Curve-Folding Polyhedra Skeletons through Smoothing

Suryansh Chandra
Zaha Hadid Architects
10 Bowling Green Lane
London, EC1R 0BQ
UK
suryansh.chandra@zaha-hadid.com

Shajay Bhooshan
Zaha Hadid Architects
10 Bowling Green Lane
London, EC1R 0BQ
UK
shajay.bhooshan@zaha-hadid.com

Mustafa El-Sayed
Zaha Hadid Architects
10 Bowling Green Lane
London, EC1R 0BQ
UK
mostafa.elsayed@zaha-hadid.com

***Figure 1**: Curve-Folded Polyhedra Skeleton made of aluminium at the AA Visiting School Bangalore 2012*

1. Introduction

The research and prototypes documented in the paper operate in the context of the exciting potential of curved-crease folding in manufacturing curved surfaces from flat sheet material. This paper describes a designer-friendly computational method to design and fabricate a class of curve-folded geometries - our proposed method generates developable surfaces that are curved-foldable, skeletal representations of a given user-defined, convex mesh.

The work draws inspiration both from the precedent works of Richard Sweeney, Ron Resch etc. as also the observed ease of physical exploration, in paper, of such geometries by students in our workshops (fig. 2). Further it also aims to address the current difficulties in recreating that ease in digital explorations. These difficulties stem both from the lack of appropriate geometric descriptions and constructive tools in ubiquitous CAD software; pointing towards the need for exploration-friendly digital methods to find and describe such geometries [Bhooshan et al. 14]. Thus, the key contributions of the paper stem from addressing these issues: An extensible, exploration-friendly digital method and the description of procedural methods to produce manufacturing data from digitally produced geometry. The paper concludes with potential avenues of architectural scale application of the method.

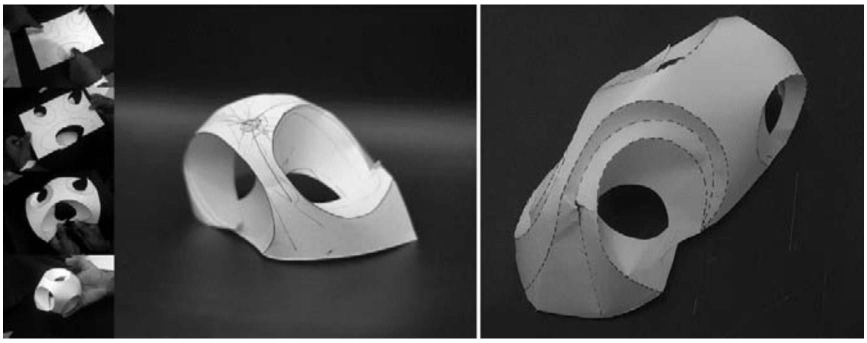

Figure 2*: Student work in our workshop*

1.1 Exploratory methods for digital geometry

Most of the precedents projects and available literature on design methods highlight the difficulty in developing an intuitive, exploratory digital-design method to generate feasible 3D geometries. Our initial survey of methods included both the iterative optimization-based method [Kilian et al. 08] and the simple constructive method – the method of reflection [Mitani and Igarashi 11]. Most methods, including the two above, presented difficulties towards incorporation within an intuitive, real-time, edit-and-observe exploratory method. Further, most of the methods aim at applicability towards a wider range of feasible geometries whereas our interest was a specific class of geometries. Lastly, they are also iterative methods that minimize one or more ‘energies’ with the aim to capture the physical behaviour of sheet-material. For an extensive overview on the precedents, and computational methods related to curved crease folding, we refer the reader to a survey [Demaine et al. 11].

Our key point of departure was to geometrically construct feasible geometry like Mitani and Igarashi as opposed to a physically-based iterative construct-and-correct solution. As such, the method benefits speedy computation and thus an edit-and-observe exploratory strategy to design. Critically however, it is applicable to a specific class of geometries, i.e. convex polyhedra. It is worth mentioning that we do implement an iterative solution to produce planarized polyhedra from non-planar meshes [Poranne et al. 13], to aid less restrictive modelling operations for the designer.

1.2 Discrete Representation for Curve Crease Folded Mesh

There are several discrete representations - *exact and inexact* - of curve-crease folded geometries. We chose to use a representation based on planar-quad meshes (PQ mesh) that additionally incorporate *developability* constraints [Kilian et al. 12] (). For a comprehensive list of representations, we refer to [Solomon et al. 12]. Given the representation, we used various mesh operations to derive a predominantly quad-faced mesh from the Form-found-mesh.

