Connectivity Editing for Quadrilateral Meshes

Chi-Han Peng[*]
Arizona State University

Eugene Zhang[†]
Oregon State University

Yoshihiro Kobayashi[‡]
Arizona State University

Peter Wonka[§]
Arizona State University /
KAUST

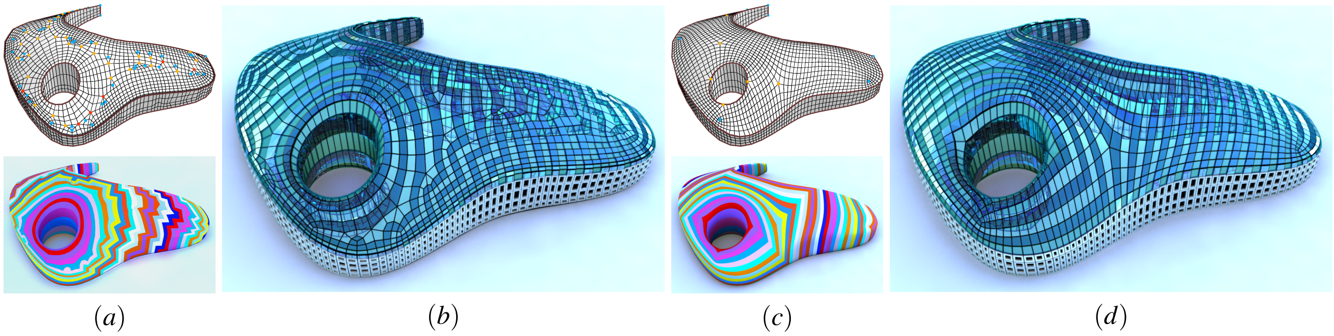


Figure 1: *We introduce connectivity editing operations to control irregular vertices in quadrilateral meshes. This can lead to improved results in the design of a glass structure: (a) top: the original mesh with irregular vertices as colored dots, (a) bottom: a stripe pattern applied to the mesh, (b) a rendering of the design as glass construction. In (c) and (d) we show the edited mesh. The glass panels on the roof are generated from the edges in the meshes.*

Abstract

We propose new connectivity editing operations for quadrilateral meshes with the unique ability to explicitly control the location, orientation, type, and number of the irregular vertices (valence not equal to four) in the mesh while preserving sharp edges. We provide theoretical analysis on what editing operations are possible and impossible and introduce three *fundamental* operations to move and re-orient a pair of irregular vertices. We argue that our editing operations are fundamental, because they only change the quad mesh in the smallest possible region and involve the fewest irregular vertices (i.e., two). The irregular vertex movement operations are supplemented by operations for the splitting, merging, canceling, and aligning of irregular vertices. We explain how the proposed high-level operations are realized through graph-level editing operations such as quad collapses, edge flips, and edge splits. The utility of these mesh editing operations are demonstrated by improving the connectivity of quad meshes generated from state-of-art quadrangulation techniques.

Keywords: quadrilateral mesh editing, irregular vertex editing, mesh optimization, mesh-based design, topology, geometry processing

Links: DL PDF WEB

[*]e-mail:pchihan@asu.edu
[†]e-mail:zhange@eecs.oregonstate.edu
[‡]e-mail:ykobaya@asu.edu
[§]e-mail:peter.wonka@asu.edu

ACM Reference Format
Peng, C., Zhang, E., Kobayashi, Y., Wonka, P. 2011. Connectivity Editing for Quadrilateral Meshes. *ACM Trans. Graph. 30*, 6, Article 141 (December 2011), 12 pages. DOI = 10.1145/2024156.2024175 http://doi.acm.org/10.1145/2024156.2024175.

1 Introduction

Quadrilateral and hexahedral meshes are popular choices in simulation and shape modeling due to the natural tensor product property that they possess. Quadrilateral meshes can also facilitate architectural modeling as well as texture and geometry synthesis. Important aspects of a quadrilateral mesh include the location, orientation, type, and number of irregular vertices. While there has been some work in quad mesh connectivity editing [Daniels et al. 2008; Bommes et al. 2011], achieving irregular vertex control is challenging and many questions about what editing operations are possible and impossible still need to be answered.

In this paper, we propose three operations that move an irregular vertex pair (two valence 3, two valence 5, or one valence 3 and one valence 5) over the mesh. To show that these are fundamental operations for quad mesh editing, we will establish the following properties:

- These editing operations impact the smallest possible region on the mesh and are therefore as local as possible (in a convex region).
- A region containing only one irregular vertex cannot be edited.
- A region containing two irregular vertices can be edited by changing the location of the irregular vertices within the region. However, they cannot be canceled. Some irregular vertex pairs can be merged while others cannot, depending on their graph distance in the initial configuration.
- A region with three irregular vertices can be edited by canceling or merging the irregular vertices.
- Our three movement operations can perform all possible edits within a (convex) region that contains two irregular vertices.

