Proceedings of the IASS 2024 Symposium
Redefining the Art of Structural Design
August 26-30, 2024, Zurich, Switzerland
Philippe Block, Giulia Boller, Catherine DeWolf,
Jacqueline Pauli, Walter Kaufmann (eds.)

Free-form surface approximation using congruent regular triangles

Yuanpeng LIU[a], Ting-Uei LEE[a], Yi Min XIE[*, a]

[*, a] Centre for Innovative Structures and Materials, School of Engineering
RMIT University, Melbourne, 3001, Australia
mike.xie@rmit.edu.au

Abstract

Free-form surfaces are increasingly used in architectural designs for their striking aesthetic appeal. While they can be assembled from prefabricated simple polygonal panels, this process is typically challenging and costly due to the need for many different panel shapes. To address this, we present an innovative design method that can approximate free-form surfaces using a single type of regular (or near-regular) triangle. The method starts with an initial triangle and gradually 'grows' new ones until the target surface is entirely covered. During the growth process, the method strategically assigns the locally optimal valence to each vertex to accommodate the surface curvature. Vertex positions are dynamically updated to keep triangles as equilateral as possible and aligned with the target surface, through an efficient localized optimization process. Tested on multiple numerical examples, the method can solve both complex organic shapes and architectural surfaces with open boundaries. The resulting mesh features seamlessly connected, near-regular triangles and smooth patterns with a small number of singularities. Via best-fit matching, each triangle can be replaced by a standard regular triangle (or the mean shape) to generate exactly congruent panels with small gaps, potentially enabling realistic manufacturing and assembly.

Keywords: architectural geometry, rationalization, mesh topology, optimization, façade design, free-form surface.

1. Introduction

Free-form surfaces create 3D shapes that possess innovative and elegant appearances. In recent years, the adoption of free-form surfaces has been rapidly increasing in architectural designs due to the development of computer-aided design technologies [1]. These surfaces, however, are inherently complex due to the double curvature, which often poses great difficulty for real-world construction. To address this, significant effort has been focused on the problem of architectural rationalization, aiming to make the construction of complex surfaces feasible and affordable [2, 3].

One major rationalization strategy is to reduce the shape variety of building components, such as beams, nodes, and panels, to reduce mold customization and facilitate production. Regarding beams, several methods have been proposed in different contexts, including architectural structures [4, 5, 6] and sphere tessellations [7, 8]. As for nodes, methods include replacing low-valence nodes with fewer types of high-valence ones [5, 6, 9], or merging different nodes by similarizing their respective shapes [10, 11]. In terms of panels, several optimization methods have been proposed to reduce the number of different molds for panel fabrication, without needing the panel shapes to be exactly the same [12, 13]. Other methods have instead explored approximating free-form surfaces using congruent groups of triangles [6, 14, 15], quadrilaterals [16], or general n-gons [17]. This study, however, focuses on the extreme case of approximating free-form surfaces using only one type of near-regular triangle. All panels being

congruent could significantly benefit fabrication. Triangular panels with regular-like shapes are also visually pleasing and desirable for architectural applications.

Few studies in architecture have attempted to design free-form surfaces using equivalent regular triangles [18, 19]. As observed in [18], the use of regular triangles reveals certain relationships between vertex valence (the number of edges meeting at a vertex) and local surface curvature: vertices with valence 5 ($V5$) tend to form convex areas, those with valence 6 ($V6$) result in flat or developable surfaces, and vertices with valence 7 ($V7$) lead to concave regions. By manipulating the number and the positions of singularities (vertices with valence different from 6), it is possible to create elegant architectural surfaces using equivalent regular triangles [18, 19]. However, existing studies mostly rely on a trial-and-error approach, which could be inefficient and even impossible for the design of highly complex geometries. An automatic approach that approximates general free-form surfaces using equivalent regular triangles remains to be developed.

In computer graphics, triangular remeshing techniques [20] share a similar goal with us, aimed at refining input meshes for various downstream applications. Particularly relevant is a method focusing on achieving equilateral triangles [21]. Given an input mesh and a target length, this method gradually splits long edges and collapses short edges to create an isotropic mesh composed of near-equilateral (i.e., near-regular) triangles. However, the resulting mesh often exhibits numerous singularities with unsmooth visual patterns, which may not be directly used for architectural applications. One may further reduce the number of singularities through a series of topological operations [22, 23], yet the resulting triangles would typically become less uniform. An approach that simultaneously achieves near-regular triangles and smooth mesh patterns with a small amount of singularities remains missing for architectural applications.