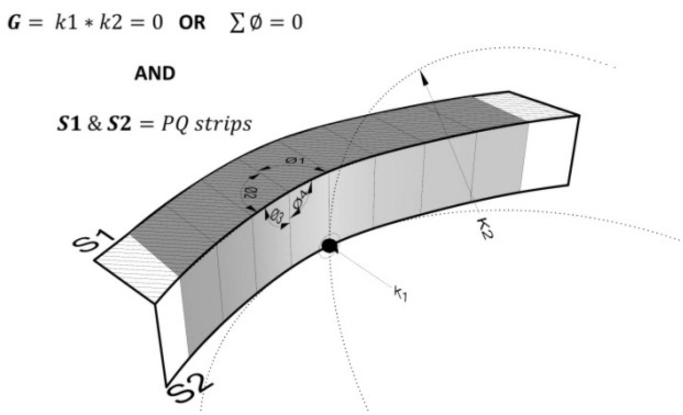

Figure 3: *Essential requirements of a discrete representation*

2. Method

Our proposed method takes a convex, user-defined, input mesh (*Base mesh*) and forces its faces towards planarity to produce a polyhedron (section 2.1). This polyhedron is subsequently used to generate developable surfaces that are its curved-foldable, skeletal representation. This process has two procedural steps - the *designed surface* (section 2.2), and the *derived surface* (section 2.3). Lastly, the resultant geometry is used to produce manufacturing information such as cut patterns, and their assemblies (section 2.4).

The designed surface (white surfaces in fig. 4) forms the base structure of the curved folded polyhedron (CFP) and is shaped through parameters external to the method, thereby allowing elaborate interactive control further to the shape of the base mesh itself. The derived surface forms the inward extrusions from the designed surface (dark gray surfaces in fig. 4), and is computed through an adaptation of the reflection method [Mitani and Igarashi 11] that is capable of generating zero Gaussian curvature surfaces. It is worth noting that by generating the derived surface in this manner, there is no explicit computing of rulings thus benefiting computing time.

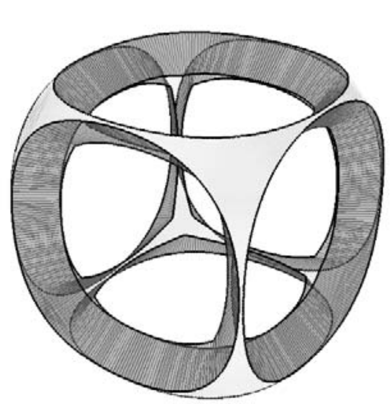

Figure 4*: Designed surface in white, derived surface in dark gray*

2.1 Planarization of Base Polyhedron

Curved folded geometry is still a very challenging domain and it is well acknowledged that planar curved folds are significantly simpler to solve [Mitani and Igarashi 11]. Further, several iterations of paper models of CFPs demonstrated that the creases remained nearly planar unless they were forced out of plane by external forces. Therefore, we run an iterative planarization routine similar to the one proposed by [Poranne et al. 13] on the faces of the Base Mesh, as all curved creases in our proposed method lie on the faces of the polyhedron.

2.2 Designed surface and parametric variations

Our method generates the design surface by the applying the following series of operations on a convex polyhedron (fig. 5):

1. Chamfer Conway operator [Hart 98] applied to all vertices
2. Bevel Conway operator [Hart 98] applied to all edges
3. Cubic Bezier curves extracted from faces of polyhedron
4. Planar polygonal surfaces generated at each vertex
5. Translational surfaces generated at each edge

In the above sequence, operations 1-3 permit a significant range of design variations and define the curves that form the edges of the designed surface. Operations 4 and 5 are surfacing operations, replacing each vertex of the original polyhedron with a planar polygonal surface and each edge with a translational surface. These translation surfaces have no Gaussian curvature by virtue of the planar faces of the initial polyhedron.