While the three vertex pair movement operations are at the core of this paper, we also introduce several supplementary operations to control the type and number of irregular vertices: splitting, merging, cancellation, and alignment. All of these operations can be realized through three graph-level editing operations: quad collapses, edge flips, and edge splits.

Our paper does not extensively deal with the geometric consequence of mesh editing. We consider feature edges in our implementation and analysis, but otherwise treat editing operations that result in isomorphic graphs as identical. Our analysis is based on recent work in triangular mesh editing [Li et al. 2010]. We adapt the definitions from this earlier work and their proof for Theorem 7.1 (shown in the additional materials). The enumeration (and parameterization) of all valid requadrangulations given a regular convex region is a contribution of this paper. Furthermore, while Li et al. [2010] demonstrate that it is possible to move an irregular vertex pair, we establish the condition under which this is possible with an explicit algorithm to requadrangulate a region.

2 Related Work

There has been much work in mesh optimization, most of which is for triangular meshes and usually focuses on preserving geometric details in the underlying surface or achieving optimal aspect ratio and size of triangles in the mesh under a reasonable triangle budget [Hoppe 1996; Desbrun et al. 1999; Alliez et al. 2002]. For a complete reference of past work in mesh processing we refer the readers to [Botsch et al. 2010]. In contrast, our work focuses on optimizing the *connectivity* of *quadrilateral* meshes.

There has been much recent work in the generation of a quad mesh given an input triangular mesh, i.e., quadrangulation. Typical approaches include tracing evenly spaced hyperstreamlines [Alliez et al. 2003; Marinov and Kobbelt 2004; Dong et al. 2005], constructing a global parameterization [Ray et al. 2006; Kälberer et al. 2007; Bommes et al. 2009; Zhang et al. 2010], and generating a patch layout on the surface that facilitates quadrangular remeshing [Dong et al. 2006; Tong et al. 2006; Myles et al. 2010]. For approaches depending on hyperstreamline tracing and global parameterizations, a guiding 4-way rotational symmetry (4-RoSy) field is needed. The quality of the remeshes depends on the quality of the fields, and this has led to work on generating geometry-aware fields either manually [Zhang et al. 2006; Zhang et al. 2007; Palacios and Zhang 2007; Ray et al. 2008] or automatically [Ray et al. 2009; Nieser et al. 2010]. In contrast to all of this work which assumes an input triangular mesh, the goal of our work is to perform connectivity editing directly on an input quadrilateral mesh, i.e., *requadrangulation*. In addition, our method provides explicit control over the location, orientation, type, and number of irregular vertices in the mesh.

There has been some work in formulating algorithms to quadrangulate an empty region with specific goals such as having fewest irregular vertices possible [Nasri and Yasseen 2009; Schaefer et al. 2004]. Our work is different in two ways. First, instead of formulating specific algorithms, our theoretical analysis provides an exhaustive enumeration of all possible quadrangulations within a simply-connected, convex region with up to two irregular vertices. Second, instead of quadrangulating the region from scratch, in our editing framework different quadrangulations are achieved by applying several local operators with minimal footprint.

Irregular vertex control is important as the location and type of irregular vertices are often intrinsically linked to the geometric features on the surface [Akleman and Chen 2006] and can impact the quality of inverse subdivision [Taubin 2002], pattern generation [Li et al. 2010], as well as remeshing [Ray et al. 2006]. Local operators have been proposed for quad mesh simplification [Tarini et al. 2010]. Surprisingly, however, local operators such as quad collapse, edge split, vertex rotation, and edge flip all increase the number of irregular vertices when applied to a single irregular vertex. Therefore, it is important to investigate operators on larger regions that do not increase the number of irregular vertices in a quad mesh [Daniels et al. 2008; Bommes et al. 2011], or a triangular mesh [Surazhsky and Gotsman 2003; Li et al. 2010]. Finally, Burgers vectors have been used in crystallography to displace irregular vertices [Hull and Bacon 2001].

3 Overview

The input to our system is a quadrilateral mesh M that represents a closed manifold surface. The *valence* of a vertex v in M, which we denote as $l(v)$, is the number of edges in the mesh incident to v. A vertex with a valence of n is denoted as vn, e.g., $v3$ and $v5$. A $v4$ vertex is considered as *regular*, and vertices of other valences are referred to as *irregular*. A pair of valence m and n vertices is denoted as $m-n$ pair.

To simplify the discussion, we only consider irregular vertices with a valence of 3 or 5. Other irregular vertices can be transformed to multiple $v3$ or $v5$ vertices through our atomic type change operations (Section 8).

