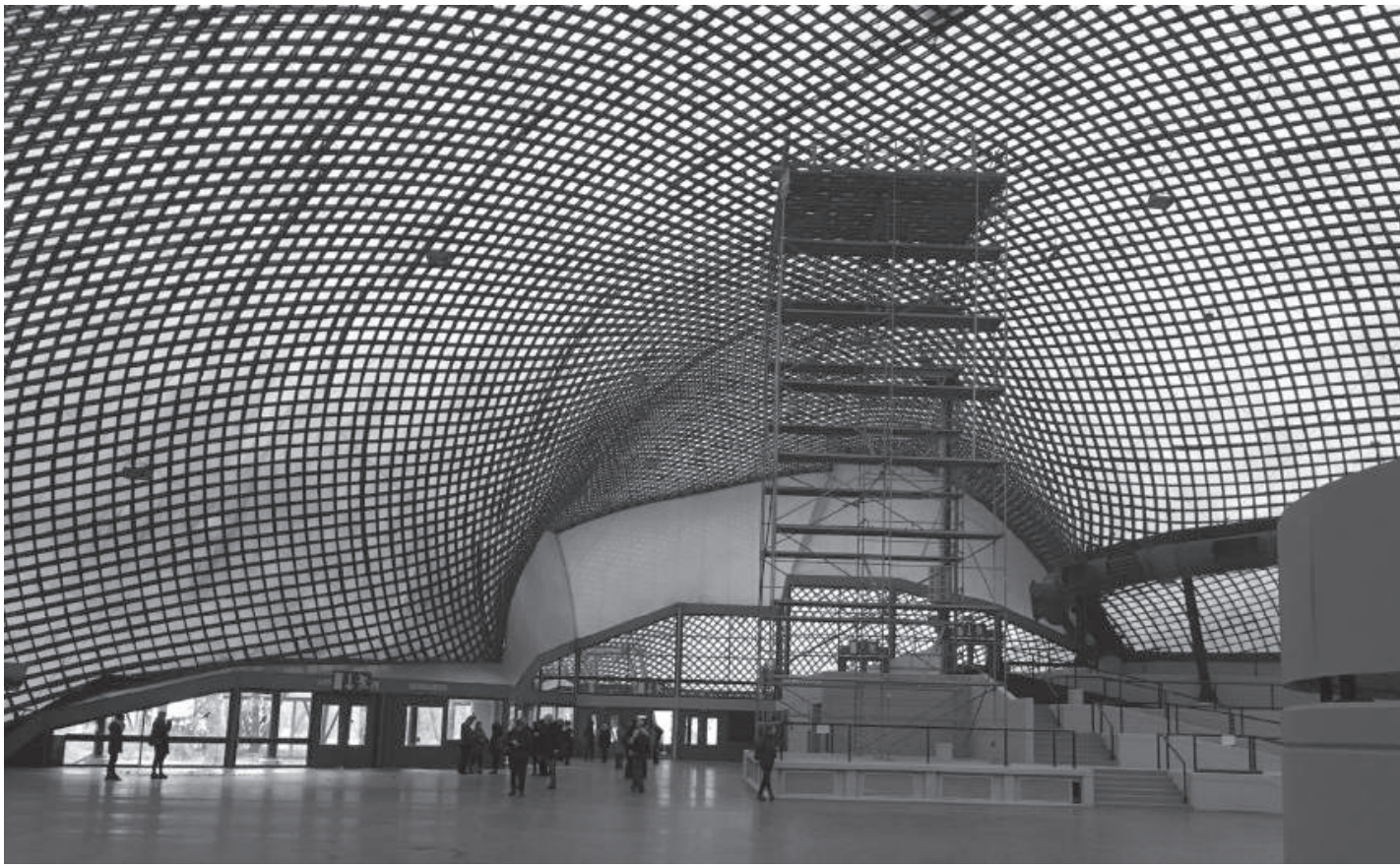


Figure 1: Mannheim Multihalle, Germany 1974. Frei Otto, Carlfried Mutschler and Winfried Langner, Ove Arup and Partners.

gridshell (**Fig. 1**) or a kitchen sieve made from a woven wire mesh. Both the Mannheim Multihalle and a sieve have a fine grid, making it appropriate to treat them as continua. The second order tensor $\mathbf{c}$ is the loading couple per unit area, which is zero in almost all practical applications, and again we can include the effect of acceleration using D'Alembert's principle.

To our knowledge this is the first time that this concept of geodesic moments has been introduced.

4. Differential geometry of a stationary surface

4.1 The base vectors and the first fundamental form

We first need to define the geometric quantities of a stationary surface before we can examine how they vary as the surface deforms. The contents of this section will be familiar to some readers, but we need

to define all our terms so that we can differentiate them with respect to time in Section 5 where we consider a deforming surface.

Imagine a surface defined by the curvilinear coordinates, θ^1 and θ^2. The position vector of a typical point on the surface and its Cartesian coordinates are given by

$$\mathbf{r}(\theta^1,\theta^2) = x(\theta^1,\theta^2)\,\mathbf{i} + y(\theta^1,\theta^2)\,\mathbf{j} + z(\theta^1,\theta^2)\,\mathbf{k} \tag{11}$$

in which $\mathbf{i}$, $\mathbf{j}$ and $\mathbf{k}$ are unit base vectors in the direction of the Cartesian axes, x, y and z. We use the surface coordinates or parameters θ^1 and θ^2 with superscripts to replace the more usual u and v so that we can use the tensor notation, which is indispensable if one is to consider both geometry and structural concepts such as stress. Following the notation in Green and Zerna (1968), the covariant base vectors, $\mathbf{a}_\alpha$, in which α is equal to 1 or 2, are given by

$$\mathbf{a}_\alpha = \frac{\partial \mathbf{r}}{\partial \theta^\alpha} = \mathbf{r}_{,\alpha}\,. \tag{12}$$

A comma will be used to denote partial differentiation. $\mathbf{a}_1$ is tangential to a curve $\theta^2 =$ constant on the surface and $\mathbf{a}_2$ is tangential to a curve $\theta^1 =$ constant. In general neither $\mathbf{a}_1$ nor $\mathbf{a}_2$ will be unit vectors, and they will not be perpendicular to each other.

The square of the distance between two adjacent points on the surface is equal to

$$\sum_{\alpha=1}^{2}\sum_{\beta=1}^{2} a_{\alpha\beta}\,d\theta^\alpha d\theta^\beta = a_{\alpha\beta}\,d\theta^\alpha d\theta^\beta \qquad a_{\alpha\beta} = a_{\beta\alpha} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta \tag{13}$$

in which we have used the Einstein summation convention for the implied summation for repeated subscripts and superscripts. $a_{\alpha\beta}$ are the covariant components of the metric tensor, also known as the coefficients of the first fundamental form. Eisenhart (1947) uses $g_{\alpha\beta}$ and Struik (1961) uses E, F and G.

The unit normal is

$$\mathbf{n} = \frac{\mathbf{a}_1 \times \mathbf{a}_2}{|\mathbf{a}_1 \times \mathbf{a}_2|} \tag{14}$$

and here our notation differs from Green and Zerna who use $\mathbf{a}_3$. We will also have occasion to use the contravariant base vectors, $\mathbf{a}^\alpha$, defined by

$$\begin{aligned} \mathbf{a}^\alpha \cdot \mathbf{a}_\beta = \delta_\beta^\alpha = 0 \text{ if } \alpha \neq \beta \\ = 1 \text{ if } \alpha = \beta \\ \mathbf{a}^\alpha \cdot \mathbf{n} = 0 \,. \end{aligned} \tag{15}$$

$\mathbf{a}^1$ lies in the tangent plane to the surface perpendicular to a curve $\theta^1 =$ constant and its magnitude is such that $\mathbf{a}^1 \cdot \mathbf{a}_1 = 1$, and similarly for $\mathbf{a}^2$.

Using $a_{\alpha\beta}$ and

$$a^{\alpha\beta} = a^{\beta\alpha} = \mathbf{a}^\alpha \cdot \mathbf{a}^\beta \tag{16}$$

we can raise and lower indices using equations such as

$$\begin{aligned} \mathbf{a}^\alpha &= a^{\alpha\beta}\mathbf{a}_\beta \\ \mathbf{a}_\alpha &= a_{\alpha\beta}\mathbf{a}^\beta \\ q_{\alpha\beta} &= a_{\alpha\lambda} q^{\lambda}_{\cdot\,\beta} \end{aligned}$$

in which the dot in $q^{\lambda}_{\cdot\,\beta}$ is used maintain the order of indices. If the second order tensor $\mathbf{q}$ is symmetric then $q_{\alpha\beta} = q_{\beta\alpha}$ and we can dispense with the dot and write q^λ_β.

The components of the permutation tensor,

$$\begin{aligned} \varepsilon_{\lambda\mu} &= -\varepsilon_{\mu\lambda} \\ \varepsilon_{11} = 0, \varepsilon_{12} &= -\varepsilon_{21} = \sqrt{\mathrm{a}}, \varepsilon_{22} = 0 \end{aligned} \tag{17}$$

in which

$$a = |\mathbf{a}_1 \times \mathbf{a}_2|^2 = a_{11}a_{22} - (a_{12})^2 \tag{18}$$

are used to perform the vector products,

$$\begin{aligned} \mathbf{a}_\alpha \times \mathbf{a}_\beta &= \varepsilon_{\alpha\beta}\mathbf{n} \\ \mathbf{n} \times \mathbf{a}^\alpha &= \varepsilon^{\alpha\beta}\mathbf{a}_\beta \,. \end{aligned} \tag{19}$$

a is not a scalar since it is a property of the coordinate system.

4.2 The second fundamental form and the Christoffel symbols

The components of the normal curvature tensor, or coefficients of the second fundamental form, are

$$b_{\alpha\beta} = b_{\beta\alpha} = \mathbf{a}_{\alpha,\beta} \cdot \mathbf{n} = \frac{\partial^2 \mathbf{r}}{\partial\theta^\alpha \partial\theta^\beta} \cdot \mathbf{n} = -\mathbf{a}_\alpha \cdot \mathbf{n}_{,\beta} \,. \tag{20}$$

Eisenhart (1947) uses $d_{\alpha\beta}$ and Struik (1961) uses e, f and g. $b_{\alpha\beta}$, together with $a_{\alpha\beta}$, contain all the information about the normal curvature and twist of the surface, including the principal curvatures and their directions. The Gaussian curvature,

$$K = \frac{b_{11}b_{22} - (b_{12})^2}{a_{11}a_{22} - (a_{12})^2} \tag{21}$$

is equal to the product of the principal curvatures and the mean curvature,

$$H = \frac{1}{2} a^{\alpha\beta} b_{\alpha\beta} = \frac{1}{2} b^\alpha_\alpha \tag{22}$$

is the average of the principal curvatures.

$b_{\alpha\beta}$ give the component of $\mathbf{a}_{\alpha,\beta}$, normal to the surface and the Christoffel symbols,

$$\Gamma^\chi_{\alpha\beta} = \mathbf{a}_{\alpha,\beta} \cdot \mathbf{a}^\chi = \frac{1}{2} a^{\chi\eta} \left(a_{\eta\alpha,\beta} + a_{\beta\eta,\alpha} - a_{\alpha\beta,\eta} \right) \tag{23}$$

give the components of $\mathbf{a}_{\alpha,\beta}$ tangent to the surface. Note that the Christoffel symbols are not the components of a tensor because they represent properties of the coordinate system rather than the surface itself.

The fundamental theorem of surface theory states that the tensor components $a_{\alpha\beta}$ and $b_{\alpha\beta}$ define the shape of a surface, but not its position and orientation in space. $a_{\alpha\beta}$ and $b_{\alpha\beta}$ are not independent since they come from differentiating 3 Cartesian coordinates with respect to the surface coordinates. Writing

$$\mathbf{a}_{\alpha,\beta\chi} = \left(b_{\alpha\beta}\mathbf{n} + \Gamma^\lambda_{\alpha\beta}\mathbf{a}_\lambda \right)_{,\chi} = \mathbf{a}_{\alpha,\chi\beta} = \left(b_{\alpha\chi}\mathbf{n} + \Gamma^\lambda_{\alpha\chi}\mathbf{a}_\lambda \right)_{,\beta} \tag{24}$$

gives the 3 conditions that the surface "fits together". These are known as Gauss's theorema egregium and the Codazzi-Mainardi equations,

$$\nabla_\lambda b_{\alpha\beta} = \nabla_\alpha b_{\lambda\beta} \tag{25}$$

in which the covariant derivative,

$$\nabla_\lambda b_{\alpha\beta} = b_{\alpha\beta,\lambda} - b_{\eta\beta}\Gamma^\eta_{\alpha\lambda} - b_{\alpha\eta}\Gamma^\eta_{\beta\lambda}\,. \tag{26}$$

4.3 Component free notation and the gradient of a tensor

It is rather unsatisfactory to only be able to talk about the components of a tensor, rather than the tensor itself, and we can write the second order normal curvature tensor, $\mathbf{b}$, as

$$\mathbf{b} = b_{\alpha\beta}\mathbf{a}^\alpha\mathbf{a}^\beta = b^\beta_\alpha\mathbf{a}^\alpha\mathbf{a}_\beta = b^{\alpha\beta}\mathbf{a}_\alpha\mathbf{a}_\beta \tag{27}$$

in which the product $\mathbf{a}^\alpha\,\mathbf{a}^\beta$, without a dot or a cross is the tensor product, or outer product, sometimes written with a $\otimes$, defined by

$$\begin{aligned}(\mathbf{de})\cdot\mathbf{g} &= \mathbf{d}\,(\mathbf{e}\cdot\mathbf{g})\\ \mathbf{g}\cdot(\mathbf{de}) &= (\mathbf{g}\cdot\mathbf{d})\,\mathbf{e}\end{aligned} \tag{28}$$

where $\mathbf{d}$, $\mathbf{e}$ and $\mathbf{g}$ are any vectors. We shall also use the double dot notation,

$$\begin{aligned}(\mathbf{de})\cdot\cdot(\mathbf{gh}) &= (\mathbf{d}\cdot\mathbf{g})\,(\mathbf{e}\cdot\mathbf{h})\\ (\mathbf{cde})\cdot\cdot\cdot(\mathbf{ghpq}) &= (\mathbf{c}\cdot\mathbf{g})\,(\mathbf{d}\cdot\mathbf{h})\,(\mathbf{e}\cdot\mathbf{p})\,\mathbf{q}\end{aligned} \tag{29}$$

which can be extended to any number of dots.

Let us now imagine that we have a vector field on a surface, that is a first order tensor field,

$$\mathbf{v} = v^\alpha\mathbf{a}_\alpha + v\mathbf{n} \tag{30}$$

with components tangential to the surface, v^α, and normal to the surface, v. We will define the gradient of this vector field as the second order tensor,

$$\begin{aligned}\nabla\mathbf{v} &= \mathbf{a}^\beta\mathbf{v}_{,\beta} = \mathbf{a}^\beta\left(v^\alpha\mathbf{a}_\alpha + v\mathbf{n}\right)_{,\beta}\\ &= \mathbf{a}^\beta\left(\left(\nabla_\beta v^\alpha - vb^\alpha_\beta\right)\mathbf{a}_\alpha + \left(v^\alpha b_{\alpha\beta} + v_{,\beta}\right)\mathbf{n}\right)\end{aligned} \tag{31}$$

in which the covariant derivative,

$$\nabla_\beta v^\alpha = v^\alpha{}_{,\beta} + v^\eta \Gamma^\alpha_{\eta\beta} \,. \tag{32}$$

The covariant derivative of the components of the metric tensor and of the permutation tensor are all zero and

$$\nabla \mathbf{n} = -\mathbf{b} \,. \tag{33}$$

The divergence of the vector $\mathbf{v}$,

$$\nabla \cdot \mathbf{v} = \mathbf{a}^\alpha \cdot \mathbf{v}_{,\alpha} = \nabla_\alpha v^\alpha - v b^\alpha_\alpha \tag{34}$$

in which

$$\nabla_\alpha v^\alpha = v^\alpha{}_{,\alpha} + v^\lambda \Gamma^\alpha_{\lambda\alpha} = \frac{(v^\alpha \sqrt{a})_{,\alpha}}{\sqrt{a}} \,. \tag{35}$$

4.4 The divergence theorem on a surface

This last result enables us to prove the divergence theorem on a surface for a vector with no normal component, $\mathbf{v} = v^\alpha \mathbf{a}_\alpha$,

$$\begin{aligned} \int_A \nabla \cdot \mathbf{v} dA &= \int_A \nabla_\alpha v^\alpha \sqrt{a} d\theta^1 d\theta^2 = \int_A (v^\alpha \sqrt{a})_{,\alpha} d\theta^1 d\theta^2 \\ &= \int_{\partial A} v^\alpha \varepsilon_{\alpha\beta} d\theta^\beta = \int_{\partial A} (d\mathbf{r} \times \mathbf{n}) \cdot \mathbf{v} \end{aligned} \tag{36}$$

in which ∂A is the boundary of the surface A. This result is central to the application of virtual work to surface structures.

5. Deformation of a surface

5.1 Velocity

Now let us imagine a moving and deforming surface defined by the curvilinear coordinates, θ^1 and θ^2, and time t. The position vector of a typical point on the surface is now given by

$$\mathbf{r}(\theta^1, \theta^2, t) = x(\theta^1, \theta^2, t)\,\mathbf{i} + y(\theta^1, \theta^2, t)\,\mathbf{j} + z(\theta^1, \theta^2, t)\,\mathbf{k} \,. \tag{37}$$

The velocity vector is

$$\mathbf{u} = u_\mu \mathbf{a}^\mu + u\mathbf{n} = \frac{\partial \mathbf{r}}{\partial t} \,. \tag{38}$$

We imagine that the coordinates move with the surface, like the laths of a gridshell.

5.2 Strain rate and angular velocity

The gradient of the velocity is

$$\begin{aligned} \nabla\mathbf{u} = \mathbf{a}^\lambda \mathbf{u}_{,\lambda} &= \left(\nabla_\lambda u_\mu - b_{\lambda\mu} u\right) \mathbf{a}^\lambda \mathbf{a}^\mu + \left(u_\mu b^\mu_\lambda + \nabla_\lambda u\right) \mathbf{a}^\lambda \mathbf{n} \\ &= \left(\gamma_{\lambda\mu} + \omega\varepsilon_{\lambda\mu}\right) \mathbf{a}^\lambda \mathbf{a}^\mu + \varepsilon_{\lambda\mu}\omega^\mu \mathbf{a}^\lambda \mathbf{n} \end{aligned} \tag{39}$$

in which the symmetric strain rate tensor,

$$\begin{aligned} \boldsymbol{\gamma} = \boldsymbol{\gamma}^{\mathrm{T}} &= \frac{1}{2}\left((\nabla\mathbf{u} - \nabla\mathbf{u}\cdot\mathbf{nn}) + (\nabla\mathbf{u} - \nabla\mathbf{u}\cdot\mathbf{nn})^{\mathrm{T}}\right) \\ &= \gamma_{\lambda\mu}\mathbf{a}^\lambda\mathbf{a}^\mu \end{aligned} \tag{40}$$

and the anti-symmetric angular velocity tensor, or vorticity tensor,

$$\begin{aligned} \boldsymbol{\omega} = -\boldsymbol{\omega}^{\mathrm{T}} &= \frac{1}{2}\left((\nabla\mathbf{u} - \mathbf{n}\nabla\mathbf{u}\cdot\mathbf{n}) - (\nabla\mathbf{u} - \mathbf{n}\nabla\mathbf{u}\cdot\mathbf{n})^{\mathrm{T}}\right) \\ &= \omega\varepsilon_{\lambda\mu}\mathbf{a}^\lambda\mathbf{a}^\mu + \varepsilon_{\lambda\mu}\omega^\mu\left(\mathbf{a}^\lambda\mathbf{n} - \mathbf{n}\mathbf{a}^\lambda\right) \,. \end{aligned} \tag{41}$$

ω is defined by only 3 quantities, ω, ω^1 and ω^2, which could be considered to be the components of a vector.

We can write

$$\nabla\mathbf{u} = \boldsymbol{\gamma} + \boldsymbol{\omega} - \mathbf{nn}\cdot\boldsymbol{\omega} \tag{42}$$

$$\mathbf{u}_{,\lambda} = \mathbf{a}_\lambda \cdot (\boldsymbol{\gamma} + \boldsymbol{\omega}) \tag{43}$$

$$\frac{\partial \mathbf{n}}{\partial t} = \mathbf{n}\cdot(\boldsymbol{\gamma} + \boldsymbol{\omega}) = \mathbf{n}\cdot\boldsymbol{\omega} \tag{44}$$

and we have the results

$$\frac{\partial a_{\alpha\beta}}{\partial t} = \mathbf{u}_{,\alpha}\cdot\mathbf{a}_\beta + \mathbf{a}_\alpha\cdot\mathbf{u}_{,\beta} = \mathbf{a}_\alpha\cdot\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^{\mathrm{T}}\right)\cdot\mathbf{a}_\beta = 2\gamma_{\alpha\beta} \tag{45}$$

$$\frac{\partial a^{\alpha\beta}}{\partial t} = -2\gamma^{\alpha\beta} \tag{46}$$

$$\frac{1}{\sqrt{a}}\frac{\partial\sqrt{a}}{\partial t} = a^{\alpha\beta}\gamma_{\alpha\beta} \,. \tag{47}$$

5.3 Rate of bending, or rate of change of normal and geodesic curvature

In order to find how the coefficients of the second fundamental form and the Christoffel symbols vary with time we need

$$\begin{aligned}\frac{\partial \mathbf{a}_{\alpha,\beta}}{\partial t} &= \frac{\partial}{\partial t}\left(b_{\alpha\beta}\mathbf{n}+\Gamma^{\lambda}_{\alpha\beta}\mathbf{a}_{\lambda}\right)\\ &= \frac{\partial b_{\alpha\beta}}{\partial t}\mathbf{n}+\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t}\mathbf{a}_{\lambda}+b_{\alpha\beta}\frac{\partial \mathbf{n}}{\partial t}+\Gamma^{\lambda}_{\alpha\beta}\mathbf{u}_{,\lambda}\\ &= \mathbf{u}_{,\alpha\beta}=\mathbf{a}_{\alpha}\cdot\left(\mathbf{a}^{\lambda}\mathbf{u}_{,\lambda}\right)_{,\beta}-\mathbf{a}_{\alpha}\cdot\mathbf{a}^{\lambda}{}_{,\beta}\mathbf{u}_{,\lambda}=\mathbf{a}_{\alpha}\cdot\left(\mathbf{a}^{\lambda}\mathbf{u}_{,\lambda}\right)_{,\beta}+\Gamma^{\lambda}_{\alpha\beta}\mathbf{u}_{,\lambda}\\ &= \left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\right)\cdot\cdot\nabla\nabla\mathbf{u}+\Gamma^{\lambda}_{\alpha\beta}\mathbf{u}_{,\lambda}\end{aligned} \tag{48}$$

and

$$\begin{aligned}\left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\right)\cdot\cdot\nabla\nabla\mathbf{u} &= \left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\right)\cdot\cdot\nabla\left(\gamma+\omega-\mathbf{n}\mathbf{n}\cdot\omega\right)\\ &= \left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\right)\cdot\cdot\nabla\left(\gamma+\omega\right)+b_{\alpha\beta}\mathbf{n}\cdot\omega\\ &= \left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\right)\cdot\cdot\nabla\left(\gamma+\omega\right)+b_{\alpha\beta}\frac{\partial \mathbf{n}}{\partial t}\end{aligned} \tag{49}$$

which mean that

$$\frac{\partial b_{\alpha\beta}}{\partial t}\mathbf{n}+\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t}\mathbf{a}_{\lambda}=\left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\right)\cdot\cdot\nabla\nabla\mathbf{u}-b_{\alpha\beta}\frac{\partial \mathbf{n}}{\partial t}=\left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\right)\cdot\cdot\nabla\left(\boldsymbol{\gamma}+\boldsymbol{\omega}\right) \tag{50}$$

$$\frac{\partial b_{\alpha\beta}}{\partial t}=\left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\mathbf{n}\right)\cdots\nabla\left(\boldsymbol{\gamma}+\boldsymbol{\omega}\right) \tag{51}$$

$$\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t}=\left(\mathbf{a}_{\beta}\mathbf{a}_{\alpha}\mathbf{a}^{\lambda}\right)\cdots\nabla\left(\boldsymbol{\gamma}+\boldsymbol{\omega}\right)\,. \tag{52}$$

Equation (51) tells us about the rate of change of normal curvature, while eq. (52) tells us about the rate of change of geodesic curvature. Equation (52) leads to the somewhat surprising conclusion that $\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t}$ are the components of a tensor, even though $\Gamma^{\lambda}_{\alpha\beta}$ are not, although in the general theory of relativity the derivatives of the Christoffel symbols lead to the components of the Riemann-Christoffel tensor.

Differentiating eq. (23) with respect to time,

$$\begin{aligned}\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t} &= \frac{1}{2}\frac{\partial a^{\alpha\beta}}{\partial t}\left(a_{\eta\alpha,\beta}+a_{\beta\eta,\alpha}-a_{\alpha\beta,\eta}\right)+\frac{1}{2}a^{\lambda\eta}\left(\frac{\partial a_{\beta\eta,\alpha}}{\partial t}+\frac{\partial a_{\eta\alpha,\beta}}{\partial t}-\frac{\partial a_{\alpha\beta,\eta}}{\partial t}\right)\\ &= -2\gamma^{\lambda\eta}a_{\mu\eta}\Gamma^{\mu}_{\alpha\beta}+a^{\lambda\eta}\begin{pmatrix}\nabla_{\beta}\gamma_{\eta\alpha}+\gamma_{\chi\alpha}\Gamma^{\chi}_{\eta\beta}+\gamma_{\eta\chi}\Gamma^{\chi}_{\alpha\beta}\\ +\nabla_{\alpha}\gamma_{\beta\eta}+\gamma_{\chi\eta}\Gamma^{\chi}_{\alpha\beta}+\gamma_{\beta\chi}\Gamma^{\chi}_{\eta\alpha}\\ -\nabla_{\eta}\gamma_{\alpha\beta}-\gamma_{\chi\beta}\Gamma^{\chi}_{\alpha\eta}-\gamma_{\alpha\chi}\Gamma^{\chi}_{\beta\eta}\end{pmatrix}\\ &= a^{\lambda\eta}\left(\nabla_{\beta}\gamma_{\eta\alpha}+\nabla_{\alpha}\gamma_{\beta\eta}-\nabla_{\eta}\gamma_{\alpha\beta}\right)\end{aligned} \tag{53}$$

which confirms that $\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t}$ are indeed the components of a tensor. But note that the rate of change of the Christoffel symbols of the first kind, $\frac{\partial \Gamma_{\alpha\beta\lambda}}{\partial t}$ are *not* the components of a tensor.

For future use we will combine the components of $\frac{\partial b_{\alpha\beta}}{\partial t}$ and $\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t}$ into the components of one third order tensor, $\boldsymbol{\beta}$,

$$\begin{aligned}\boldsymbol{\beta} &= \mathbf{a}^{\alpha}\mathbf{a}^{\beta}\left(\beta_{\alpha\beta}\mathbf{n}+\beta_{\alpha\beta}^{\cdot}{}^{\lambda}\mathbf{a}_{\lambda}\right)=\mathbf{a}^{\alpha}\mathbf{a}^{\beta}\left(\frac{\partial b_{\alpha\beta}}{\partial t}\mathbf{n}+\frac{\partial \Gamma^{\lambda}_{\alpha\beta}}{\partial t}\mathbf{a}_{\lambda}\right)\\ &= \mathbf{a}^{\alpha}\mathbf{a}^{\beta}\left(\mathbf{a}_{\alpha}\mathbf{a}_{\beta}\right)\cdot\cdot\nabla\left(\boldsymbol{\gamma}+\boldsymbol{\omega}\right)\end{aligned} \tag{54}$$

which we might call the rate of bending tensor. It includes both the rate of change of the normal curvature as well as the rate of change of the geodesic curvature of lines on the surface.

6. Puissances virtuelles – rate of virtual work being done on a surface structure

6.1 The rate of virtual work

The rate of virtual work being done on a surface A with boundary ∂A as it moves and deforms is

$$W=\int_{\partial A}(d\mathbf{r}\times\mathbf{n})\cdot(\boldsymbol{\sigma}\cdot\mathbf{u}+\mathbf{m}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega}))+\int_{A}(\mathbf{p}\cdot\mathbf{u}+\mathbf{c}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega}))\,dA\,. \tag{55}$$

We do not have to justify this equation in any way, except to say that we assume that W only depends upon $\mathbf{u}$ and $(\boldsymbol{\gamma}+\boldsymbol{\omega})$ and the loading on the surface itself and its boundary. Equation (55) is the definition of the vector $\mathbf{p}$, the second order tensors $\mathbf{c}$ and $\boldsymbol{\sigma}$ and the third order tensor $\mathbf{m}$, whose physical interpretation were given in Section 3.

We could add other terms, particularly those to include shear deformation as in a Timoshenko (1921) or Cosserat (1909) beam or shell. Such deformation was important in the design of the Mannheim Multihalle gridshell because of the flexible connection between the upper and lower parallel members. However, while not difficult to add such deformation, it introduces further complexity which is not relevant to this paper.

We can stipulate that

$$\mathbf{n}\cdot\boldsymbol{\sigma}=0 \tag{56}$$
$$\mathbf{n}\cdot\mathbf{m}=0 \tag{57}$$

because $(d\mathbf{r}\times\mathbf{n})$ lies in the plane of the surface. We will also stipulate that

$$(\mathbf{a}_\alpha\mathbf{n})\cdot\cdot\mathbf{m}=0 \tag{58}$$
$$\mathbf{n}\cdot\mathbf{c}=0 \tag{59}$$

and include the work done by that part of $\boldsymbol{\omega}$ containing $\mathbf{n}$ from the "right hand part" of the third order tensor $\mathbf{m}$. Thus we can write the components

$$\boldsymbol{\sigma}=\mathbf{a}_\alpha\left(\sigma^{\alpha\beta}\mathbf{a}_\beta+\sigma^{\alpha}\mathbf{n}\right) \tag{60}$$
$$\mathbf{m}=\mathbf{a}_\alpha\mathbf{a}_\beta\left(m^{\alpha\beta}\mathbf{n}+m^{\alpha\beta}_{\cdot\cdot\ \chi}\mathbf{a}^\chi\right)\ . \tag{61}$$

$\boldsymbol{\sigma}^{\alpha\beta}$ are the components of membrane stress and $\boldsymbol{\sigma}^{\alpha}$ are the components of normal shear force. $m^{\alpha\beta}$ are the components of normal bending and twisting moment and $m^{\alpha\beta}_{\cdot\cdot}$ are the components of the geodesic bending moments.

We can now use the divergence theorem, eq. (36), to change the boundary integral in eq. (55) to a surface integral,

$$W=\int_A\nabla\cdot(\boldsymbol{\sigma}\cdot\mathbf{u}+\mathbf{m}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega}))\,dA+\int_A(\mathbf{p}\cdot\mathbf{u}+\mathbf{c}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega}))\,dA \tag{62}$$

in which

$$\begin{aligned}\nabla\cdot(\boldsymbol{\sigma}\cdot\mathbf{u}+\mathbf{m}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega}))&=\mathbf{a}^\alpha\cdot(\boldsymbol{\sigma}\cdot\mathbf{u}+\mathbf{m}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega}))_{,\alpha}\\&=\nabla\cdot\boldsymbol{\sigma}\cdot\mathbf{u}+\nabla\cdot\mathbf{m}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega})\\&+\boldsymbol{\sigma}\cdot\cdot(\mathbf{a}^\alpha\mathbf{a}_\alpha\cdot\nabla\mathbf{u})+\mathbf{m}\cdot\cdot\cdot(\mathbf{a}^\alpha\mathbf{a}_\alpha\cdot\nabla(\boldsymbol{\gamma}+\boldsymbol{\omega}))\\&=\nabla\cdot\boldsymbol{\sigma}\cdot\mathbf{u}+\nabla\cdot\mathbf{m}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega})\\&+\boldsymbol{\sigma}\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega})+\mathbf{m}\cdot\cdot\cdot\boldsymbol{\beta}\end{aligned} \tag{63}$$

so that

$$W=\int_A((\nabla\cdot\boldsymbol{\sigma}+\mathbf{p})\cdot\mathbf{u}+(\boldsymbol{\sigma}+\nabla\cdot\mathbf{m}+\mathbf{c})\cdot\cdot(\boldsymbol{\gamma}+\boldsymbol{\omega})+\mathbf{m}\cdot\cdot\cdot\boldsymbol{\beta})\,dA\ . \tag{64}$$

6.2 The equilibrium equations

If $\mathbf{u}$ and $\boldsymbol{\omega}$ are such that we have a rigid body motion, then $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are both zero so that

$$W = \int_A ((\nabla \cdot \boldsymbol{\sigma} + \mathbf{p}) \cdot \mathbf{u} + (\boldsymbol{\sigma} + \nabla \cdot \mathbf{m} + \mathbf{c}) \cdot\cdot \boldsymbol{\omega})\, dA\,. \tag{65}$$

We now postulate that no net work can be done in any rigid body motion of a structure in equilibrium, so that $\mathbf{W = 0}$, even if we change the location the boundary relative to the surface, adding or removing parts of the surface to just include some arbitrary part of the structure. This leads to the equation of equilibrium of forces,

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{p} = 0 \tag{66}$$

and of moments

$$(\boldsymbol{\sigma} + \nabla \cdot \mathbf{m} + \mathbf{c}) = (\boldsymbol{\sigma} + \nabla \cdot \mathbf{m} + \mathbf{c})^{\mathrm{T}} \tag{67}$$

because $\boldsymbol{\omega}$ is anti-symmetric.

6.3 The rate of work being absorbed

Finally we have the rate of work being absorbed by the structure,

$$\begin{aligned} W &= \int_A ((\boldsymbol{\sigma} + \nabla \cdot \mathbf{m} + \mathbf{c}) \cdot\cdot \boldsymbol{\gamma} + \mathbf{m} \cdots \boldsymbol{\beta})\, dA \\ &= \int_A ((\boldsymbol{\sigma} + \mathbf{c}) \cdot\cdot \boldsymbol{\gamma} + \mathbf{m} \cdots \nabla\boldsymbol{\omega})\, dA + \int_{\partial A} (d\mathbf{r} \times \mathbf{n}) \cdot \mathbf{m} \cdot\cdot \boldsymbol{\gamma}\,. \end{aligned} \tag{68}$$

The term $\mathbf{c} \cdot\cdot \boldsymbol{\gamma}$ is there because the laths of a gridshell like that of the Mannheim Multihalle can undergo different angular velocities about an axis normal to the surface, like a pair of scissors, and the loading couples $\mathbf{c}$ can therefore do work.

It is important to realize that $\dfrac{\partial b_{\alpha\beta}}{\partial t}$ may be non zero even when $\nabla\boldsymbol{\omega}$ is zero. For example when a spherical shell undergoes a uniform expansion there are no rotations but $b_{\alpha\beta}$ change because of the change in the magnitude of $\mathbf{a}_\alpha$.

6.4 The equilibrium equations and rate of work being absorbed in terms of components

We ought, perhaps, to write the equilibrium equations, eq. (66) and eq. (67), in terms of components when they lose their essential simplicity,

$$\nabla_\alpha \sigma^{\alpha\beta} - \sigma^\alpha b_\alpha^\beta + p^\beta = 0 \tag{69}$$

$$\sigma^{\alpha\beta} b_{\alpha\beta} + \nabla_\alpha \sigma^\alpha + p = 0 \tag{70}$$

$$\varepsilon_{\alpha\beta}\left(\sigma^{\alpha\beta} - m^{\eta\alpha} b_\eta^\beta + \nabla_\eta m^{\eta\alpha\beta} + c^{\alpha\beta}\right) = 0 \tag{71}$$

$$\sigma^\beta + \nabla_\alpha m^{\alpha\beta} + \left(m^{\lambda\mu\beta} + m^{\lambda\beta\mu}\right) b_{\lambda\mu} + c^\beta = 0\,. \tag{72}$$

These equations are identical to equations (10.4.4) to (10.4.7) of Green and Zerna(1968), if one makes the following changes to the notation, $n^{\alpha\beta} = \sigma^{\alpha\beta}$, $q^\alpha = \sigma^\alpha$,$\tilde{p}^\beta = -c^\beta$, the sign of $m^{\alpha\beta}$ are reversed, $|_\alpha$ is used instead of ∇_α for the covariant derivative and the components $m^{\eta\alpha\beta} = 0$ and $c^{\alpha\beta} = 0$. Green and Zerna derive their equations from the 3 dimensional equations of equilibrium and integrating through the thickness of a thin shell, but this is rather unsatisfactory for gridshells or ribbed shells.

The rate of work being absorbed, eq. (68), becomes

$$W = \int_A \left(\left(\sigma^{\alpha\beta} - m^{\eta\alpha} b_\eta^\beta + \nabla_\eta m^{\eta\alpha\beta} + c^{\alpha\beta}\right)\gamma_{\alpha\beta} + m^{\alpha\beta}\beta_{\alpha\beta} + m^{\alpha\beta}_{\cdot\cdot\cdot\chi}\beta^{\cdot\cdot\cdot\chi}_{\alpha\beta}\right) dA\,. \tag{73}$$

7. Elastic surfaces

We define an elastic surface as any surface whose strain energy per unit mass is a function of $a_{\alpha\beta}$, $b_{\alpha\beta}$ and $\Gamma^\lambda_{\alpha\beta}$. Such structures include soap films, fabric structures, shells and gridshells, as well as surfaces which minimize the Willmore energy, that is the surface integral of $H^2 - K$ (Velimirovi et al., 2011; Williams, 1987).

For an elastic surface the virtual work formulation becomes identical to the mini- mization of the sum of the strain energy of the structure and the potential energy of the loads, assuming that they admit a potential. The same results could be obtained using the calculus of variations, which would need exactly the same geometric relationships and how they change under an increment of displacement, or better a velocity allowing us to differentiate with respect to time.