This study presents a novel mesh-growth algorithm for approximating free-form surfaces using a single type of regular (or near-regular) triangle. Given a target surface, this algorithm gradually appends new triangles to the front of the existing mesh until the entire surface is covered (see Figure 1). Specifically, faces are gradually added around a chosen vertex to achieve a locally optimal valence, guided by the local geometry. Vertex positions are dynamically updated to keep triangles as equilateral as possible and aligned with the target surface, through an efficient localized optimization process. Specific considerations are included to manage potential self-intersection of the front and ensure proper alignment with the surface boundaries, as detailed in Section 2. Various numerical examples are presented in Section 3 to demonstrate the effectiveness of our method. The slightly different triangles in the obtained mesh can

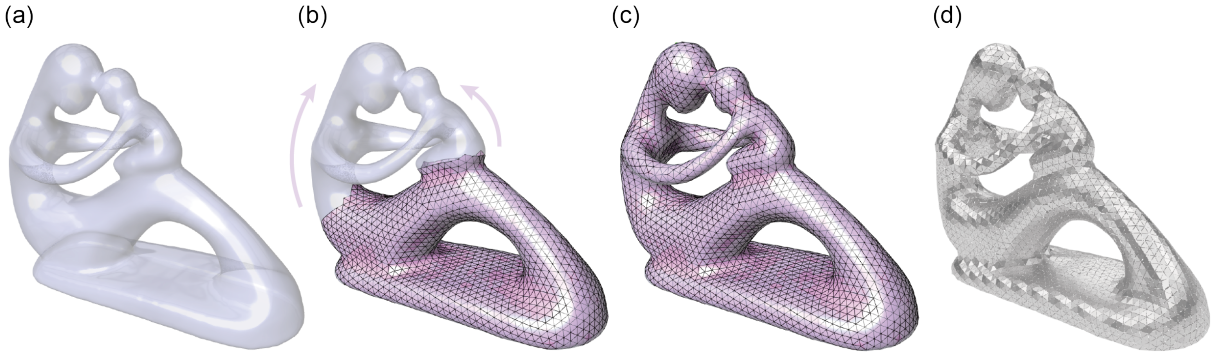

Figure 1: Overview of the mesh-growth algorithm. Given a target surface (a), the proposed algorithm gradually adds new faces to the front of the existing mesh (b), until covering the entire surface (c). Faces of the mesh may be further replaced by the mean shape, creating a discrete form composed of exactly congruent faces to enable realistic fabrication (d).

each be replaced by a standard regular triangle (or the mean shape) to create exactly congruent panels with small gaps for real construction (see Figure 1(d)). The final conclusions are given in Section 4.

2. Methodology

2.1. Overview

The overall framework of our method is a mesh growth process. The core idea is to gradually grow new triangles along a target surface while explicitly assigning the optimal valence for each vertex being processed. Essentially, our algorithm is a special type of *advancing-front method*, with two novel features, local topological operations and localized geometry optimization, specifically tailored for regular triangles, as further detailed in Sections 2.2 and 2.3, respectively. The general process of the algorithm is summarized below:

(1) Input a target surface, a target length for edges (denoted by $l^\star$), and an initial triangle as the growth starting point.

(2) Pick up one vertex at the front and perform corresponding topological operations depending on the local geometry. If there is no valid vertex, stop the iteration.

(3) Perform localized geometry optimization to update vertex positions towards the specified geometric properties.

(4) Return to step 2.

2.2. Local topological operation

For each iteration, we first need to find a position (i.e., a vertex) for processing. Inspired by the work of Zimmer et al. [9], we select the vertex with the largest valence at the front. This is to prioritize the treatment of concave region, which could help improve the robustness of the algorithm. Moreover, if the selected vertex is close enough to the boundary of the target (open) surface (with the closest distance smaller than $0.5l^\star$ for the default setting), we omit it and search for the next.