The focus of this paper is to introduce and analyze the following three operations: $3-5$, $3-3$, and $5-5$ pair movement operations. To realize these operations, we develop a *three-level hierarchy* of editing operations: basic operations, atomic semantic operations, and the three pair-wise movement operations.

1. Basic operations (Section 4) include quad collapse, edge split, and edge flip. All semantic operations are achieved by a combination of these three types of operations.
2. Atomic semantic operations (Section 5) are single basic operations with semantic interpretations. Users can generate and remove irregular vertices as well as move, split, and merge adjacent irregular vertices.
3. Pair-wise movement operations (Section 6) include the aforementioned $3-5$, $3-3$, and $5-5$ pair movement operations. These operations are accomplished by multiple atomic semantic operations.

In Section 7 we provide theoretical analysis of the proposed operations. We first prove that it is impossible to requadrangulate a convex region containing a single irregular vertex without introducing additional irregular vertices. Consequently, possible local operations require at least two irregular vertices. By analyzing all possible connectivity edits for a pair of irregular vertices within a convex region, we conclude that the three proposed irregular vertex pair movement operations are sufficient to generate all possible local edits within a convex region containing two irregular vertices.

In Section 8 we present several useful operations built upon the three pair-wise movement operations. These additional operations can be used to reduce the number of irregular vertices, align $3-3$ and $5-5$ pairs, and merge one $3-3$ or $5-5$ pair into one $v2$ or $v6$ vertex. The aforementioned editing operations are *topological* by design and may lead to a loss of geometric details in the underlying surface and quadrilaterals with poor aspect ratios. We provide several geometric operations to remedy such problems. In Section 9 we compare our editing operations to the irregular vertex editing operations for triangular meshes proposed in [Li et al. 2010]. In Section 10 several applications of our editing operations are shown.

4 Basic Operations

Our set of basic operations has the following desirable properties: First, the support for these operations is local. Second, their implementations are relatively easy with a low computational cost. Third, it is straightforward to combine multiple basic operations, incurring no limitations and special cases (Figure 2).

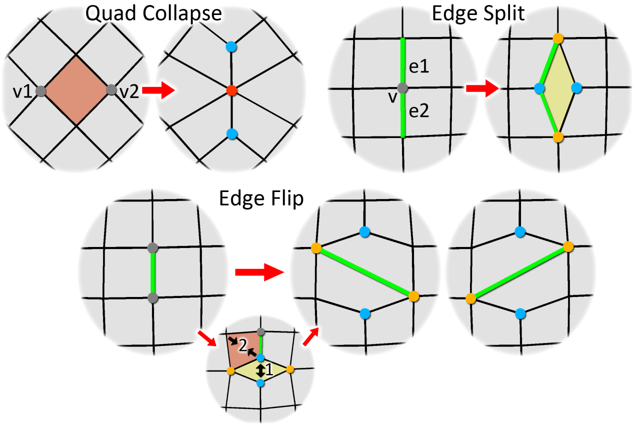


Figure 2: *Quad collapse: a pair of diagonally opposing regular vertices (gray) is merged. The face to be deleted is shown in red. The $v3$ vertices are shown in blue and the $v6$ vertex in red. Edge split: a pair of connected edges (green) is bloated into a face (yellow). The $v5$ vertices are shown in orange. Edge flip: an edge (green) is flipped in a counter-clockwise and clockwise direction. One way to realize an edge flip is by one edge split followed by one quad collapse (shown below).*

Quad Collapse: Quad collapse can be intuitively understood as merging a pair of diagonally opposing vertices (v_1, v_2) sharing the same face. The face they share will be deleted, and the valence of the merged vertex will be $l(v_1)+l(v_2)-2$, while the valence of the other pair of diagonally opposing vertices will be decreased by one per vertex.

Edge Split: Edge split can be intuitively understood as bloating a pair of connected edges (e_1, e_2) into one face and the central vertex v between them is split into two. After an edge split, a new face and a new vertex will be created. The pair of edges (e_1, e_2) separates the remaining edges incident to v_1 into two groups containing d_1 and d_2 edges, respectively. After the edge split, the valences of the two vertices are d_1+2 and d_2+2, respectively. The valence of the other two vertices involved in the operation will be increased by one.

Edge Flip: Edge flip can be intuitively understood as rotating an edge in either the counter-clockwise or clockwise direction. The valence of the two vertices on the edge will be decreased by one, while the valence of the other two involved vertices will be increased by one.

Quad collapse and edge split are inverse to each other; they are both atomic in the sense that each cannot be realized by any combination of the other two basic operations. On the other hand, the inverse of an edge flip is an edge flip in the other direction; it is not atomic because it can be realized by one edge split and one quad collapse.