It is preferable to use the strain energy per unit mass than the strain

energy per unit area because mass is conserved, whereas area changes. If Q is the strain energy per unit mass and ρ is the mass per unit area, we can write

$$\begin{aligned}\rho\frac{\partial Q}{\partial t} &= (\boldsymbol{\sigma}+\nabla\cdot\mathbf{m}+\mathbf{c})\cdot\cdot\boldsymbol{\gamma}+\mathbf{m}\cdots\boldsymbol{\beta}\\ &= \left(\sigma^{\alpha\beta}-m^{\eta\alpha}b_{\eta}^{\beta}+\nabla_{\eta}m^{\eta\alpha\beta}+c^{\alpha\beta}\right)\frac{1}{2}\frac{\partial a_{\alpha\beta}}{\partial t}\\ &+m^{\alpha\beta}\frac{\partial b_{\alpha\beta}}{\partial t}+m^{\alpha\beta}_{\ \ \chi}\frac{\partial\Gamma^{\chi}_{\alpha\beta}}{\partial t}\end{aligned} \tag{74}$$

in which

$$\frac{\partial}{\partial t}\left(\rho\sqrt{a}\right)=0 \tag{75}$$

$$\frac{1}{\rho}\frac{\partial\rho}{\partial t}=-\frac{1}{\sqrt{a}}\frac{\partial\sqrt{a}}{\partial t}=-a^{\alpha\beta}\gamma_{\alpha\beta}\,. \tag{76}$$

7.1 Surfaces with constant mean curvature, including minimal surfaces

The strain energy per unit mass of a surface with a constant isotropic surface tension T is

$$Q=\frac{T}{\rho}\,. \tag{77}$$

Thus Q does not depend upon $b_{\alpha\beta}$ or $\Gamma^{\lambda}_{\alpha\beta}$ and so the moments are zero giving

$$\boldsymbol{\sigma}=\sigma^{\alpha\beta}\mathbf{a}_{\alpha}\mathbf{a}_{\beta}=\sigma^{\beta\alpha}\mathbf{a}_{\alpha}\mathbf{a}_{\beta} \tag{78}$$

and eq. (74) becomes

$$\frac{\partial}{\partial t}\left(\frac{T}{\rho}\right)=-\frac{T}{\rho^{2}}\frac{\partial\rho}{\partial t}=\frac{T}{\rho}a^{\alpha\beta}\gamma_{\alpha\beta}=\frac{1}{\rho}\sigma^{\alpha\beta}\gamma_{\alpha\beta}\,. \tag{79}$$

Thus the membrane stress

$$\begin{aligned}\sigma^{\alpha\beta}&=Ta^{\alpha\beta}\\ \boldsymbol{\sigma}&=T\mathbf{J}\end{aligned} \tag{80}$$

corresponding to the uniform surface tension.

$$\mathbf{J}=a^{\alpha\beta}\mathbf{a}_{\alpha}\mathbf{a}_{\beta} \tag{81}$$

is the unit tensor on the surface.

Figure 2: Surface with constant mean curvature on a plane elliptical boundary.

The surface in figure 2 was found using dynamic relaxation (Day, 1965). The boundary is a plane ellipse and the surface has the minimum surface area for a given enclosed volume, like an inflated soap film. Dynamic relaxation was also used to find the pressure necessary to enclose a fixed volume. The pressure cannot be kept constant for a surface such as this because the pressure *decreases* with increasing volume, once it is inflated beyond a certain point. It was found necessary to damp the pressure change with both the rates of change of pressure and volume.

7.2 Surfaces which minimize the integral of the mean curvature subject to a constant volume

If we assume that the strain energy per unit mass is equal to the mean curvature divided by the density, then

$$Q = \frac{H}{\rho} = \frac{a^{\alpha\beta} b_{\alpha\beta}}{2\rho} \tag{82}$$

and

$$\begin{aligned} \rho \frac{\partial Q}{\partial t} &= \frac{1}{2} \frac{\partial a^{\alpha\beta}}{\partial t} b_{\alpha\beta} + \frac{1}{2} a^{\alpha\beta} \frac{\partial b_{\alpha\beta}}{\partial t} - \frac{H}{\rho} \frac{\partial \rho}{\partial t} \\ &= \gamma^{\alpha\beta} \left(H a_{\alpha\beta} - b_{\alpha\beta} \right) + \frac{1}{2} a^{\alpha\beta} \frac{\partial b_{\alpha\beta}}{\partial t} \\ &= \left(H a^{\alpha\beta} - b^{\alpha\beta} \right) \frac{1}{2} \frac{\partial a^{\alpha\beta}}{\partial t} + \frac{1}{2} a^{\alpha\beta} \frac{\partial b_{\alpha\beta}}{\partial t} \, . \end{aligned} \tag{83}$$

Then comparison with eq. (74) shows that

$$m^{\alpha\beta} = \frac{1}{2} a^{\alpha\beta} \,. \tag{84}$$

Assuming that the loading couples, $\mathbf{c}$ are zero, the equations of equilibrium of moments show that

$$\sigma^{\alpha\beta} = \sigma^{\beta\alpha} \tag{85}$$

$$\sigma^{\alpha} = 0 \tag{86}$$

so that

$$\sigma^{\alpha\beta} - m^{\eta\alpha} b^{\beta}_{\eta} = H a^{\alpha\beta} - b^{\alpha\beta} \tag{87}$$

and therefore

$$\begin{aligned} \sigma^{\alpha\beta} &= H a^{\alpha\beta} - b^{\alpha\beta} + \frac{1}{2} a^{\eta\alpha} b^{\beta}_{\eta} \\ &= \frac{1}{2} \left(b^{\eta}_{\eta} a^{\alpha\beta} - b^{\alpha\beta} \right) \,. \end{aligned} \tag{88}$$

or

$$\boldsymbol{\sigma} = H \mathbf{J} - \frac{\mathbf{b}}{2} = \frac{\boldsymbol{\varepsilon} \cdot \mathbf{b} \cdot \boldsymbol{\varepsilon}}{2} \,. \tag{89}$$

We therefore have

$$\nabla_{\alpha} \sigma^{\alpha\beta} = \frac{1}{2} \left(\nabla_{\alpha} b^{\eta}_{\eta} a^{\alpha\beta} - \nabla_{\alpha} b^{\alpha\beta} \right) = \frac{1}{2} \left(\nabla_{\alpha} b^{\eta}_{\eta} - \nabla_{\eta} b^{\eta}_{\alpha} \right) a^{\alpha\beta} = 0 \tag{90}$$

from the Codazzi-Mainardi equations, eq. (25), and

$$\begin{aligned} \sigma^{\alpha\beta} b_{\alpha\beta} &= \frac{1}{2} \left(b^{\eta}_{\eta} a^{\alpha\beta} - b^{\alpha\beta} \right) b_{\alpha\beta} \\ &= \frac{1}{2} \left(b^{\alpha}_{\alpha} b^{\beta}_{\beta} - b^{\alpha}_{\beta} b^{\beta}_{\alpha} \right) \\ &= K \,. \end{aligned} \tag{91}$$

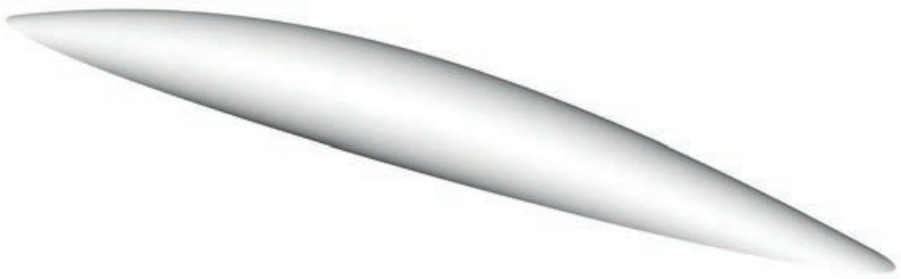


Figure 3: Surface with constant positive Gaussian curvature on a plane elliptical boundary.

Thus the tangential components of load

$$p^{\alpha} = 0 \tag{92}$$

are zero and the normal component of load,

$$p = -K \tag{93}$$

where K is the Gaussian curvature.

Thus if we minimize, or possibly maximize, the surface integral of the mean curvature subject to the enclosed volume being constant we find that we need an internal, or possibly external pressure, which must be a constant and therefore the Gaussian curvature is also constant. One would imagine that this simple fact must have been known before.

The moments in the surface do not affect equilibrium, and can therefore be dispensed with and are purely a phantom, we only need the membrane stresses for equilibrium.

The mean membrane stress,

$$\frac{1}{2} a_{\alpha\beta} \sigma^{\alpha\beta} = \frac{1}{2} b^{\eta}_{\eta} = H \tag{94}$$

and therefore for a sphere with an outwards pointing normal the mean stress is compressive, and we need an external pressure. On the other hand if we choose to have the normal pointing inwards we have a tensile mean stress and an internal pressure.

Figure 3 show a surface of constant positive Gaussian curvature on the same plane elliptic boundary as that in fig. 2. The numerical procedure uses flat triangular facets and the stress in the surface is represented by forces in each fold proportional to the angle of the fold from flat. Those familiar with the Airy stress function (Timoshenko, 1934) will realize that this is equivalent to the shell being its own stress function, but with no projection onto the plane. The equilibrium shape was again found using dynamic relaxation. Note that this procedure only controls the shape of the surface, not the position of the nodes upon it and therefore some other constraint is required. In this case the folds were given an additional constant force density, but the normal component of the extra resultant force was removed before moving the nodes. This technique is commonly used to find geodesics on a surface for fabric structures (Williams, 1980).

8. Simultaneous conjugate directions for membrane stress and curvature

8.1 Non-orthogonal directions

In the stress state in eq. (89) the principal stress directions and principal curvature directions coincide. That means we could construct a gridshell structure with an orthogonal quadrilateral mesh with no bending moments and flat panels, subject to a pressure loading. In this section we shall relax the pressure loading requirement, because we are only really interested in the state of stress and the curvature. We can also relax the condition on the mesh being orthogonal.

We can write any state of membrane stress as

$$\begin{aligned} \boldsymbol{\sigma} &= A\mathbf{x}\mathbf{x} + B\mathbf{y}\mathbf{y} \\ \mathbf{x}\cdot\mathbf{x} &= 1 \\ \mathbf{y}\cdot\mathbf{y} &= 1 \\ \sigma^{\alpha\beta} &= Ax^{\alpha}x^{\beta} + By^{\alpha}y^{\beta} \end{aligned} \tag{95}$$

corresponding to two monoaxial stresses of magnitude A and B in the directions of the unit vectors $\mathbf{x}$ and $\mathbf{y}$ tangent to the surface. There are 3 values of the components σ^{11}, $\sigma^{12} = \sigma^{21}$ and σ^{22}, but 4 quantities A, B and the directions $\mathbf{x}$ and $\mathbf{y}$, see fig. 4. We therefore need 1 further condition, often taken as $\mathbf{x}\cdot\mathbf{y} = 0$, which leads to the principal stresses and their directions, but we will not make that assumption at this juncture.

Now let us postulate that we can write the normal curvature tensor $\mathbf{b}$ as

$$\begin{aligned} \mathbf{b} &= C(\boldsymbol{\varepsilon}\cdot\mathbf{x})(\boldsymbol{\varepsilon}\cdot\mathbf{x}) + D(\boldsymbol{\varepsilon}\cdot\mathbf{y})(\boldsymbol{\varepsilon}\cdot\mathbf{y}) \\ b_{\alpha\beta} &= \varepsilon_{\alpha\lambda}\varepsilon_{\beta\mu}\left(Cx^{\lambda}x^{\mu} + Dy^{\lambda}y^{\mu}\right) \end{aligned} \tag{96}$$

with the same unit vectors x and y. The reason for the in eq. (96) is that we want the stresses to coincide with the directions of “folding”, rather than the directions of curvature.

Then if $\boldsymbol{\sigma}$ and $\mathbf{b}$ are known we have 6 equations in the unknowns A, B, C, D and the directions of $\mathbf{x}$ and $\mathbf{y}$. These equations can be solved by introducing the orthogonal

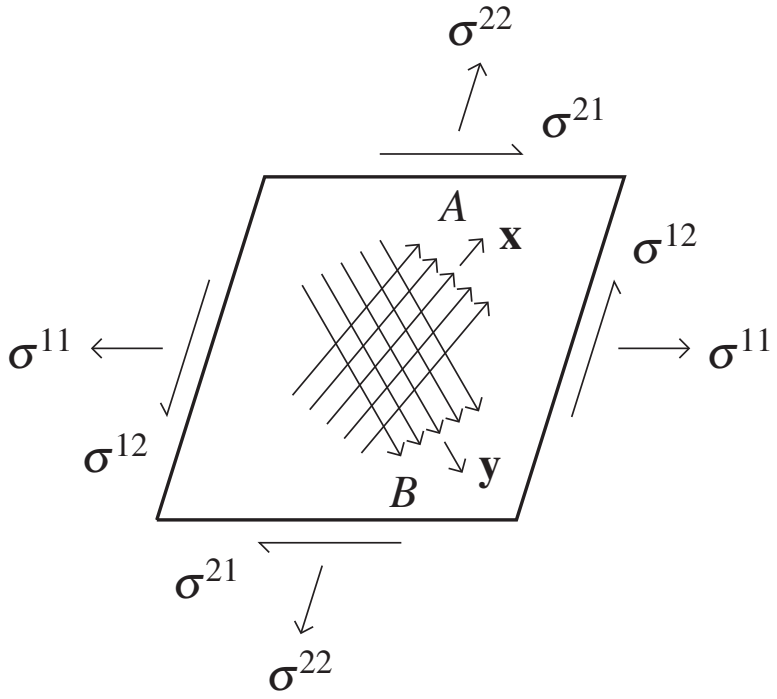


Figure 4: Monoaxial stress state of magnitude A and B acting in the directions of unit vectors **x** and **y** tangent to the surface on a small element of shell.

unit vectors **X** and **Y** such the

$$\begin{aligned} \mathbf{x} &= \mathbf{X}\cos\frac{\theta}{2} + \mathbf{Y}\sin\frac{\theta}{2} \\ \mathbf{y} &= \mathbf{X}\sin\frac{\theta}{2} + \mathbf{Y}\cos\frac{\theta}{2} \\ \mathbf{X}\cdot\mathbf{X} &= 1 \\ \mathbf{X}\cdot\mathbf{Y} &= 0 \\ \mathbf{Y}\cdot\mathbf{Y} &= 1 \end{aligned} \tag{97}$$

and substituting into eqs. (95) and (96) to give

$$\begin{aligned} \boldsymbol{\sigma} = {} & \frac{(A+B)}{2}(\mathbf{XX}+\mathbf{YY}+(\mathbf{XY}+\mathbf{YX})\sin\theta) \\ & + \frac{(A-B)}{2}(\mathbf{XX}-\mathbf{YY})\cos\theta \end{aligned} \tag{98}$$

$$\begin{aligned} -\boldsymbol{\varepsilon}\cdot\mathbf{b}\cdot\boldsymbol{\varepsilon} = {} & \frac{(C+D)}{2}(\mathbf{XX}+\mathbf{YY}+(\mathbf{XY}+\mathbf{YX})\sin\theta) \\ & + \frac{(C-D)}{2}(\mathbf{XX}-\mathbf{YY})\cos\theta\,. \end{aligned} \tag{99}$$

Hence

$$H\boldsymbol{\sigma} + S\boldsymbol{\varepsilon}\cdot\mathbf{b}\cdot\boldsymbol{\varepsilon} = \left(H\frac{(A-B)}{2} - S\frac{(C-D)}{2}\right)\cos\theta\,(\mathbf{XX}-\mathbf{YY}) \tag{100}$$

where S is the mean stress. We can find $\mathbf{X}$ and $\mathbf{Y}$ by observing that $(\mathbf{X} + \mathbf{Y})$ and $(\mathbf{X} - \mathbf{Y})$ are the eigenvectors of $(H\boldsymbol{\sigma} + S\boldsymbol{\varepsilon} \cdot \mathbf{b} \cdot \boldsymbol{\varepsilon})$. Having found $\mathbf{X}$ and $\mathbf{Y}$ we can find all the other unknowns and further study shows that $|\cos\theta| \leq 1$ and $|\sin\theta| \leq 1$. Thus it would appear that there is always a solution, except for the case when H and S are both zero.

8.2 Orthogonal directions

A special case of the previous section is when the curvature and membrane stress are such that they share the same principal directions, so that $\mathbf{x}$ and $\mathbf{y}$ are orthogonal.

Figure 5: Plan and cross-section of shell with coincident principal membrane stress and principal curvature directions, so that x and y are orthogonal.

Then we can generalize the state of stress in eq. (89) to

$$\begin{aligned} \boldsymbol{\sigma} &= \phi\mathbf{J} + \psi\boldsymbol{\varepsilon} \cdot \mathbf{b} \cdot \boldsymbol{\varepsilon} \\ \sigma^{\alpha\beta} &= \phi a^{\alpha\beta} - \psi\varepsilon^{\alpha\lambda}\varepsilon^{\beta\mu}b_{\lambda\mu} \end{aligned} \tag{101}$$

where ϕ and ψ are scalar fields.

Using the Codazzi-Mainardi equations, eq. (25), the equilibrium equations become

$$(\nabla \cdot \boldsymbol{\sigma} + \mathbf{p}) \cdot \mathbf{J} = \nabla\phi + \nabla\psi \cdot \boldsymbol{\varepsilon} \cdot \mathbf{b} \cdot \boldsymbol{\varepsilon} + \mathbf{p} \cdot \mathbf{J} = 0 \tag{102}$$

in the plane of the surface and

$$(\nabla \cdot \boldsymbol{\sigma} + \mathbf{p}) \cdot \mathbf{n} = 2\phi H + 2\psi K + p = 0 \tag{103}$$

normal to the surface.

Let us assume that we have a load corresponding to a known weight per unit area w, then

$$\mathbf{p} = p^{\beta}\mathbf{a}_{\beta} + p\mathbf{n} = -w\mathbf{k} \tag{104}$$

where $\mathbf{k}$ is a unit vector in the z direction. The equilibrium equations are then

$$\nabla\phi + \nabla\psi \cdot \boldsymbol{\varepsilon} \cdot \mathbf{b} \cdot \boldsymbol{\varepsilon} = w\nabla z \tag{105}$$

$$2\phi H + 2\psi K = w\mathbf{k} \cdot \mathbf{n} \,. \tag{106}$$

The equilibrium equations in the plane of the surface, eq. (105), have one possible very simple solution.

$$\psi = \text{constant} \tag{107}$$

$$\phi = \phi(z) \tag{108}$$

$$w = w(z) = \frac{d\phi}{dz} \tag{109}$$

where $\phi(z)$ is a function we can choose. Having done this the equilibrium equation normal to the surface eq. (106) can be used to find the geometry of the surface, in exactly the same way as in sections 7.1 and 7.2. Aish et al. (2015) consider the special case $\psi = 0$.

If we assume that

$$\frac{\phi}{w} = -L = \text{constant} \tag{110}$$

so that the isotropic part of the membrane stress is proportional to the weight per unit area, then the isotropic stress and weight per unit area reduce with height,

$$\phi = \phi_0 e^{-(z-z_0)/L} \,. \tag{111}$$

Figure 5 shows a plan and cross-section of a shell corresponding to this isotropic stress and loading state plus the stress associated with a

constant ψ. It can be seen that there is a concentration of vertical stress at the centre support which could not have been obtained with the isotropic stress state on its own. The high vertical stress is associated with ψ and the large curvature in the horizontal plane.

Having defined the surface we need to construct the orthogonal grid of principal curvature directions, which coincide with the directions of the principal membrane stresses. This is not a trivial task, particularly in ensuring that the variation in spacing of the lines is satisfactory (Sun et al., 2016).

9. Conclusion

We have derived the shell equilibrium equations using virtual work, which enables many formfinding methods to be reformulated as a minimization using the calculus of variations. This method of deriving the equilibrium equations naturally introduces the concept of geodesic bending moments for the analysis of gridshells and kitchen sieves via the Christoffel symbols, which become the components of a tensor upon differentiation with respect to time.

We have also demonstrated that minimizing the surface integral of the mean curvature subject to a constraint on enclosed volume gives a surface of constant Gaussian curvature, although one would imagine that this simple fact must have been known before.

These studies lead us to examine the conditions under which principal stress and principal curvature coincide and how this can be incorporated into a formfinding process.

Some of the ideas introduced in this paper could lead to further numerical studies and practical application.

References

Aish, F., S. Joyce, S. Malek, and C. J. K. Williams (2015, April). The use of a particle method for the modelling of isotropic membrane stress for the form finding of shell structures. *Computer-Aided Design 61* (C), 24–31.

Axelrad, E. L. and F. A. Emmerling (1988). Vector and tensor form of intrinsic shell-theory relations. *International Journal of Non-Linear Mechanics 23* (1), 9–23.

Calladine, C. (1982). The theory of shell structures aims and methods. *International Journal of Mechanical Sciences 24* (4), 219–230.

Capecchi, D. (2012). *History of Virtual Work Laws*. Springer-Verlag Italia.

Cosserat, E. and F. Cosserat (1909). *Théorie des corps déformables*. A. Hermann et fils.

Day, A. S. (1965). An introduction to dynamic relaxation. *The Engineer 219*, 218–221.

Diamanti, O., A. Vaxman, D. Panozzo, and O. Sorkine-Hornung (2014, August). Designing n-polyvector fields with complex polynomials. *Comput. Graph. Forum 33* (5), 1–11.

Dirac, P. A. M. (1975). *General Theory of Relativity*. Wiley.

Eisenhart, L. P. (1947). *An introduction to differential geometry, with use of the tensor calculus*. Princeton University Press.

Flügge, W. (1973). *Stresses in Shells.* Berlin: Springer-Verlag.

Green, A. E. and W. Zerna (1968). *Theoretical elasticity* (2nd ed.). Oxford: Oxford University Press.

Jiang, C., C. Tang, A. Vaxman, P. Wonka, and H. Pottmann (2015, October). Polyhedral patterns. *ACM Trans. Graph.* 34 (6), 172:1–172:12.

Kupferman, R., E. Olami, and R. Segev (2017, Jun). Continuum dynamics on manifolds: Application to elasticity of residually-stressed bodies. *Journal of Elasticity 128* (1), 61–84.

Monaghan, J. (2012). Smoothed particle hydrodynamics and its diverse applications. *Annual Review of Fluid Mechanics 44* (1), 323–346.

Silling, S. and R. Lehoucq (2010). Peridynamic theory of solid mechanics. In H. Aref and E. van der Giessen (Eds.), *Advances in Applied Mechanics*, Volume 44 of *Advances in Applied Mechanics*, pp. 73–168. Elsevier.

Struik, D. J. (1961). *Lectures on Classical Differential Geometry*. Addison-Wesley.

Sun, X., C. Jiang, J. Wallner, and H. Pottmann (2016). Vertex normals and face curvatures of triangle meshes. In A. I. Bobenko (Ed.), *Advances in Discrete Differential Geometry*, pp. 267–286. Berlin, Heidelberg: Springer Berlin Heidelberg.

Timoshenko, S. and S. Woinowsk-Krieger (1959). *Theory of plates and shells* (2nd ed.). McGraw-Hill.

Timoshenko, S. P. (1921). Lxvi. On the correction for shear of the differential equation for transverse vibrations of prismatic bars. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 41* (245), 744–746.

Timoshenko, S. P. (1934). *Theory of Elasticity* (1st ed.). Engineering Societies Monographs. New York and London: McGraw-Hill book company, inc.

Velimirovi , L. S., M. S. iri , and N. M. Velimirovi (2011). On the Willmore energy of shells under infinitesimal deformations. *Computers & Mathematics with Applications 61* (11), 3181–3190.

Vouga, E., M. Höbinger, J. Wallner, and H. Pottmann (2012, July). Design of self-supporting surfaces. *ACM Trans. Graph. 31* (4), 87:1–87:11.

Williams, C. J. K. (1980). Form finding and cutting patterns for air-supported structures. In *Air-supported structures: the state of the art*, London, pp. 99–120. The Institution of Structural Engineers.

Williams, C. J. K. (1987). Use of structural analogy in generation of smooth surfaces for engineering purposes. *Computer-Aided Design 19* (6), 310–322.

Yang, Y.-L., Y.-J. Yang, H. Pottmann, and N. J. Mitra (2011). Shape space exploration of constrained meshes. In P*roceedings of the 2011 SIGGRAPH Asia Conference*, SA '11, New York, NY, USA, pp. 124:1–124:12. ACM.

Topologic: Tools to explore architectural topology

Robert Aish, Wassim Jabi, Simon Lannon, Nicholas Mario Wardhana, Aikaterini Chatzivasileiadi

Robert Aish
robert.aish@ucl.ac.uk
Bartlett School of Architecture, University College London, United Kingdom

Wassim Jabi
jabiw@cardiff.ac.uk
Welsh School of Architecture, Cardiff University, United Kingdom

Simon Lannon
lannon@cardiff.ac.uk
Welsh School of Architecture, Cardiff University, United Kingdom

Nicholas Mario Wardhana
wardhanan@cardiff.ac.uk
Welsh School of Architecture, Cardiff University, United Kingdom

Aikaterini Chatzivasileiadi
chatzivasileiadia@cardiff.ac.uk
Welsh School of Architecture, Cardiff University, United Kingdom

Keywords:
Non-manifold topology, idealised model, material model

Abstract

Buildings enclose and partition space and are built from assemblies of connected components. The many different forms of spatial and material partitioning and connectedness found within buildings can be represented by topology. This paper introduces the "Topologic" software library which integrates a number of architecturally relevant topological concepts into a unified application toolkit.

The goal of the Topologic toolkit is to support the creation of the lightest, most understandable conceptual models of architectural topology. The formal language of topology is well-matched to the data input requirements for applications such as energy simulation and structural analysis. In addition, the ease with which these lightweight topological models can be modified encourages design exploration and performance simulation at the conceptual design phase.

A challenging and equally interesting question is how can the formal language of topology be used to represent architectural concepts of space which have previously been described in rather speculative and subjective terms?

1. Introduction

This paper focusses on the conceptual issues surrounding the use of topology in architecture. It builds on previous research and proof of concept studies (Aish and Pratap 2013; Jabi 2014; Jabi et al. 2017). Other concurrently published papers describe in greater detail the implementation of the Topologic toolkit and specific applications of Topologic in building analysis and simulation (Jabi et al. 2018; Chatzivasileiadi, Lannon, et al. 2018; Wardhana et al. 2018).

Topology and in particular non-manifold topology are vast subjects that span algebra, geometry and set theory. It is beyond the scope of this paper to delve into the mathematical constructs and proofs that precisely define non-manifold topology. Topology has applications in biology, medicine, computer science, physics and robotics among others. Since the motivation for this research is to address the needs of architects and engineers, this research focusses on a specific application of non-manifold topology in the representation of significant spatial relationships in the design of buildings using computer-aided three-dimensional geometric processing.

We can contrast this approach with more conventional representations of buildings as a collection of physical building components, typically modelled as manifold solids, as demonstrated by Building Information Modelling (BIM) applications. While BIM can be used to model the physical structure of the building, architecture is usually conceived in terms of an overall form and a series of related spatial enclosures (Curtis 1996). This spatial conceptualization is a key aspect of architectural design because it directly anticipates how the resulting building will be experienced. However, there are no practical design tools which support the creation of this spatial representation of architecture. Non-manifold topology is ideally suited to create a lightweight representation of a building as an external envelope and the subdivision of the enclosed space into separate spaces such as rooms, building storeys, cores, atria, etc. This lightweight representation also matches the input data requirements for important analysis and simulation applications, such as energy analysis, (Ellis, Torcellini, and Crawley 2008).

Conventional BIM applications, in contrast, do not explicitly model the enclosure of space. Although it might be possible to indirectly infer the enclosed spaces from the position of the physical building components, the fidelity of this representation depends on the precise connectivity of the bounding physical components, which cannot be relied upon. Even if this approach was viable, the level of detail of BIM models is often too complex for this type of analysis (Maile et al. 2013). Detailed BIM models are also cumbersome to change which may inhibit design exploration at the conceptual design stage.

One option might be to explore spatial modelling with existing solid modelling applications. However most of these applications are based on conventional manifold modelling techniques and do not support non-manifold topology. Indeed, many regular manifold modelling applications treat non-manifold topology as an error condition.

The objective of this research is to develop design tools based on precise topological principles but presented in ways which are understandable by architectural users who may have little previous experience of topology. The intention is that Topologic can be an effective intermediary between the abstract world of topology and the practical world of architecture and building engineering.

2. Background

2.1 The distinction between manifold and non-manifold Topology

In a previous paper (Aish and Pratap 2013) the following distinctions were made between manifold and non-manifold topology:

"A 3D manifold body has a boundary that separates the enclosed solid from the external void. The boundary is composed of faces, which have (interior) solid material on one side and the (exterior) void on the other. In practical terms, a manifold body without internal voids can be machined out of a single block of material."

"A non-manifold body also has a boundary [composed of faces] that separates the enclosed solid from the external void. Faces are either external [separating the interior (enclosed space) from the exterior (void)] or internal [separating one enclosed space (or cell) from another]. Furthermore, a non-manifold solid can have edges where more than two faces meet."

2.2 The distinction between an idealized and a material model

One of the key themes which runs through this research is the distinction between an "idealised" model (of a building) and a "material" model of the physical building components. An early demonstration of this principle was made in 1997 (Aish 1997) and further developed (Hensen and Lamberts 2012).

Typically, idealised models are far less detailed than material models, therefore lighter and more easily edited. In addition, the different topological components of the idealised model (faces, edges, vertices) can be used as the "supports" for related building components in the material model. The connectivity of the components in the material model need not be directly modelled. Instead this connectivity can be represented through the topology of the idealised model.

2.3 Previous research

The case for non-manifold topology as well as its data structures and operators for geometric modelling were comprehensively set out by (Weiler 1986). In his introduction, Weiler explains why non-manifold topology is needed:

"*A unified representation for combined wireframe, surface, and solid modelling by necessity requires a non-manifold representation, and is desirable since it makes it easy to use the most appropriate modelling form (or combination of forms) in a given application without requiring representation conversion as more information is added to the model.*"

Non-manifold topology allows an expansion of the regular Boolean operations of union, difference, and intersection. This expanded set includes operators such as merge, impose, and imprint. For a full description of non-manifold operators, please consult (Masuda 1993).

Representing space and its boundary was the focus of early research into BIM (Björk 1992; Chang and Woodbury 1997) and into "product modelling" (PDES/STEP) (Eastman and Siabiris 1995) and was proposed as an approach to the representation of geometry definition for input to Building Performance Simulation in the early design stages (Hui and Floriani 2007; Jabi 2016). However, this is not emphasised in modern BIM software where the building fabric is represented through manifold geometry and energy models from are derived from the fabric models.

Separately, non-manifold topology has been successfully used in the medical field to model complex organic structures with multiple internal zones (Nguyen 2011; Bronson, Levine, and Whitaker 2014).

Our focus is to create a schema which separates abstract topological concepts from domain specific and pragmatic concerns of architecture, engineering and construction. We maintain this separation, but also explore important connections: how buildings can be represented by topology and how a topological representation can potentially assist architectural users in the conceptualisation and analysis of new buildings. Therefore, our focus is not to create new non-manifold data structures, but rather to harness existing geometry and topology kernels in an innovative way; indeed, it is completely feasible that the Topologic schema could be implemented with different data structures or with different kernels.

A comprehensive and systematic survey of topological modelling kernels, which support non-manifold topology, was carried out by the authors and published elsewhere (Chatzivasileiadi, Wardhana, et al. 2018). Features and capabilities of kernels were compared in order to make an informed decision regarding what underlying kernel to use. Popular geometric kernels, such as CGAL, were discounted due to their inability to represent higher dimensional entities such as *CellComplexes* and for their more limited set of irregular Boolean operations.

3. The Topologic toolkit

The core Topologic software is developed in C++ using Open Cascade (https://www.opencascade.com/) with specific C++/CLI variants developed for different visual data flow programming environments (Wardhana et al. 2018). Topologic integrates a number of architecturally relevant topological concepts into a unified application toolkit. The features and applications of Topologic are summarised in Figure 1 and Figure 2.

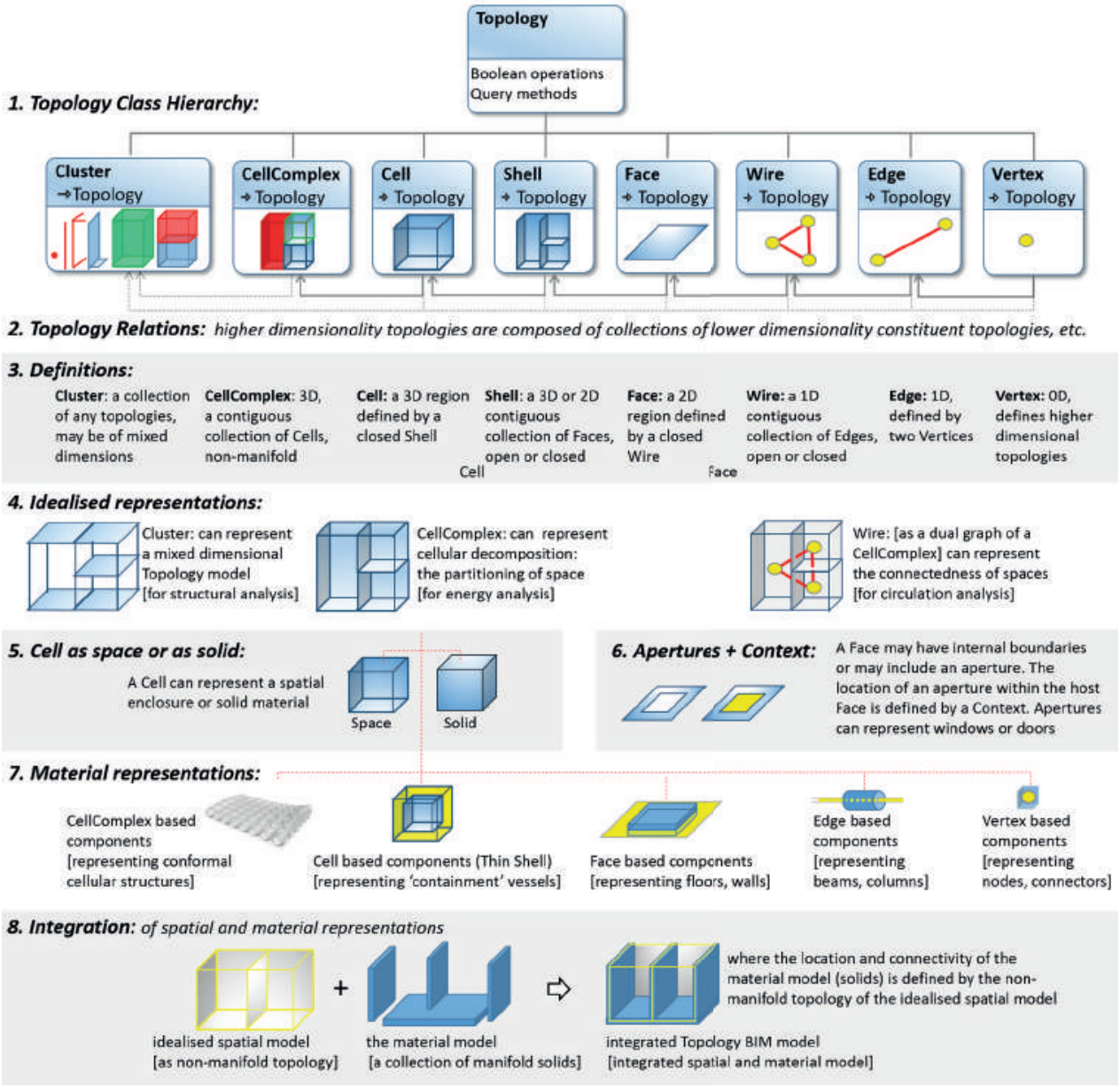
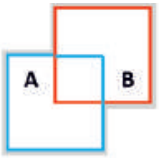
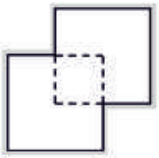


Figure 1: The Topologic application toolkit summarised in eight key points.



Input: Starting configuration where A and B intersect.

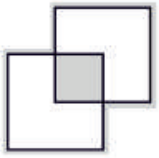


Merge: All parts of A are intersected with all parts of B. The parts of A are split at the intersection with B and the parts of B are split at the intersections with A to form the merge result. The result is non-manifold.

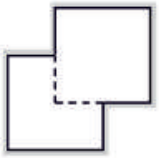
Union: All parts of A which are within the region of B and all parts of B which are within the region of A are removed.

Difference: [A difference B] All parts of B which are not within A are removed.

Intersect: All parts of A which are not within the region of B and all parts of B which are not within the region of A are removed.



XOR: Only those parts of A which are not within the region of B and those parts of B which are not within the region of A are retained.

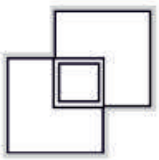


Impose: [A Impose B] All parts of A which are within the region of B are removed. The result is non-manifold.

Imprint: [A Imprint with B] All parts of B that are not within A are removed. The result is non-manifold.

Slice: [A slice with B] A is a higher dimensional and B is typically a collection of topological entities of lower dimensionality. The result is non-manifold.

Trim: A is a topological entity of higher dimensionality and B is typically a topological entities of lower dimensionality, returns the part of A on a specified side of B.



Unmerge: [which is the converse of a Boolean operation] returns a Cluster of unconnected manifold entities by duplicating those entities which were previously shared within the input topology.

Figure 2: Boolean Operations implemented in Topologic.

3.1 Class hierarchy

The Topologic class hierarchy is designed to provide the architectural end-user with a conceptual understanding of topology. It also functions as an "end-user programmers' interface" (EDPI). This user-oriented class hierarchy is distinct to the implementation-oriented class hierarchy within the Topologic core.