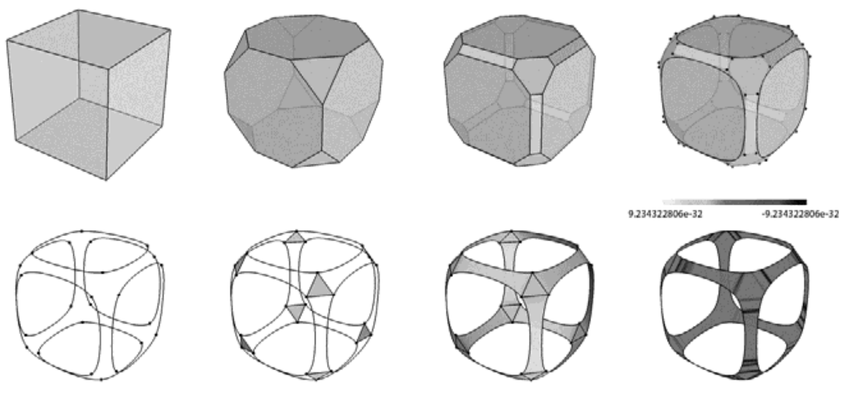

***Figure 5**: Geometric operations to extract designed surface from polyhedron*

2.3 Derived surface

Our early attempts to compute the derived surface involved using the method of reflection [Mitani and Igarashi 11], but we had limited success with it as the reflected surfaces tend to intersect each other (fig. 6). Subsequently, we used an energy minimization method iteratively converging towards zero-Gaussian curvature geometries that we describe in [Bhooshan et al. 14]. Eventually, we found a non-iterative, closed form method specific to the geometry of CFPs that is significantly faster to compute over the energy minimization method. This is the method described in this paper.

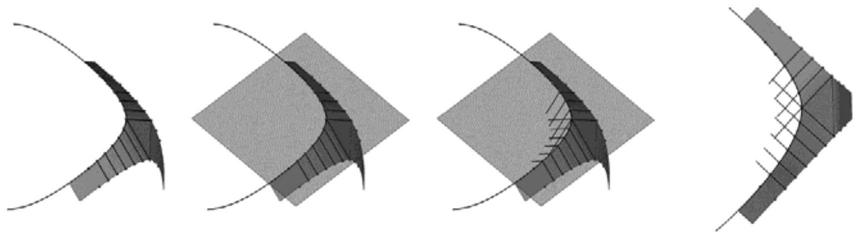

***Figure 6**: The method of reflection produces intersecting surfaces*

Our proposed method operates on a discretized mesh representation of the designed surface, and computes an extrusion vector at each vertex such that the sum of angles projected by the existing faces and the new faces sums up to 360 degrees. In fig. 7, consider P, A and B to be border vertices and the line PC to be an interior edge on the designed surface. A solution set {Q} exists such that every vertex Q_i:

$\angle APC + \angle BPC + \angle APQ + \angle BPQ = 2\pi$

Or:

$$\theta_1 + \theta_2 + \theta_x + \theta_y = 2\pi$$

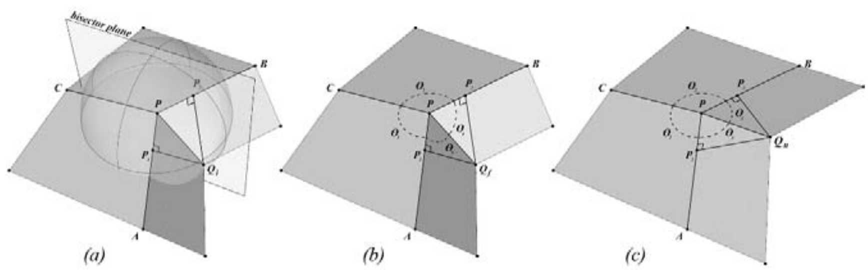

***Figure 7**: Calculating the derived surface*

As **{Q}** contains an infinite number of solutions, we constraint our solution space to the bisector plane of angle ∠APB (fig. 7a) and its intersection with a unit sphere centered at P, which limits the number of solutions to two: the folded-state with vertex Q_f (fig. 7b), and a the unfolded-state with vertex Q_u (fig. 7c). We could also make assumption of other linear and polynomial relations between θ_x and θ_y, which will result in higher order roots for Q_i. If no assumption is made regarding the relation between the two angles, the solution space **{Q}** would be the intersection of a quartic surface and the unit sphere centred at P. The formulation noted in [Kilian et al. 08] would imply a similar result.

If P_1 and P_2 are points on edges PA and PB respectively nearest to Q_i, and d_1 & d_2 are the distances from P to P_1 and P_2 respectively we can state that:

$$d_1 = d_2 = \sqrt{(Cos(2\pi - \theta_1 - \theta_2) + 1) * 0.5}$$

Thus the solution can be understood as the intersection of a unit sphere centred at P, with a line connecting Q_u and Q_f. Further, this line is the intersection of two planes, one centred at P1 with PA as its normal and the other centred at P2 and PB as normal. This results in a system with quadratic roots. This makes the length of the extrusion vector the only designer controllable parameter in the derived surface.