5 Atomic Semantic Operations

The influence of the aforementioned basic operations on the valence is non-trivial and lacks semantic meaning, which makes it difficult to use them directly to control irregular vertices. By providing semantic interpretations for basic operations we define a collection of atomic semantic operations.

Adjacent $3-5$ Pair Movement: A pair of adjacent $v3$ and $v5$ vertices can be moved in the direction of its six adjacent vertices (Figure 3).

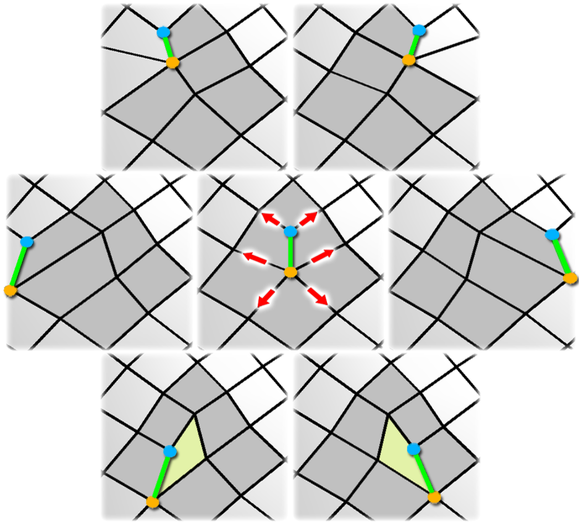


Figure 3: *Possible directions of an adjacent $3-5$ pair movement operation. Moving to the upper-left and upper-right corners are achieved by one quad collapse. Moving to the lower-left and lower-right corners is achieved by one edge split. Moving to the left and right is achieved by one edge flip. Faces adjacent to the $3-5$ pair are shown in gray to assist comparisons. In the lower-left and lower-right cases faces created by edge split are shown in yellow.*

$v3(v5)$ **Movement and** $3-5$ **Pair Generation:** A $v3$ vertex can be moved to one of its adjacent locations, and one adjacent $3-5$ pair is created. Each direction can be achieved by two kinds of quad collapses and two kinds of edge flips. Similarly, a $v5$ vertex can be moved to one of its five adjacent locations, and one adjacent $3-5$ pair is created. Each direction can be achieved by two kinds of edge splits and two kinds of edge flips. Figure 4 shows one possible moving direction for a $v3$ and a $v5$ vertex.

$v3(v5)$ **Movement and** $3-5$ **Pair Removal:** This operation is the inverse of the operation described above. It can be understood by reading Figure 4 in the reverse direction.

$3-3-5-5$ **generation/removal:** By one edge flip two $v3$ and two $v5$ vertices can be generated. Inversely two $v3$ and two $v5$ vertices can be removed by one edge flip in the other direction.

Type Change Operations: Type change operations increase or decrease vertex valences. As mentioned before, all irregular vertices other than $v3$ or $v5$ can be transformed into a set of $v3$ or $v5$ irregular vertices by using type changing operations.

A vertex's valence can be increased by applying a quad collapse to one of its adjacent faces, at the cost of creating adjacent irregular vertices. For example, applying a quad collapse to a $v2$ vertex with a regular diagonally opposing vertex can transform the $v2$ vertex into a regular vertex, while decreasing valence of the other pair of diagonally opposing vertices by one (Figure 5a). Similarly, a vertex's valence can be decreased by applying an edge split to one of the adjacent edge pairs between it, at the cost of creating adjacent irregular vertices. For example, splitting a $v6$ vertex along an edge pair that evenly separates its six adjacent edges will create two regular new vertices, while increasing the other two vertices' valence by one (Figure 5b).

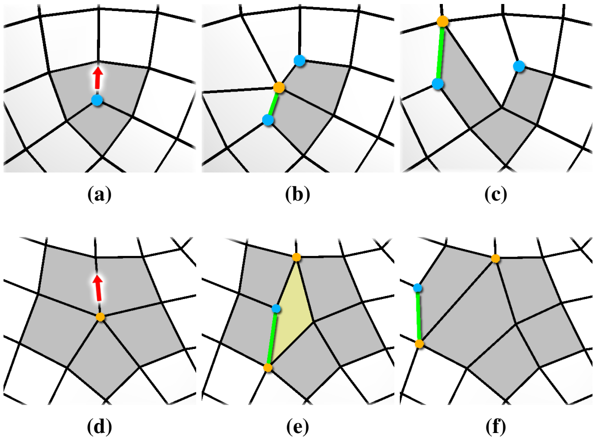


Figure 4: *Examples of moving an irregular vertex upward from its current location. (a) to (c): the* v3 *vertex is moved upward by either (b) collapsing the face on the left or (c) flipping the left edge clockwise. It can also be achieved by collapsing the face on the right or flipping the right edge counter-clockwise. (d) to (f): the* v5 *vertex is moved upward by either (e) splitting the edge pair on the left or (f) flipping the left edge counter-clockwise. It can also be achieved by splitting the edge pair on the right or flipping the right edge clockwise. Note that in all these scenarios a $3-5$ irregular vertex pair is created as a result of the movement operation.*