The Topologic superclass (**Fig. 1**, section 1) is abstract and implements constructors, properties and methods including a set of Boolean operators. These operators can be used with both manifold and non-manifold topology (**Fig. 2**). Topologic implements the expected concepts such as: *Vertex, Edge, Wire, Face, Shell*, and *Cell*. The interesting additional topological concepts are:

CellComplex which is a contiguous collection of *Cells* and is non-manifold.

Cluster which is a universal construct and allows any combination of topologies, including other "nested" *Clusters*, to be represented. A *Cluster* may represent non-contiguous, unrelated topologies of different dimensionalities.

3.2 Topological relationships

Topologic supports the building and querying of three different types of topological relationships (**Fig. 1**, section 2)

Hierarchical relationships: between topological entities of different dimensionality. These relationships are created when a higher dimensional topology construct is composed from a collection of lower dimensional topologies. Subsequently the compositional relationships may be queried:

```
cellComplexes =
     vertex.Edges.Wires.Faces.Shells.Cells.
     CellComplexes;
```

Conversely, the decompositional relationships may also be queried, for example from higher dimensional topologies down to the constituent collections of lower dimensional topologies:

```
vertices = cellComplex.Vertices;
```

or

```
vertices =
     cellComplex.Cells[n].Shells[n].Faces[n].Wires[n].
     Edges[n].Vertices;
```

Lateral relationships: these occur within a topological construct when the constituents share common topologies of a lower dimensionality.

```
adjacentCells = cellComplex.Cells[n].AdjacentCells;
adjacentFaces = shell.Faces[n].AdjacentFaces;
```

Connectivity: The path between two topologies can be queried.

```
path = topology.PathTo(otherTopology);
```

3.3 Idealised representations

Three different idealized models are considered (**Fig. 1**, section 4)

Energy Analysis: a *CellComplex* can represent the partitioning and adjacency of spaces and thermal zones.

Structural Analysis: a *Cluster* can be used to represent a mixed-dimensional model, with *Faces* representing structural slabs, blade columns and shear walls, *Edges* representing structural columns and *Cells* representing building cores.

Digital Fabrication Analysis: a *CellComplex* can represent the design envelope where topology can inform the shape and interface between deposited material (Jabi et al. 2017).

Circulation Analysis: a dual graph of a *CellComplex* can represent the connectedness of spaces.

3.4 Cell as a space or as a solid

A *Cell* is defined as a closed collection of faces, bounding a 3D region. However, this same topology can represent two distinctly different application concepts: a Solid and a Space (**Fig. 1**, section 5). A Solid is interpreted as a single homogeneous region of material and its boundary defines where the material ends and the void begins. This is the interpretation of the *Cell* as used in "Solid Modelling" and BIM applications.

A Space is a more abstract concept and may include an implied conceptual distinction between the material which is "contained" (represented by the enclosed 3D region of the *Cell*) and the "container" (represented by the Faces of the *Cell*). A Face may represent a boundary which is intended to be materialized with a defined thickness or may represent a "virtual" (e.g. adiabatic) barrier which is not intended to be materialized.

Solids and Spaces have exactly the same *Cell* topology, but the domain specific semantics and expected behaviour of this topology may be different. Consider a boolean "difference" operation representing a hole drilled into a *Cell* (as a solid). A new part of the *Cell* boundary would be created, but the result would still be a *Cell*.

What result would the user expect if the same *Cell* represented a Space? Would the boolean "difference" only apply to a specific Face (as part of the Space's boundary)? Would the user expect the boolean operation to create an internal boundary within the selected Face? Would the user expect this operation to destroy the integrity of the enclosure, changing the *Cell* into an open *Shell*?

This example helps to explain the difference between a material model (the *Cell* as a Solid) and an idealised model (the *Cell* as a Space). More generally this example demonstrates the need for the architectural users to customise the application of abstract topological concepts with the domain semantics which suits their purpose.

This relationship between application semantics and abstract concepts works both ways. Sometimes more generally applicable concepts emerge by abstracting ideas from other specialist domains. For example, the concept of a topological *Cell* may have originated as an abstracted analogy of a biological cell, with similarities in terms of the homogeneity and continuity of the contained 3D region and the role of the cell wall as a closed container with selective permeability (**Fig. 3**).

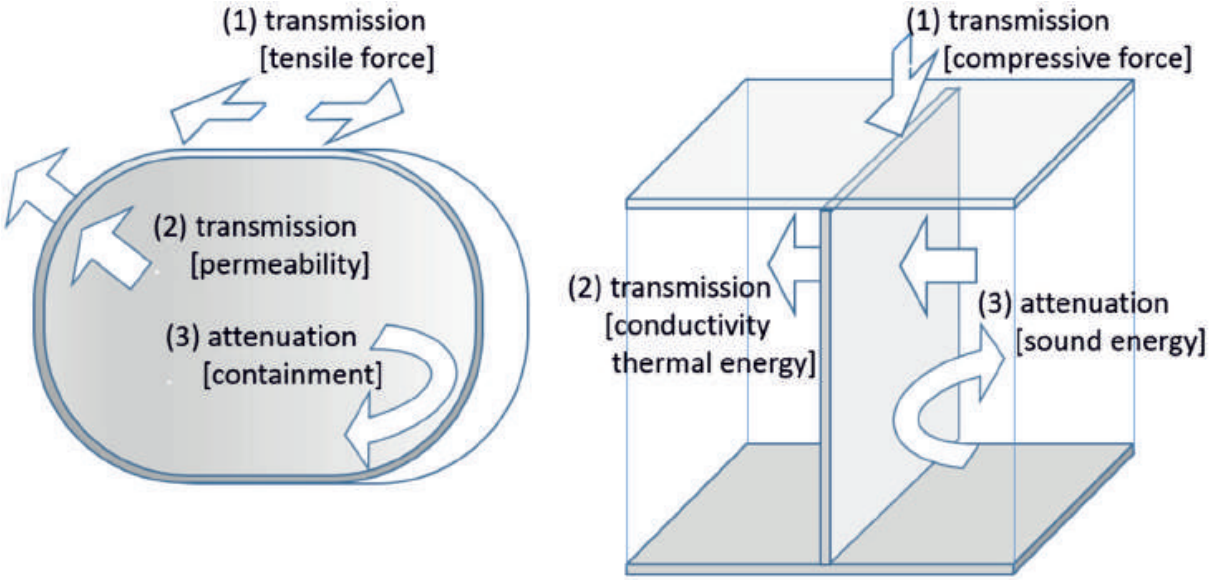


Figure 3: The cell wall as a separator and as a connector, in biology and in architecture (with acknowledgement to Wix, 1994).

3.5 Apertures and Contexts

A Face may have internal boundaries which may represent an aperture. The location of an aperture within the host *Face* is defined by a Context. Apertures can represent windows or doors. (**Fig. 1**, section 6) (The representation of Apertures is discussed in more detail in section 4.4 "Regional Topology")

3.6 Material representations

While all *Cells* have a common topology (a closed 3D region bounded by *Faces*) different configurations of *Cells* may be generated from different types of foundational topologies using different geometric operations (**Fig. 1**, section 7), for example:

Point location connector components: may be based on *Vertices*.

Linear components such as columns or beams: may be based on *Edges* (or *Wires*) using operations where a cross section *Wire* is extruded along a path.

Area based components such as slabs, floors, walls may be based on *Faces*: using offset operations with a specified thickness and direction.

Volume based components such as a containment vessel may be based on *Cells* using thin-shell operations and a specified wall thickness.

Conformal cellular structures, used in 3D printing, may be based on *CellComplexes*.

Complex sub-assemblies of material components can be modelled as *Clusters*.

3.7 Integration of idealized and material models

The integrated BIM model uses the idealized non-manifold spatial model to define the location and connectivity of the material model. (**Fig.1**, section 8). The defining centre lines or centre faces of walls and floors of the material model may be offset from the edges and faces of the idealized model. We can now appreciate the difficulty of attempting to reverse the direction of the arrow to recover an idealized spatial model from a material model.

In traditional BIM, the 3D material representation is the defining model while the drawings are the derived models. With architectural topology the idealized non-manifold topological representation becomes the defining model and the 3D material representation is now a derived model.

The idealised non-manifold spatial model acts as a useful conceptual and practical intermediary between the user and the material model (**Fig 4**). In this workflow the user is not manually placing specific material components on specific *Faces* or *Edges* of the idealised model. If such a workflow had been adopted, then any change in the idealised topology might have removed these specific *Face* and *Edge* and orphaned (or potentially deleted) the material components. Also such a change to the idealised topology might have created new *Faces* and *Edges* which the user would be required to populate with material components.

Instead, the populating of the idealised topology is rule-based using the Visual Data Flow programming tools available in the host application. The rule-based generation of the material model allows alternative building configurations to be easily explored via the manipulation of the idealised spatial model as previously suggested (Aish and Pratap 2013).

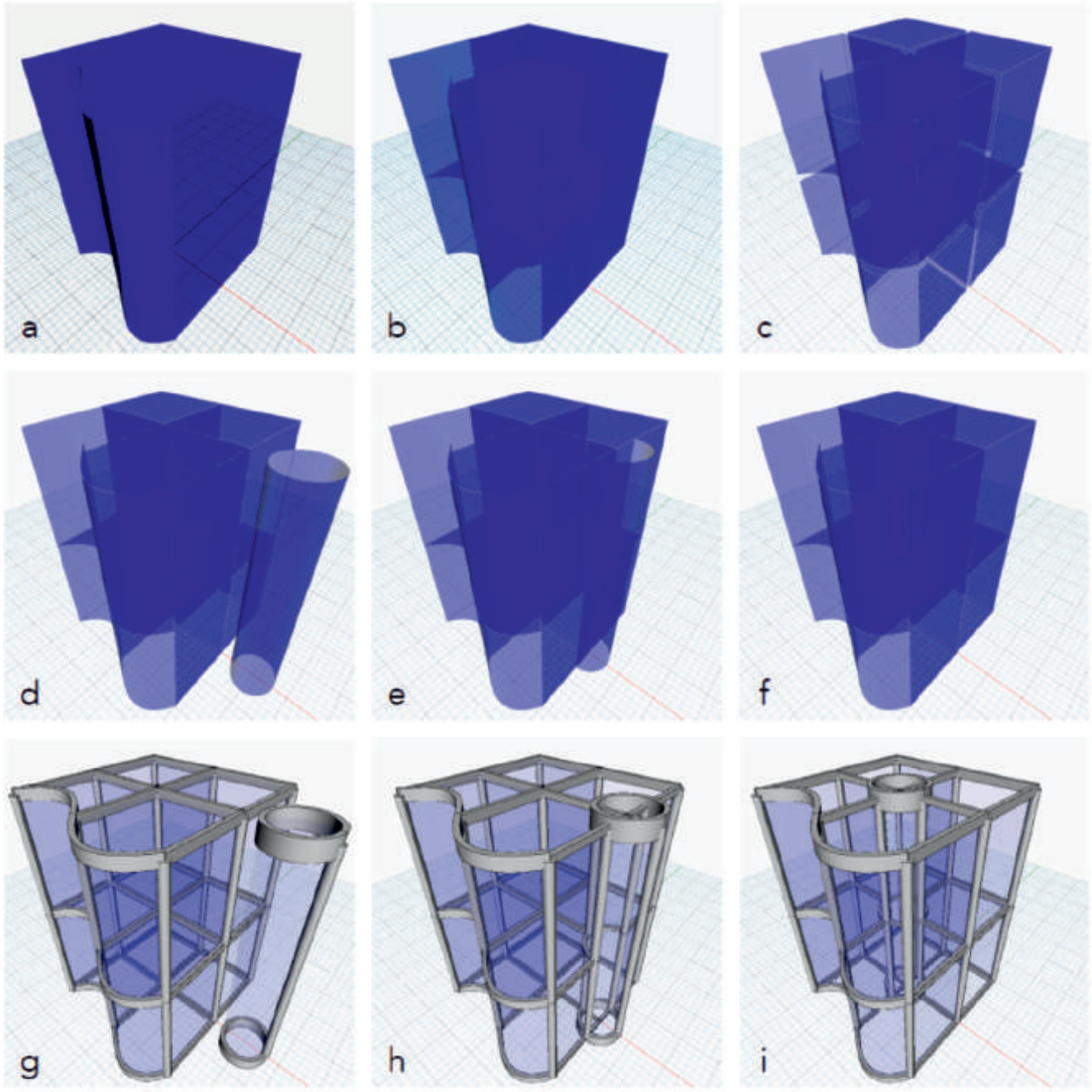


Figure 4: An idealised spatial model built with non-manifold topology can be used as a convenient intermediate representation to manipulate a material model, involving:

a. creating a cell from a lofted solid.
b. dividing the cell using several faces, resulting is a CellComplex.
c. the individual cells can be derived from the CellComplex.
d. introduce a cylinder outside the CellComplex.
e. move the cylinder into and imposed on the CellComplex: new cells are created.
f. move the cylinder further into the centre: the cells update accordingly.
g. h. i. corresponding material models are derived from the NMT models in d, e, f.

The workflow includes detecting vertical and non-vertical edges, sweeping a circle along vertical edges to create cylindrical columns

and a rectangle along non-vertical edges to create rectangular beams. The depth of the beams are parametrically computed according to their length. For visualisation purposes, the surfaces are thickened slightly into solids and made translucent.

4. Using non-manifold topology to represent relevant architectural concepts

Non-manifold topology embraces five concepts with architectural relevance:

4.1 Non-manifold Cell

A non-manifold *Cell* may contain internal Faces which are not part of the external *Cell* boundary. Both sides of such internal Faces point to the same enclosed region. The concept of a non-manifold *Cell* is required to model internal "semi-partitions" of architectural spaces which do not fully divide the cell. (**Fig. 5**)

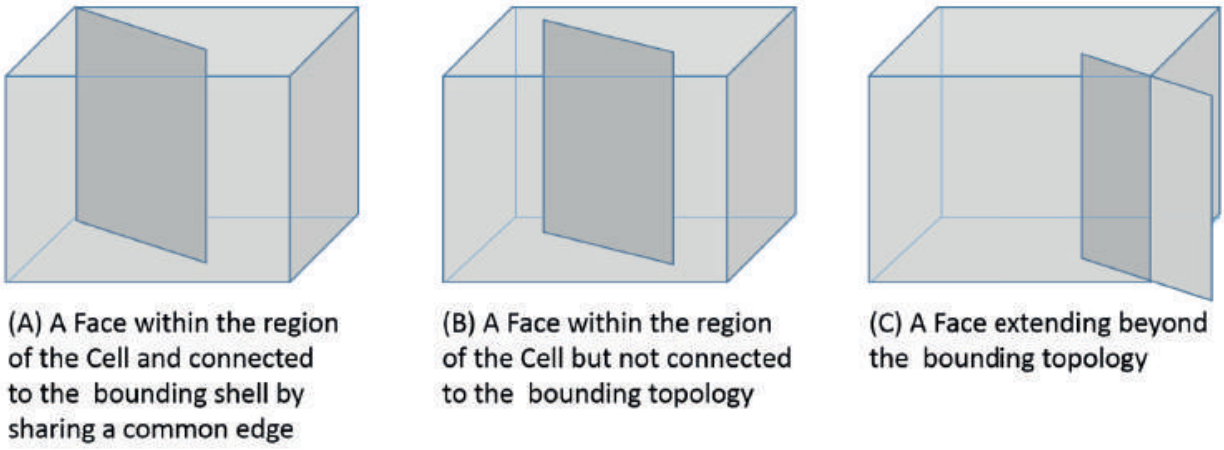


Figure 5: Different configurations of non-manifold Cells.

4.2 Cellular Topology

Cellular Topology is implemented as a *CellComplex*, where some Faces of the *Cell* are also the external boundary, while other Faces form the boundary between adjacent *Cells.* Cellular Topology can be used to model a building which is partitioned into different architectural spaces (**Fig. 6**).

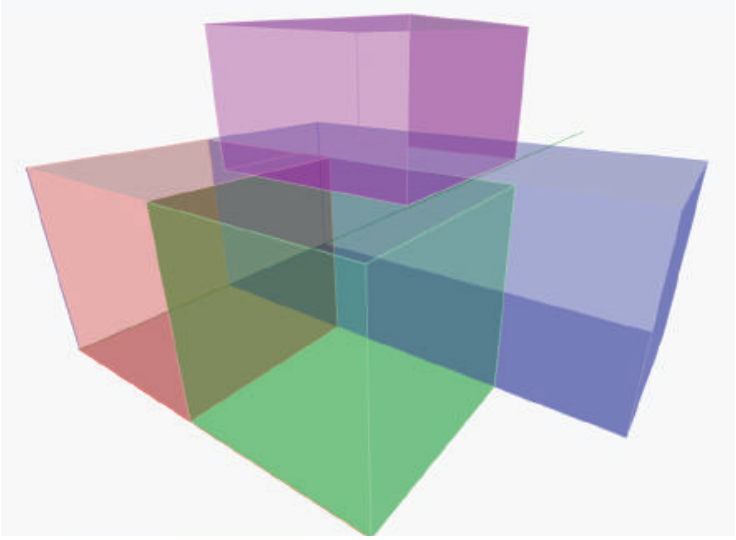


Figure 6: Cellular Topology modelled as a CellComplex.

4.3 Mixed dimensionality Topological models

In non-manifold topology it is possible to construct a single topological model composed of entities of different types and dimensionality. The concept of a mixed dimensionality topology is implemented as a *Cluster* and can be used to create an idealized model of the structure of a building (**Fig. 7**).

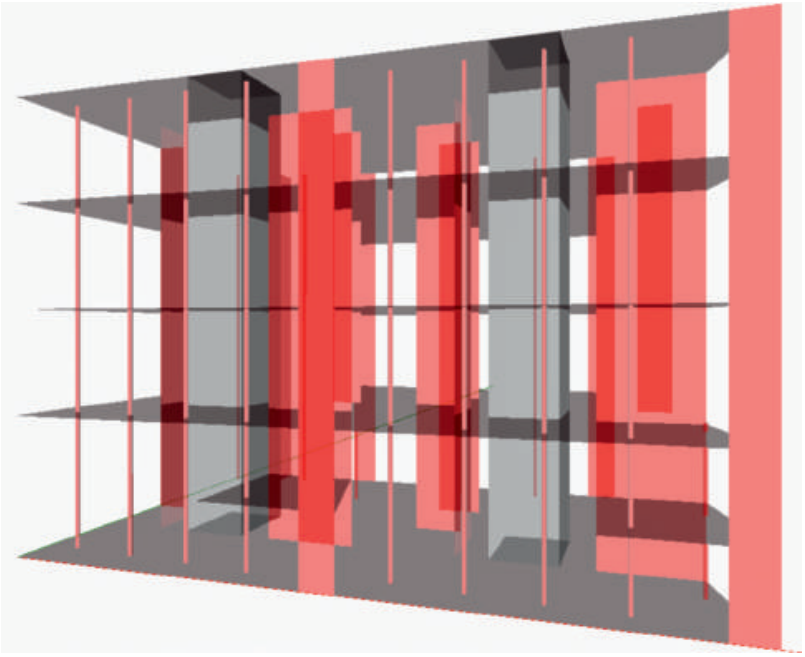


Figure 7: A mixed dimensional model with Edges representing the column centre lines and Faces representing floor slabs, blade columns and shear walls. Cells are used to represent the building cores.

4.4 Regional Topology

In conventional topological modelling, higher dimensional topological entities are constructed from lower dimensional ones. Higher dimensional topological entities are connected because they share common lower dimensional entities. For example, adjacent *Cells* within a *CellComplex* may share a common *Face*.

However, in the domain of architecture there are other forms of connectedness which cannot be directly expressed in this way. For example, a column can be idealised as an *Edge*. A floor or ceiling can be idealised as a *Face*. We intuitively understand that a column (*Edge*) may connect a floor (*Face*) to a ceiling (*Face*), but how can this be described if the column is in the middle of the floor and when there is no topology within the definition of the floor and ceiling *Faces* which is shared with the *Vertices* defining the column's *Edge*? (**Fig. 8**).

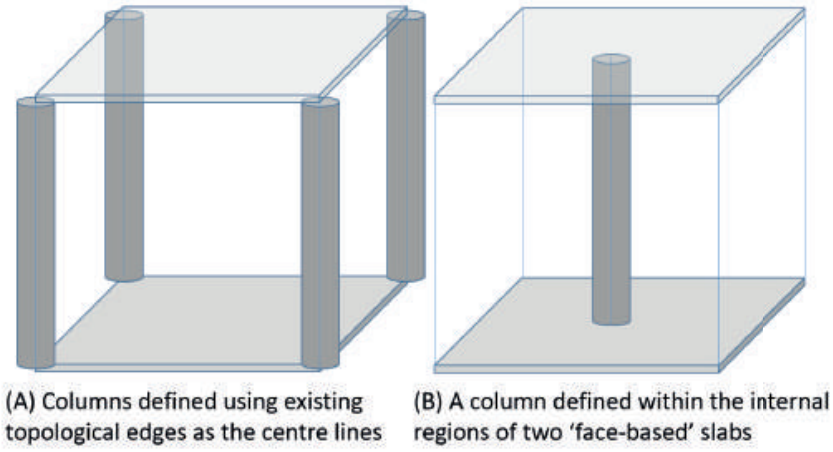


Figure 8: Defining the "Context" to describe the connectedness of two topologies where one entity exists within the region of the other entity and when the two entities do not share any common constituent topology.

Similar issues arise when we consider an internal boundary within a *Face*. For example the *Face* may represent a wall and the internal boundary may define an Aperture such as a window or a door. We intuitively understand that the Aperture (as a single 2D region) is contained within the 2D region of the *Face*, with no shared topology.

To address these issues, Topologic introduces the concept of a *context* to represent the connectivity between two topological entities which do not otherwise share common topology. In this example, the Aperture is the *subject* (representing a window) and is defined *within the region* (or *context*) of the *host Face* (representing the wall). The

user may optionally specify that the *context* defines a locational "link" between the *subject* and the *host*. Here the vertices of the subject are defined in the parameter space of the host and are now dependent on any changes which are applied to the host. (**Fig. 9**).

The *context* with parametric coordinates is only used when there is no shared topology connecting the two entities (**Fig. 10**).

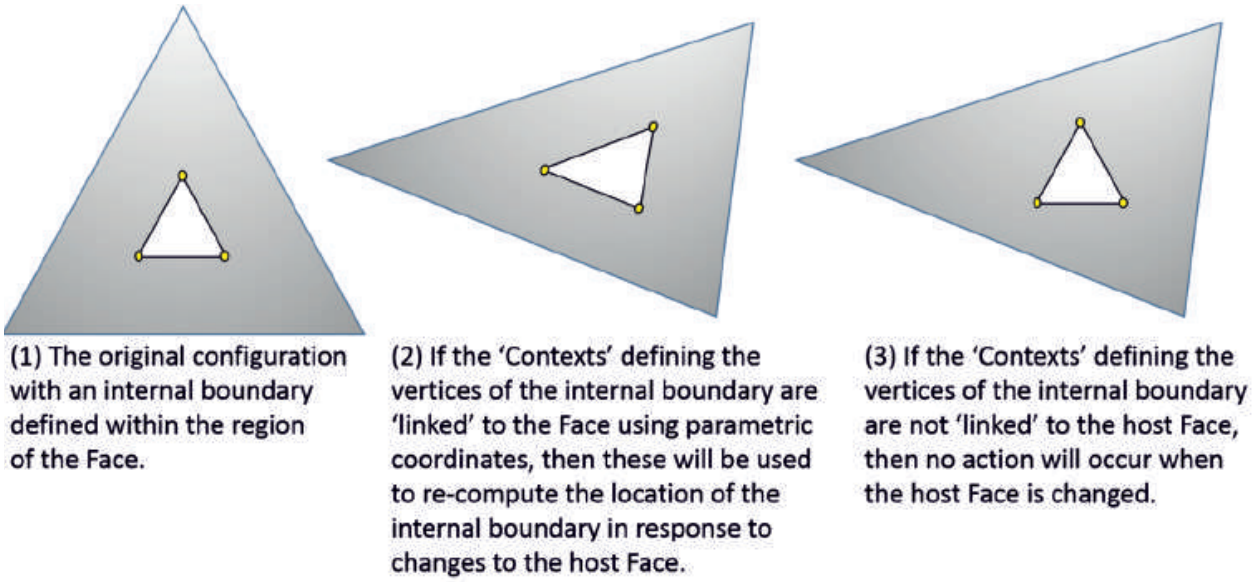


Figure 9: The option to "link" the subject topology to the host topology.

4.5 Variable topology

In architecture, spatial divisions may be "hardcoded" as distinct rooms separated by physical walls. While buildings appear to be solid, one of the central tenets of architecture is that the use of space within a building is or should be flexible. We think of multi-use or reconfigurable spaces.

There appears to be no established architectural methodology which prescribes how the topology of a building emerges. In fact, the architectural design process is quite imprecise. It may start with an occupancy model and a description of the anticipated activities of the occupants. Activities may vary in time and space. Activities may overlap. Alexander (1965) noted that neither activities nor space could be adequately described by a simple hierarchical decomposition. The process by which activities get translated into specific spatial enclosures and the choice as to which boundaries of these enclosures are actually materialised as walls or are left as purely virtual, is often a matter of contention (**Fig. 11**). Virtual partitions may also be used in the topological representation of

other building sub-systems. For example, an atrium may be considered as a single continuous space, or it may be considered to be subdivided into different air conditioning zones without physical partitions. Depending on the simulation parameters, virtual Faces could be inserted and can be represented in the analytical model either as adiabatic or diathermic.

More generally, architecture is often characterized by degrees of spatial partitioning and connectedness. How can these different and sometimes ambiguous architectural concepts of space be represented with topology? Topology provides a formal way to represent connectedness, but

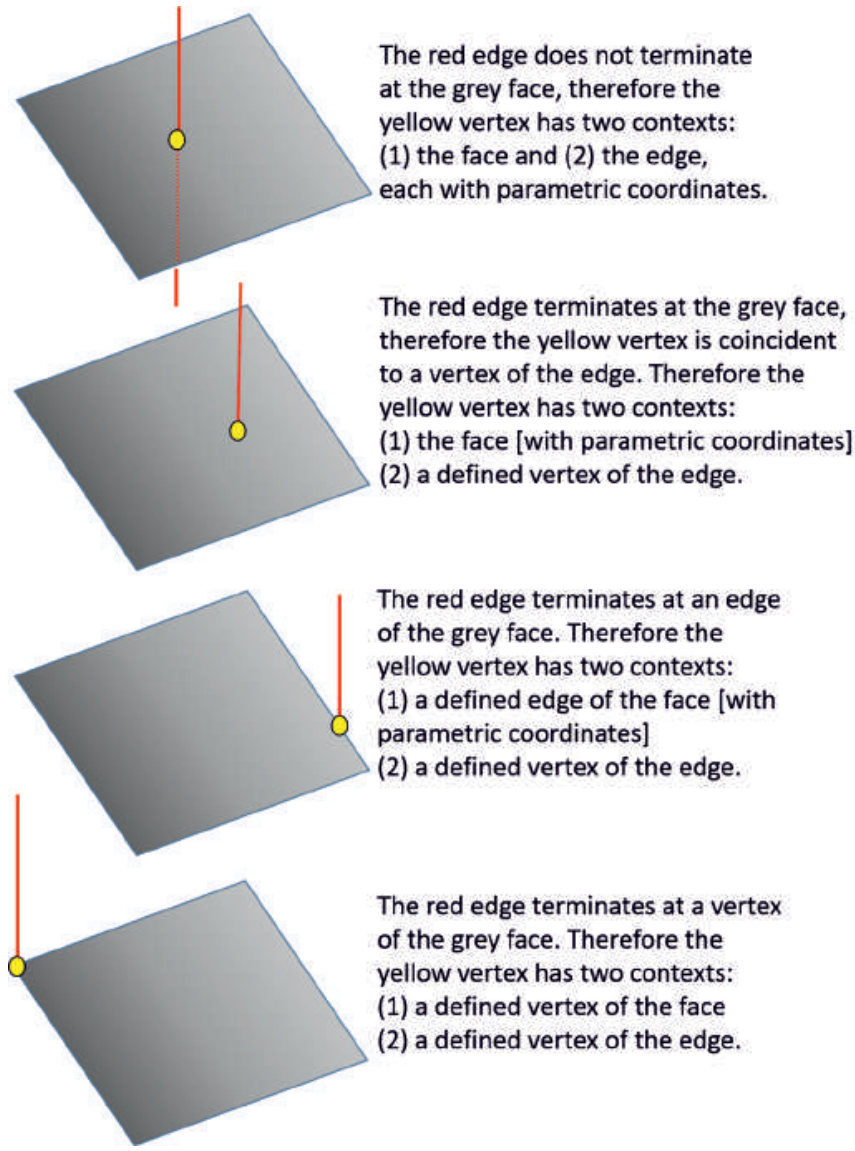


Figure 10: Given the intersection of an Edge (red) and a Face (grey) in different configurations, then the concept of the context (with parametric coordinates) is used when the resulting Vertex occurs within a region of the intersecting topologies.

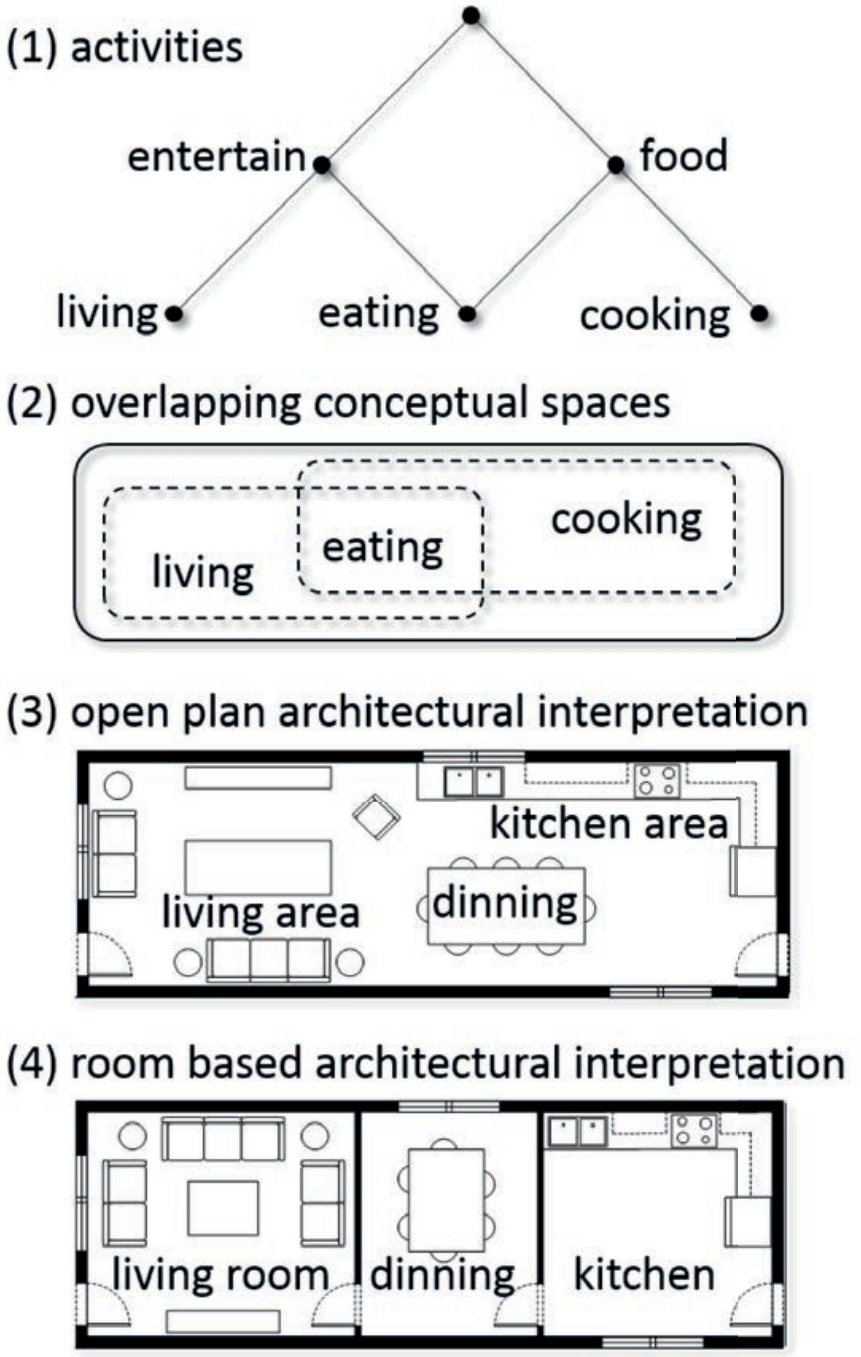


Figure 11: The choice of spatial configuration often starts with identifying underlying activities of the occupants (1). These activities and their spatial requirements may overlap. It may be inappropriate to describe these as a simple hierarchical decomposition (with acknowledgement to Alexander, 1965). The process by which activities are translated into defined conceptual spaces (2) and are further translated into recognisable enclosures (3) or into specific rooms (4) often reflects architectural intuition rather than a defined methodology.

when applied to architecture, it requires the user to choose what is being connected.

If two adjacent regions have exactly the same contents with the same behaviour and are so intimately connected that there is no effective barrier between them, then perhaps they should be considered as a single region. So, the ultimate form of connectedness is the unification of two adjacent regions into a single region or *Cell*. Therefore, a *Cell* is more than just a continuous 3D region. It also implies that what is contained represents a level of homogeneity, which has appropriate meaning within the application domain.

If *Cells* represent spaces and *Faces* represent walls (or partitions) then operations which add or remove the *Faces* of *Cells* within a *CellComplex* can radically change the topology. The result of a modelling operation to an existing topological construct may change the "type" of that construct. The advantage of Topology is that it tells the architectural users exactly what has been modelled in terms of partitioning and connectedness and the type of the result (**Fig. 12**).

The general conclusion is that, where possible, the user should define a single canonical non-manifold topology model describing the maximal partitioning of space. Different subdivisions may be combined to represent the spaces required for different activities. Different dual graphs can be constructed as required by different analysis and simulation applications (**Fig. 13**).

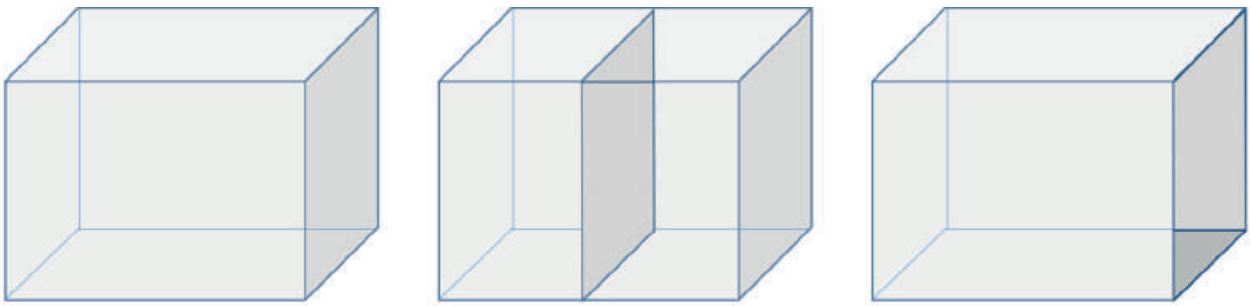


Figure 12: Editing operations to add or remove topological components can have a radical affect, including changing the type of topological construct.

5. Applying topology in analysis, simulation and fabrication

Vitruvius distinguished between the practical aspects of the architecture (*fabrica*) and its rational and theoretical foundation (*ratiocination*) (Pont 2005). Establishing topological relationships was found to be an essential component of the setting out of the conceptual principles of a design project (Jabi et al. 2017). Non-manifold topology was also found to be a consistent representation of entities that can be thought of as loci, axes, spaces, voids, or containers of other material.

This concept was previously explored by the authors in the context of energy analysis, façade design, and additive manufacturing of conformal cellular structures (Jabi 2016; Fagerström, Verboon, and Aish 2014; Jabi et al. 2017).

5.1 Energy analysis

A proof of concept implementation of non-manifold topology for energy analysis allowed the user to create simple regular manifold polyhedral geometries and then segment them with planes and other geometries to create a non-manifold *CellComplex* (Chatzivasileiadi, Lannon, et al. 2018; Wardhana et al. 2018). The tool can create complex geometry that produces outputs that are highly compatible with the input requirements for energy analysis software. *Cells* within the *CellComplex* are converted to spaces with surfaces, and bespoke glazing sub-surfaces, and set to their own thermal zones.

5.2 Digital fabrication

A proof of concept implementation of non-manifold topology for digital fabrication allowed a *CellComplex* to be conformed to a NURBS-based design envelope (Jabi et al. 2017).The resulting model used topological and geometric queries amongst adjacent *Cells* to create rules for depositing material. These query results were used to identify boundary conditions and to deposit material only where needed. This improved the material efficiency and resulted in a higher mechanical and structural profile for the 3D printed model.