$\boldsymbol{Q_u} = \boldsymbol{P_0} + \lambda_u \boldsymbol{V}$ and $\boldsymbol{Q_f} = \boldsymbol{P_0} + \lambda_f \boldsymbol{V}$

Where:

$$\lambda_u = \frac{-b - \sqrt{b^2 - 4ac}}{2a} \quad \text{and} \quad \lambda_f = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$\boldsymbol{V} = \boldsymbol{PA} \times \boldsymbol{PB}$

$a = \boldsymbol{V} \cdot \boldsymbol{V}$

$b = 2(\boldsymbol{V} \cdot \boldsymbol{P})$

$c = (\boldsymbol{P_0} \cdot \boldsymbol{P_0} - 1)$

$\boldsymbol{P_0} = < x_0, y_0, 0 > \quad x_0 = \frac{d_1 y_2 - y_1 d_2}{x_1 y_2 - y_1 x_2} \quad y_0 = \frac{x_1 d_2 - d_1 x_2}{x_1 y_2 - y_1 x_2}$

$$d_1 = d_2 = \sqrt{(Cos(2\pi - \theta_1 - \theta_2) + 1) * 0.5}$$

In the equations above, bold face indicates vectors, × is cross product and · is dot product of two vectors.

2.4 Manufacturing Information

The resultant geometry is used to generate manufacturing information, in the form of unfolded CAD/ CAM data.

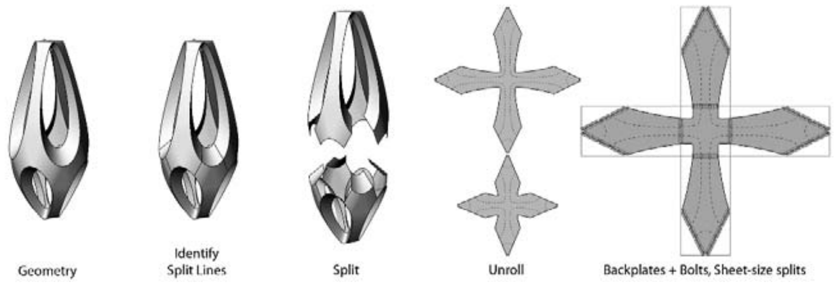

***Figure 8**: Process of CAD/CAM info generation*

3. Results & Discussion

The resulting geometry is a Zero-Gaussian curvature mesh. The method currently does not solve the mesh faces for planarity. For the purpose of architectural installations in paper and sheet aluminium, we have found the resulting geometry to be within acceptable tolerance for fabrication, as further illustrated by the reasonably precise edge alignment in a cluster of 1 metre tall polyhedra in (fig. 9) made from 1.5mm card paper.

Figure 9: *Precise edge alignment in clusters of CFPs*

Unfolding a resultant quad mesh produces unfold error due to the non-planarity of mesh faces. However, since we were working with a discrete representation of a curve-foldable mesh that relied on a PQ mesh with a develop-ability constraint, we addressed the planarity issue in the 2d unfolded pattern. This means that the physical object will not match the digital 3d model. However, as the sculpture was a free-standing object and due to constraints of time this issue was not addressed. In order to quantify the error in the geometry, we digitally unfolded different mesh resolution variants of the same base polyhedron (table 1). We observed that lower mesh resolution computed to significantly higher accuracy.

	DISCRETIZATION LEVEL 1	DISCRETIZATION LEVEL 2	DISCRETIZATION LEVEL 3	DISCRETIZATION LEVEL 4
				
AVG EDGE LENGTH	15.0	11.6	8.8	6.8
MINIMUM ERROR	0.076	0.093	0.124	0.129
MAXIMUM ERROR	0.137	0.164	1.05	1.54

Table 1: *Key results from unfold error measurements*

4. Conclusion

The paper described a method of generating a class of curve folded geometry – smoothed skeleton of convex polyhedra, as also producing the necessary fabrication information from the generated geometries. Further, this paper also noted the intuitive nature of using polyhedral meshes and their manipulation using ubiquitous mesh modelling tools. This then enables speedy exploration of variations and the edit-and-observe strategies of designers (fig. 10).