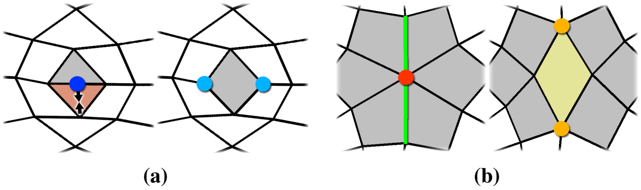


Figure 5: *Applying type change operations to convert* v2 *and* v6 *vertices into* v3 *and* v5 *vertices. (a) A* v2 *vertex is converted to two* v3 *vertices by one quad collapse. (b) A* v6 *vertex is converted to two* v5 *vertices by one edge split.*

6 Pair-wise Movement Operations

While the aforementioned atomic operations only have local influences, we can move non-adjacent pairs of irregular vertices ($3-5$, $3-3$, and $5-5$) by a combination of multiple atomic operations. We also show how the pair can be moved even when the shortest path between the vertices intersects some feature edge that has to be preserved. These operations result in quads being inserted into and deleted from the mesh. Therefore, we also need to interleave smoothing operations with these operations to avoid degrading the mesh quality.

$3-5$ **Pair Movement:** A non-adjacent $3-5$ pair can be moved together in the same direction in the following three-step pipeline:

1. Move the $v3$ vertex to a user-specified adjacent location by applying a $v3$ movement and $3-5$ pair generation operation.

2. Apply multiple adjacent $3-5$ pair movement operations to transport the generated $3-5$ pair toward the $v5$ vertex until they become adjacent.

3. Apply a $v5$ and $3-5$ pair removal operation to remove the $3-5$ pair and shift the $v5$ vertex. The relative position of the two vertices remains the same.

Alternatively the pipeline can be executed reversely by moving the $v5$ vertex first and then colliding the generated $3-5$ pair toward the $v3$ vertex. Each movement has four moving directions, which can be understood as moving on a 2D Cartesian grid with a nearby regular vertex as the origin. After one movement the relative distance between the $3-5$ pair, defined by the number of edges of their two connecting separatrices (Definition 7.7) will be preserved. In Figure 6 the four moving directions for $3-5$ pairs in different configurations are analyzed.

$3-5$ **Pair Movement with Sharp Features:** The transportation of the generated $3-5$ pair in the second step does not need to follow the shortest path between the irregular vertices. Figure 7 illustrates this with an example. If the generated $3-5$ pair is transported through the shortest path, the sharp feature will be modified. Alternatively, the generated $3-5$ pair can be transported through a longer path to avoid modifying the sharp feature.

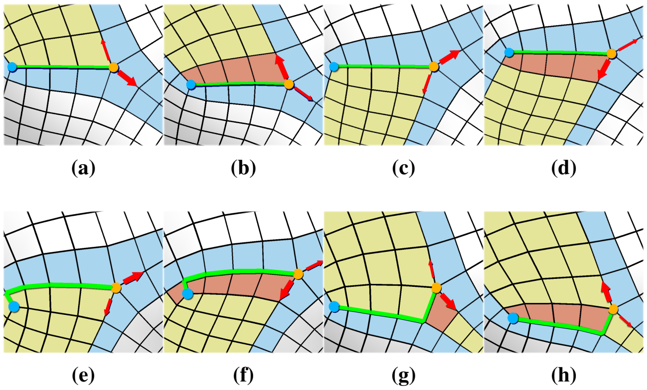


Figure 6: *Moving a $3-5$ pair. The four moving directions (red arrows) for $3-5$ pairs with aligned (top row) and mis-aligned (bottom row) separatrices. Each green line denotes a shortest path between the pair. The blue faces denote the nearest unchanged quad strips that enclose the affected region. The yellow and red faces are generated or deleted depending on which direction the $3-5$ pair moves, while red faces denote the ones that are immediately to be created or deleted. The length of the connecting path determines the number of faces created or deleted in one movement.*

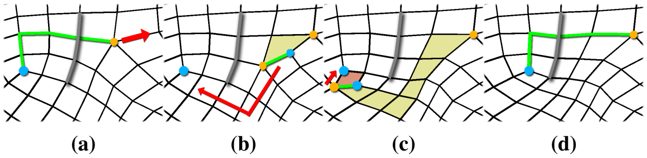
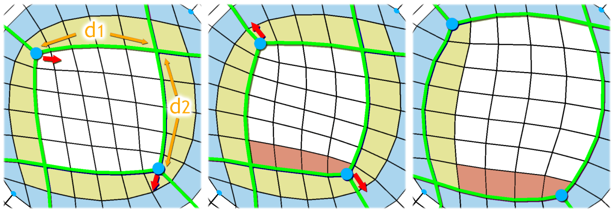