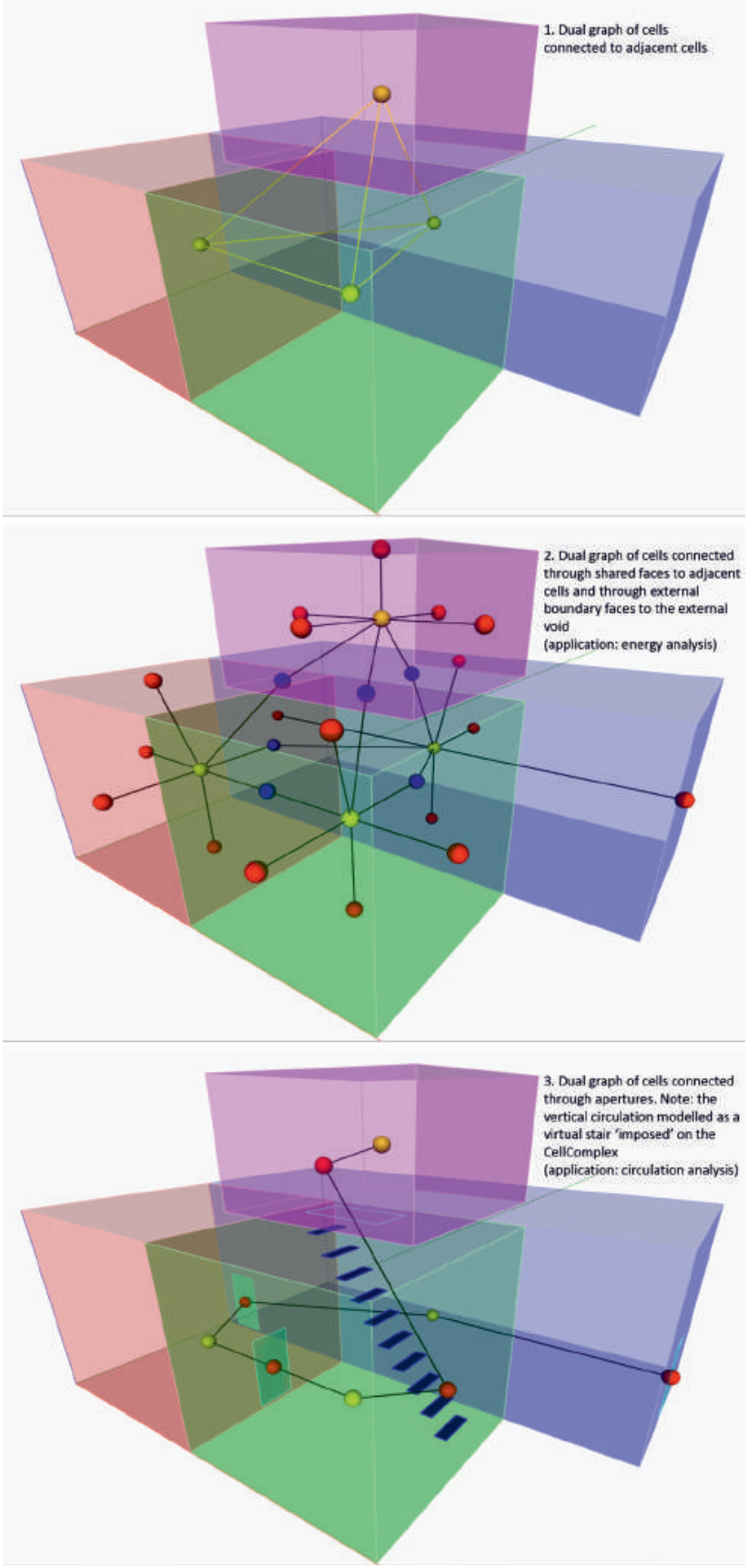


Figure 13: Dual graphs can be constructed which describe alternative connectivity of the *Cell*s representing architectural spaces and used as different analytical models.

6. Conclusions

New design technologies often emerge in response to the limitations of existing technologies and have the potential to benefit the architectural design process. Understandably, the founding concepts and terminology may be unfamiliar to architectural practitioners which may inhibit adoption of these technologies.

The challenge in developing Topologic has been to maintain the theoretically consistent use of topological concepts and terminology, yet relate these to the more ambiguous concepts of space and "connectedness" found in architecture. The application of topology as a direct link between architectural conceptual modelling and relevant analysis applications is becoming established. A more challenging task is to explore how topology can contribute to the way in which architecture as the "enclosure of space" can be conceptualised.

Acknowledgments

The Topologic project is funded by a Leverhulme Trust Research Project Grant (Grant No. RPG-2016-016).

References

Alexander, Christopher. 1965. "A City is Not a Tree." Architectural Forum, 122 (1): 58–62.

Aish, Robert. 1997. "MicroStation/J." In *Next Generation CAD/ CAM/CAE Systems, NASA Conference Publication 3357,* edited by Ahmed K. Noor and John B. Malone, 167–80. Hampton, Virginia: National Aeronautics and Space Administration. Accessed June 25 2018 https://www.researchgate.net/publication/320507941_MicroStationJ

Aish, Robert, and Aparajit Pratap. 2013. "Spatial Information Modeling of Buildings Using Non-Manifold Topology with ASM and DesignScript." In *Advances in Architectural Geometry 2012*, edited by Lars Hesselgren, Shrikant Sharma, Johannes Wallner, Niccolo Baldassini, Philippe Bompas, and Jacques Raynaud, 25–36. Vienna: Springer Vienna. doi:10.1007/978-3-7091-1251-9.

Björk, Bo-Christer. 1992. "A Conceptual Model of Spaces, Space Boundaries and Enclosing Structures." Automation in Construction 1 (3): 193-214. doi:10.1016/0926-5805(92)90013-A.

Bronson, Jonathan, Joshua A. Levine, and Ross Whitaker. 2014. "Lattice Cleaving: A Multimaterial Tetrahedral Meshing Algorithm with Guarantees." IEEE Transactions on Visualization and Computer Graphics 20 (2): 223–37. doi:10.1109/TVCG.2013.115.

Chang, T.W., and Robert Woodbury. 1997. "Efficient Design Spaces of Non-Manifold Solids." In *Second Conference on Computer-Aided Architectural Design Research in Asia,* 335–44. Taiwan: CAADRIA.

Chatzivasileiadi, Aikaterini, Simon Lannon, Wassim Jabi, Nicholas Mario Wardhana, and Robert Aish. 2018. "Addressing Pathways to Energy Modelling through Non- Manifold Topology." In *SIMAUD 2018: Symposium for Architecture and Urban Design*, edited by Daniel Macumber, Forrest Meggers, and Siobhan Rockcastle. Delft, the Netherlands: Society for Modeling & Simulation International (SCS).

Chatzivasileiadi, Aikaterini, Nicholas Mario Wardhana, Wassim Jabi, Robert Aish, and Simon Lannon. 2018. "A Review of 3D Solid Modeling Software Libraries for Non-Manifold Modeling." In CAD'18 - the *15th Annual International CAD Conference.* Paris, France: CAD Solutions, LLC.

Curtis, William J. R. 1996. *Modern Architecture* Since 1900. London: Phaidon Press.

Eastman, Charles M., and Anastassios Siabiris. 1995. "A Generic Building Product Model Incorporating Building Type Information." Automation in Construction 3 (4): 283–304. doi:10.1016/0926-5805(94)00028-L.

Ellis, Peter G., Paul A. Torcellini, and Drury B. Crawley. 2008. "Energy Design Plugin : An EnergyPlus Plugin for SketchUp." In I*BPSA-USA SimBuild 2008 Conference*, 238–45. Berkeley, California. http://www.nrel.gov/docs/fy08osti/43569.pdf.

Fagerström, Gustav, Erik Verboon, and Robert Aish. 2014. "Topo-Façade: Envelope Design and Fabrication Planning Using Topological Mesh Representations." In *Fabricate 2014*, edited by Fabio Gramazio, Matthias Kohler, and Silke Langenberg, 36–43. Zürich: GTA Publishers.

Hensen, J.L.M., and R. Lamberts. 2011. *Building Performance Simulation for Design and Operation*. London: Spon Press.. doi:10.4324/9780203891612.

Hui, Annie, and Leila De Floriani. 2007. "A Two-Level Topological Decomposition for Non-Manifold Simplicial Shapes." In SPM '07: *Proceedings of the 2007 ACM Symposium on Solid and Physical Modeling*, 355–60. New York, NY, USA. doi:10.1145/1236246.1236297.

Jabi, Wassim. 2014. "Parametric Spatial Models for Energy Analysis in the Early Design Stages." In SIMAUD 2014: *Symposium for Architecture and Urban Design,* edited by D Gerber and R Goldstein, 17–24. San Diego, CA: Simulations Councils, Inc. .

Jabi, Wassim. 2016. "Linking Design and Simulation Using Non-Manifold Topology." Architectural Science Review 59 (4): 323–34. doi:10.1080/00038628.2015.1117959.

Jabi, Wassim, Robert Aish, Simon Lannon, Aikaterini Chatzivasileiadi, and Nicholas Mario Wardhana. 2018. "Topologic: A Toolkit for Spatial and Topological Modelling." In *Computing for a Better Tomorrow, Proceedings of the 36th eCAADe Conference,* edited by Anetta K pczy ska-Walczak, Sebastian Białkowski, and Mateusz Pankiewicz. Lodz, Poland: eCAADe.

Jabi, Wassim, Shwe Soe, Peter Theobald, Robert Aish, and Simon Lannon. 2017. "Enhancing Parametric Design through Non-Manifold Topology." Design Studies 52 (May): 96–114. doi:10.1016/j.destud.2017.04.003.

Maile, Tobias, James T. O'Donnell, Vladimir Bazjanac, and Cody Rose. 2013. "BIM - Geometry Modelling Guidelines for Building Energy Performance Simulation." In *BS2013: 13th Conference of International Building Performance Simulation Association, Chambéry, France*, 3242–49. International Building Performance Simulation Association.

Masuda, H. 1993. "Topological Operators and Boolean Operations for Complex-Based Nonmanifold Geometric Models." Computer-Aided Design 25 (2): 119–29. doi:10.1016/0010-4485(93)90097-8.

Nguyen, Trunc Duc. 2011. "Simplifying The Non-Manifold Topology Of Multi-Partitioning Surface Networks." MA Thesis, Washington University. http://openscholarship.wustl.edu/etd/510/.

Pont, Graham. 2005. "The Education of the Classical Architect from Plato to Vitruvius." Nexus Network Journal 7 (1): 76–85. doi:10.1007/s00004-005-0008-0.

Wardhana, Nicholas Mario, Aikaterini Chatzivasileiadi, Wassim Jabi, Robert Aish, and Simon Lannon. 2018. "Bespoke Geometric Glazing Design for Building Energy Performance Analysis." In *MonGeometrija 2018*, edited by Vesna Stojakovi . Novi Sad, Serbia.

Weiler, Kevin J. 1986. "Topological Structures for Geometric Modeling." PhD Thesis, Rensselaer Polytechnic Institute.

Wix, Jeffrey. 1994 "Interoperability: a future context". Paper presented at the CAMP Autodesk Developer Group Europe Conference 1994, Autodesk.

Topology finding of structural patterns

Robin Oval, Matthias Rippmann, Romain Mesnil,
Tom Van Mele, Olivier Baverel, Philippe Block

Robin Oval
robin.oval@enpc.fr
Ecole Nationale des Ponts et Chaussées, France/ETH Zurich, Switzerland

Matthias Rippmann
rippmann@arch.ethz.ch
ETH Zurich, Switzerland

Romain Mesnil
romain.mesnil@enpc.fr
Ecole Nationale des Ponts et Chaussées, France

Tom Van Mele
van.mele@arch.ethz.ch
ETH Zurich, Switzerland

Olivier Baverel
olivier.baverel@enpc.fr
Ecole Nationale des Ponts et Chaussées, France

Philippe Block
block@arch.ethz.ch
ETH Zurich, Switzerland

Keywords:
Structural design, discrete shells, structural patterns, topology finding

Abstract

This paper presents a strategy for the exploration of the topology of structural patterns, such as beam grids for gridshells or voussoir tessellations for masonry vaults. The authors define *topology finding*, by analogy and in complement to form finding, as the design of the connectivity of patterns in relation to architectural and structural requirements. The method focuses on the design of the singularities in the pattern through the automatic generation and subsequent rule-based editing of a coarse quad mesh that encodes the properties of the singularities and their relationships before mesh densification, pattern mapping, geometrical exploration and performance assessment.

(a)

(b)

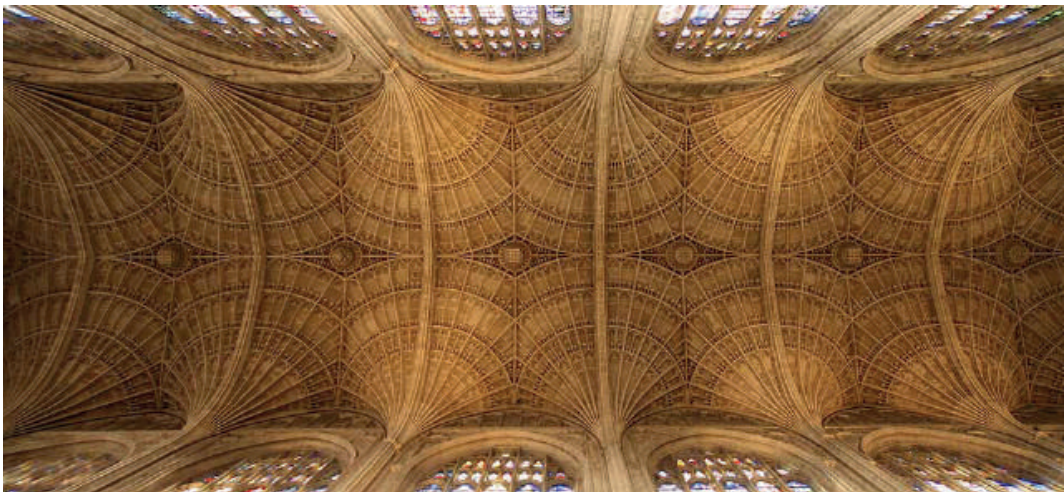


Figure 1: Two examples of structural patterns for shell-like structures: (**a**) beam grid of the Hippo House in Berlin, Germany (photo credit: sbp.de) and (**b**) voussoir tessellation of the King's College Chapel in Cambridge, England (photo credit: kings.cam.ac.uk).

1. Introduction

1.1 Structural patterns

Shell structures span large areas efficiently thanks to their double curvature that provides geometrical stiffness. These structures are often discretised in a pattern, which constitutes the load-bearing system once fabricated and assembled. Beam grids for gridshells or voussoir tessellations for masonry vaults are such examples of structural patterns for shells (**Fig. 1**). The design of structural patterns is a complex and rich process influenced by many aspects of the projects, such as aesthetics, statics, fabrication, assembly, as well as sustainability and cost.

1.2 Topology of patterns

Reciprocally, the choice of pattern greatly influences these criteria since a pattern performs better for some criteria and worse for others. More specifically, the topology – or connectivity – of a structural pattern matters because it sets the bounds of the geometrical design space, within the general design space, for form finding and other geometrical design approaches. This geometrical design space, which represents all the possible geometries for a given topology, may not contain efficient or even feasible designs. Indeed, the topology, and more specifically the set of singularities in a pattern, define the qualitative degrees of freedom for design and optimisation, as illustrated by Schiftner and Balzer (2010) for competing statics and fabrication requirements. For this reason, designers need conceptual and practical tools to allow them to flexibly and efficiently explore the topology of structural patterns during early-stage design, as stated by Harding et al. (2012), and make design choices which balance the different performance requirements.

1.3 Contributions and outline

This research introduces topology finding of structural patterns. Topology finding deepens the available design space for geometrical exploration. The authors use a specific design space structure and focus on the design of the singularities through a coarse quad mesh.

This new practical design tool is implemented as a package for COMPAS (Mele et al. (2017)), an open-source computational Python framework for researchers, professionals and students working in the fields of architecture, engineering and digital fabrication.

Section 2 shows the design space structure used for topological design of patterns. Section 3 develops an automated generation scheme for the singularities of the pattern using the medial axis of a shape. Section 4 presents a rule-based editing method for the singularities of the pattern using a grammar. Section 5 illustrates how to visualise designs from a topological space based on their relative performance.

2. Design spaces

Topology finding relates to the exploration of the three topology-related design spaces: singularities, density and pattern, in complement to the geometrical design space. These design spaces derive from each other as shown in Figure 2 for the gridshell of the Great Court at the British Museum in London, England, whose geometry has been analytically defined by Williams (2001), and which serves as main example throughout this paper.

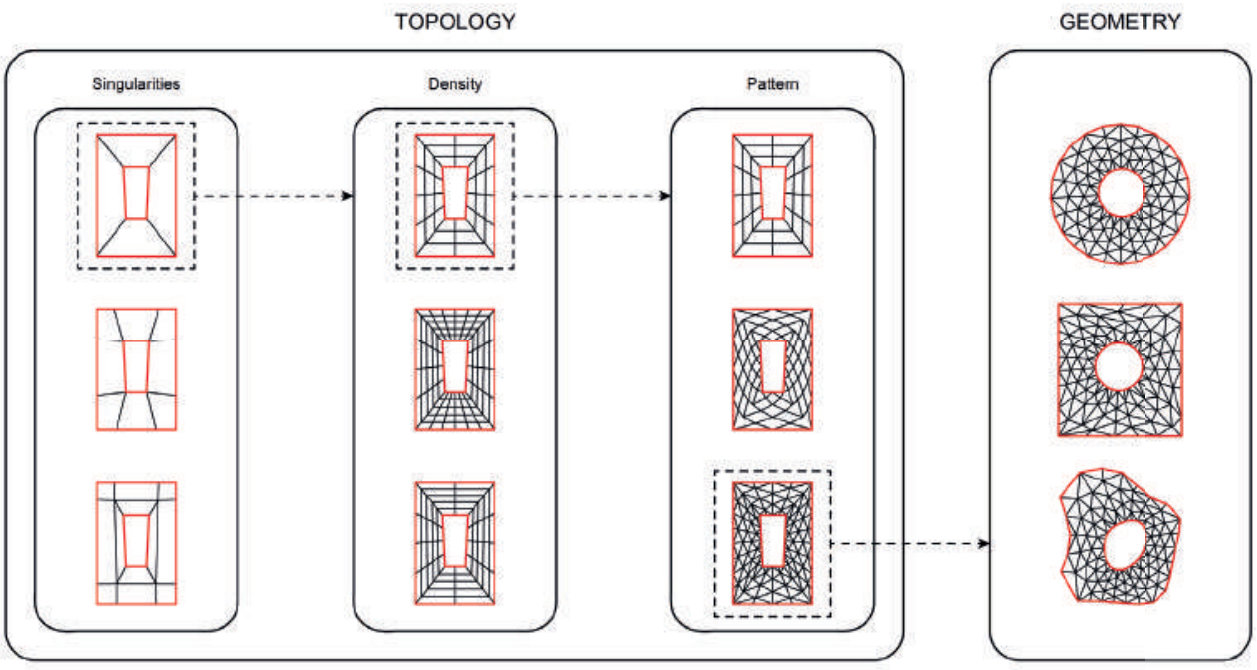


Figure 2: The design space structure for topological exploration of patterns.

The design of the singularities in a pattern is handled at the level of a coarse quad mesh (which can also be referred to as control mesh or

patch set), which includes the vertices which represent the singularities and whose edges represent their connectivity in the pattern, and which defines the parameterisation directions of the shape.

The density is set through densification of the coarse quad mesh into a quad mesh and the pattern is derived from a transformation of the quad mesh elements, for instance through global conversion of the initial quad mesh into its dual mesh, its diagonal mesh or through triangulation, supplemented by local modifications.

The geometry of the pattern is explored through smoothing, form finding, form optimisation etc. Although presented linearly, the designer can move downstream and upstream the design space structure during the design process.

All the patterns that are presented in this paper are untrimmed and characterised as being aligned with the boundaries, which benefits the aesthetics, favors loads paths parallelly or perpendicularly to the boundaries and avoids the creation of irregular elements to fabricate and assemble. However, this characteristic can induce practical limitations when performing planarisation with constrained straight boundaries, as mentioned by Tang et al. (2014).

The first challenge is about how to enter the design space, manual drawing of a coarse quad mesh requiring time and experience.

3. Automated generation

We describe a scheme to automatically generate an initial coarse quad mesh, and a corresponding pattern, on a NURBS surface input. The input surface is initially mapped to the plane based on its UV-parameterisation, then the coarse quad mesh is generated on the planar map before being remapped back onto the surface. The density and the pattern are set, before being relaxed on the surface and further processed.

3.1 Singularities

The singularities are derived from the medial axis of the surface, also known as its topological skeleton, introduced by Blum (1967), which consists in a dimensional reduction of the the surface into a set of curves called medial branches. The steps of the process to obtain a coarse quad mesh are show in Figure 3.

The input NURBS surface is mapped to the plan and its boundaries are subdivided into a set of vertices for a Delaunay triangulation. The key points from the Delaunay mesh are: the singular points S at the centroids of singular faces (faces adjacent to three other faces), the boundary points B at the vertices of singular faces, and the corner points C (two-valent boundary vertices). The medial axis is consituted by the branches connecting the circumcentres of the adjacent Delaunay faces: the S-S and S-C branches. The medial axis defines a natural decomposition of the surface with singularities stemming from its topology. Three simple heuristics based on the connectivity of the Delaunay mesh, which relate to Rigby (2003), are used to generate a coarse quad mesh from the medial axis: pruning to remove the S-C branches; grafting to add the S-B branches; and closing to add the B-B and B-C branches. The extracted connectivity of these branches defines the coarse quad mesh.

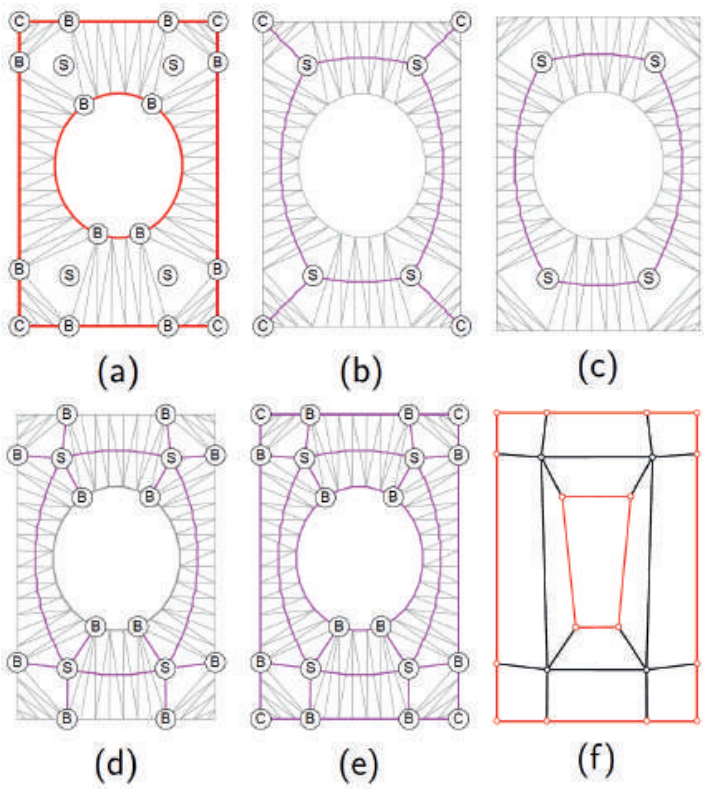


Figure 3: Medial-axis-based automated generation of a coarse quad mesh from the planar map of an input surface: (**a**) triangulation, (**b**) skeletonisation, (**c**) pruning, (**d**) grafting, (**e**) closing, (**f**) extraction.

3.2 Pattern

From the automatically generated coarse quad mesh, a smooth pattern can be directly generated on the input surface, as shown in Figure 4. Once the singularities are generated and the coarse quad mesh is mapped back onto the input surface, the density and the pattern can be chosen.

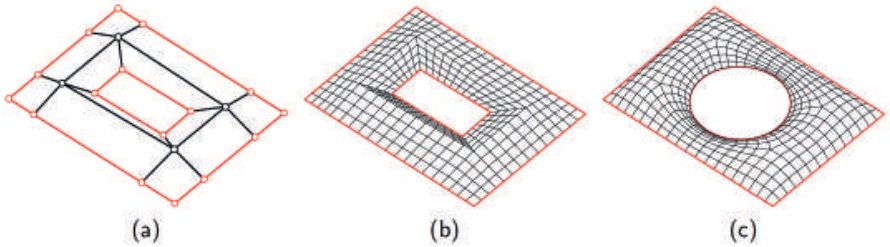


Figure 4: Automated generation of a smooth quad mesh on a surface from a coarse quad mesh: (**a**) remapping, (**b**) densification, (**c**) relaxation.

The density of the pattern is controlled per quad face strip in the coarse quad mesh, because each pair of opposite edges in each quad face share the same density parameter. The dependent edges are grouped in independent groups that correspond to the density parameters per face strip. The designer controls all of these degrees of freedom and can automatically compute subdivision parameters based on a target length and the average length of the edges in each group. The coarse quad mesh is then densified into a quad mesh that sets the density of the pattern.

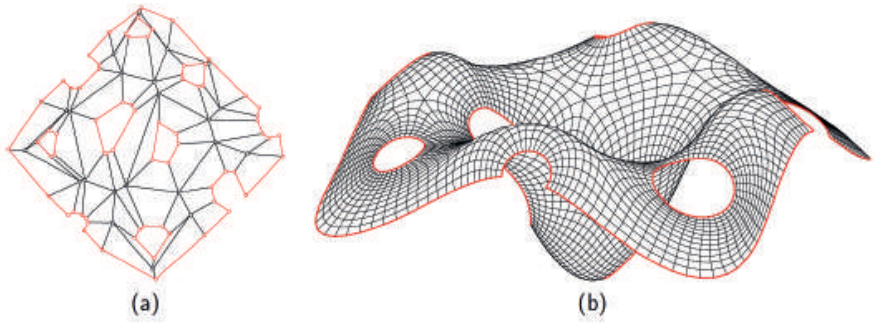


Figure 5: Automated generation of a smooth quad mesh on an input free-form surface: (**a**) coarse quad mesh, (**b**) smooth quad mesh.

The pattern is relaxed on the surface with constraints at boundary corners and along boundary curves using a smoothing algorithm, such as (area-weighted) Laplacian smoothing (Botsch et al. (2010)), to provide

a smooth starting geometry for further exploration. Figure 5 shows an example with stronger double curvature after automated generation of a coarse quad mesh and tuned densification and relaxation, necessary to compensate the distortions between the coarse quad mesh and the NURBS surface, even though the vertices of the mesh lie on the surface.

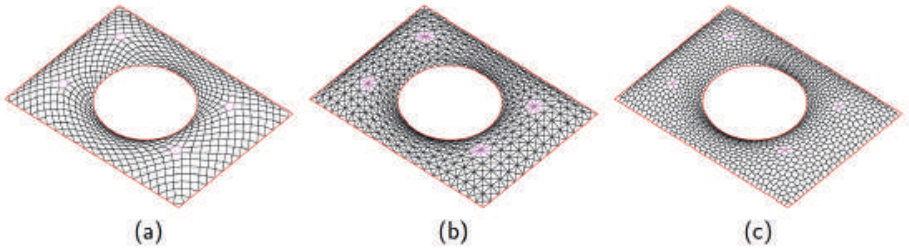


Figure 6: Automated generation of smooth patterns on a surface using Conway operators: (**a**) ambo, (**b**) kis, (**c**) gyro, with singular elements highlighted in magenta.

The pattern is derived from global transformation of the quad mesh and its elements, for instance using the operators by Conway et al. (2016), as shown in Figure 6 applied to the quad mesh in Figure 4, and already investigated by Shepherd and Pearson (2013) adn applied to the original pattern of the British Museum. The singularities from the quad mesh are converted into irregular vertices or faces in the pattern. Another round of constrained relaxation on the surface provides smoothness to the pattern.

3.3 Form finding

The input surface provides a starting geometry with the main topological information. Figure 7 shows a thrust network resulting from a funicular form finding process using RhinoVAULT (Rippmann and Block (2013)) after conversion of the relaxed pattern into a form diagram projected to the XY plane. The input curved surface serves as design intent and helps to reduce the element distortion due to the slope.

Nevertheless, a planar input surface is sufficient to generate a pattern, as shown for the design in Figure 8, inspired by the Solemar baths in Bad Dürrheim, Germany. The quad mesh structure, the smoothness, the low number of singularities and the alignment to the boundaries permit clear readability between the reciprocal form and force diagrams, a key aspect of interactive graphical design methods and empowers the designer to perform force-based geometrical exploration.

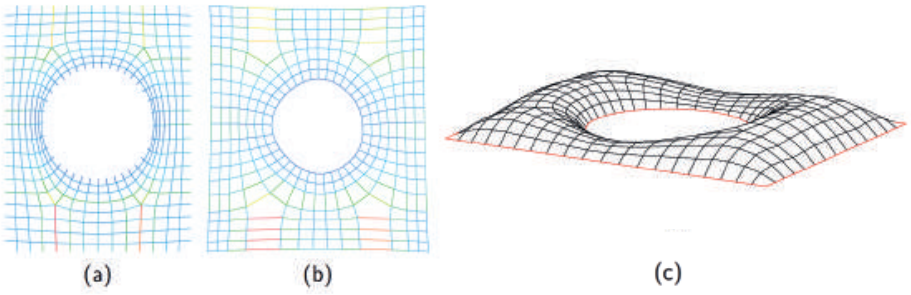


Figure 7: Funicular form finding revisiting the geometry of the British Museum: (**a**) form diagram, (**b**) force diagram (rotated by 90°), (**c**) thrust network.

3.4 Pole points

Pole points are a special type of singularities whose valency depends on the density of the pattern. Poles are integrated in the coarse quad mesh by allowing pseudo-quad faces which are geometrically as triangles but topologically as quads with a double vertex at the pole location, as shown in Figure 9.

Point features complete the input data. These points are added to the set of vertices of the Delaunay mesh, which displays thereby additional singular faces around the point features. The resulting coarse quad mesh includes pseudo-quad faces around the point features, which serve as double vertices, marked as filled dots. Thus, the generated pattern features additional singularities and poles.

Poles can stem from statics reasons such as concentrated forces, loads or reactions, or geometrical reasons such as umbilical points. The design in Figure 10 revisits the ribbed slabs of Pier Luigi Nervi by showing a smooth planar quad mesh with multiple point features: although the pattern does not derive from the integration of principal stress directions for a load combination, the design is informed by the statics system by heuristically adding poles at the location of columns to provide a high number of load paths towards the supports.

The second challenge is about how to move in the design space, since the singularities from the medial axis may not be the best choice regarding the relevant requirements, though they naturally derive from the topology of the boundaries.

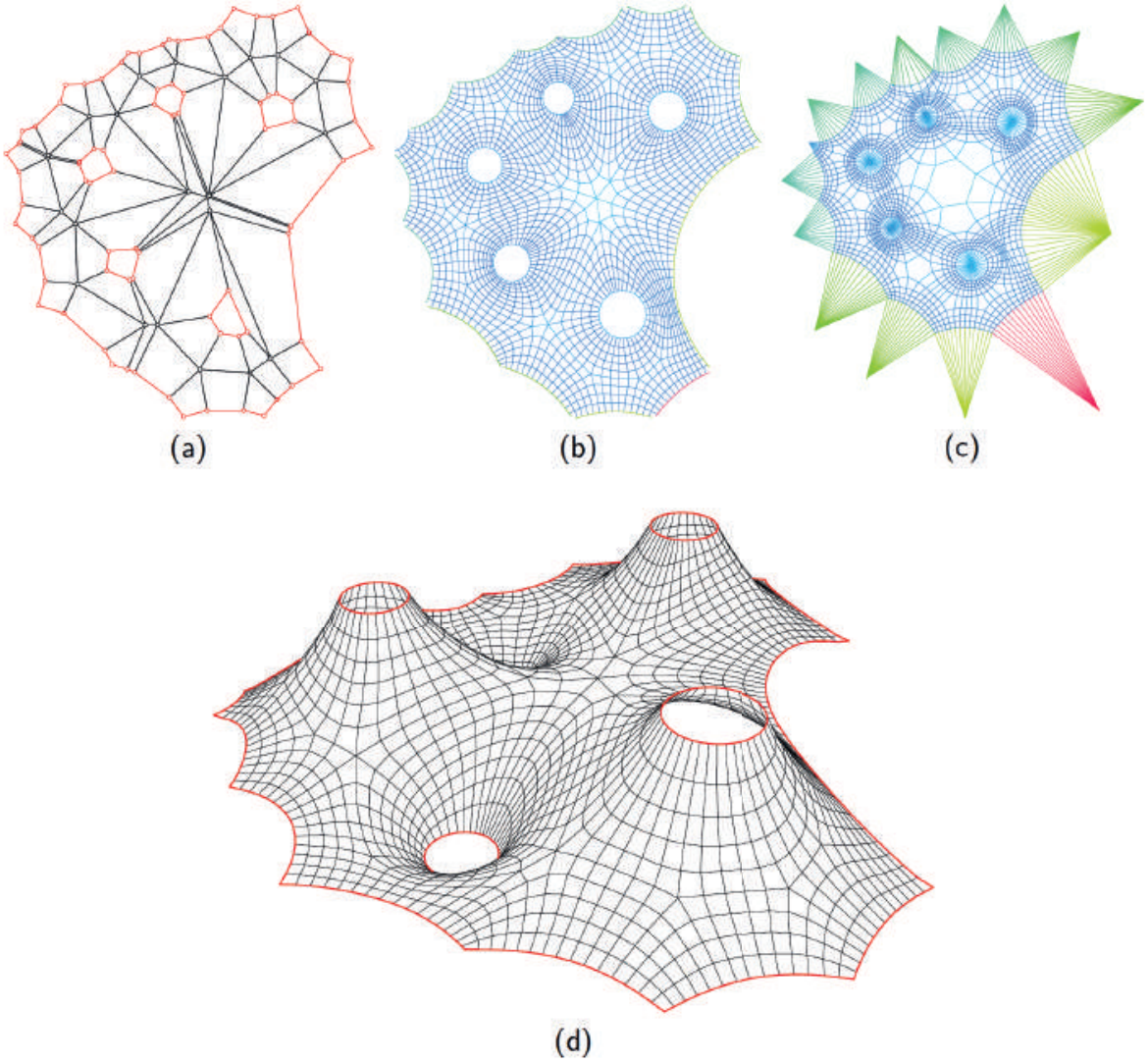


Figure 8: Automated topological generation and funicular form finding revisiting the Solemar baths in Bad Dürrheim, Germany: (**a**) coarse quad mesh, (**b**) form diagram, (**c**) force diagram (rotated by 90°), (**d**) thrust network.

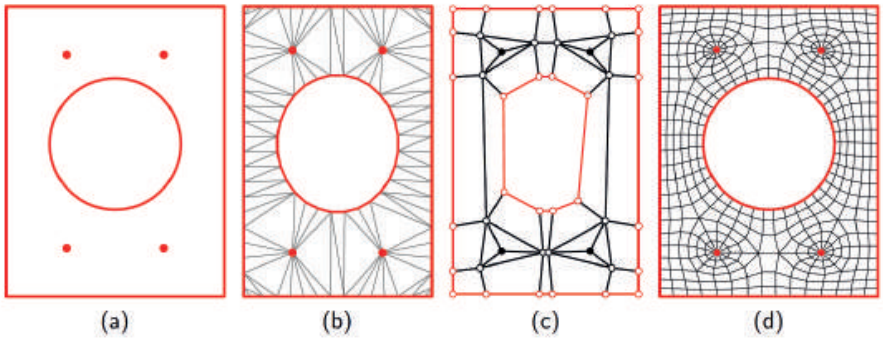


Figure 9: Automated generation of a smooth quad mesh with poles: (**a**) input with point features, (**b**) Delaunay mesh, (**c**) coarse quad mesh, (**d**) smooth quad mesh with poles.

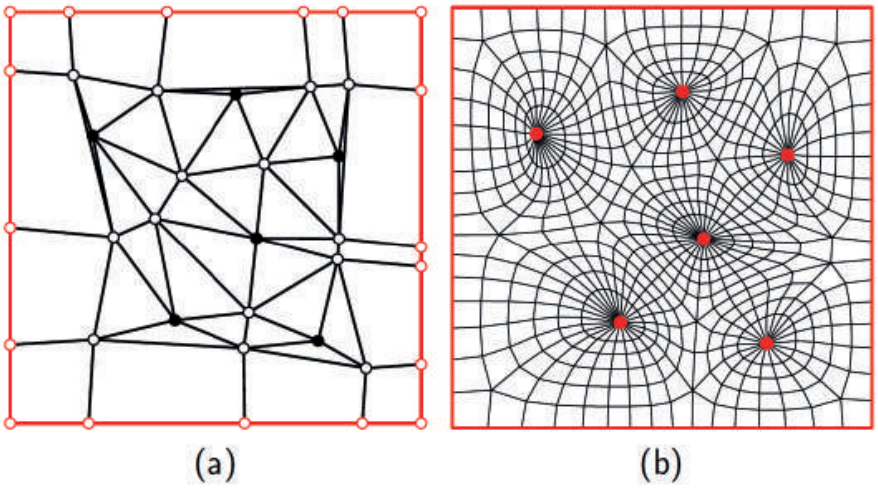


Figure 10: Automated generation of a smooth quad mesh with multiple poles: (**a**) coarse quad mesh with pseudo-quad faces, (**b**) smooth quad mesh with poles.

4. Rule-based editing

The exploration of the design space related to the singularities in the pattern is performed through topological modifications of the coarse quad mesh using grammar rules.

4.1 Topological spaces

Indeed, topological spaces are more general than geometrical spaces, which have a metric that allows thorough exploration using continuous-valued design parameters. Nevertheless, topological spaces can be explored using grammars to perform topological transformations in a rule-based design approach. For structural design, original shape grammars evolved into functional grammars (Mitchell (1991)) and then into structural grammars (Mueller (2014)) to include non-geometrical data related to structures. Specific to shell structures, Shea and Cagan (1997) introduce a grammar for the design of structural patterns for geodesic domes, which are triangulated meshes. The grammar required for exploration of singularities is specific for the editing of coarse quad meshes, optionally including pseudo-quads, where most vertices are singular.

4.2 Grammar rules

Infinite combinations exist to modify a set of quads into another. The practical grammar introduced in Figure 11 has been developed based on

practice and experience to achieve certain designs with certain goals. This grammar represents a set of tools for the designers and is meant to be further enriched.