At the selected vertex, we grow new triangles around it until reaching the locally optimal valence. The main goal here is to snap the valence of the current vertex to the closest integer of the sum of the face angles at that vertex divided by $60°$. This can be achieved through three topological operations, including *merge*, *connect*, and *add* based on the local geometry, as shown in Figure 2. Specifically, if the remaining angle $\theta \leq 30°$, we merge the two end points. If $30° < \theta \leq 90°$, we connect the points with an edge and fill in a new triangle. If $\theta > 90°$, we add a new regular triangle connected to one adjacent edge of this

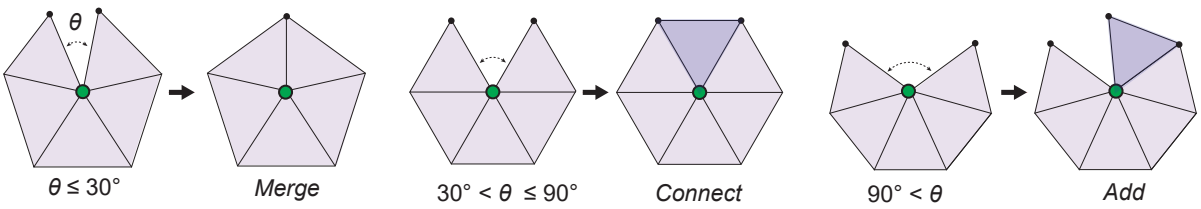

Figure 2: Local topological operations. The selected vertex for processing is denoted in green, and the involved vertices in each operation (for the initialization of the subsequent optimization) are denoted in black.

vertex, orthogonal to the vertex normal direction. In addition, we try to avoid the appearance of vertices with $V < 5$ or $V > 7$, due to their undesired visual effects. Thus, when $V < 5$, we simply *add*, and if $V > 7$, we *merge*. In addition, to manage the potential front meeting, if two non-adjacent vertices at the front are close enough to each other (distance smaller than $0.5l^\star$), these two vertices will first be merged to avoid mesh overlapping.

2.3. Localized geometry optimization

After each topological operation, an optimization process is followed to update vertex positions for keeping triangles as equilateral as possible and aligned with the target surface. The corresponding objective function is given in Equation (1), where E_i denotes each edge length, and $V^\star$ represents the closest point for each vertex V_i on the target surface. Note that for vertices positioned within a distance less than $0.5l^\star$ from the target surface's boundary or corners, their designated target points are specifically chosen from these boundary or corner to maintain the geometric features.

$$F = \sum (E_i - l^\star)^2 + \sum (V_i - V^\star)^2 \tag{1}$$

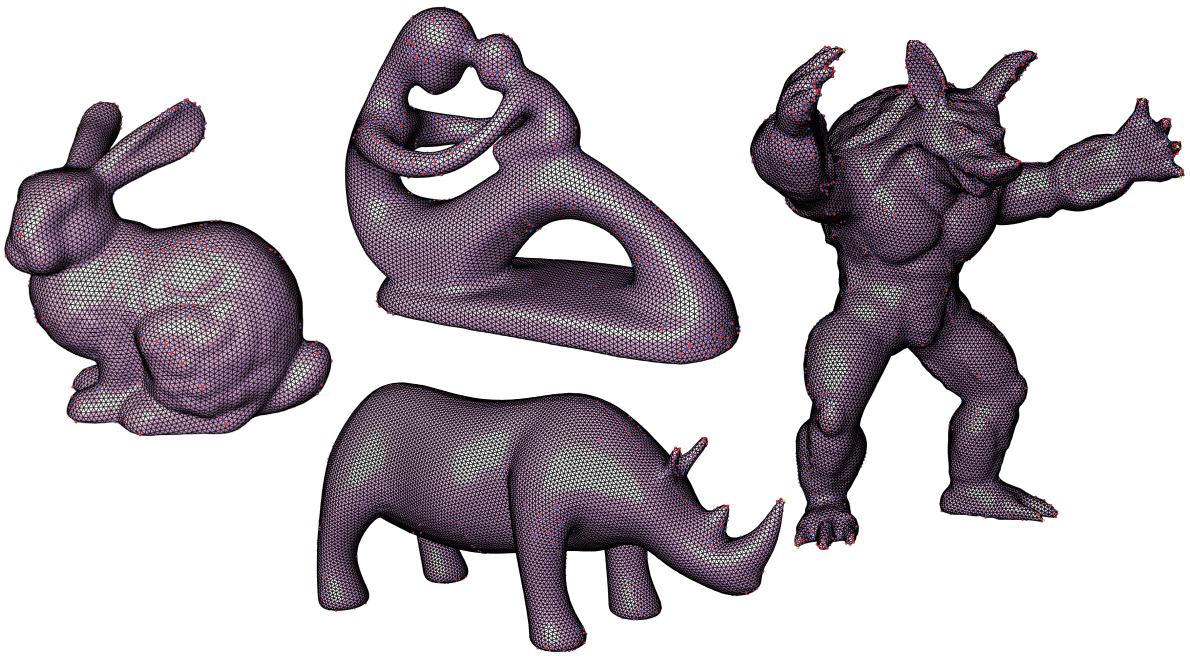

Figure 3: Test results of complex organic shapes with intricate details.