Figure 7: *Moving a $3-5$ pair whose shortest path intersects sharp features. Faces generated in the process are shown in yellow and faces deleted are shown in red. (a) The connecting path, shown in green, intersects the sharp feature shown in grey. (b) to (d) The generated $3-5$ pair is transported toward the* v3 *vertex through an alternate path that avoids the sharp feature.*

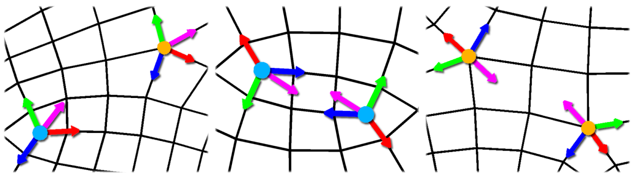
$3-3$ **Pair Movement:** The mechanics of moving a non-adjacent $3-3$ irregular vertex pair is identical to moving a $3-5$ pair. There are also four directions of movement. The major difference is that the graph distance between the two irregular vertices changes. We define the number of edges on each of the two connecting separatrices as d_1 and d_2 (See Figure 8). If the two vertices are directly connected by a separatrix one of these two values is equal to zero. One step of movement can be labeled by the changes to (d_1, d_2) in one of the four possible ways: $(+1,+1)$, $(+1,-1)$, $(-1,+1)$, and $(-1,-1)$.



(a) (b) (c)

Figure 8: *Examples of $3-3$ pair movement. (a) to (b) The $3-3$ pair is moved in the $(-1,+1)$ direction (shown as red arrow). (b) to (c) The $3-3$ pair is moved again in the $(+1,+1)$ direction. Generated faces are shown in red.*

$5-5$ **Pair Movement:** This operation is similar to moving a $3-3$ pair in that there are four directions of movement and the graph distance between the vertices changes in the same fashion.



(a) (b) (c)

Figure 9: *UI implementation of the three pair-wise movement operations. Each of the four possible moving directions is shown as one pair of arrows with the same color.*

UI Implementation: Figure 9 shows our UI implementation of the three pair-wise movement operations. The four moving directions for $3-5$, $3-3$, and $5-5$ pair movements are shown as colored arrow pairs to help the user predict the effect of a movement before actually executing it.

7 Topological Analysis

In this section we provide theoretical analysis on why the three pair-wise movement operations are fundamental. Unlike vector and tensor field editing in which it is possible to move a singularity and cancel a singularity pair with opposite singularity indexes, it is impossible to move an irregular vertex (Theorem 7.1), and topological changes between a pair of irregular vertices have to follow specific constraints. For example, an irregular vertex pair whose discrete Gaussian curvature sum to zero cannot be canceled. The following theoretical discussion explains all possible edits within a convex region containing one or two irregular vertices ($v3$ or $v5$). This discussion explains why our suggested operations are the simplest possible operations that do not increase the number of irregular vertices. Note that some terminologies used in Theorem 7.1, 7.2, 7.3, and 7.4 are defined in Definitions 7.5, 7.6, and 7.7.

Theorem 7.1 *Consider a convex region R that contains exactly one irregular vertex v_0 in its interior. When $l(v_0)$, the valence of v_0, is not a multiple of 4, it is impossible to remesh the interior quadrilaterals of R to have a different configuration that still contains only one irregular vertex.*

The proof of Theorem 7.1 is adapted from [Li et al. 2010] and is given as additional material. The following three theorems deal with two irregular vertices in a convex region. A convex N-sided polygonal region is *regular* if all N-sides have equal length measured in graph distance. In this case we call the length of each side to be the *side length* of the region.

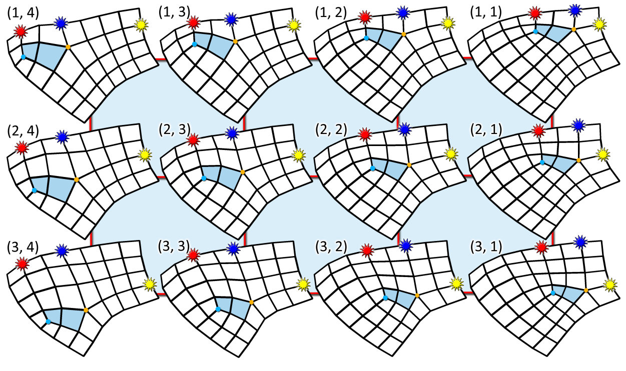


Figure 12: *All possible configurations in a region that contains a $3-5$ pair. Each configuration is labeled by its parameterization (m,n). Notice they correspond to translations over a regular 2D grid parameterized by the locations of the red and yellow stars. Every node has at most four neighbors, indicating four possible moving directions given a configuration.*