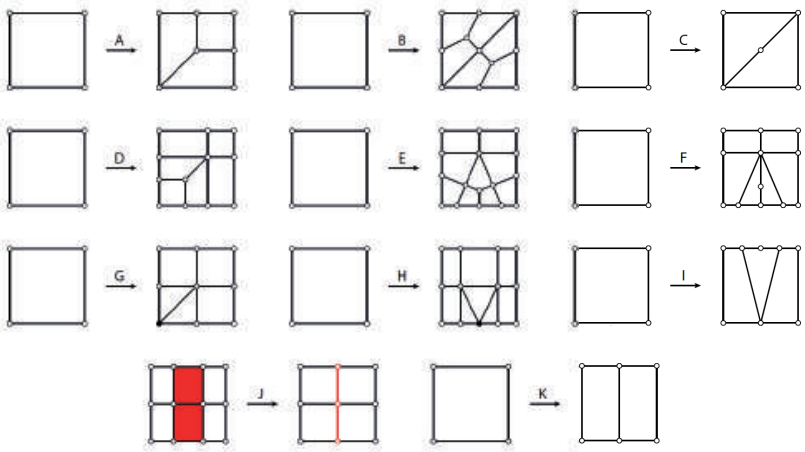


Figure 11: A practical grammar for rule-based editing of coarse quad meshes for singularities in patterns.

Rules A to F split a single quad into multiple quads by adding different sets of singularities, which change the edge flow. Rules G and H add pseudo-quads at a vertex or an edge of a face, respectively. Rule I inserts convex singularities (valency < 3) or concave singularities (valency > 3) at the boundary. Rule J is a coarsening operation which collapses a quad strip (in red), corresponding to one density parameter. Rule K subdivides a quad in two quads without introducing singularities as a utility rule.

After applying one or several of these local rules, a global propagation procedure ensures the validity of the coarse quad mesh. For instance, if a face is modified and one new vertex added on an edge, the adjacent quad face becomes a pentagon, which must be split into two quads, and so on.

4.3 Exploration

A set of designs with different singularities are edited in Figure 12. Starting from the automatically generated topology 0, fifteen other topologies are constructed. The edited coarse quad meshes are projected back onto the input surface. All these designs result from open exploration, without any algorithmic approach, and represent a small set of this unstructured design space.

Topologies 1 to 3 result from coarsening of topology 0 using rule J: topologies 2 and 3 appear as the two simplest ones among the sixteen, with topology 0 as a compromise between them, hence the relevance to start with the medial-axis singularities. Rule K is used for topology 1 to avoid collapsing the opening to only two vertices. Topologies 4 to 6 result from the applications of rules A to C to the corner quads of topology 0, respectively. Topologies 7 and 8 result from the applications of rule D to the corner quads of topology 3 with two different orientations. Topologies 9 and 10 result from the applications of rules E and F to the top and bottom quads of topology 2, respectively. Topology 11 results from the application of rule I to the top, bottom, left and right quads of topology 0. Topologies 12 and 13 result from the application of rule G to the corner quads of topology 0 to add poles, without and with coarsening using rule J, respectively. Topologies 14 and 15 result from the application of rules J and H to the outer boundary quads of topology 0 to add poles, without and with coarsening using rule J, respectively.

The third challenge is about how to visualise the design space, in spite of the lack of continuous-valued parameters structuring the design space.

topology	rule sequence
0	-
1	K-K-J-J
2	J-J-J-J
3	J
4	A-A-A-A
5	B-B-B-B
6	C-C-C-C
7	D-D-D-D
8	D-D-D-D
9	J-J-J-J-E-E
10	J-J-J-J-F-F
11	I-I-I-I
12	G-G-G-G
13	G-G-G-G-J-J-J-J
14	G-G-G-G-H-H-H-H
15	G-G-G-G-H-H-H-H-J-J-J-J-J-J-J-J

Table 1: Sequences of rules applied per topology starting from the automatically generated topology.

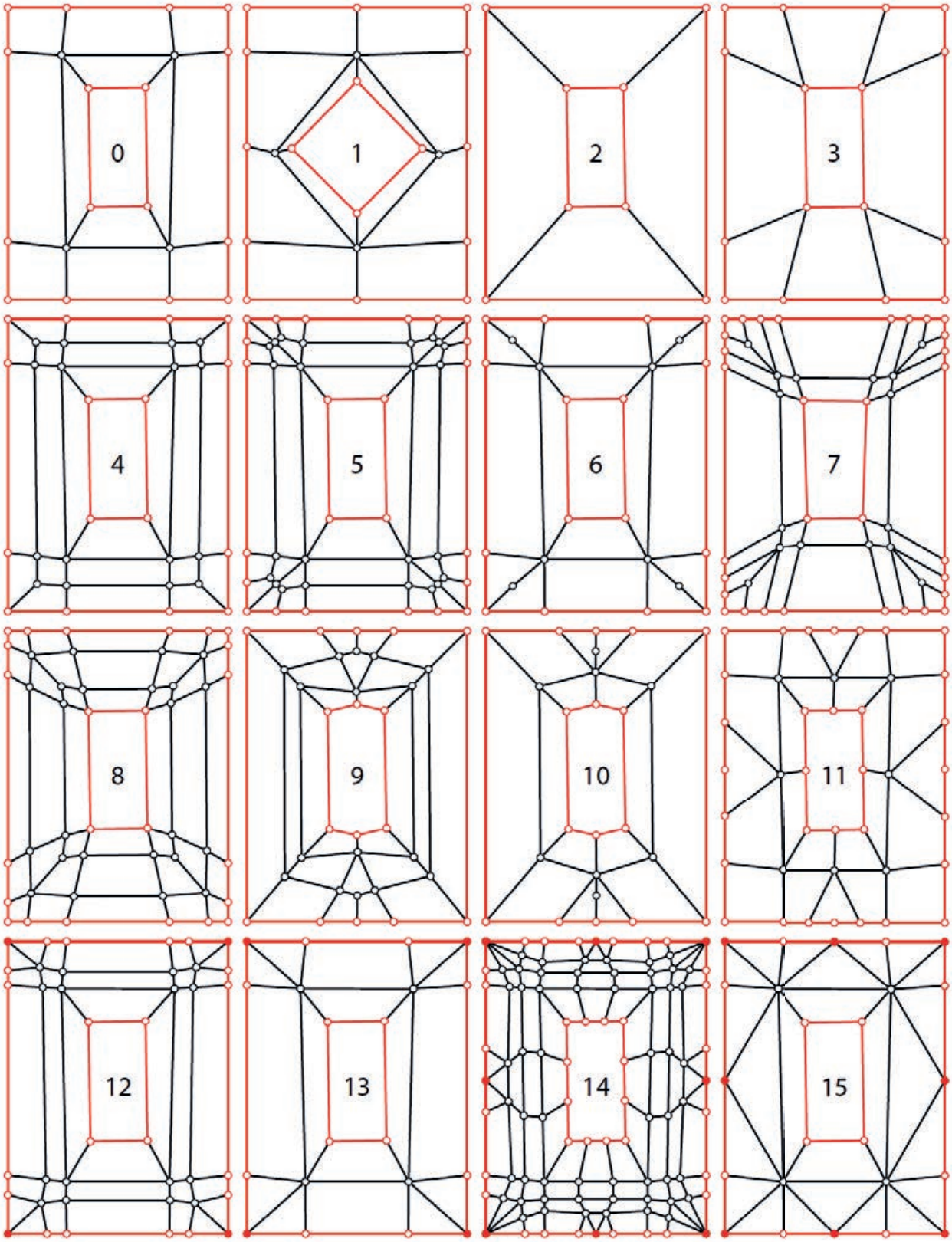


Figure 12: Rule-based exploration of the topological space of singularities in patterns.

5. Performance-driven visualisation

Instead of visualising the design space based on the topology of the pattern, using the performance of each topology for one or several criteria allows to shift the problem to the visualisation of a (potentially high-dimensional) metric space and inform the designer on the relative performance of the different topologies and understand their advantages and drawbacks.

5.1 Designs

All the topologies of the coarse quad meshes in Figure 12 are converted into smooth quad meshes as design patterns shown in Figure 13, which relaxed on the input surface. The density design space formed by the density parameters of the quad strips of the coarse quad mesh is a space in itself which can be subjected to optimisation. Here, the same target length is used to define the density parameters, though it results in differences on the edge length sum, particularly because of poles.

5.2 Performance metrics

The design of a steel and glass gridshell such as the British Museum must integrate a wide range of requirements, among which from fabrication and statics. A few classic of them are considered here.

5.2.1 Fabrication

A first criterion is the planarity of the panels, i.e. the face curvature must be minimised to avoid expensive bending processes. The metric is computed as:

$$C_{mesh} = \frac{1}{A_{mesh}} \sum A_{face} C_{face}, \quad (1)$$

with

$$C_{face} = \frac{dL}{(L_1 + L_2)/2}, \quad (2)$$

where L_1 and L_2 are the lengths of the diagonals of a quad face and dL the shortest distance between them.

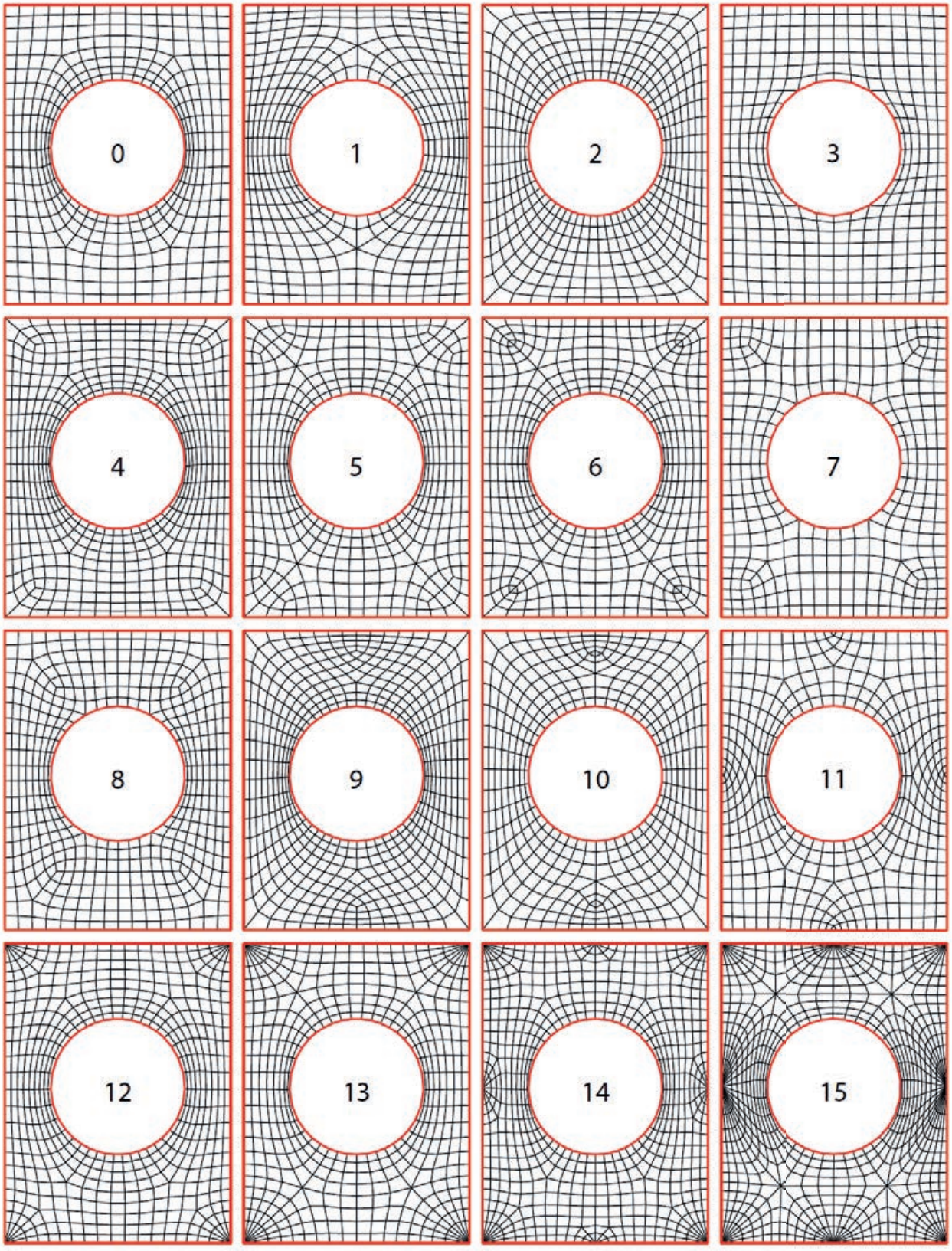


Figure 13: Gridshell pattern designs for the British Museum.

A second criterion is the skewness of the panels, i.e. the face skewness must be minimised to reduce material loss when cutting the panels. The metric is computed as:

$$S_{mesh} = \frac{1}{A_{mesh}} \sum A_{face} S_{face}, \tag{3}$$

with

$$S_{face} = \max\left(\frac{\theta_{max} - 90}{90}, \frac{90 - \theta_{min}}{90}\right), \tag{4}$$

where θ_{min} and θ_{max} are the minimal and maximal angles between two consecutive edges in the quad face.

A third criterion is the regularity of the edges, i.e. the variation of edge lengths must be minimised to avoid fabrication of too long or too short elements. The metric is computed as the standard deviation of the edge length in the mesh L_{mesh}.

The optimal parameterisation regarding these fabrication criteria follows the lines of principal curvature (Monge (1798), Liu et al. (2006)). However, the input surface and these lines evolve during geometrical exploration such as form finding and do not relate to structural efficiency.

5.2.2 Statics

Structural efficiency of a pattern depends mainly on the edge flow and its relevance for the considered statics system. Three support conditions shown in Figure 14 are taken into account to highlight this sensitivity: the structure is always vertically supported along its outer and inner boundaries, but thrust is applied either all along its boundaries, at 4 points only (the poles in design 14) or at 8 points only (the poles in design 12). The metrics are computed as the structural mass M_1, M_2 and M_3 after sizing optimisation for each support conditions, respectively.

Some hypothesis are made for comparative analysis.

The S355 steel profiles all have the same tubular cross section and are clamped at the nodes. The tube diameter is minimised for each design with a fixed wall thickness ($t = 40mm$). This choice of unique cross section is meant to favor designs with the most homogeneous stiffness distribution for the considered statics system. The considered load cases are the self-weight G, a vertical downward permanent loading $G' = 1kN/m^2$ and a vertical projected downward snow loading $S = 1kN/m^2$, either

on the whole structure (S_0) or on one fourth of the structure (S_1, S_2, S_3, S_4). The SLS and ULS load combinations are $1.0(G + G') + 1.0S_i$ and $1.35(G + G') + 1.5S_i$, respectively.

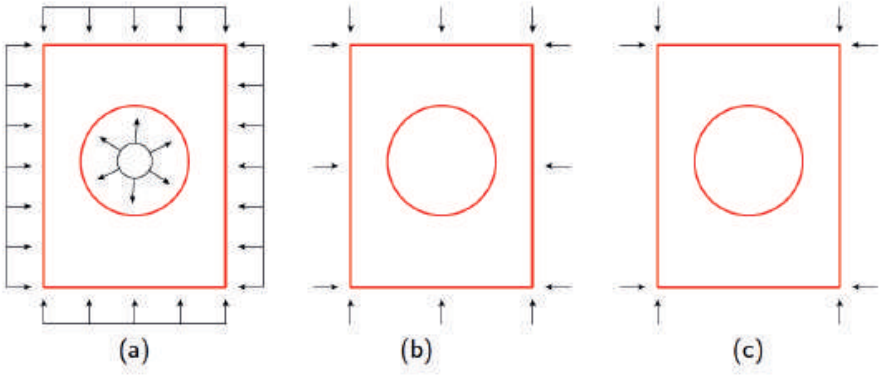


Figure 14: Considered thrust conditions: (**a**) full boundary thrust for M_1, (**b**) thrust on 8 points for M_2 and (**c**) 4 points only for M_3.

The structural analysis and sizing optimisation are performed in a framework well known by architects and engineers: Rhino3D and Grasshopper3D using the finite element software Karamba and the gradient-free optimisation library Goat. Constrained optimisation of the structural self-weight is expressed in Equation (5), similarly to Mesnil et al. (2017). Constraints apply on the maximum SLS deflection f_d (60 mm or 1/500[e] of the span), the maximum ULS cross-section utilisation ratio u_d (100 %) and the minimum ULS buckling load factor p_d (1). Eventually, deflection ends up being the governing constraint, with utilisation ratios staying below 80 % and first buckling load factors above 4 for all the designs.

$$\underset{f \le f_d, u \le u_d, -p \le -p_d}{\text{minimize}} \quad M_i \tag{5}$$

5.3 Self-organising maps

The raw results are displayed in Table 2 with the value of each metric for each design and its rank among all the sixteen designs: the lower the metric, the lower the rank and the more efficient the design regarding the metric. As expected, design 14 performs the best for the 8 thrust point support condition thanks to the poles, but more surprisingly, design 7 performs the best for the other support conditions. The mean and the

standard deviation are also computed to show the distribution per metric. The metrics M_2 and M_3 featuring different means but similar standard deviations show that the sensitivity to the 8 thrust point support condition is higher than to the one with 4.

The design space can be be visualised based on these results using self-organising maps. Self-organising maps are a neural network technique for dimensionality reduction of a N-dimensional space to a lower dimension. Using the implementation by Harding (2016), the initial 6-dimensional performance space is reduced to a 2-dimensional map, as shown in Figure 15.

	C_{mesh} [-]	S_{mesh} [-]	L_{mesh} [-]	M_1 [t]	M_2 [t]	M_3 [t]
0	0,012 (11)	0,091 (2)	0,27 (4)	714 (6)	1818 (15)	2532 (10)
1	0,015 (13)	0,170 (13)	0,30 (9)	864 (15)	1969 (16)	2798 (16)
2	0,009 (7)	0,169 (12)	0,32 (12)	639 (3)	1490 (6)	2448 (7)
3	0,016 (16)	0,082 (1)	0,15 (1)	784 (10)	1755 (14)	2413 (6)
4	0,009 (6)	0,150 (7)	0,27 (5)	758 (7)	1378 (3)	2240 (4)
5	0,007 (4)	0,123 (5)	0,32 (11)	806 (12)	1612 (11)	2534 (11)
6	0,008 (5)	0,167 (11)	0,27 (6)	778 (8)	1379 (4)	2299 (5)
7	0,016 (15)	0,099 (3)	0,20 (2)	610 (1)	1495 (7)	2106 (1)
8	0,016 (14)	0,107 (4)	0,22 (3)	629 (2)	1748 (13)	2552 (13)
9	0,010 (9)	0,188 (14)	0,35 (15)	780 (9)	1561 (9)	2731 (15)
10	0,011 (10)	0,200 (15)	0,33 (13)	656 (4)	1415 (5)	2520 (9)
11	0,014 (12)	0,158 (9)	0,29 (7)	712 (5)	1730 (12)	2544 (12)
12	0,005 (1)	0,146 (6)	0,30 (8)	804 (11)	1531 (8)	2450 (8)
13	0,006 (2)	0,160 (10)	0,33 (14)	816 (14)	1373 (2)	2189 (2)
14	0,007 (3)	0,150 (8)	0,31 (10)	813 (13)	1284 (1)	2225 (3)
15	0,010 (8)	0,310 (16)	0,47 (16)	968 (16)	1579 (10)	2597 (14)
mean	0,011	0,154	0,29	758	1570	2449
st. dev.	0,004	0,053	0,07	95	191	194

Table 2: Performance of each design per metric as value and rank.

The performance of each design i is displayed as a bar chart using dimensionless metrics X_i^*:

$$X_i^* = \frac{X_i}{\max_i X_i}. \tag{6}$$

The Voronoi diagram between the designs mark their influence on the underlying map of six-dimensional vectors.

The closer two designs, the more similar their respective performances. Thereof, performance clusters appear: designs 12, 13 and 14 perform well regarding face curvature but badly regarding edge length, because of the poles, and are structurally efficient for thrust at eight or four points; designs 3, 7 and 8 perform well regarding face skewness

Figure 15: Self-organising map for performance-driven visualisation of the topological design space of singularities.

and edge length but badly regarding face curvature, because of the singularities on the inner boundaries, and are structurally efficient for full thrust; designs 2, 9 and 10 strike a compromise between all the metrics.

This map helps understanding the consequences of a choice of singularities along the design process and illustrates the necessary trade-off between competing require ments which have to be balanced.

6. Conclusion

This paper introduces topology finding of structural patterns, complementary to form finding. Design and exploration of the topology of a pattern and its singularities is approached through automated generation and rule-based editing of coarse quad meshes. Design space exploration can be informed via performance-driven visualisation.

Implemented in a practical tool, these design strategies allow the architect and the engineer to explore efficiently the topological design space.

Future work should focus on algorithmic exploration of the design space. A challenge is to shift from performance-informed exploration and resorting to experience and heuristic rules (poles at the location of concentrated forces for structural efficiency, alignment with principal curvature directions for panel planarity…) to guided exploration to well-performing parts of the design space.

Another challenge is the development of automated generation schemes for more general topological shapes. Indeed, the NURBS surfaces here are all disc-homotopic, potentially with perforations, and do not need seams for planar mapping, on the contrary to closed shapes like spheres and non-null genus shapes like tori, which are also part of the topological family of orientable compact manifolds.

References

Blum, H. (1967). A transformation for extracting new descriptors of shape. In *Models for Perception of Speech and Visual Forms*, pp. 362–380. MIT Press.

Botsch, M., L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy (2010). *Polygon mesh processing*. CRC press.

Conway, J. H., H. Burgiel, and C. Goodman-Strauss (2016). *The symmetries of things*. CRC Press.

Harding, J. (2016). Dimensionality reduction for parametric design exploration. In S. Adriaenssens, F. Gramazio, M. Kohler, A. Menges, and M. Pauly (Eds.), *Advances in Architectural Geometry 2016*, pp. 274–87. vdf Hochschulverlag AG.

Harding, J., S. Joyce, P. Shepherd, and C. Williams (2012). Thinking topologically at early stage parametric design. In L. Hesselgren, S. Sharma, J. Wallner, N. Baldassini, P. Bompas, and J. Raynaud (Eds.), *Advances in Architectural Geometry 2012*, pp. 67–76. Springer.

Liu, Y., H. Pottmann, J. Wallner, Y.-L. Yang, and W. Wang (2006). Geometric modeling with conical meshes and developable surfaces. *ACM Transactions on Graphics 25* (3), 681–689.

Mele, T. V., A. Liew, T. Mendéz, M. Rippmann, et al. (2017). COMPAS: A frame- work for computational research in architecture and structures. http://compas- dev.github.io/compas/.

Mesnil, R., C. Douthe, and O. Baverel (2017). Non-standard patterns for gridshells: fabrication and structural optimization. *Journal of the International Association for Shell and Spatial Structures 58* (4), 277–286.

Mitchell, W. J. (1991). Functional grammars: An introduction. In *Proceedings of the Annual Conference of the ACADIA 1991*. CUMINCAD.

Monge, G. (1798). *Géométrie descriptive: leçons données aux écoles normales, l'an 3 de la république*. Baudouin, imprimeur du corps legislatif et de l'institut national.

Mueller, C. T. (2014). *Computational exploration of the structural design space*. Ph. D. thesis, Massachusetts Institute of Technology.

Rigby, D. (2003). Topmaker: A technique for automatic multi-block topology generation using the medial axis. In *ASME/JSME 2003 4th Joint Fluids Summer Engineering Conference*, pp. 1991–1997.

Rippmann, M. and P. Block (2013). Funicular shell design exploration. In *Proceedings of the Annual Conference of the ACADIA 2013*.

Schiftner, A. and J. Balzer (2010). Statics-sensitive layout of planar quadrilateral meshes. In C. Ceccato, P. Hesselgren, M. Pauly, H. Pottmann, and J. Wallner (Eds.), *Advances in Architectural Geometry 2010*, pp. 221–236. Springer.

Shea, K. and J. Cagan (1997). Innovative dome design: Applying geodesic patterns with shape annealing. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 11* (5), 379–394.

Shepherd, P. and W. Pearson (2013). Topology optimisation of algorithmically generated space frames. In *Proceedings of the International Association for Shell and Spatial Structures Annual Symposium.*

Tang, C., X. Sun, A. Gomes, J. Wallner, and H. Pottmann (2014). Form-finding with polyhedral meshes made simple. *ACM Transactions on Graphics 33* (4), 70:1–70:9.

Williams, C. J. (2001). The analytic and numerical definition of the geometry of the british museum great court roof. In *Mathematics & Design*, pp. 434–440.

Using non-linear logic for construction of self-stabilizing hammock structure

Shohei Furuichi, Shuntaro Nozawa, Machiko Asahara, Priya Murugeswaran, Yuqing Shi, Ziyi Wu, Yusuke Obuchi, Jun Sato

Shohei Furuichi
furuichi@arch1.t.u-tokyo.ac.jp

Priya Murugeswaran
priyamu90@gmail.com

Yuqing Shi
useko515@qq.com

Ziyi Wu
18817369198@163.com

Yusuke Obuchi
obuchi@arch.t.u-tokyo.ac.jp
Department of Architecture, Graduate School of Engineering, the University of Tokyo, Japan

Shuntaro Nozawa
nozawa@global.c.u-tokyo.ac.jp
Graduate School of Arts and Sciences, the University of Tokyo, Japan

Machiko Asahara
kyuzizim@gmail.com

Jun Sato
junsato@edu.k.u-tokyo.ac.jp
Graduate School of Frontier Sciences, the University of Tokyo, Japan

Keywords:
Non-liner structure, computational design, construction method, algorithm design, inverse analysis, structural optimization

Abstract

This paper explores our project of constructing a pavilion applying the self-stabilizing hammock structure on site. This structure was proposed to utilize its non-linear logic through iterative method of fabrication. It was as deployed on site by a base geometry of thin, steel fixed-base columns, connected loosely by individual meshes, which are then stiffened by loading coconut fibers upon the meshes. We developed an algorithm to compute the amount of fibers to be deposited on meshes at each loading sequence, aiming to make all initially inclined columns upright. First, this paper outlines the non-linear feature of the hammock structure through assessment of its buckling length, and shows the loading algorithm designed based on an inverse analysis to define a load-displacement relationship, and the result of iterative loading simulation. Second, the design and entire installation procedures of the hammock-structure pavilion, and the implementation of the developed algorithm are described. It was realized through daily feedback processes, where column coordinates obtained by using a set of motion captures were input into CAD and structural analysis software, to update and reconfigure the loading sequence for the next day. Following this, we illustrate a natural frequency test conducted during the dismantling of the pavilion. Finally, we evaluate recorded column coordinate data, as well as actual weights of loaded meshes measured in dismantling. This allows us to pinpoint potential improvements to the loading algorithm, particularly the modelling of peripheral meshes and columns.

1. Introduction

The hammock structure was proposed to utilize its potential of self-stability as well as non-linear nature per se as a logic for its fabrication. The main components of its frame are thin, cantilever columns, and meshes that at first, loosely connect the heads. The general idea of its self-stabilization is that, when tension is generated after applying load onto the meshes, the columns are stiffened against buckling. This structure was scaled up and materialized in our experimental project of the installation of a full-scale pavilion, which consisted of a number of fine steel columns, coir coconut rope meshes, and coconut fibers used as load (**Fig. 1**). These columns in their initial position were inclined; some were cambered to prevent undesired deflections, and the others were inclined

randomly at a slight angle. For this installation a loading algorithm was designed to calculate loading target meshes and the requisite amount of fibers to be deposited at each loading sequence. Each loading sequence took place each day, and the whole structure was scanned every night, in order to structurally optimize the load distribution. We intended that at the end of the construction process all columns become vertical.

This paper explores the design and operations of the loading algorithm based upon the fundamental concept of the hammock structure, and evaluates how it functioned in the pavilion installation. An underlying mechanism for this structure is the stiffening effect of the external load. While Chen and Kawaguchi (2012) investigate the static properties of the hammock structure in the context of a negatively pressurized pneumatic structure (see also Chen (2014)), we intend to utilize this effect directly for supporting the weight of the roof, and apply its non-linear features to construction method. We also investigate the dynamic behavior under lateral loads.

Our project is an attempt at linking the essence of geometry exploration by means of evolutionary algorithms and structural optimization tools in the initial stage of design (e.g. structural engineer Mutsuro Sasaki's works characterized by repetitive non-linear analysis procedures; see Januszkiewicz and Banachowicz 2017), with an area of research on cyber-physical design (e.g. Doerfler et al., 2017; Lopez et al., 2016; Yoshida et al., 2015). The proposed self-stabilization method predetermines a structural form, like a hammock and is conditioned by a reciprocal relationship with the structural characteristics of the form. This mutuality of structural properties, geometry, and manners of building is seen in the quest for integration of structural, material and geometric computing that presumes novel, digital fabrication techniques (e.g. Deuss et al. 2014; Frick et al. 2016; Veenendaal et al. 2014).

First, this paper examines the logic and characteristics of the hammock structure, assessing some structure models by buckling length. It is followed by exploration of inverse analysis on the relationship between load and column displacement for designing the loading algorithm, and demonstration of how iterative loading processes gradually make columns straight in simulation. Second, our installation of a pavilion with the hammock structure is outlined. We show the way that the developed algorithm was implemented by making full use of motion captures, CAD, structural analysis software, and our unique handheld bazooka projec-

tile tool incorporated with auditory guidance for shooting and loading fibers. A natural frequency test held in demolition of the pavilion is also illustrated. Finally, this paper revisits the algorithm by evaluating the data of column inclination and the actual amount of load observed during and after the installation.

The hammock structure as a fabrication technique has potential to be developed further as a fast and easy construction system of a temporary structure. This is a method whereby any temporary support during the construction is omitted. Unlike conventional tensile structures, including membranes and cable nets, our proposal does not require any strict control of tension.

Figure 1: Birds-eye view of the overall geometry of the hammock-structure pavilion exhibited between November 28th and December 10th, 2017.

2. Structural and loading principles

2.1 Column Stiffness

The hammock structure relies on the idea of stiffening thin columns fixed at their base against buckling by applying load. The column heads are initially loosely connected by meshes which are considered to be hinge joints when modelled. This initial condition without load (Model (D-0) in **Fig. 2**) is equivalent to the model of a cantilevering column (Model (A) in **Fig. 2**). The columns of the hammock structure can bear a larger load than independent columns because of the stiffening effect of the tension in the meshes. When they are loaded, the columns are likely to sway horizontally. This leads to the change in sag and tension in adjoining meshes which causes a restoring force to act and move the columns back to their original positions.

The outcome of our two-dimensional, non-linear analysis allows us to confirm our hypothesis that the hammock structure becomes self-stabilized by loading. The co-rotation method as shown in Krenk (2009), was employed to assess the geometrically non-linear behavior of a column when loading, enabling us to understand the relationship between axial force and vertical displacement. As Figure 3 shows, buckling is observed when the axial force reaches approximately 100 N, which is equivalent to a buckling length, L_k of 0.79L, where L is the actual column length (see **Fig. 2**). This simulation outcome is in agreement with Chen's study (2014), and is close to the buckling length of a column with a hinge-fixed head and a fixed base, $L_k = 0.7L$ (Model (B) in **Fig. 2**). Buckling strength is inversely proportional to the square of buckling length; therefore, the loading-self-stabilizing process makes the hammock structure approximately 6 times stronger than a cantilevering column, $L_k = 2.0L$ (Model (D-0)).

According to our non-linear analysis, supplementary membrane bracing enables the hammock structure to be further strengthened. Our unique idea is to add an auxiliary, lower mesh linking the upper middle parts of columns with the bottom faces of meshes (**Fig. 4**). As well as the main upper meshes, it is initially placed loosely, and its function as a brace is achieved by loading it (Model (E) in **Fig. 2**). In our analysis, the hammock structure column braced in this way exhibits approximately 170 N buckling strength (**Fig. 3**), which is equivalent to $L_k = 0.61L$. It indicates that the strength performance of this braced model is comparable to

that of a fixed-ended column, whose buckling length is 0.5L (Model (C) in **Fig. 2**), and approximately 11 times to that of a cantilever column (Model (A)).

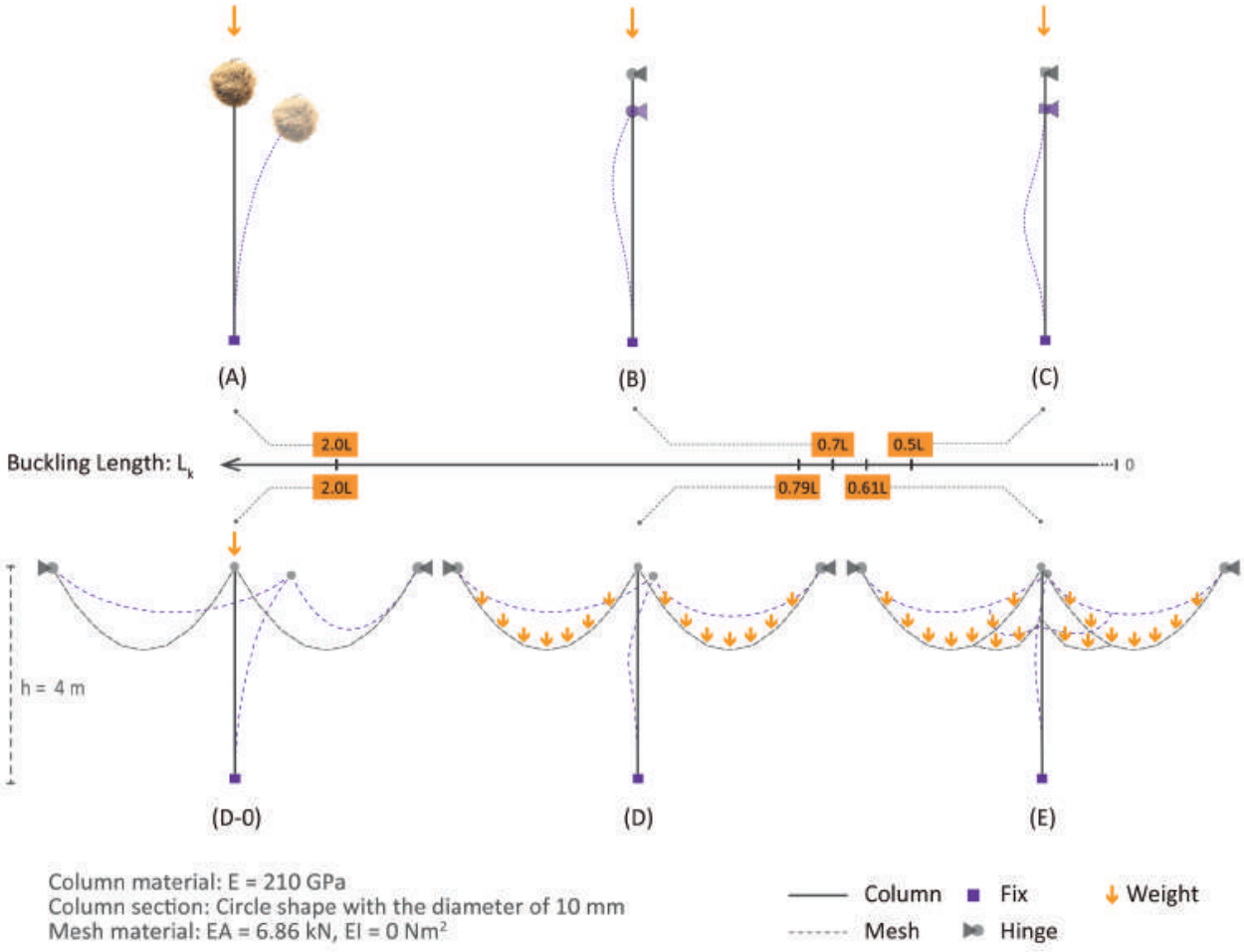
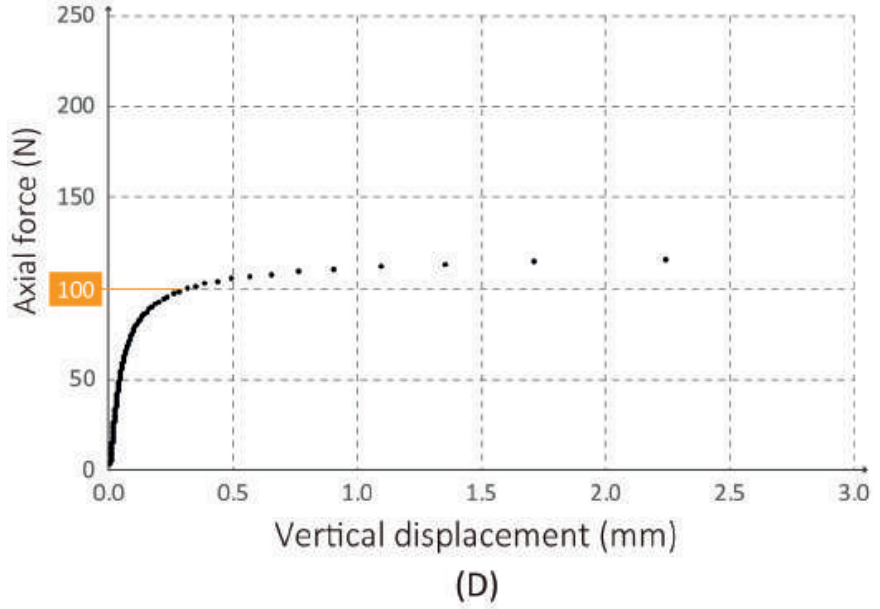


Figure 2: Buckling length (L_k) of the hammock structure. The bottom left Model (D-0) represents a hammock structure column without load, which is equivalent to a cantilevering column in Model (A). Model (D) represents Model (D-0)'s status after loading, which is similar to a column with a hinge-fixed head and a fixed base like Model (B). In order to strengthen the hammock structure, as proposed in Model (E), supplementary membrane bracing is added as an auxiliary, lower mesh linking the upper middle parts of columns with the bottom faces of meshes. This strengthening performance is comparable to that of a fixed-fixed column as Model (C).