During the mesh growth process, as the mesh is increasingly expanding, directly optimizing the full mesh at each iteration would drag the whole algorithm highly inefficient. Since the error is mostly localized at the vertices involved in the topological operation (see Figure 2), only updating a small region near these vertices would be adequate. In this regard, inspired by Bommes et al. [24], we develop a localized optimization strategy, as detailed below:

(1) *Initialization*. Input a list of the vertices initially influenced by the topological operations (see Figure 2).

(2) *Local optimization*. Perform local optimization only on the vertices involved in the list.

(3) *List update*. Calculate the displacement of each vertex in the list. If the displacement is greater

than a certain threshold (e.g., 1e-3$l^\star$ by default), add the neighbors of this vertex to the list; otherwise, remove this vertex from the list.

(4) *Convergence check.* If the list is empty, end the iteration; otherwise, return to step 2.

3. Results and discussion

Various geometries are tested using our method, as illustrated in Figures 3 and 4. The results in Figure 3 demonstrate that our method can handle highly complex organic shapes with intricate details. Yet, to capture these local features, a reasonably small target length is required. Meanwhile, Figure 4 highlights the capacity of our method in addressing open free-form architectural surfaces while preserving the features of boundaries and corners. Nevertheless, triangles close to boundaries are typically less regular due to the imposed additional constraints. Remarkably, all numerical tests are finished within a few seconds, attributed to the localized nature of the geometry optimization. The resulting meshes feature near-regular triangles and smooth patterns with a small number of singularities, owing to the proper selection of the vertex valence. Yet, chains of singularities may appear in areas where fronts meet, potentially creating locally unsmooth patterns.

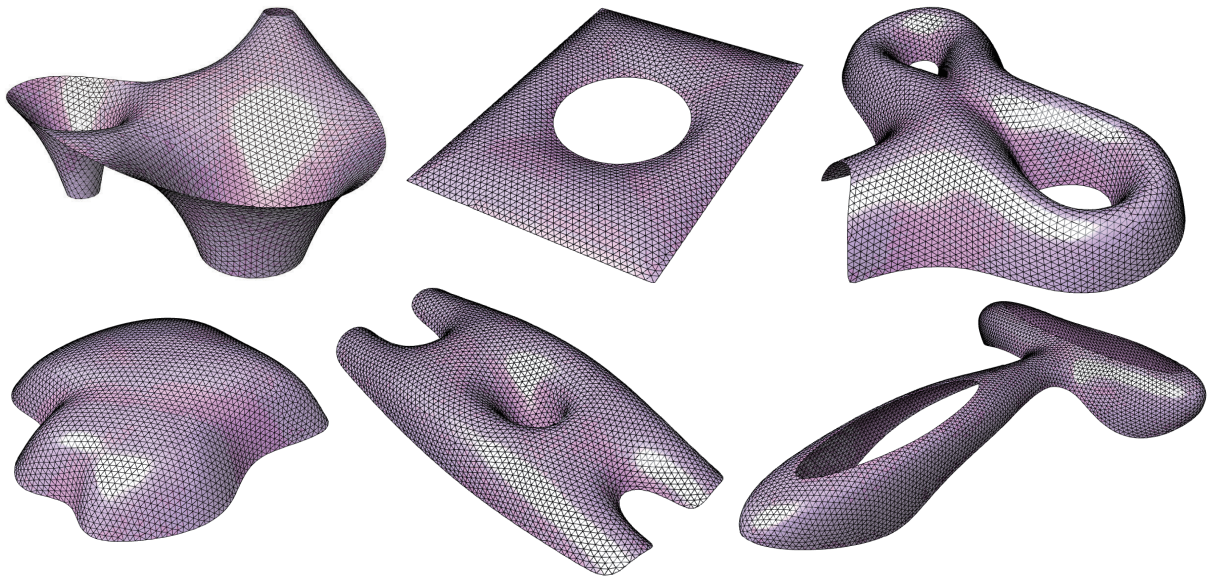

Figure 4: Test results of architectural surfaces with open boundaries.

In the resulting meshes, all faces are near-regular, seamlessly connected, but slightly differ in their shapes. For fabrication, each triangle may be replaced by either the mean shape, or a standard regular triangle, via best-fit mapping [17, 25], as shown in Figure 5. The first case creates less errors, hence more uniform gaps, while the latter case would allow the use of the same regular triangles for the fabrication of multiple different structures.