Theorem 7.2 *Let R be a regular convex digon with a side length L. If R encloses a $3-3$ irregular vertex pair, there are $\tau(L)$ mutually distinctive requadrangulations of R where $\tau(L) = N(N-1)$ when L is even and N^2 when L is odd. Here $N = \lfloor \frac{L}{2} \rfloor$. These configurations can be parameterized by the set $\{(m,n) \mid m \geq 0, n \geq 0, 0 < m+n \leq L-2\}$ (Figure 10). Any one-step requadrangulation from the case (m_0, n_0) must be one of the following possible scenarios: (1) (m_0+1, n_0-1), (2) (m_0-1, n_0+1), (3) (m_0+1, n_0+1), and (4) (m_0-1, n_0-1). The operations can correspond to distance-preserving mutual spinning as well as distance-varying movement. Furthermore, when R is not regular, the number of valid configurations is at most $\tau(L)$ where L is the side length of the smallest regular digon that contains R.*

Theorem 7.3 *Let R be a regular convex hexagon with a side length L. If R encloses a $5-5$ irregular vertex pair, there are $\tau(L)$ mutually distinctive requadrangulations of R where $\tau(L) = 3N(N-1)$ when L is even and $3N^2$ when L is odd. Here $N = \lfloor \frac{L}{2} \rfloor$. These configurations can be parameterized by the set $\{(m,n,p) \mid m \geq 0, n \geq 0, 0 < m+n \leq L-2, 1 \leq p \leq 3\}$ (Figure 11). Any one-step requadrangulation from the case (m_0, n_0, p_0) must be one of the following possible scenarios: (1) (m_0+1, n_0-1, p'_0), (2) (m_0-1, n_0+1, p'_0), (3) (m_0+1, n_0+1, p'_0), and (4) (m_0-1, n_0-1, p'_0). Here p'_0 can be p_0, p_0-1, or p_0+1, depending on the situation. The operations can correspond to distance-preserving mutual spinning as well as distance-varying movement. Furthermore, when R is not regular, the number of valid configurations is at most $\tau(L)$ where L is the side length of the smallest regular hexagon that contains R.*

Theorem 7.4 *Given a convex region R that encloses a $3-5$ pair connected by separatrices of lengths d_1 and d_2, there are exactly $K = MN$ mutually distinctive requadrangulations of R that still contain a $3-5$ pair. Here M and N are constants derived from the side lengths E_i's of R and d_1 and d_2. The set of the requadrangulations is parameterized by the following set: $\{(m,n) \mid 1 \leq m \leq$*