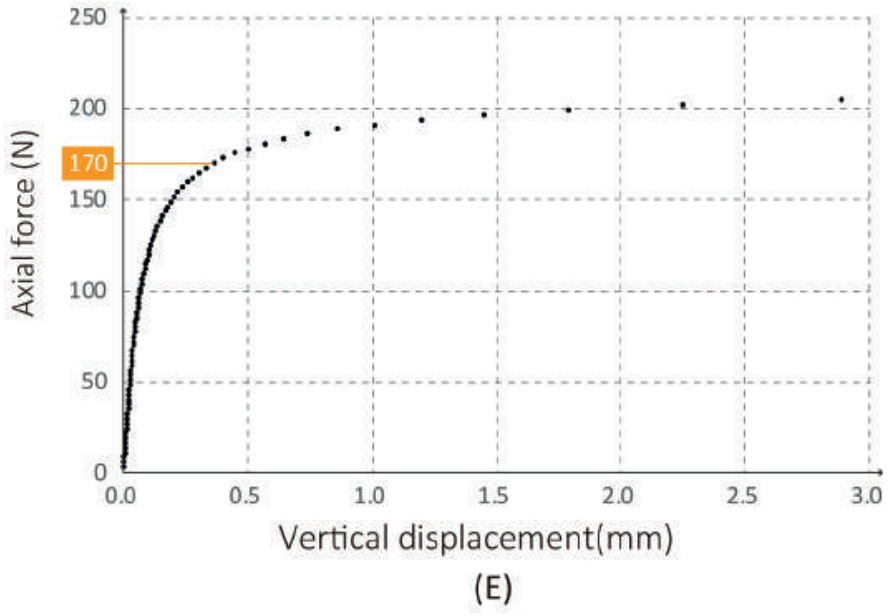


Figure 3: Relationship of vertical displacement and axial force of hammock structure, showing force-displacement relationship in Model (D), and Model (E) (right). According to our analysis, loaded hammock structure in Model (D) exhibits a buckling strength of approximately 100 N, while with bracing in Model (E), its buckling strength increased to 170 N (right).

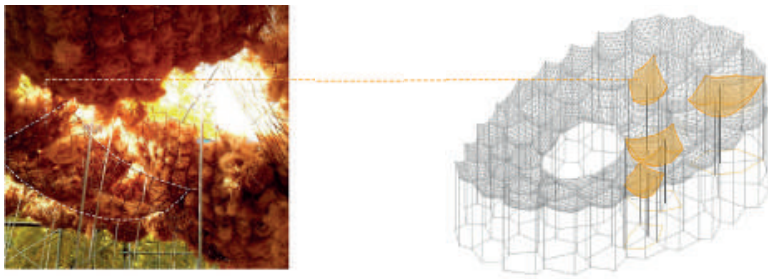


Figure 4: Supplementary membrane braces. These were installed as an auxiliary, lower mesh connecting the upper middle parts columns with the bottom faces of meshes (left) in the four positions indicated in the right image.

2.2 Inverse analysis and loading simulation

The loading algorithm that was implemented in the pavilion construction was designed presuming that self-stabilizing property of the hammock structure can be effectively achieved by iterations of loading. We presumed that it enables us to maintain the balance of distributed load through the loading process. The developed algorithm was a product of the calculation of the optimal load distribution to attain a desirable column deformation. For this calculation, two three-dimensional models (Model (A_0) and Model (T_0) in **Fig. 5**) were created in line with the specifications of columns and mesh used for the installation, and a design principle for sectioning its geometry: the Voronoi diagram. The behavior of a mesh in loading was replaced with that of a pseudo-spring of equivalent vertical stiffness as shown in Model (T_0). This way of modelling enabled us to assess the target weight necessary to make the connected columns vertical as this load is equivalent to the compression force in the spring.

The inverse analysis is performed in two steps. First, Model (A_0) in Figure 5 is used to determine the Young's modulus of the pseudo-spring. The model consists of four columns connected with the same number of strings that represent a mesh and meet at the Voronoi cell center. On the condition that the meshes do not extend under vertical loading at the intersection of the strings, the stiffness of the pseudo-spring is determined from the vertical displacement of the intersection point. The Young's modulus of the spring (E) can be calculated by a formula relating vertical force (V) and deformation (Δ), with the length (l) and cross-sectional area (A) as follows:

$$E = Nl/A\Delta \qquad (1)$$

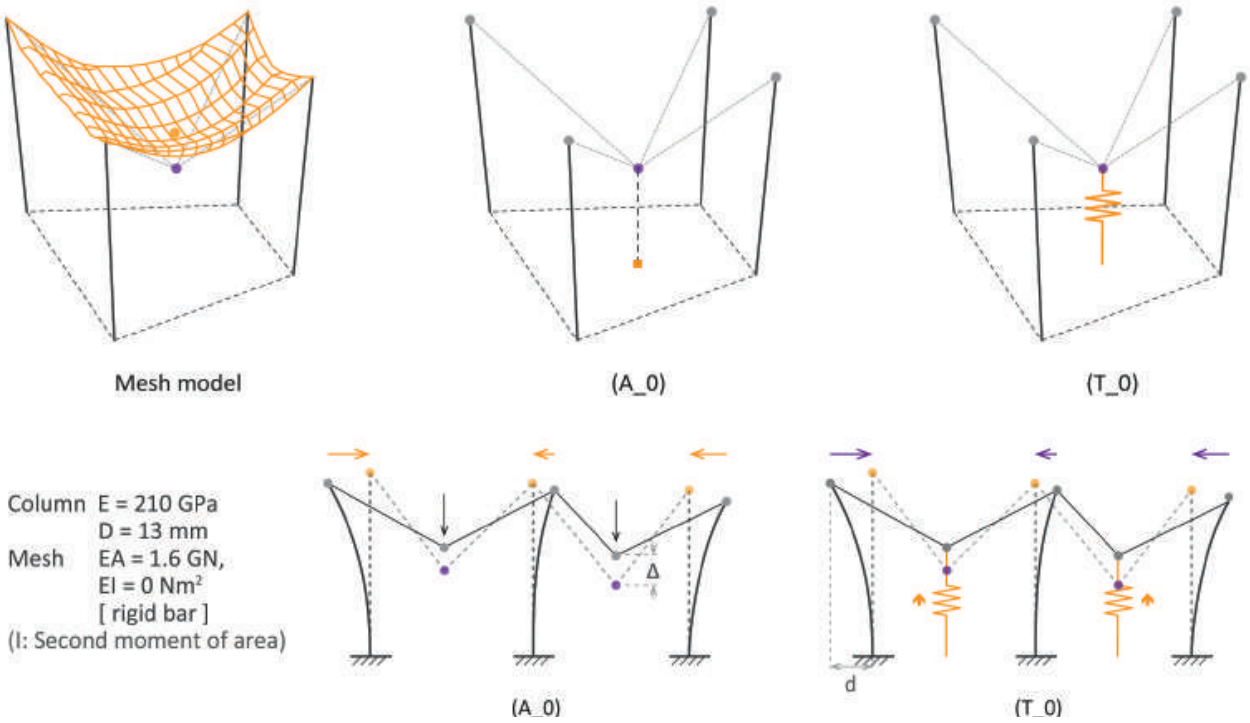


Figure 5: Voronoi section mesh models for loading simulation created to analyze and define the relationship between a load and lateral displacement of a column. In the mesh model shown top left, there is a difference between the actual mesh center (orange) and the mesh force simulation center (purple). This difference exists because in the mesh forces simulation, the direction lines are drawn as tangentially to the mesh corner surface, and converge at a point lower than actual mesh center. Model (A_0) is used to determine the Young's modulus of the pseudo-spring in Model (T_0), which replaced the loading with the spring. Section mesh models of Model (A_0) and Mode (T_0) in inverse analysis, schematizing the relationship of two Voronoi cell structure with a triad of a load, mesh, and columns.

Second, horizontal displacements are applied to the heads of the columns in Model (T_0) as shown in Figure 5, to make all columns upright. Then, compression forces corresponding to loads required to vertically straighten the columns are introduced into the pseudo-springs. It is considered that due to the non-linearity of the hammock structure, the columns are unlikely to be vertical by applying the required loads just once. Therefore, the loading process needs to be iterated several times. Note that the Young's modulus of the pseudo-springs is assumed to be the same values through iteration.

The models we used for this analysis are made up of two 3 m high steel columns with a diameter of 9 mm, and five pseudo-springs corresponding to loading points as well as the Voronoi centers of the meshes

(Model (A_0) and Model (T_0) in **Fig. 6**). The bottoms of the columns are fixed, and the column heads, springs and meshes are modelled as joints hinged in all directions. Prior to the simulation, the Young's modulus of each spring was calculated by applying a 10 N load vertically downward on each Voronoi center in Model (A_0). As a result of the vertical deformation the Young's modulus was calculated for each spring as shown in Table 1, and for simplicity the five obtained values of the Young's modulus were approximated to 2.5, 10.0 and 24.0 kN/cm².

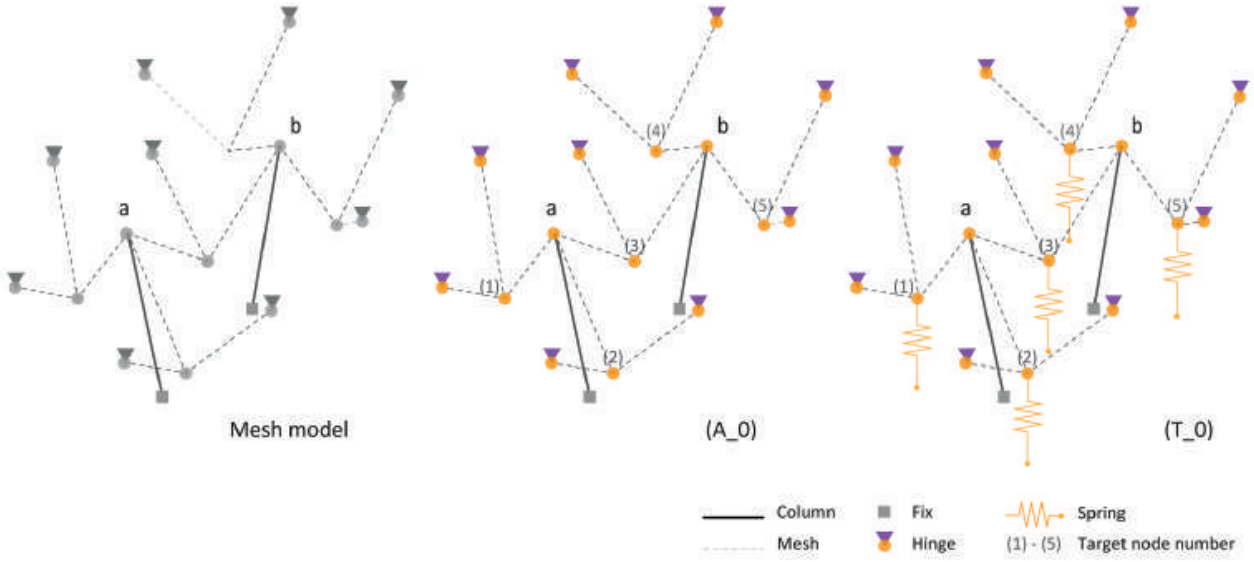


Figure 6: Prototype loading algorithm models with two 3 m high, 9 mm-diameter columns and five pseudo-springs representing the load on the Voronoi centers of the meshes (left). Model (A_0) is used to determine the Young's modulus of the pseudo springs in Model (T_0).

Shooting point	Deformation Δ [cm]	Spring length l [cm]	Sectional area A [cm²]	Young's modulus E [kN/ cm²]	
				Estimated	Approximate
(1)	0.89	170	0.79	2.35	2.5
(2)	0.09	170	0.79	23.32	24.0
(3)	0.81	170	0.79	2.65	2.5
(4)	0.22	170	0.79	9.70	10.0
(5)	0.75	170	0.79	2.84	2.5

Table 1: Young's modulus of pseudo-springs.

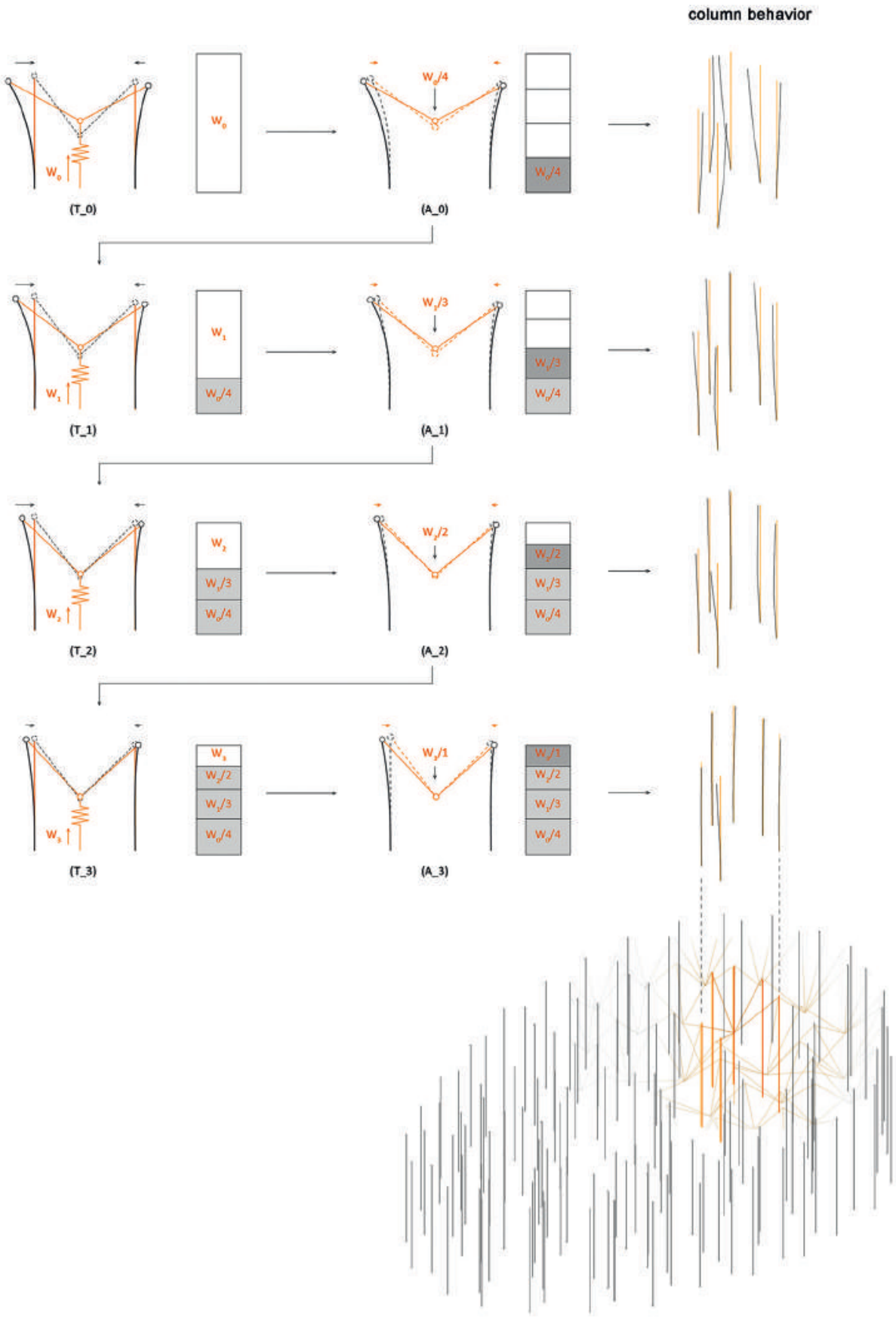


Figure 7: Iterative loading simulation of actual workflow during the construction of pavilion by observing a part of the whole geometry that includes six columns and one mesh.

Finally, we conducted the simulation to test the prototype of the loading algorithm, which empirically divided a loading process into four steps. As Figure 7 describes, one fourth of a load (Wo) estimated from Model (T_0) was input into Model (A_0), followed by iterated calculations of second, third and fourth loads (W_1, W_2, W_3) and applications of $W_1/3$, $W_2/2$ and W_3 to its loading point. A load given to the loading point in each step is shown in Table 2. Figure 8 reveals the behaviors of the two columns through this manner of loading, demonstrating that the gaps with their upright positions was gradually reduced step by step. The tested algorithm was ultimately put in practice in the pavilion construction, whose loading procedures, contrary to those of the simulation, were defined by the maximum amount of load per day (50 kg), rather than a predetermined number of loading iterations. In fact, its underlying principle, iterative loading, was also favorable for its implementation; an interval between two loading steps as such was suited to the procedures for scanning column inclinations embedded in the daily feedback routine.

Shooting point	W1 [N]		W2 [N]		W3 [N]		W4 [N]		Total [N]
		1/4		1/3		1/2		1/1	
(1)	-77.7	0.0	-31.4	0.0	-26.6	0.0	-17.3	0.0	0.0
(2)	629.7	157.4	-4.4	0.0	35.3	17.7	16.9	16.9	192.0
(3)	241.8	60.5	97.6	32.5	58.1	29.0	29.2	29.2	151.3
(4)	142.2	35.5	99.4	33.1	68.1	34.0	38.8	38.8	141.5
(5)	-62.1	0.0	-15.9	0.0	-27.2	0.0	-5.8	0.0	0.0

Table 2: Loads in iterative loading.

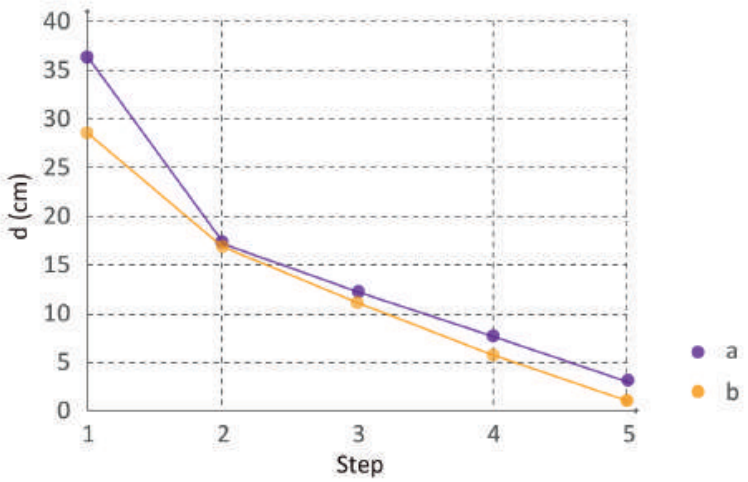


Figure 8: Horizontal gap between column base and head in each step, demonstrating the behaviors of the two columns during step-by-step interactive loading.

3. Case study

3.1 Approach to Construction and Design

The developed loading algorithm was experimentally put into practice in the installation of a full-scale pavilion with the hammock structure. This attempt was intertwined with our enthusiasm for suggesting and demonstrating a novel method of computer-aided, human-centric fabrication. It presumes the integration of design and construction procedures through the application of up-to-date sensors, human capacity augmentation technologies, various software in design and simulation, and so on. For this installation we invented a bazooka projectile tool incorporated with an auditory guidance system, which guides a worker using three-dimen-

Figure 9: Acoustically guided aiming and shooting process of the pavilion installation using the bazooka projectile tool located in one of the shooting points.

sional sound sources, corresponding to target positions for depositing fibers (**Fig. 9**). The positions of the column heads were detected by daily column inclination scanning, and taken into account at the next step of shooting through structural optimization. This iterative loading that underlies the logic of algorithm design fits with the feedback process of fabrication. The implementation of the algorithm was fully embedded in the use of the auditory guidance system, allowing (re-)calculation of target loading positions and weights to eventually make all columns straights.

The hammock-structure pavilion was built from November 6th to 26th, 2017. The platform for this temporary architecture was 12 m by 12 m, and 72.87 m^2 was covered with the hammock. The hammock's plan was elliptical with a hole, reflecting not only aesthetic consideration, but also the relationship with shooting points as well as tool performance (**Fig. 10**). Coconut fibers were shot from three openings in scaffolds around the site and a workstation located in the hole, and the maximum shooting range of 7 m covers the whole pavilion area. The major axis was designed to run from north to south, because we expected that a seasonal north wind served as a supplementary projectile force to transport the fibers.

The plan was partitioned into 48 Voronoi cells, and 128 thin, steel columns were placed at the vertices of these cells. There were three column sizes: 13, 16 and 19 mm diameter steel rods, and their heights ranged from 2.6 to 4.0 m. Their sizes were calculated to be as thin as possible based on the total target load, which was around 500 kg. The 19 mm columns were set at the periphery, and the 16 mm columns erected on the inner edge. Prior to installation, these columns were cambered, between 302 and 394 mm, so to ensure the columns finally straighten. This allowed the pavilion to be free from the use of any supplementary tension cables, which would prevent them from bending inward after loading. The cambers were set to be equal to estimated displacements. 48 coir rope mesh sheets initially hung loosely from the heads of all columns, with varying surface areas between 0.80 m^2 and 2.87 m^2.

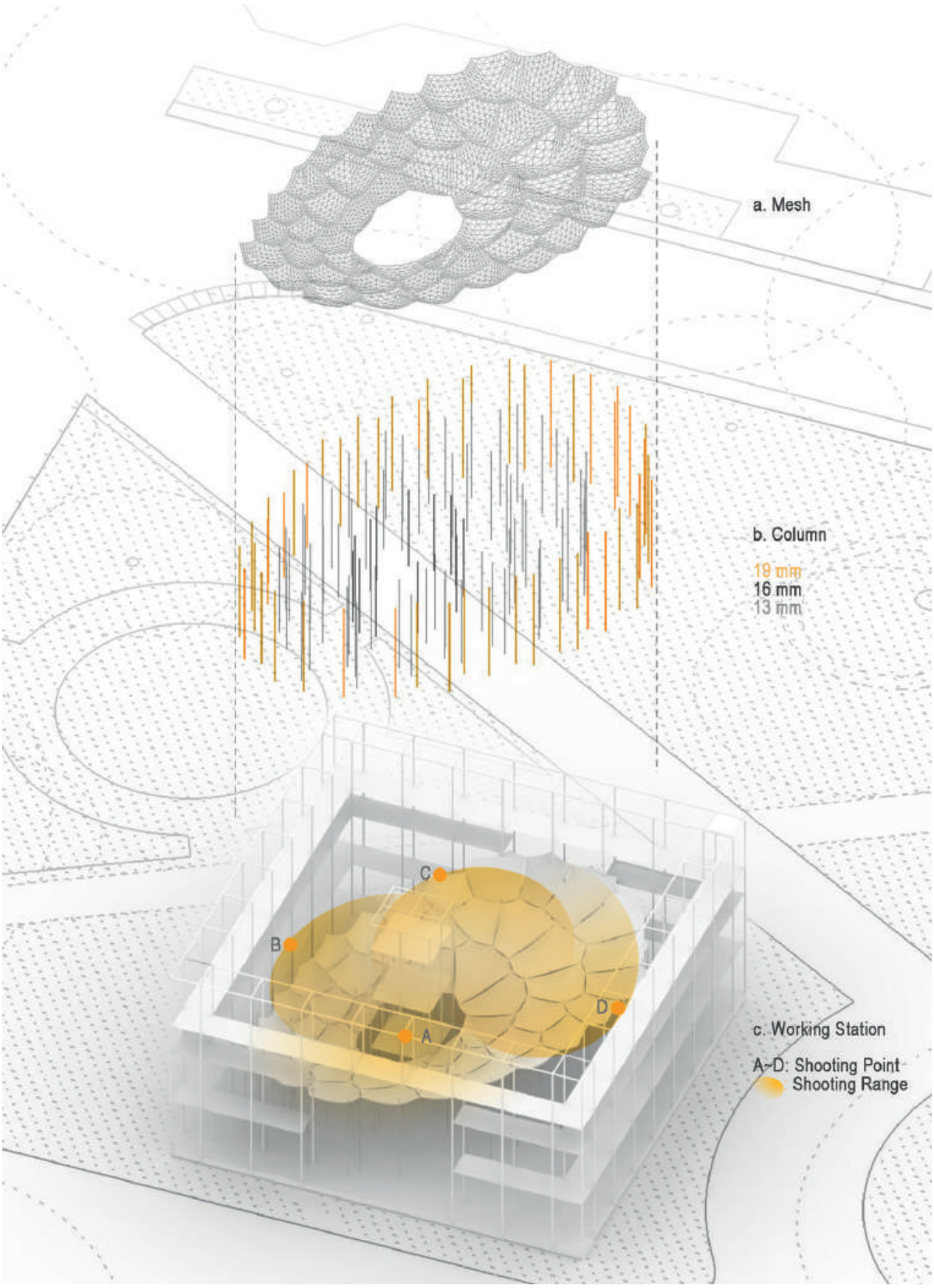


Figure 10: Key compositions of the pavilion, including Voronoi-patterned meshes, diameter distribution of steel columns, and shooting range and shooting points based on a central working station and scaffolds.

3.2 Loading algorithm implementation

The initial load of 1kg fibers was equally distributed onto each mesh by hand. This was intended not only to prevent large deformation of the meshes, but it was also necessary to make a fiber screen on the surface of the meshes to prevent shot fibers from passing through any holes. Following this, we started loading using the developed sound guidance system as well as loading algorithm. The application of the sound guidance system involved the use of a set of motion captures, namely the HTC Vive, which include two sensors to track and control, and two base stations to follow motions and sensor coordinates in real time. The tracker was put on the top of a worker's helmet, and the controller was installed on the body of the bazooka. The base stations were placed around a shooting point of the scaffolds and workstation. The closest point to a target mesh was selected for shooting. Before loading, a mass of compressed coconut fibers was processed and split into a number of 4-6-g small pieces. Once a worker detected the source of continuous note, he/she was expected to aim the projectile tool at it, and continue to shoot fibers until required loading was complete.

Every night the scanning of the as-built conditions of the hammock structure took place and the distribution of the fibers for the next day was calculated through structural optimization of the verticality of the columns (**Fig. 11**). First, the inclinations of all columns were recorded utilizing the HTC Vive controller (**Fig. 12**). The coordinates obtained were of the bottom and two points at heights of 1 and 2 m. The coordinate data was used to redraw the column and mesh lines automatically by Grasshopper, a plug-in of Rhinoceros, allowing us to monitor deformations of the columns (**Fig. 13**). Second, we ran the developed loading algorithm, in order to simulate how to straighten the columns. Finally, calculation of the amounts of fibers to be loaded onto target meshes for the next day was made with our original structural analysis software. This series of daily feedback was initially conducted on November 17th, and 6 consecutive days between 20th and 25th.

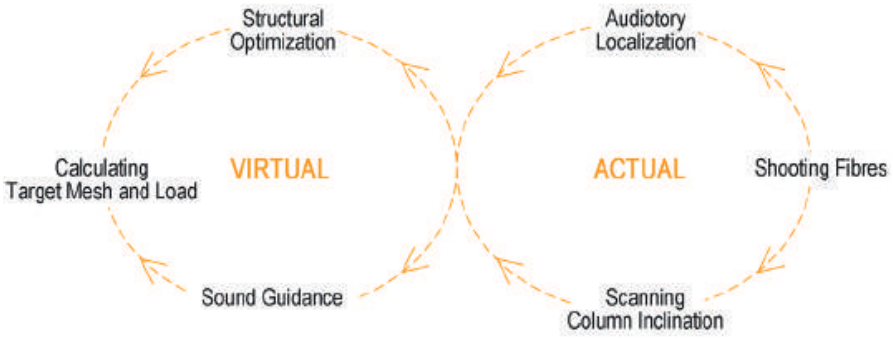


Figure 11: Loading process feedback loop, presenting the procedures and relationship between actual construction process and virtual simulation and optimization.

Figure 12: Daily column inclination scanning and tracking. The coordinate data of a Vive controller can be input into Grasshopper by means of its plug-in (left). Every night the inclinations of all columns were measured and tracked by scanning the coordinates of their bottoms, and two points at heights of 1 and 2 m. As shown in the right image, a 3D-printed, curved plate was installed in the controller, to make scanning more precise.

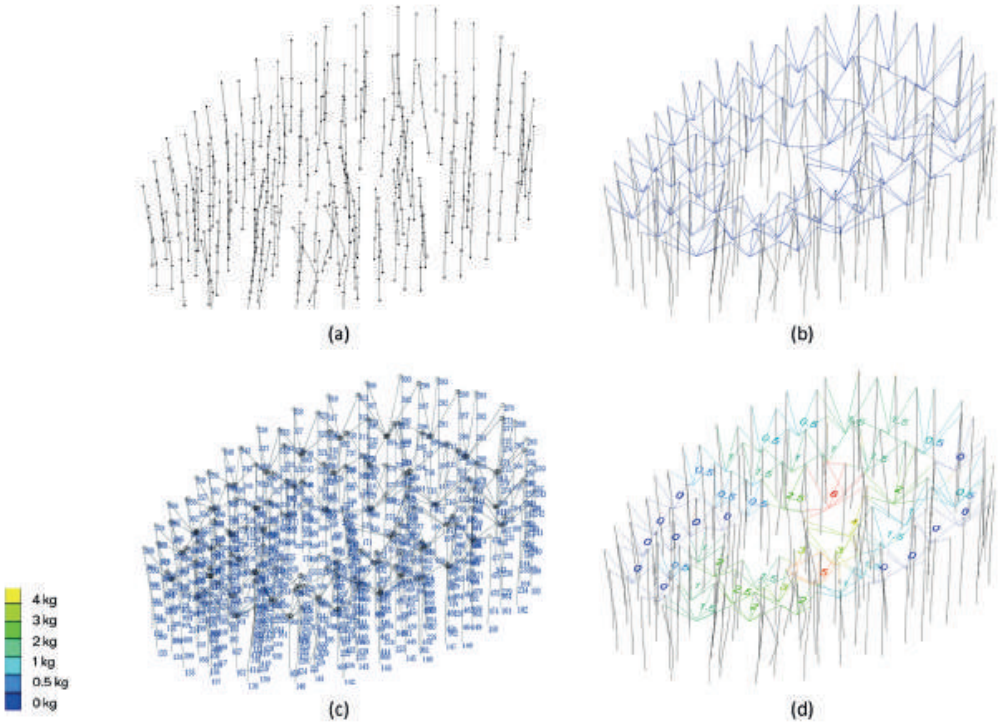


Figure 13: Structural and loading sequence optimization. (a) shows the Grasshopper generated columns based on inclination scanning. It allows mesh lines to be recreated as shown in (b). (c) displays estimated spring forces, which are used to calculate target loads and loading points for the next day (see subfigure (d)).

3.3 Natural frequency test

Dismantling of the pavilion was also part of our research. After a two-week exhibition we carried out a natural frequency test, where we physically shook it several times with some collaborators (**Fig. 14**). Seven people stood evenly around the pavilion and shook together in unison. We kept shaking the pavilion until it reached a steady state and vibrated in the natural frequency in the first mode. As damping was relatively large, we continued to keep shaking while measuring the length of time which it vibrated ten times. This test was conducted three times, and the results were 29.97, 29.53 and 29.90 sec. The time for one vibration can be regarded as a natural frequency and was 2.97 sec on average. Note that the amplitude of vibration every part of the pavilion showed was apparently different. It is thought that this dispersion was ascribed to the variety of the columns in specifications (e.g. height and diameter). The columns around 3.4 m in height experienced large amplitude; therefore, we assumed this size of column for further analysis (see Chapter 4).

Following this, fibers loaded on the meshes were disassembled

and measured by scales. Due to the lack of equipment to quantify the amount of fibers in shooting, the actual weights on the meshes were unknown during the installation. As well as the observation of the natural frequencies, this direct measurement of the weights enables us to assess the structural behavior under horizontal seismic force.

(a)

(b)

(c)

Figure 14: Natural frequency test. The pavilion was manually shaken three times to measure the length of time which it vibrated ten times.

4. Results

4.1 Iterative loading and column inclination

Figure 15 demonstrates the relationship between the total amount of load per mesh that the loading algorithm suggested through the shooting process, and the actual weights directly measured in dismantling of the pavilion, indicating that the algorithm worked properly. Though some construction errors were likely to exist, most points are plotted along a 45-degree line. The dispersion of the points could be minimized if we knew the weight of the loaded meshes each day.

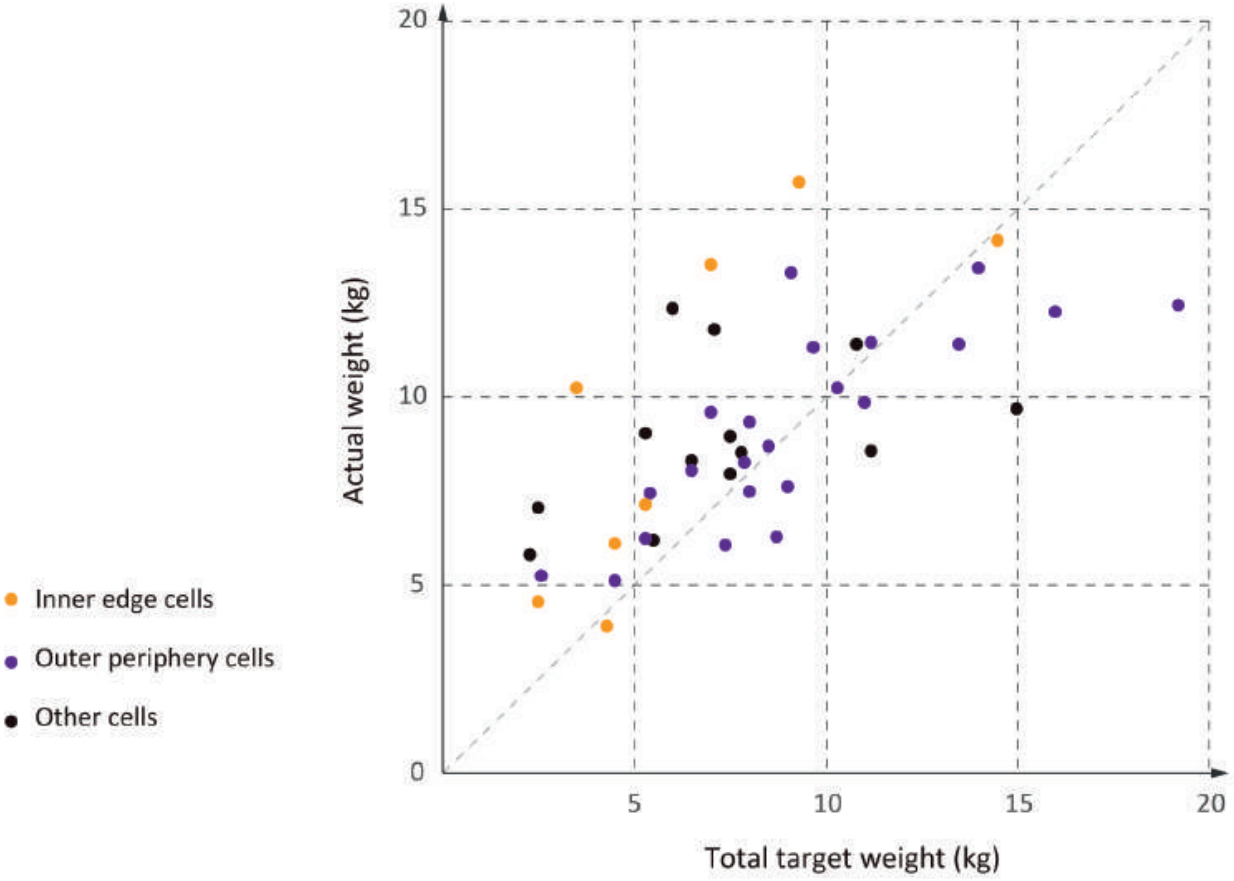


Figure 15: Relationship between target of load suggested by algorithm and actual weight of fibers, which was measured during the dismantling of the pavilion. Most points demonstrate the effectiveness of the loading algorithm in construction.

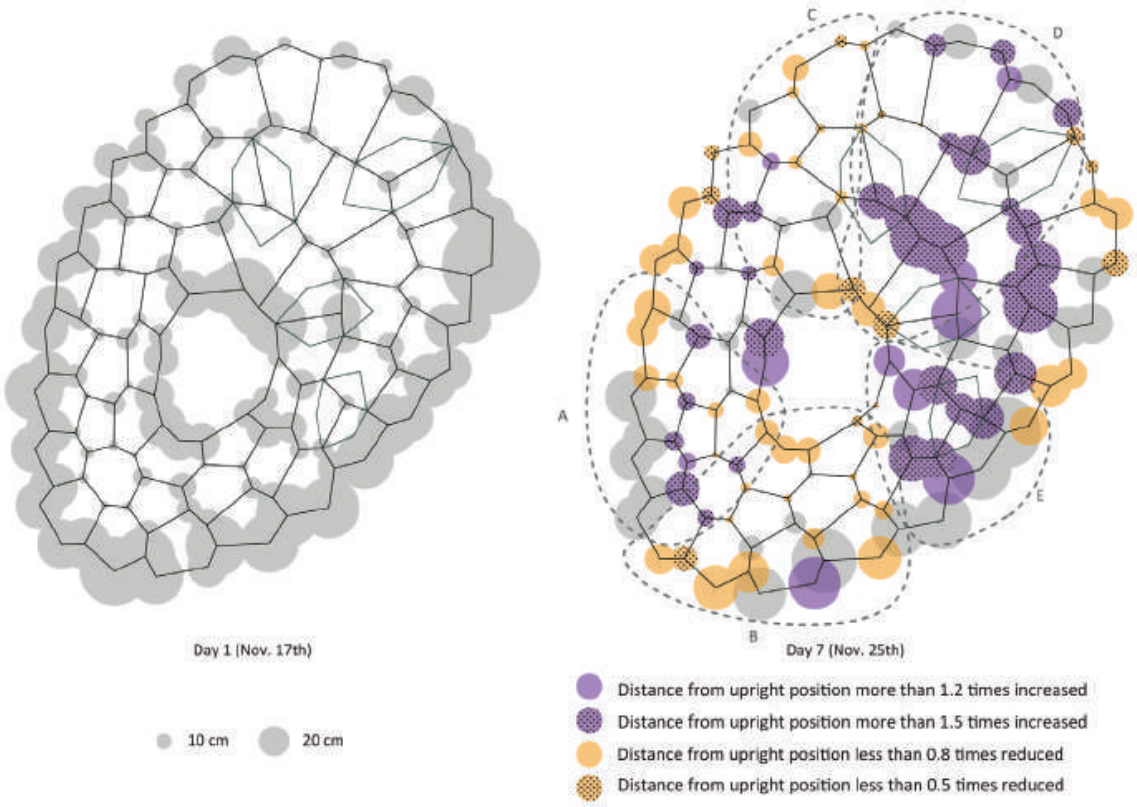


Figure 16: Column deformation. The left image shows the column inclination on Day 1, with the appropriate pre-cambering of the columns on the periphery columns. The right image exhibits the deformation of Day 7, indicating that the optimization algorithm is effective in most of the cases.