Note that within our algorithm, the final error is fixed based on the given inputs. Consequently, our method does not guarantee to satisfy any given error threshold. For projects where smaller errors are required, further processing would be necessary, such as increasing the group number [17]. Since the meshes created using our algorithm already contain faces with similar shapes, they could serve as good initialization, potentially lowering the final group number of different faces needed to meet the given error threshold.

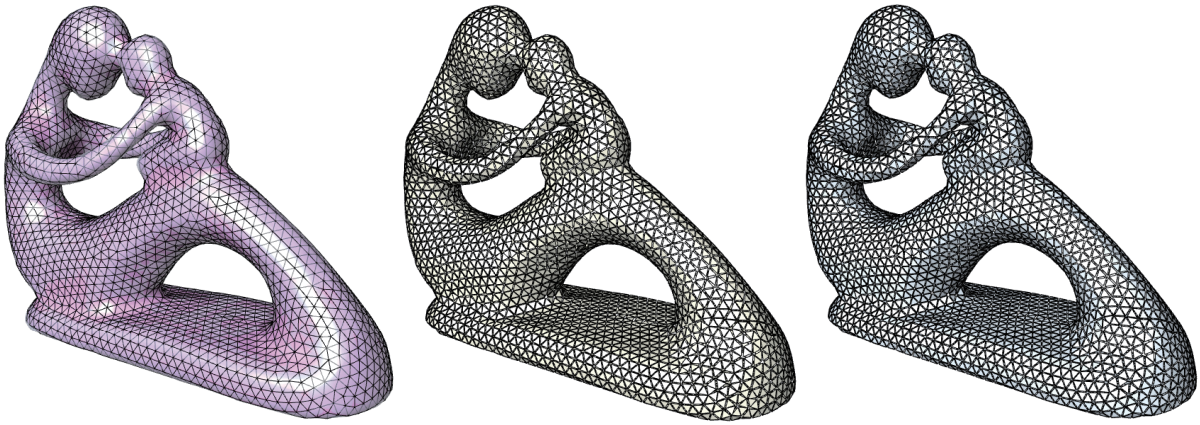

Figure 5: With the obtained mesh (Left), each face can be replaced by (Middle) a regular triangle or (Right) the mean shape, creating exactly congruent faces with small gaps to enable realistic fabrication and assembly.

4. Conclusion

This study presents a mesh-growth method to approximate free-from surfaces using a single type of regular (or near-regular) triangle. Specifically, we propose a novel strategy of explicitly assigning the locally optimal valence for each vertex during the growth process via defined topological operations. Also, we introduce an efficient localized optimization strategy to dynamically update vertex positions to keep triangles as equilateral as possible and aligned with the target surface. Various numerical examples are presented to demonstrate the effectiveness of the method, including complex organic shapes and architectural surfaces with open boundaries. All cases are completed within a few seconds, and the generated meshes feature near-regular triangles and smooth patterns with a small number of singularities. By replacing each face with a standard regular triangle (or the mean shape), we can create exactly congruent panels with small gaps that may enable realistic fabrication and assembly. For future research, considering the alignment with an underlying vector field would be a promising approach to further enhance the outcome.

Acknowledgments

This work was supported by the Australian Research Council (FL190100014 and DP200102190).

References

[1] S. Adriaenssens, P. Block, D. Veenendaal, and C. Williams, *Shell structures for architecture: form finding and optimization*. Routledge, 2014.

[2] H. Pottmann, M. Eigensatz, A. Vaxman, and J. Wallner, "Architectural geometry," *Computers and Graphics*, vol. 47, pp. 145–164, Apr. 2015.

[3] G. Austern, I. G. Capeluto, and Y. J. Grobman, "Rationalization methods in computer aided fabrication: A critical review," *Automation in Construction*, vol. 90, pp. 281–293, Jun. 2018.

[4] H. Lu and Y. M. Xie, "Reducing the number of different members in truss layout optimization," *Structural and Multidisciplinary Optimization*, vol. 66, no. 3, p. 52, 2023.

[5] J. Brütting, G. Senatore, and C. Fivet, "Design and fabrication of a reusable kit of parts for diverse structures," *Automation in Construction*, vol. 125, pp. 103614, May 2021.