Figure 10: *Parametric variations of the designed and derived surface computed from the same base polyhedron*

Apart from the obvious benefit of using sheet material to produce curved surfaces, we also noted the fabrication benefits of such geometry in general. They ease the on-site description of geometry and assembly of parts due to their capacity to be formed with minimal effort. In our workshops, we have been able to build 2 metre tall CFPs from sheet aluminium with teams of 10-15 students within a matter of 6-8 hours (fig. 11).

Figure 11: *2m long aluminium CFPs folded and assembled in 6 hours by a group of 15 students*

Based on the prototypes and research thus far, we can envision the use of the method in the production of architectural scale assembly of skeletal geometries. Such folded geometries, especially from smooth sheet material such as aluminium or plastic, could also be beneficial in their use as false-work or lost form-work to cast concrete and other hardening structural materials [Larsen et al. 13].

5. Future work

A significant aspect of curve-folded geometries in general is that, barring a few exceptions [Rohim et al. 13] not much is known in regards to their structural behaviour. Some of difficulties of using conventional FEM methods on such geometries stems from complexities in computationally describing features such as removal of material along crease lines, tendencies of spring-back in the folded shape, etc. Thus, an exciting and critical aspect of future work that we envision is the development of discrete (i.e. bar-and-node) form-finding methods that also incorporate structural parameters alongside the exploratory design and fabrication parameters that we currently investigate.

6. Acknowledgement

We would like to thank Zaha Hadid Architects and the AA School of Architecture for providing platforms to conduct this research. Further, we would like to thank AA Visiting School Bangalore teams from 2012 and 2013, and the Fragile Beasts 2014 workshop team from Lodz University of Technology for helping with the design and execution of the installations.

7. References

[Bhooshan et al. 14] Shajay Bhooshan, Mustafa El-Sayed and Suryansh Chandra. "Design-friendly strategies for computational form-finding of curved-folded geometries: a case study". In *Proceedings of the Symposium on Simulation for Architecture & Urban Design,* SimAUD 2014, edited by Dr. David Gerber and Rhys Goldstein, pp. 117-124. San Diego, CA, USA: Society for Computer Simulation International, 2014.

[Demaine et al. 11] Eric Demaine, Martin Demaine, Duks Koschitz, and Tomohiro Tachi. "Curved crease folding: a review on art, design and mathematics." In *Proceedings of the IABSE-IASS Symposium: Taller, Longer, Lighter (IABSE-IASS2011).* pp. 20-23. London, England, UK: 2011.

[Hart 98] "Conway Notation for Polyhedra". In http://www.georgehart.com/virtual-polyhedra/conway_notation.html [retrieved June 10, 2014]

[Kilian et al. 08] Martin Kilian, Simon Flöry, Zhonggui Chen, Niloy J. Mitra, Alla Sheffer, and Helmut Pottmann. "Curved folding". In *ACM SIGGRAPH 2008 Papers*, pp. 75-83. New York, NY, USA: ACM, 2008.

[Larsen et al. 13] Niels Martin Larsen, Ole Egholm Pedersen, and David Pigram. "Realisation of complex precast concrete structures through the integration of algorithmic design and novel fabrication techniques". In http://aarch.dk/fileadmin/filer/webmaster/PDF/Realisation_of_complex_precast_concrete_structures.pdf [retrieved Oct 10, 2014]

[Mitani and Igarashi 11] Jun Mitani and Takeo Igarashi. "Interactive Design of Planar Curved Folding by Reflection." In *Proceedings of the Pacific Conference on Computer Graphics and Applications,* Pacific Graphics (2011) edited by Bing-Yu Chen, Jan Kautz, Tong-Yee Lee, and Ming C. Lin, pp. 77-81. Kaohsiung, Taiwan: The Eurographics Association, 2011.

[Rohim et al. 13] Rohamezan Rohim, Kok Keong Choong and Thavaraj Jeevaratnam. "Structural Behaviour of Origami Inspired Folded Shell Surface". In *World Applied Sciences Journal 24(4)*. pp. 497-502. 2013.

[Poranne et al. 13] Roi Poranne , Elena Ovreiu, and Craig Gotsman. "Interactive planarization and optimization of 3D meshes." In *Computer Graphics Forum*, vol. 32, no. 1, pp. 152-163. Blackwell Publishing Ltd, 2013.

[Solomon et al. 12] Justin Solomon, Etienne Vouga, Max Wardetzky, and Eitan Grinspun. "Flexible developable surfaces." In *Computer Graphics Forum*, vol. 31, no. 5, pp. 1567-1576. Blackwell Publishing Ltd, 2012.