A comparison of the differences of column coordinates on the first and last days of loading allows us to examine how some underlying assumptions in our model building affected the outcomes of the implementation of the loading algorithm. As Figure 16 shows, many columns in Zone A, B and C experienced a decrease in distances from their upright positions. This can be attributed to the effective operation of the loading algorithm, as well as to the appropriate pre-cambering of the columns on the periphery and inner edge. In theory, the algorithm was developed based on models where peripheral meshes were fixed; in practice, they were linked with the 19 mm and 16 mm columns which are stiffer than the others with the diameter of 13 mm. In order to make the boundary condition of the edge meshes closer to fixed ends, the columns on the periphery and inner edge were pre-cambered instead. The amounts of cambering were calculated according to estimated loads to make them vertical at the end of loading processes. It can be considered that by appropriate cambering, a similar boundary condition was created to fixed ends in the areas where the algorithm worked well. (**Fig. 16**).

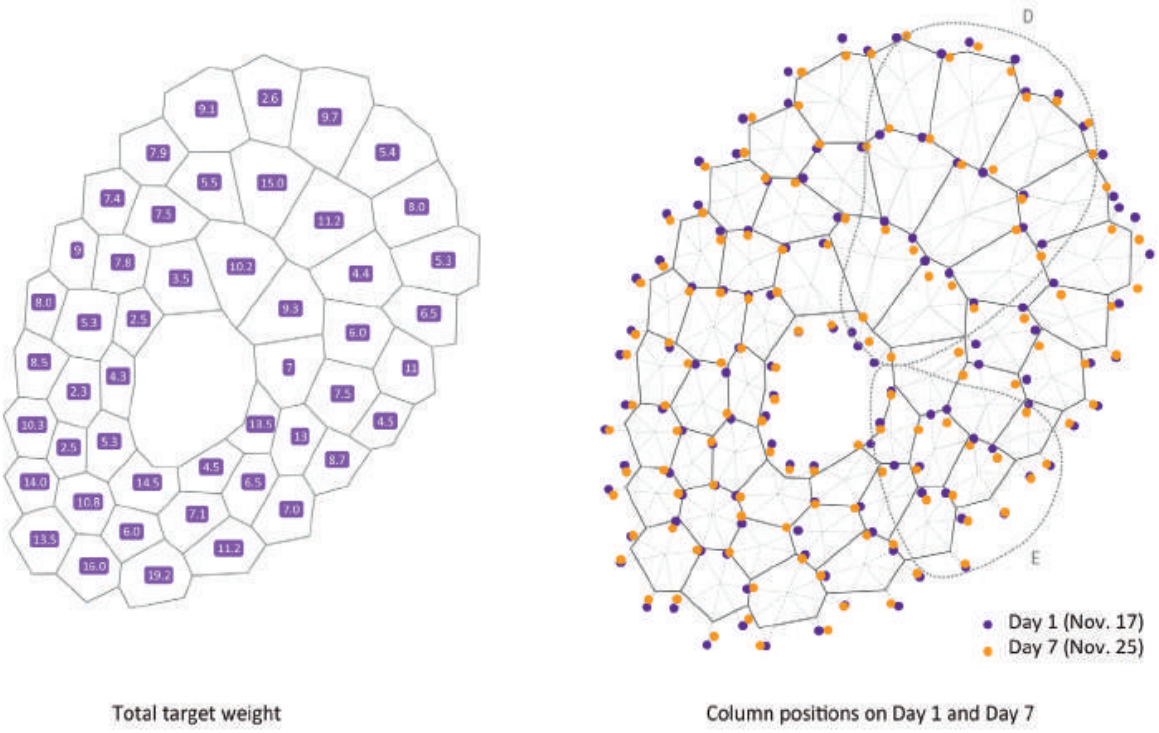


Figure 17: Final loading and deformation outcome indicating large deviation in Zones D and E on the right image, which indicates the ineffectiveness of the algorithm in these two zones. Left: total target weight on each mesh. Right: comparison of column positions on Day 1 and Day 7.

By contrast, Figure 17 reveals that the loading algorithm was not so effective in Zone D and E. Most columns on the periphery rarely moved through a series of iterative loading. It is thought that this had something to do with the fact that no fiber was loaded on peripheral meshes for many days. When the heads of adjoining columns are inclined towards the same direction, compression force of a pseudo-spring can be negative in our model (Model (a) in **Fig. 18**). As the negative compression corresponds not to loading but to taking away the loads, our algorithm was designed to estimate the amount of load for a next shooting step is 0 kg as Model (b) in Figure 18 shows. If tension in a pseudo-spring were taken into account in calculation of load in some way or other, the loading algorithm would have worked better by loading the larger amount of fibers on the peripheral meshes in Zone D and E.

In addition, because the columns were connected just to the target points and not connected each other along the boundary of each mesh, some of the periphery columns were connected to only one target point with a string which represents a mesh. Consequently, when the direction

of pre-cambering was far off from the string, for example in Zone D in Figure 16, the columns were just twisted around their axes keeping the cambered shape.

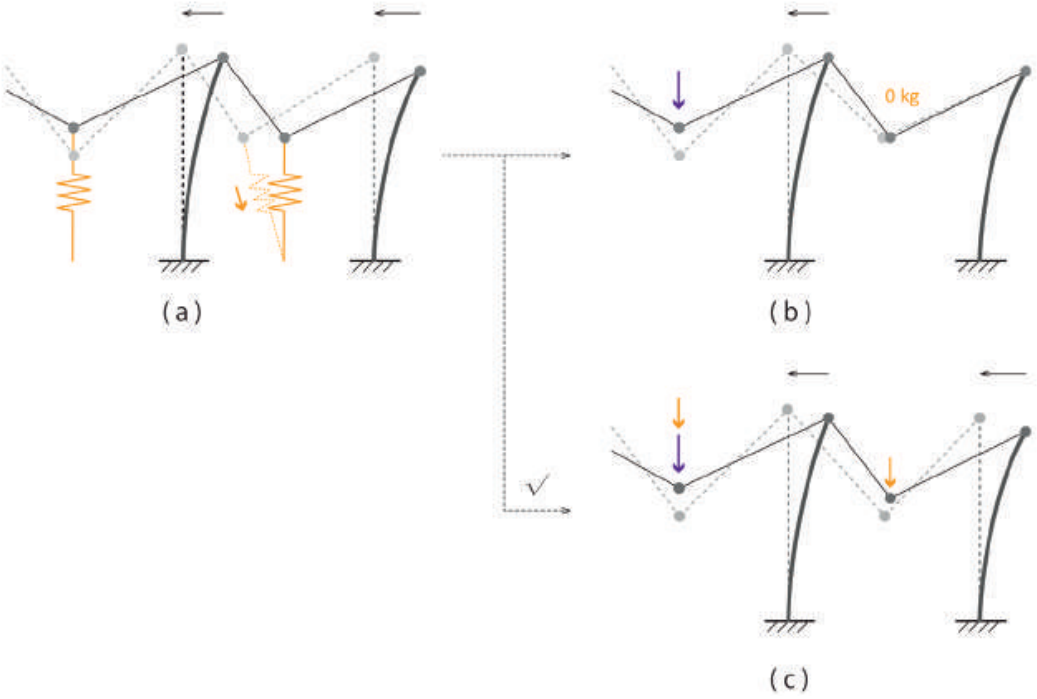


Figure 18: Section models of peripheral meshes and columns, analyzing the cause of deviation in peripheral areas. The images in the first row indicate the deformation in actual cases while the bottom right indicates the ideal case.

4.2 Natural frequency

When calculating the natural period of such a structure, non-linearity should be taken into consideration. A 3.4 m cantilevering 13 mm column is chosen as a representative column. For comparison we applied a 10 N lateral load incrementally on three simple models; a cantilevering column, a cantilevering column with constant 42 N vertical load and a hammock structure with constant 42 N vertical load on each mesh (**Fig. 19** (a)-(c) respectively). The mesh properties are as defined in Section 2.1 and the vertical load is the average weight of the actual fibers supported by each column. It is assumed that the upper half of the column's mass contributes to the vibration and the bottom half of the column's mass is supported by the base and so is disregarded in the vibration calculation. The total mass at the column head, including the mass of the fibers, is 5.9 kg.

For (a), the lateral stiffness is easily calculated as $3EI/L^3 = 22.5\,N/m$, however, for (b) when the weight on the column is relatively heavy, the $P - \Delta$ effect cannot be ignored for large displacements. As shown in Figure 20, when displacements are small the initial stiffness of (a) and (b) are similar, but as displacement increases, the stiffness of (a) remains constant but that of (b) reduces. At a displacement of 400 mm, the lateral stiffness of (b) is 14.6 N/m and the corresponding natural period is 4.00 sec, which is much longer than the measured period 2.97 sec.

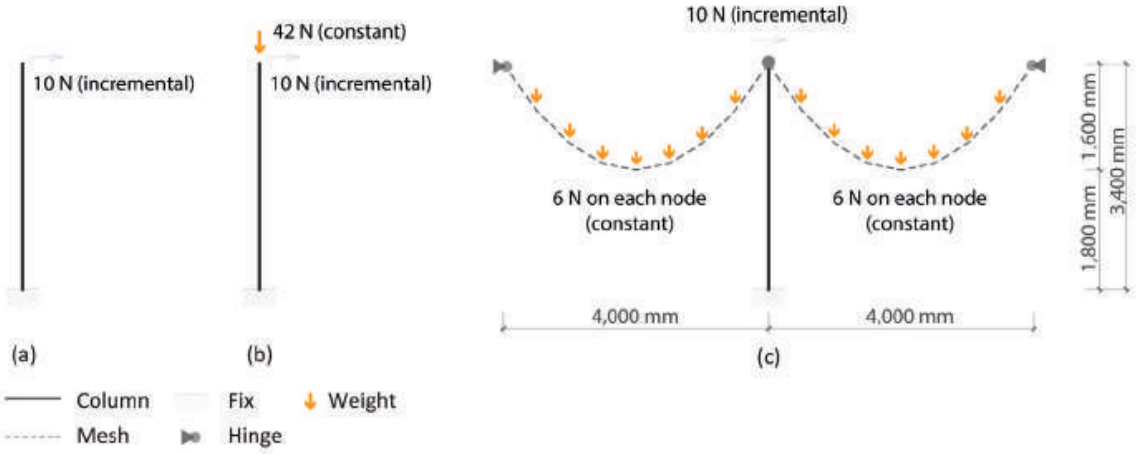


Figure 19: Analytical models used to simulate lateral load. (a) Cantilevering column model with lateral incremental load. (b) Cantilevering column with constant weight load on top and lateral incremental load. (c) Model of hammock structure with constant weight on the mesh and lateral incremental load.

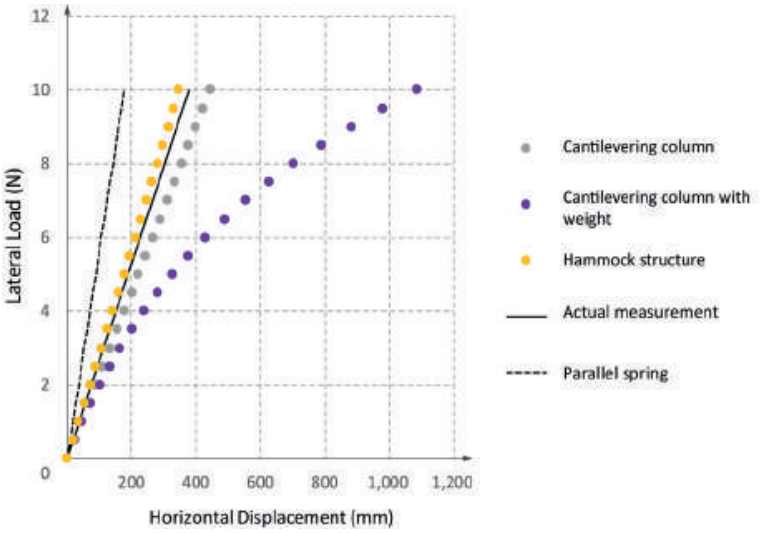


Figure 20: Relationship of lateral load and horizontal displacement of three types of column structure model, along with that of the whole pavilion of actual measurement and on the assumption of a parallel spring.

For the hammock structure (c), the restoring force is generated by the change in sag of the meshes. This stiffening effect only is non-linear, but it cancelled with the above-mentioned P – Δ effect and the hammock structure as a whole behaves very linear. The lateral stiffness is 28.8 N/m and the natural period is 2.80 sec which is close to the measured period. It can be said that the hammock structure not only stiffens the column but also allows a large deformation against the P – Δ effect by cancelling it with its own non-linearity.

Furthermore, assuming simple harmonic motion we can calculate the lateral stiffness of the pavilion from the measured natural period (2.97 sec) and the average mass on an individual column head (5.9 kg). The result is 26.4 N/m which is showed as a black solid line in Figure 20. For comparison the natural period can be approximately calculated assuming that all the columns act like a parallel spring as a whole like when all the columns are connected with a solid roof or tightly tensioned cables and that each column behaves as a column with constant weight on their head like (b) in Figure 20. The total lateral stiffness is calculated in the same way as 7,317 N/m and the total mass is 764 kg (438 kg for the fibers, 326 kg for the upper half of the columns). The natural period is calculated as 2.00 sec and the corresponding stiffness for the mass of 5.9 kg is 56.5 N/m which is showed as a black dotted line in Figure 20. As these two stiffnesses are relatively close compared with the cantilevering column with weight, it can be said that just by flexibly connecting the columns together you can get rather stiff structure. It can be said that the lateral stiffness of the pavilion is somewhere between the two ideal conditions; a single cantilevering column with weight and a single hammock structure with weight, because the 19 mm and 16 mm columns as the actual boundary condition are somewhere between pin supports and roller supports which correspond the two conditions.

5. Conclusion

The development of the loading algorithm was a process of exploring the final outcome of self-stabilization of the hammock structure, and seeking the way to incarnate and scale up it on site. The inverse analysis, loading simulation, and the pavilion installation as a full-scale experiment allowed us to explore the structural property of the hammock structure in natural frequency, as well as its non-linear, self-stabilizing nature.

The application of a set of motion captures to daily 3D scanning was crucial to putting the algorithm into practice. A plug-in of this powerful tool made column coordinate data gained by using it importable to Rhinoceros and Grasshopper in real time, allowing us not only to run the algorithm, but also to examine the input data in CAD as well as structural analysis software.

The structural characteristics of the hammock structure was examined for both vertical and lateral load. For vertical loads, the stiffening effect was verified to make the buckling load more than 6 times larger than a cantilevering column. Furthermore, utilizing supplementary membrane bracing it can be even more than 10 times larger. For lateral loads, our analysis revealed that the lateral stiffness of a single-column hammock structure is almost constant regardless of the displacement, unlike a normal cantilevering column under the P-effect. The stiffening effect was also detected experimentally by manually shaking the whole structure.

In short, the analysis of a set of measured data in load and column deformation revealed that in the pavilion installation, the loading algorithm worked properly, particularly in the areas whose boundary conditions were closer to the models underlying it. This enabled us to find some causes of its partial failure associated with the way of modelling, and examine for future improvement.

Acknowledgement

This project was sponsored by our industrial partner: Taisei Corporation, and conducted in close collaboration with them. We would thank Taisei for their support. We are grateful to Shingi Tarirah, Jun Sato Structural Engineers Co., Ltd. for proofreading our paper.

References

Chen, Kun. 2014. "Elastic stability of arches with buckling constraint components and their applications." PhD diss., University of Tokyo.

Chen, Kun, and Ken'ichi Kawaguchi. 2012. "Ch rokuzai keiyu no saika ni yoru zakutsu hog koka ni tsuite no kisoteki k satsu" [Fundamental Study on Stiffening Effect of Tensioned Components under Load], *Research Report on Membrane Structures* 26: 13–19.

Deuss, Mario, Daniele Panozzo, Emily Whiting, Yang Liu, Philippe Block, Olga Sorkine-Hornung, and Mark Pauly. 2014. "Assembling self-supporting structures." *ACM Transactions on Graphics (TOG) – Proceedings of ACM SIGGRAPH Asia* 33 (6). doi: 10.1145/2661229.2661266.

Doerfler, Kathrin, Sebastian Ernt, Luka Piskorec, Jan Willmann, Volker Helm, Fabio Gramazio, and Matthias Kohler. 2017. "Remote Material Deposition: Exploration of Reciprocal Digital and Material Computational Capacities." In *What's the Matter - Materiality and Materialism at the Age of Computation, International Conference, COAC, ETSAB, ETSAV*, Barcelona, Spain, September 4–6.

Frick, Ursula, Tom Van Mele, and Philippe Block. 2016. "Data management and modelling of complex interfaces in imperfect discrete-element assemblies." In *Spatial Structures in the 21st Century: Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium*, Tokyo, Japan, September 26–30.

Januszkiewicz, Krystyna, and Marta Banachowicz. 2017. "Nonlinear Shaping Architecture Designed with Using Evolutionary Structural Optimization Tools." In the *World Multidisciplinary Civil Engineering-Architecture-Urban Planning Symposium*, Prague, Czech Republic, June 12–16. doi:10.1088/1757-899X/245/8/082042

Krenk, Steen. 2009. *Non-linear Modeling and Analysis of Solids and Structures*. Cambridge: Cambridge University Press.

Lopez, Deborah, Hadin Charbel, Yusuke Obuchi, Jun Sato, Takeo Igarashi, Yosuke Takami, and Toshikatsu Kiuchi. 2016. "Human Touch in Digital Fabrication." In *Posthuman Frontiers: Data, Designers and Cognitive Machines: Proceedings of the 36th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)*, 382–93, Michigan, US, October 27–29.

Veenendaal, Diederik, Mile Bezbradica, David Novak, and Philippe Block. 2014. "Controlling the geometry and forces of a hybrid cable-net and fabric formwork for thin concrete shells." *In Shells, Membranes and Spatial Structures: Footprints: Proceedings of the IASS-SLTE 2014 Symposium,* Brasilia, Brazil, September 15–19.

Yoshida, Hironori, Takeo Igarashi, Yusuke Obuchi, Yosuke Takami, Jun Sato, Mika Araki, Masaaki Miki, Kosuke Nagata, Kazuhide Sakai, and Syunsuke Igarashi. 2015. "Architecture-Scale Human-Assisted Additive Manufacturing." *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH* 34 (4). doi: 10.1145/2766951.

Volumetric modelling for 3D printed architecture

Mathias Bernhard, Michael Hansmeyer,
Benjamin Dillenburger

Mathias Bernhard
bernhard@arch.ethz.ch
ETH Zurich, Switzerland

Michael Hansmeyer
hansmeyer@arch.ethz.ch
ETH Zurich, Switzerland

Benjamin Dillenburger
dillenburger@arch.ethz.ch
ETH Zurich, Switzerland

Keywords:
3D-printed architecture, volumetric modelling, function representation, binderjet printing

Abstract

Powder-based additive manufacturing strategies such as binderjet 3D printing are increasingly attractive and promising for architecture due to their fine resolution, their capacity to precisely distribute material in three dimensions and the availability of very large scales. Building elements can no longer just be designed only by their outer shape, but throughout their entire volume. Existing design software based on a boundary representation of the geometry is unable to fully exploit the geometric freedom of this technology.

We outline the particular geometrical features for additive manufacturing and identify existing limits of conventional CAD systems. We describe the alternative representation of geometry based on volumetric modelling, present specific volumetric operations for design and optimization of 3D printed elements and highlight their potential for architecture. We present two applications in the context of 3D printing for architecture at different scales. One of them operates on a micro-scale for designing specific object properties by geometry. The other one is at an architectural scale, the processing of high-resolution mesh inputs for the preparation of production data of large 3D printed bricks for an architectural structure.

1. Introduction

1.1 Geometry for large scale additive manufacturing

Additive manufacturing bears the great potential of not being constrained by tool accessibility constraints, as subtractive manufacturing is. Instead, objects can be differentiated throughout in three dimensions, not only on the outer surface but also on the inside. Binderjet 3D printers further add to these benefits their high resolution and large scale. The biggest available printer, developed by the company Voxeljet, offers a resolution of 40 000 × 16 000 × 3 000 individually addressable voxels at a size of 4 × 2 × 1 m. In this printing process, the unbound sand of already produced layers always supports the next layer. This allows for printing of cantilevering parts, hollow structures, internal voids, etc. Material can be placed – or rather: solidified – precisely according to functional needs and basically at a detail level of a grain of sand.

1.2 Potential of volumetric modelling vs. BREP

Most CAD packages describe geometric objects as a collection of individual surfaces that are joint along their edges to form the object's boundary. Boundary representations (BRep) are highly efficient and offer a lot of flexibility. The flip side of this representation is that objects can end up with holes, cracks, self-intersections and non-orientable faces. Even for perfectly closed BReps, problems can occur once they need to be offset to form a shell of a certain thickness. While such 3D models can be sufficient for visualization purposes or for subtractive manufacturing CAM software, they are unsuitable for an additive manufacturing process.

As its name suggests, volumetric modelling does not primarily deal with free-floating surfaces but with the definition of the entire space and describes shapes as volumes. The 0-level iso-surface always unambiguously divides inside from outside, and therefore slicers for the preparation of job files have no problem in determining where to apply binder. The bed of powder, which is always full, is conceptually close to volumetric modelling and we therefore consider them a promising pairing.

1.3 Function representation

Many algorithms used in volumetric modelling, e.g. marching cubes (Lorensen and Cline 1987) for the generation of an iso-surface mesh, have their origins in medical imaging. With CT and MRI scans, this domain provides data as a stack of 2D images that together form a volume of voxels. The methods presented in chapter 3 also start by defining a three-dimensional grid of voxels at the required resolution first and then profit from the organisation of this data-structure to perform geometric operations on high resolution input geometry at speed.

Volumetric modelling however includes also the description of shapes as the result of a function (function representation, FRep) that converts any point coordinate into a real value, $\mathbb{R}^3 \rightarrow \mathbb{R}$. Many geometric primitives (sphere, box, cone, cylinder, plane, platonic solids, etc.) can be described not explicitly by placing vertices in space and connecting them with edges and faces, but instead as a function $v=f(x,y,z)$. In this formula, v is the distance of point (x,y,z) to the shape's surface. All the point locations where this function evaluates to $v=0$ form the skin of the object. All those resulting in a negative value lie inside the object, all those with positive values lie outside of it. Such a function is called a

distance function and space is defined as a signed distance field (SDF).

The combination of objects is now not a geometrical problem of difficult intersection calculations, but purely a matter of arithmetic (**Fig. 1**). If object A is defined by $a=f(p)$ and object B is defined by $b=g(p)$ $(p \in \mathbb{R}^3)$, then the Boolean union $A \cup B$ is defined as $min(a,b)$, the intersection $A \cap B$ as $max(a,b)$ and the subtraction $A-B$ as $max(a,-b)$. Many more combinations with a and b are possible e.g. to produce smooth blends, chamfer angles, stepped transitions or V-shaped grooves along the intersection curve. To convert a solid object into a shell along its surface, it suffices to calculate the new value v' as $v'=abs(v+(f-0.5)*d)-d/2$, where d is the thickness of the shell and f is a position factor that is $f=0$ if the shell is offset to the outside, $f=1$ if offset to the inside and $f=0.5$ if half of d is on either side of the original surface.

Figure 1: Distance fields and Boolean operations; *top left*, circle; *bottom left*, rectangle; *top middle*, circle plus rectangle; *bottom middle*, circle rectangle intersection; *top right*, circle minus rectangle; *bottom right*, rectangle minus circle

With function representation, objects can be combined into arbitrarily complex constructive solid geometry (CSG) trees. These definitions are completely resolution independent. A discretisation of space e.g. into a

three-dimensional grid of voxels for the generation of a mesh approximation or into a two-dimensional slice for layer-wise feeding of a 3D printer can be done by querying the CSG tree for every point in the grid at the required resolution.

1.4 Related work

Volumetric modelling has always been developed in parallel to surface modelling in the history of computational geometry, ever since the early days of computer graphics (Blinn 1982). Some authors distinguish implicit from parametric surfaces (Bloomenthal et al. 1997). We follow the naming function representation FRep (as opposed to BRep) proposed by Pasko et al. (1995).

The work we present draws from many different branches and industries – from pure mathematics over game design to aerospace engineering – and unites many of the methods in a comprehensive framework. There are many commercial and open source software packages available that integrate some of the concepts of volumetric modelling. Most of them do not specifically address architecture, and all of them have specific shortcomings for a direct application in the design and fabrication of 3D printed architecture.

Monolith (Michalatos and Payne 2013) which puts the main focus really on the voxel (3D pixel) aspect and is an attempt to create a 3D image editor by offering many freehand editing tools like swirls and smear. With the multi-channel property of the voxels, the main target output of Monolith is multi-material 3D printing. The other project is Symvol by Norwegian start-up Uformia (Vilbrandt, Pasko, and Vilbrandt 2009). Their mission is mainly to provide a software package, that allows the generation of shapes which are closer to natural shapes, not homogenous and solid either but internally differentiated.

2. Performative microstructures

Making 3D printed parts lighter can often lead to significant cost savings, both in production (expensive raw materials), and in operation (e.g. aerospace industry). The central question is where material can most efficiently be omitted while still maintaining the required performance? On the scale of the overall form, topology optimisation (Bendsøe and

Kikuchi 1988, Bendsøe and Sigmund 2003) has become a valuable solution, because AM finally allows the production of the resulting shapes without the need to redesign due to fabrication constraints such as tool head access or demoulding drafts.

On a smaller scale within the part itself, various degrees of porosity can be introduced. For configurations of linear sticks combined into spatial trusses, the term lattice has become prevalent. We also present a second approach based on continuous convoluted surfaces for which we employ the more general term microstructures. Any porous geometry is constrained by the need for both the solid and the void part being fully connected – the solid part for optimal force flow and the void part to enable unbound material removal.

2.1 Lattices

The first approach defines the microstructure as a spatial truss of skeletal lines (hence the name lattice), that are then thickened by pipes along these axes. Any collection of lines is in theory possible, from completely random over stochastic (e.g. foam-like, along the edges of 3D Voronoi cells) to strictly ordered in a triply periodic orthogonal grid. For the triply periodic structures in Figure 2 (left), only one octant of the unit cell needs to be defined. This octant is then mirrored across all the three planes XY, YZ and XZ before being array-copied to fill the entire space.

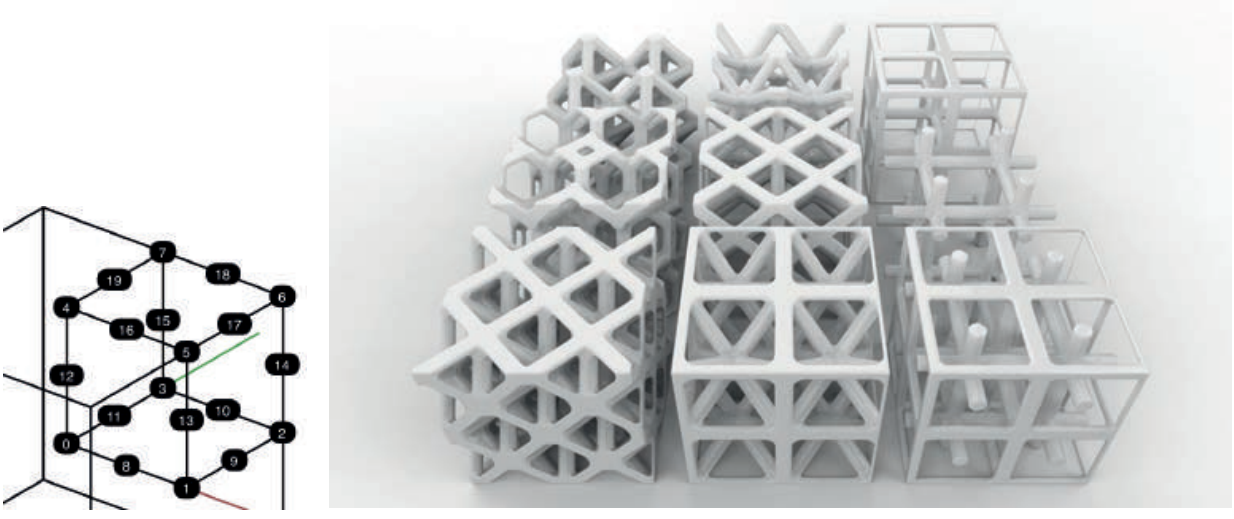


Figure 2: *right*, 9 examples of skeleton-based microstructures, each 8 (2 × 2 × 2) unit cells, front left: octet truss; *left*, positive octant, point 0 at the center of the unit cell, 1 on X-axis, 3 on Y-axis, 4 on Z-axis, example node locations.

For the thickening of the skeletal lines, two different methods are compared here. The construction based on BRep typically constructs two rings of points at either end of each line with a predefined offset from its end points and connects the rings to form cylinders. For every node, all the rings' points are collected, a convex hull polyhedron is calculated, and the faces between the cylinders are added to the final mesh. This works well for many cases and produces reasonably small meshes. Possible sources of failure are very acute angles between two connected lines or small surface areas within loops of lines, which result in self-intersecting meshes.

The construction based on FRep treats the thickening as the Boolean union of multiple cylinder functions. Whenever internodal areas become too small or the pipes too thick for an opening, the topology is just altered without causing any error (see **Fig. 3**). The lines do not need to connect end-to-end but can also form T-joints. The resulting microstructure defines an SDF and can be combined in the CSG tree just as any other object.

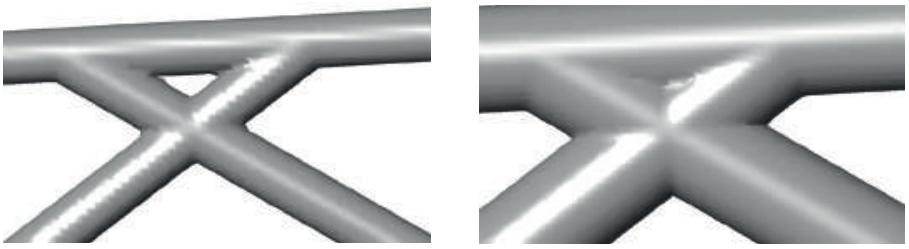


Figure 3: Skeleton thickened with FRep cylinders; the hole just turns into a solid node as diameter increases, without causing topological problems.

2.2 Microstructures

The second approach uses not distance but trigonometric functions for the determination of $v=f(x,y,z)$. A large number of triply periodic minimal surfaces (TPMS) have been discovered since the first was described by Schwarz (1871) and more added to the set by Schoen (1970). As trigonometric functions oscillate between -1 and 1, the 0-level iso-surface divides space into solid and void at equal shares (porosity 0.5). Instead of this split, the surface can be given a thickness (shell) which results in two highly intertwined but never touching volumes of air. A practical use case thereof could be the internal chamber of a heat exchanger. Other parameters that can be varied are the wavelength or – by applying

an inverse transformation matrix to the query point – the rotation and starting point of the grid.

The collection in Figure 4 shows 3D printed 50 × 50 × 50 mm samples of some of these surfaces, with different wavelengths, orientations, and with or without shell, and various spatial noise functions (Perlin, cubic, cellular, etc.) forming the control group. They all have a porosity of 0.5 in common and are the basis for the tests described in chapter 2.3.

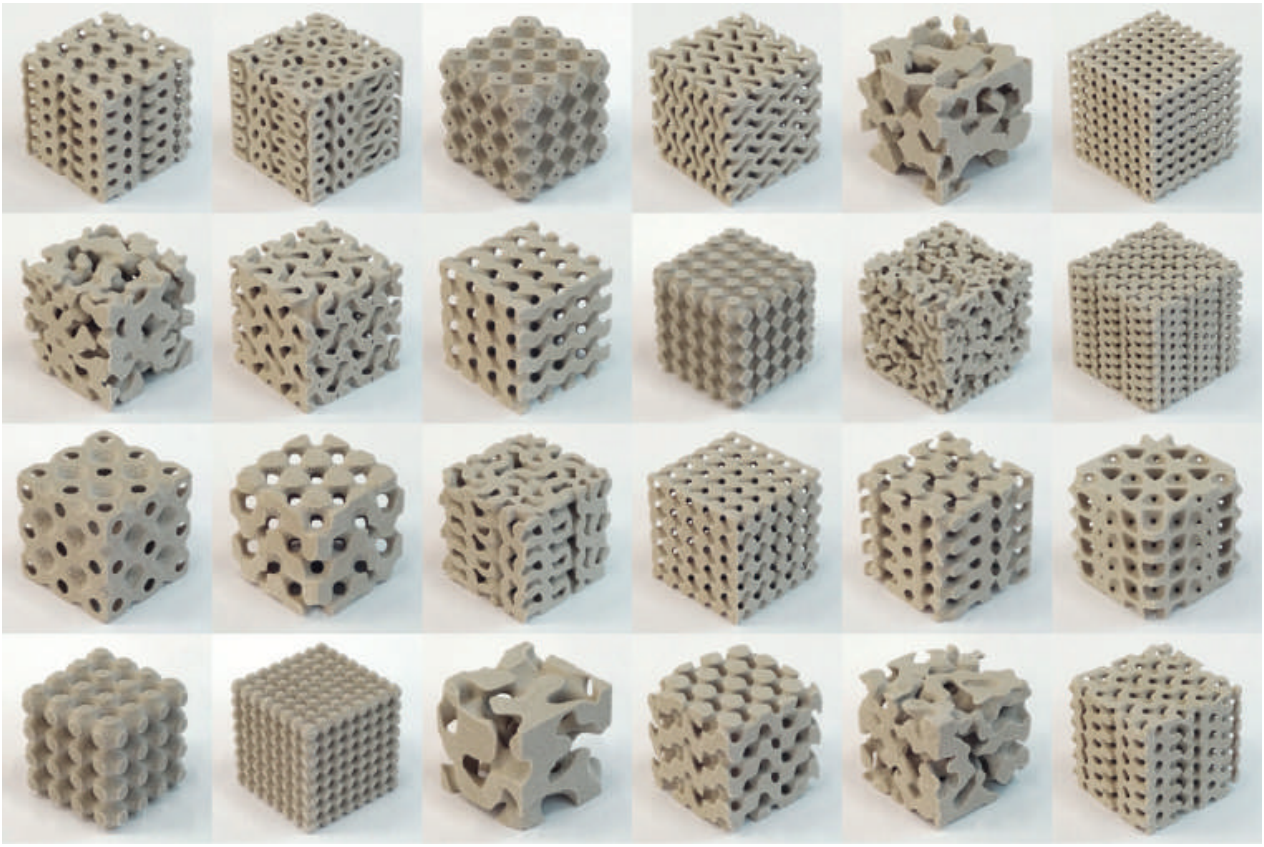


Figure 4: 3D printed samples of various micro-structures, including TPMS and noise.

The addition of microstructures to the interior of parts can alter their physical properties, e.g. the behavior under compressive load. Physical tests have shown that completely solid 3D printed sandstone parts (porosity 0) resist forces of 6–8 kN, before a diagonal crack leads to total failure (see **Fig. 5**). Porous parts differentiated with a micro-structure as those shown in Figure 4 stand more than twice as much deformation until building up a resisting force and do not fail completely but continue resisting with ±60 % of the maximum force while being squished.

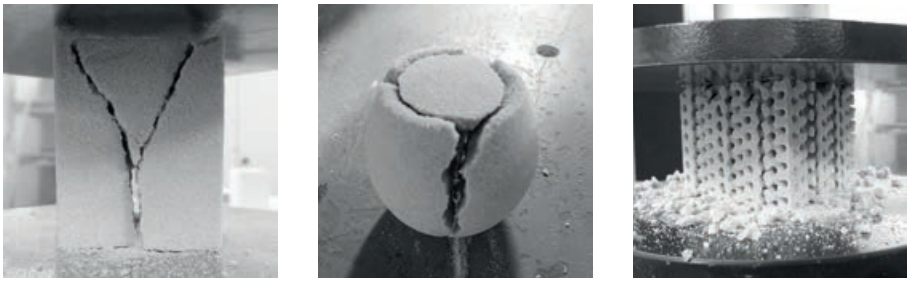


Figure 5: Different failure behavior under compressive load; *left* and *middle*, total failure crack of solid volume; *right*, sponge-like squishing of microstructure.

2.3 Designing with microstructures

The lattices and micro-structures (described in the previous two chapters) represent a function $v=f(x,y,z)$ to define a distance field as well, just as any other primitive. Their degree of porosity can continuously be blended from massive to lofty along a spatial gradient or based on local structural needs. They can be combined with other objects using a variety of operations. Basic Boolean operations (union, subtraction and intersection) but also smooth blends between sharp and round features by exponential or logarithmic functions enable the designer to create part geometries (**Fig. 6**) difficult to achieve in conventional CAD packages.

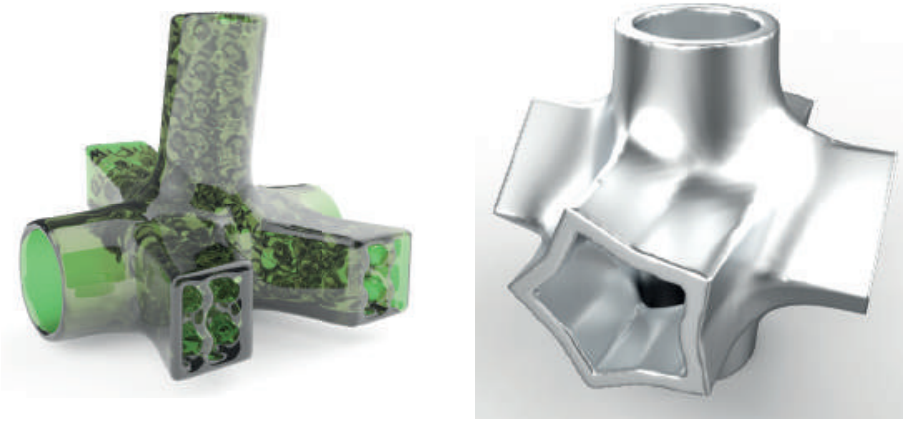


Figure 6: Complex nodes joining different pipe objects, shelled, partially filled with a micro-structure; generated from simpler primitives assembled through the CSG tree.