[6] M. Bi, Y. Liu, T. Xu, Y. He, J. Ma, Z. Zhuang, and Y. M. Xie, "Clustering and optimization of nodes, beams and panels for cost-effective fabrication of free-form surfaces," *Engineering Structures*, vol. 307, p. 117 912, 2024.

[7] Y. Liu, T.-U. Lee, A. Rezaee Javan, and Y. M. Xie, "Extending Goldberg's method to parametrize and control the geometry of Goldberg polyhedra," *Royal Society Open Science*, vol. 9, no. 8, pp. 220675, 2022.

[8] T.-U. Lee, Y. Liu, and Y. M. Xie, "Dividing a sphere hierarchically into a large number of spherical pentagons using equal area or equal length optimization," *Computer-Aided Design*, vol. 148, pp. 103259, Jul. 2022.

[9] H. Zimmer and L. Kobbelt, "Zometool rationalization of freeform surfaces," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 10, pp. 1461–1473, 2014.

[10] A. Koronaki, P. Shepherd, and M. Evernden, "Rationalization of freeform space-frame structures: Reducing variability in the joints," *International Journal of Architectural Computing*, vol. 18, no. 1, pp. 84–99, 2020.

[11] Y. Liu, T.-U. Lee, A. Koronaki, N. Pietroni, and Y. M. Xie, "Reducing the number of different nodes in space frame structures through clustering and optimization," *Engineering Structures*, vol. 284, p. 116 016, 2023.

[12] M. Eigensatz, M. Kilian, A. Schiftner, N. J. Mitra, H. Pottmann, and M. Pauly, "Paneling architectural freeform surfaces," *ACM Transactions on Graphics*, vol. 29, no. 4, pp. 1–10, Jul. 2010.

[13] D. Pellis, M. Kilian, H. Pottmann, and M. Pauly, "Computational design of weingarten surfaces," *ACM Transactions on Graphics*, vol. 40, no. 4, Jul. 2021.

[14] M. Singh and S. Schaefer, "Triangle surfaces with discrete equivalence classes," *ACM Transactions on Graphics*, vol. 29, no. 4, pp. 1–7, Jul. 2010.

[15] Z. Liu, Z. Zhang, D. Zhang, C. Ye, L. Liu, and X.-M. Fu, "Modeling and fabrication with specified discrete equivalence classes," *ACM Transactions on Graphics*, vol. 40, no. 4, pp. 1–12, Jul. 2021.

[16] C.-W. Fu, C.-F. Lai, Y. He, and D. Cohen-Or, "K-set tilable surfaces," *ACM Transactions on Graphics*, vol. 29, no. 4, pp. 1–6, Jul. 2010.

[17] Y. Liu, T.-U. Lee, A. Rezaee Javan, N. Pietroni, and Y. M. Xie, "Reducing the number of different faces in free-form surface approximations through clustering and optimization," *Computer-Aided Design*, vol. 166, p. 103 633, 2024.

[18] M. Huard, M. Eigensatz, and P. Bompas, "Planar panelization with extreme repetition," in *Advances in Architectural Geometry 2014*, Cham, Switzerland: September 2014, London, England. Springer, 2015, pp. 259–279.

[19] A. Lobel, *Lobel Frame*. 2005, accessed on 2024-03-30. [Online]. Available: `https://www.equilatere.net`.

[20] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy, *Polygon mesh processing*. CRC Press, New York., 2010, ISBN 978-1568814261.

[21] M. Botsch and L. Kobbelt, "A remeshing approach to multiresolution modeling," in *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, ser. SGP '04, Nice, France: Association for Computing Machinery, 2004, pp. 185–192.

[22] Y. Li, E. Zhang, Y. Kobayashi, and P. Wonka, "Editing operations for irregular vertices in triangle meshes," *ACM Transactions on Graphics*, vol. 29, no. 6, Dec. 2010.

[23] V. Surazhsky and C. Gotsman, "Explicit Surface Remeshing," in *Eurographics Symposium on Geometry Processing*, L. Kobbelt, P. Schroeder, and H. Hoppe, Eds., The Eurographics Association, 2003, ISBN: 3-905673-06-1.

[24] D. Bommes, H. Zimmer, and L. Kobbelt, "Mixed-integer quadrangulation," *ACM Transactions on Graphics*, vol. 28, no. 3, Jul. 2009.

[25] O. Sorkine-Hornung and M. Rabinovich, *Least-squares rigid motion using SVD*. 2017, accessed on 2023-02-17. [Online]. Available: `https://igl.ethz.ch/projects/ARAP/svd_rot.pdf`.