2.4 Data structures and Implementation

Three-dimensional objects, for applications like rendering or additive manufacturing, are defined by their skin separating inside from outside. To generate that limit surface using a marching cubes algorithm, only the distance values for the voxels that contain a snippet of the surface need to be calculated at the maximum resolution. Calculation and storage of distance values for dense voxel grids is computationally very expensive and multiplies by a factor of eight upon cutting the voxel's edge length in half. Depending on the application, for extensive regions of the volume a calculation of a precise distance value is not required and only a coarse approximation of the value will suffice. One solution to this challenge is to organize the data in a so-called sparse voxel octree (SVO), which works as follows:

1. The root cell is created to comprise the region to be queried (e.g. as the minimum bounding cube of all objects).

2. Each cell stores its center point, edge length e, a list of child cells (empty at first) and its subdivision level n. The root cell has level $n=0$.

3. The cell's center location is fed in as query point into the cumulated distance functions of the CSG tree.

4. If the returned distance is smaller than $\sqrt{3}/2 * e$ (half the cell's room diagonal), eight child cells are created by splitting the original cell in two along all axes X, Y and Z. The level of the child nodes is $n+1$ and their edge length $e/2$.

5. The process from step 3 is recursively repeated for all the eight child nodes, until some specified maximum subdivision level is reached.

Figure 7 shows the result of this procedure as the 2D equivalent, a quadtree subdivision. The color is given by the distance of the cell's center point to the boundary of the shape, the number in the squares indicates n, the level of subdivision (omitted above 5 for readability, maximum level 8).

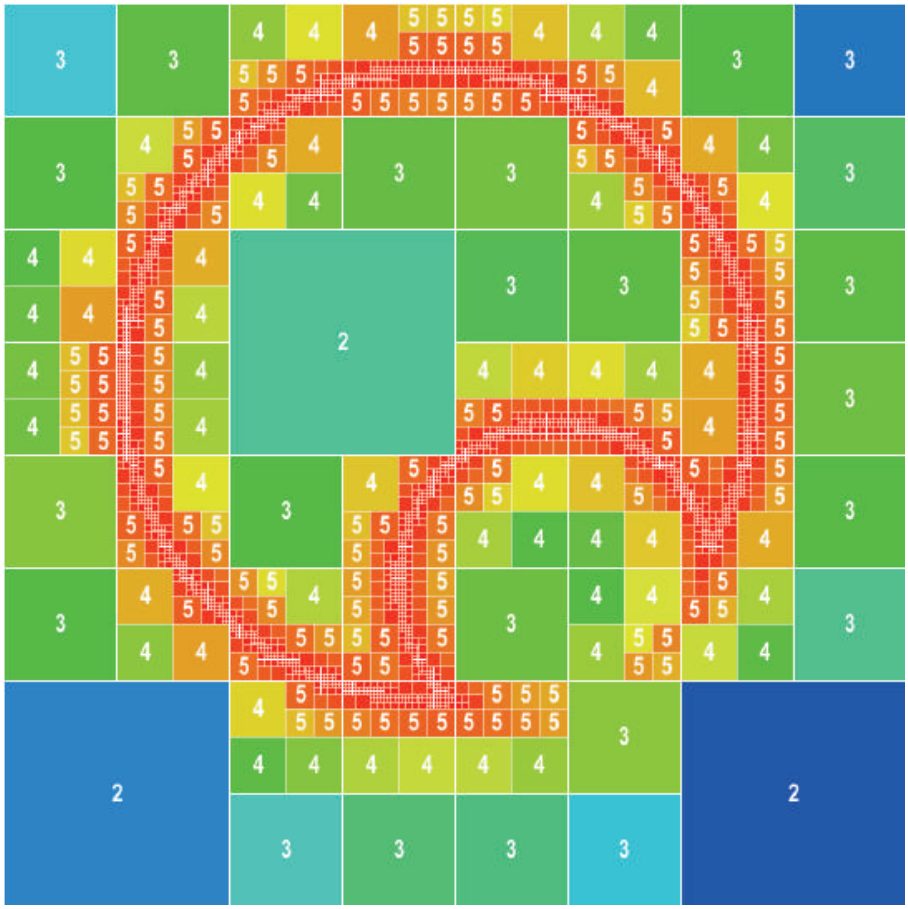


Figure 7: Quadtree visualization.

The following table (**Tab. 1**) compares the octree method with a full voxel space numerically in terms of number of elements. The test geometry consists of the union of two spheres from which a third sphere is subtracted. For other objects, these numbers would differ, depending on the ratio between their surface and the occupied volume. The last column shows that after 7 steps, there are already 15 times as many voxels as there are octree cells and 60 times as many after 9 steps.

The column "growth rate of octree cells" in Table 1 shows by what factor the total NO increases. In the first two division steps, this is close to eight as the shape passes through most of the cells. From the third step on, this number quickly stabilizes around four (see highlight and **Fig. 8**). A doubling of the resolution therefore only increases the number of octree cells by $2^2=4$ while the number of voxels grows by a steady rate of $2^3=8$. This is because the tree is only refined along the surface (2D) and not the entire volume (3D).

division step	number of cells per axis	number of voxels NV	number of octree cells NO	growth rate of octree cells	ratio NO:NV
1	2	8	9	9.000	0.889
2	4	64	73	8.111	0.877
3	8	512	529	7.247	0.968
4	16	4'096	2'177	4.115	1.881
5	32	32'768	8'509	3.909	3.851
6	64	262'144	33'913	3.986	7.730
7	128	2'097'152	138'857	4.095	15.103
8	256	16'77'216	551'377	3.971	30.428
9	512	134'217'728	2'210'033	4.008	60.731

Table 1: numerical comparison between voxel space and octree subdivision.

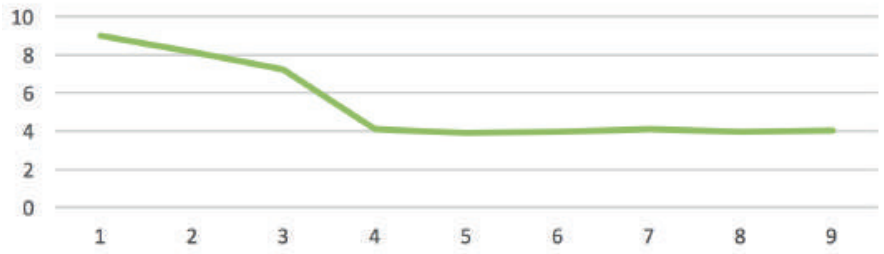


Figure 8: growth rate of the number of octree cells from one subdivision level to the next.

Timed test runs over 20 iterations of distance field calculations returned an average of 483 ms for the voxel grid and 91 ms for the octree, a speedup by a factor of more than five.

3. From high resolution meshes to 3D printed bricks

Architectural designs are usually not generated in voxel space but constructed in CAD Software based on BRep. The data is not represented in a 3D printable format. Multiple entities might be self-intersecting, have holes, contain overlapping or duplicate surfaces, as well as non-aligned normal orientations. One cannot assume clean two-manifold models but rather a “triangle soup”.

This chapter introduces volumetric procedures to turn these meshes into printable volumes. The voxelization of high resolution meshes is

discussed and a novel approach to create stiffening ribs in voxel space is presented. It is shown that by using only simple Boolean operations and voxel propagation algorithms, all necessary geometric operations can be done in a robust way, allowing optimized print data to be created. The results are demonstrated using the example of the printed bricks of the project *Digital Grotesque*.

3.1 Voxelization of high resolution polyhedral meshes

Volumetric representation is not only useful for the creation of shapes from the combination of geometric primitives but can also be derived from complex polyhedral meshes with many individual facets. The process of voxelization converts such data into a three-dimensional volume of data values. One challenge of this process is that the computation time rises in relation to both the number of triangles as well as the resolution of the voxel space.

Therefore, instead of calculating distance values for all data points, we only calculate the shell distance field in an exact way, leading for small triangles to similar optimization effects as with the octree data-structure described above (Cohen-Or and Kaufman 1995, Jones, Baerentzen, and Sramek 2006). For each triangle of the input mesh, we only measure the distance to voxels within a predefined threshold. In order to detect those data points in fast manner, we apply several conditional filters in sequence. Only certain points are analyzed (see **Fig. 9**), namely those which are:

1. Within the bounding box of the triangle.

2. Close enough to the plane defined by the triangle.

3. Close enough to the triangle itself. 3 cases are distinguished; projected point is closest:

    a. to an edge
    b. to a vertex
    c. to the plane of the triangle

As each triangle is only inspected once, this procedure can be optimized by representing this triangle in a form which allows a quick analysis of the distance to a point.

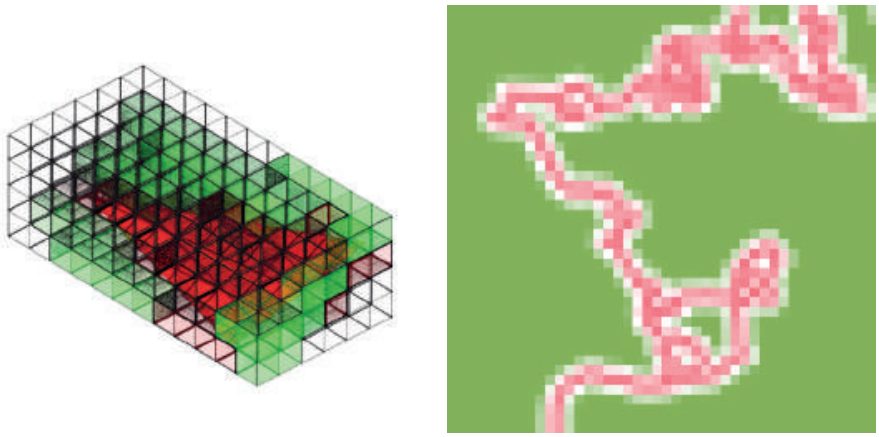


Figure 9: *left*, Implementation of distance to triangle measurement; *right*, the resulting distance shell around the mesh in section.

3.2 Definition of interior regions

Another challenge of converting non-manifold meshes into printable data lies in defining interior and exterior regions. While several strategies exist to determine whether a point lies inside or outside a boundary of a polyhedron, these methods fail when meshes are not clean (closed and manifold).

Therefore, we detect interior regions within the volumetric representation. As a voxel-space can also be regarded as a network of cells, network algorithms to calculate reachability using connectivity methods apply here. We classify the inside and outside of the voxel model with a flood fill approach (Khudeev 2005). Initially, only a single starting point on the outside needs to be defined. Every voxel that can be reached without crossing a voxel that is closer than a certain threshold to the surface will be outside as well.

With this method, it is also possible to turn an open surface into a volume (**Fig. 10** and **Fig. 11**). An additional boundary volume is introduced. Every data point which cannot be reached without crossing either the voxelized surface or the boundary surface and which is contained by this boundary volume is considered to be inside.

Figure 10: Flood fill operation with a cube as border-constraint.

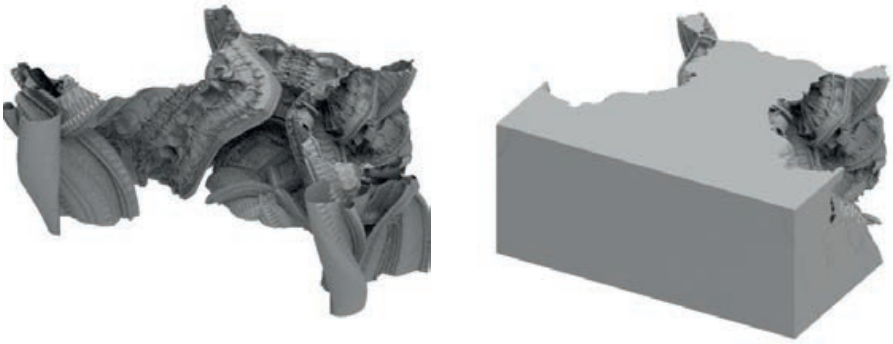


Figure 11: Conversion of a mesh surface into a solid volume.

3.3 Offset operations

Offset operations are often needed in architecture, for example to turn a solid volume into a shell with a certain thickness, or to turn a surface into a solid volume. Offset operations are challenging in mesh representation (Farouki 1985), because offsets can self-intersect, leading to changes in topology. These operations are also slow and not robust due to rounding errors.

In volumetric space, offset operations are easier. In a continuous field in which all data points contain the distance value, variable offsets can be extracted at any time with a Boolean operation: everything within a certain distance is considered to be inside (see also 1.3).

One challenge is that the complete calculation of a continuous distance field for a high-resolution mesh is computationally expensive. Therefore, we work with an approximation based on the shell distance, described above. Different approaches are documented for calculating an approximated distance field (Jones, Baerentzen, and Sramek 2006). We use a scheme based on the chamfer distance transform (Rosenfeld and Pfaltz 1966), in which the distance is calculated using the voxel network, avoiding expensive Euclidean distance calculations (see **Fig. 12**). Here, for short distances, the precision is sufficient for architectural applications.

We implemented sweeping and wave front schemes, which have different running speeds and memory consumption depending on the geometric features and the computational implementation. The sweeping scheme travels through the entire voxel field in two specific directions, once from the lower left front corner to the upper right back corner, and once in the opposite direction. The wave front approach

$\sqrt{2}$	1	$\sqrt{2}$
1	0	1
$\sqrt{2}$	1	$\sqrt{2}$

Figure 12: 2D Chamfer distance values

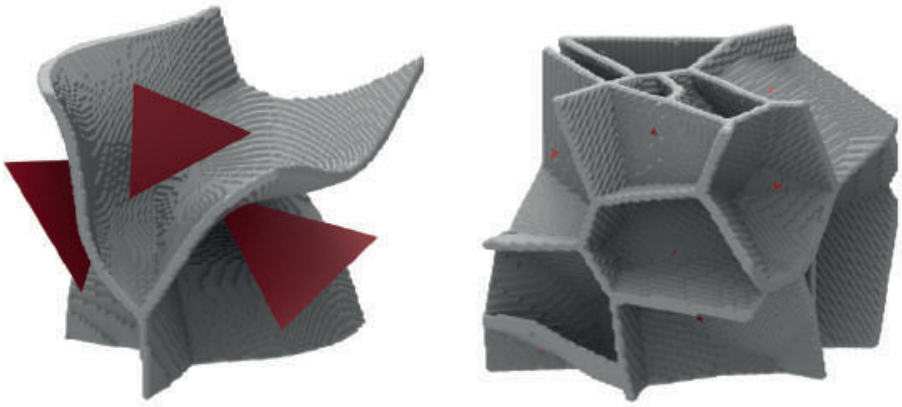


Figure 13: Voxel-based 3D Voronoi diagrams of two different sets of triangles.

requires a more complex data handling and travels from close points to the furthest regions.

It is possible to calculate an approximated 3D Voronoi tessellation based on the chamfer distance transformation without additional calculations. Each cell not only assigns the next distance to its neighbors, but also passes the information to which point or object this distance is measured. At the end, regions which have the same closest object are within a common Voronoi cell.

One way of calculating locally adapted offsets is to give different distance functions to separated entities (for example to each triangle), or to measure the distances between cells in a different manner according to their spatial position. In addition, the direction can be taken into account, creating variable offsets to horizontal or vertical elements.

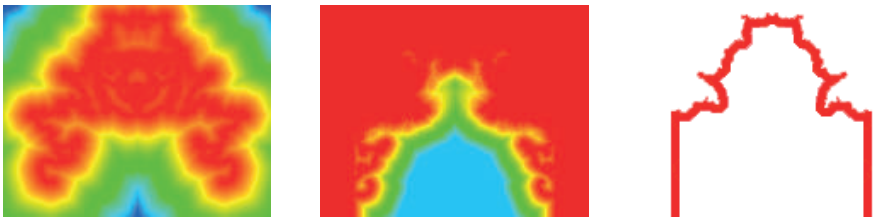


Figure 14: Continuous distance field based on chamfer distance transform for offset operations.

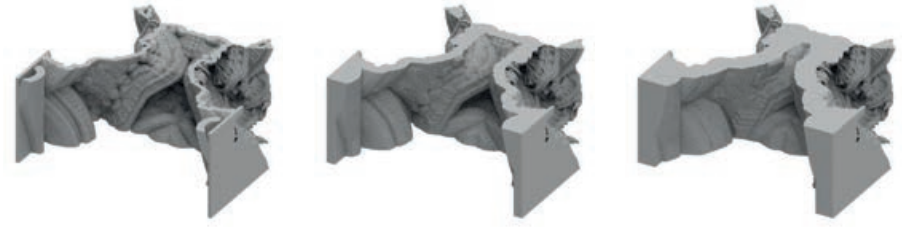


Figure 15: Robust offset operations on complex meshes.

3.4 Stiffening-structures

In order to stabilize 3D prints without using too much material, stiffening structures such as ribs can be introduced. On irregular mesh surfaces, this requires complex operations in order to maintain a manifold mesh. As these stiffening structures are often ideally oriented along the normal of the surfaces, it is not trivial how this can be done within a voxel-space. In this section we outline an approach which consists of several steps based on the flood fill and Voronoi tessellation processes described above.

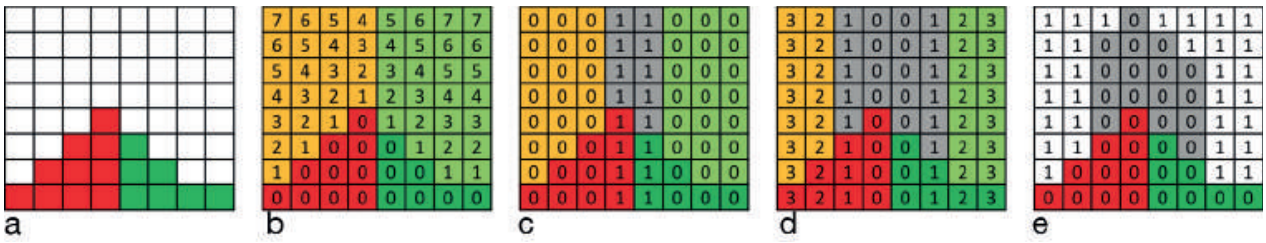


Figure 16: *a*: Classification of regions, *b*: Voxel-based Voronoi, *c*: Voronoi borders, *d*: Border offset, *e*: Border crop (simplified distance measurement).

Regions of the surface are classified according to the desired rib layout (**Fig. 16a**). From these regions, a 3D Voronoi tessellation is created (**Fig. 16b**). The borders of these tessellations are perpendicular to the surfaces of the regions. In voxel-space, border voxels can be identified by the number of neighbors belonging to a different Voronoi region (**Fig. 16c**). Once these borders are selected, an offset operation inside the voxel-space defines the desired thickness of the ribs (**Fig. 16d**). The desired depth of the ribs can be cropped after an additional offset operation based on the original surface.

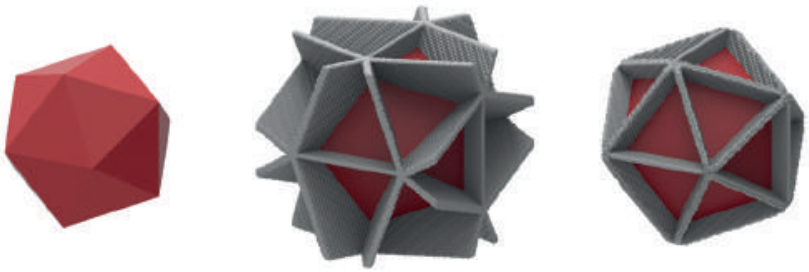


Figure 17: *left*, exemplary input mesh; *middle*, Voronoi borders of each triangle; *right*, border crop.

These computational steps can be calculated in linear time, and always result in a manifold geometry. The downside of this approach is that the approximated distance calculation leads to some imprecisions. In the case of the stiffening ribs for the interior of objects, this imprecision is acceptable for most applications.

Additional operations that are meaningful in the context of architectural applications can be easily performed in voxel-space. Calculation of the overall mass, the volume, and the gravity center are based on simple arithmetic operations. Within a continuous distance field, the calculation of a skeleton, or the detection of thin parts are also fast to calculate.

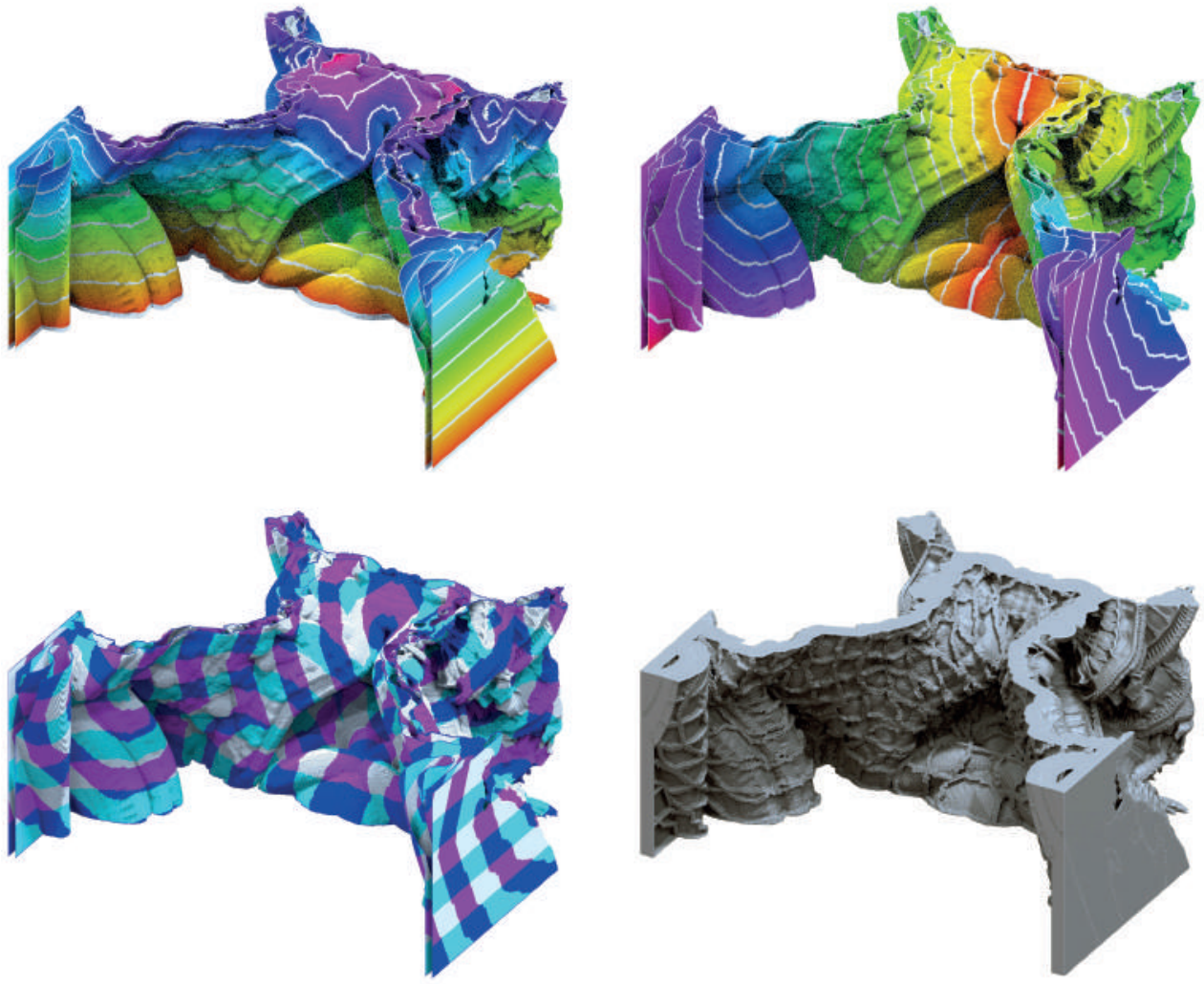


Figure 18: *top left*, voxel-based surface distance from ground; *top right*, voxel-based surface distance from central axis; *bottom left*, segmentation of a volumetric model based on surface distances; *bottom right*, stiffening rib structure based on 3D Voronoi calculated on volumetric distance field.

Figure 19: Voxel-based offset and stiffening ribs of 3D printed shells.

3.5 3D printed bricks

The described volumetric operations have already successfully been used for the generation of 3D printable geometry for multiple large-scale 3D printed structures (Hansmeyer and Dillenburger 2014). The project *Digital Grotesque* showcases the most radical application, as here the initial mesh data carried extreme details and complex topological structures.

In *Digital Grotesque*, the initial mesh geometry consists of 260 million triangles. In order to create 3D printable data, a volumetric model in the resolution of 25 billion voxels was generated.

Using this volumetric model, all offset operations and Boolean operations for the inner details were calculated according to the methods described above. From the volumetric model, printable layer data could

be extracted in the slice-format CLI, directly feeding the binderjet printing system.

In the case of the *Digital Grotesque*, both the strength and the limitations of the presented approach can be demonstrated. No existing commercial software was able to perform the necessary operation of turning the mesh into a manifold geometry. Through the help of volumetric modelling, a large architectural structure could be discretized into 3D printable elements. Those printed stones could be optimized, reducing the material thickness to minimum and strategically stiffening the shell with additional ribs. Alignment cones and lifting details positioned according to the center of gravity could be integrated (see **Fig. 20**).

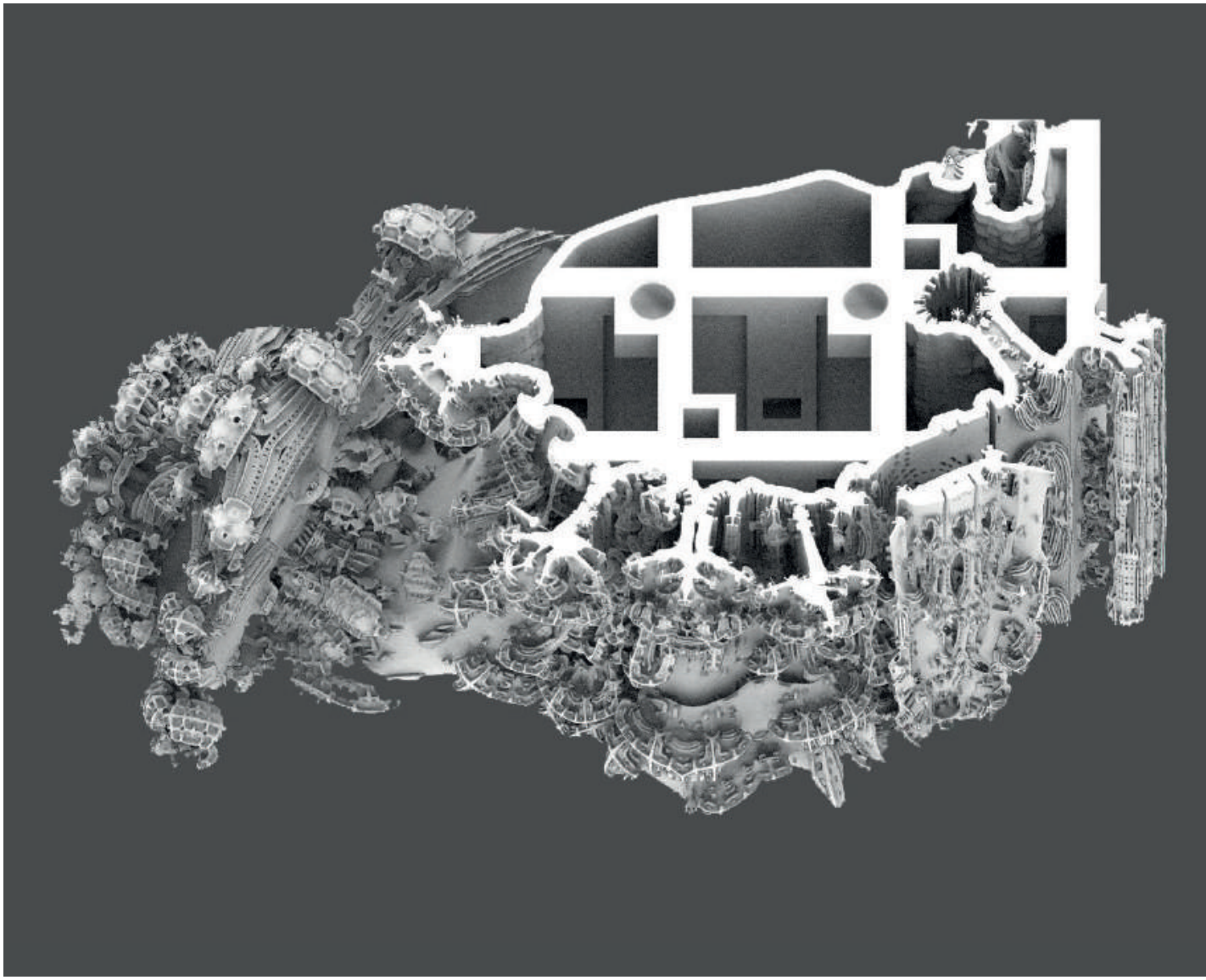


Figure 20: 3D Printed Brick. An initial mesh is detailed in voxel-space.

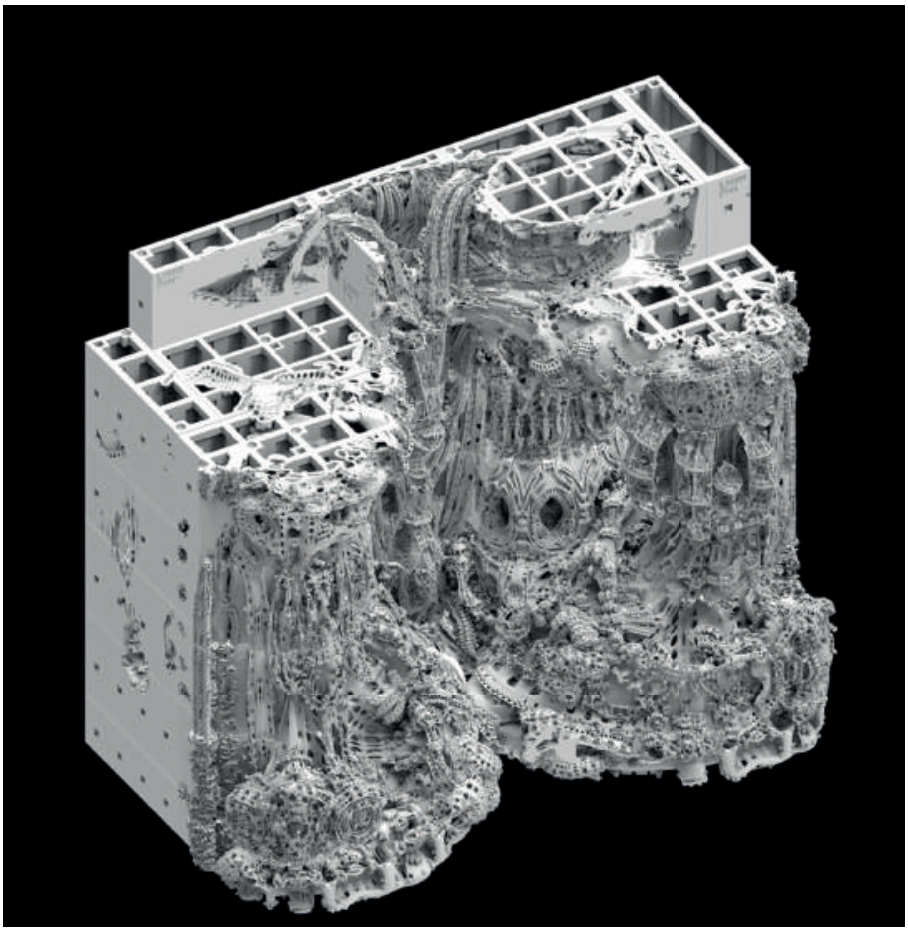


Figure 21: Assembly sequence of multiple 3D printed bricks, all detailed in voxel-space

Figure 22: Detail of a 3D printed brick, showcasing interior structure of volumetric offset operation

3.6 Discussion

Although the resolution of the voxel model was extremely high, the detail of the initial mesh form could not be translated to the printed parts to 100%. Specifically, filigree features with sharp edges in the original became thickened and smoothened within a fraction of millimeters. On a positive note, this slight thickening protected the most delicate features from cracking.

On the interior of the printed bricks, the pattern of the approximated distance calculation is still readable, leaving traces of the volumetric approach (see **Fig. 22**). On the exterior, some of the original triangulated facets remained visible throughout the entire process. Last but not least, certain regions, depending on the orientation within the print-box displayed the traces of the layer height 0.3 mm of the printers.

4. Conclusion

We are facing a new situation in architecture: with large 3D printers, we might soon materialize almost anything. Currently, our architectural design tools do not allow us to address this potential. Volumetric modelling gives us a new perspective. As architects, we define space by designing the boundaries between the inside and the outside. But this design should not stay on the surface of the building elements only. The interior structure of building elements is becoming more and more relevant. Instead of a binary separation of space into either solid or void, various degrees of porosity can be tailored to specific needs. If we want to optimize building elements and reduce the amount of consumed material, we need tools to differentiate both their external and internal structures. This paper describes fundamental geometric strategies to do so. A plethora of further instruments can be realized in volumetric representation. It has been shown how volumetric modelling can help to materialize designs. We believe that in the future it will also help to generate new ideas and spawn unforeseeable designs.

References

Bendsøe, Martin Philip, and Noboru Kikuchi. 1988. "Generating Optimal Topologies in Structural Design Using a Homogenisation Method." *Computer Methods in Applied Mechanics and Engineering* 71 (2): 197–224.

Bendsøe, Martin Philip, and Ole Sigmund. 2003. Topology Optimization: *Theory, Methods, and Applications. Engineering*. 2nd ed. Springer Berlin Heidelberg. https://doi.org/10.1063/1.3278595.

Blinn, James F. 1982. "A Generalization of Algebraic Surface Drawing." *ACM Transactions on Graphics* 1 (3): 235–56. https://doi.org/10.1145/357306.357310.

Bloomenthal, Jules, Chandrajit Bajaj, Jim Blinn, Marie-Paule Cani-Gascuel, Alyn Rockwood, Brian Wyvill, and Geoff Wyvill. 1997. *Introduction to Implicit Surfaces*. Edited by Jules Bloomenthal. San Francisco, California: Morgan Kaufmann Publishers, Inc.

Cohen-Or, Daniel, and Arie Kaufman. 1995. "Fundamentals of Surface Voxelization." *Graphical Models and Image Processing* 57 (6): 453–61. https://doi.org/10.1006/GMIP.1995.1039.

Farouki, R.T. 1985. "Exact Offset Procedures for Simple Solids." *Computer Aided Geometric Design* 2 (4): 257–79. https://doi.org/10.1016/S0167-8396(85)80002-9.

Hansmeyer, Michael, and Benjamin Dillenburger. 2014. "Printing Architecture – Castles Made of Sand." In *FABRICATE*, edited by Fabio Gramazio, Matthias Kohler, and Silke Langenberg, 92–97. Zurich.

Jones, M.W., J.A. Baerentzen, and M. Sramek. 2006. "3D Distance Fields: A Survey of Techniques and Applications." *IEEE Transactions on Visualization and Computer Graphics* 12 (4): 581–99. https://doi.org/10.1109/TVCG.2006.56.

Khudeev, R. (2005). A New Flood-Fill Algorithm for Closed Contour. In *2005 Siberian Conference on Control and Communications* (pp. 172–176). Tomsk, Russia. https://doi.org/10.1109/SIBCON.2005.1611214

Lorensen, William E., and Harvey E. Cline. 1987. "Marching Cubes: A High Resolution 3D Surface Construction Algorithm." *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques* - SIGGRAPH '87 21 (4): 163–69. https://doi.org/10.1145/37401.37422.

Michalatos, Panagiotis, and Andrew O. Payne. 2013. “Working with Multi-Scale Material Distributions.” *Acadia 2013: Adaptive Architecture*, 43–50.

Pasko, Alexander A, Valery D Adzhiev, Alexei I Sourin, and Vladimir V Savchenko. 1995. “Function Representation in Geometric Modeling: Concepts, Implementation and Applications.” *The Visual Computer*, no. 11: 429–46.

Rosenfeld, Azriel, and John L. Pfaltz. 1966. “Sequential Operations in Digital Picture Processing.” *Journal of the ACM* 13 (4): 471–94. https://doi.org/10.1145/321356.321357.

Schoen, Alan H. 1970. “Infinite Periodic Minimal Surfaces without Self-Intersections.” Cambridge, Massachusetts.

Schwarz, Hermann Amandus. 1871. *Bestimmung Einer Speziellen Minimalfläche.* Dümmler. https://books.google.ch/books?id=7HEXcgAACAAJ.

Vilbrandt, Turlif, Alexander Pasko, and Carl Vilbrandt. 2009. “Fabricating Nature.” *Technoetic Arts: A Journal of Speculative Research* 7 (2): 165–74. https://doi.org/10.1386/tear.7.2.165/1.

Advances in Architectural Geometry 2018

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

Book design
Marie Kalmnäs – Kalmnäs formgivning AB
Oscar Borgström – Layout

The Advances in Architectural Geometry (AAG) symposia are a unique forum where developments in the design, analysis and fabrication of building geometry are presented. With participation of academics and professionals, each symposium aims to gather and present practical work and theoretical research that responds to contemporary design challenges and expands the opportunities for architectural form.

The sixth edition of the AAG symposia was hosted by Chalmers University of Technology, Gothenburg, Sweden in September 2018.

This book contains the proceedings from the AAG2018 conference and offers detailed insight into current and novel geometrical developments in architecture. The 18 diverse, peer-reviewed papers present recent innovations in the fields of mathematics, computer graphics, software design, structural engineering, and the design and construction of architecture.

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

www.chalmers.se
info@chalmers.se

Klein Publishing GmbH (Ltd.)
kpv@gmx.at

ISBN 978-3-903015-13-5