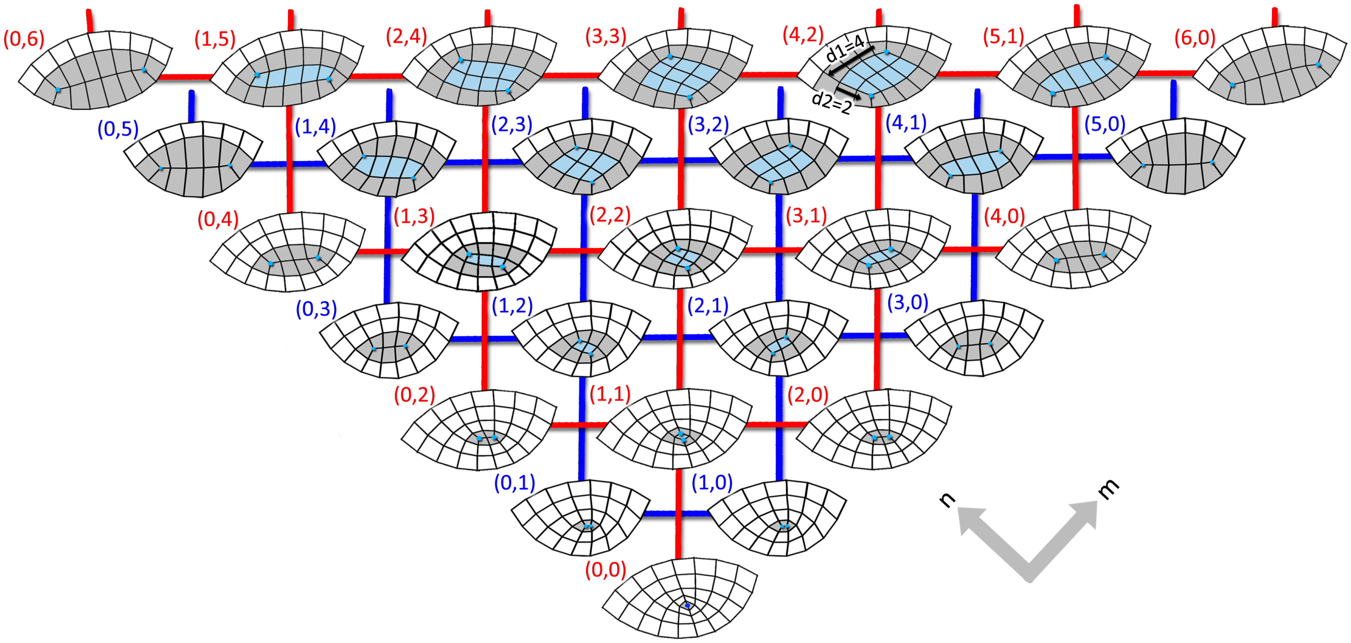


Figure 10: *All possible configurations of a region that contains a $3-3$ pair. Note that this figure extends to infinity. All possible requadrangulations of a specific region would be a sub-grid (either blue or red) that is cut along a line $n+m<c$, where c is a constant. R_α, the smallest region between the pair, is shown in blue, and the smallest regular digons containing R_α are shown in grey. Each configuration is labeled by its parameterization (m,n) and represented by a node positioned at (m,n) in a 2D coordinate system rotated by $\frac{\pi}{4}$. Every node has at most four neighbors, indicating four possibilities of movement given a configuration. Note that each pair of nodes $(x,0)$ and $(0,x)$ on the boundaries (not including the v2 case) represent the same degenerate case parameterized differently, thus each boundary configuration can also have at most four neighbors. Note that the grids of configurations with even (red) and odd (blue) L are disconnected, i.e., it is impossible to requadrangulate a configuration with even L to any configuration with odd L and vice versa. Note that the grids of configurations with odd and even L are dual to each other, i.e., each face corresponds to a vertex and each pair of adjacent faces corresponds to an edge.*

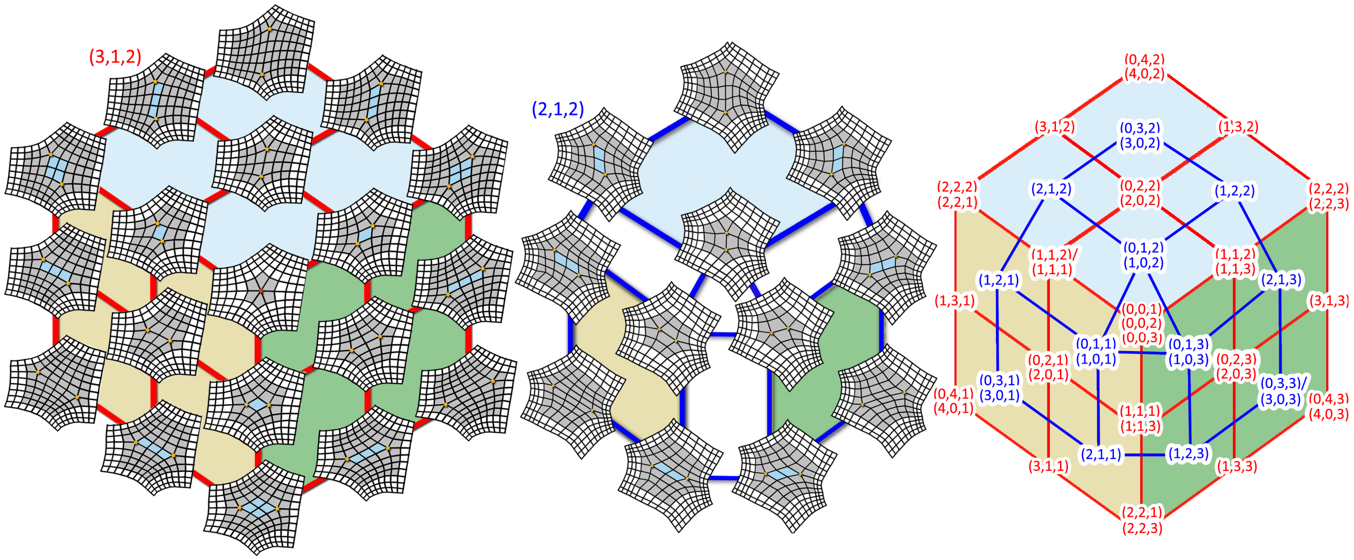


Figure 11: *All possible configurations within a regular hexagon of side length six (left) and five (middle) that contains a $5-5$ pair. Each configuration is labeled by its parameterization (m,n,p). For brevity we show all parameterizations in the right. Note that the grids of configurations with odd and even L are dual to each other, i.e., each face corresponds to a vertex and each pair of adjacent faces corresponds to an edge.*

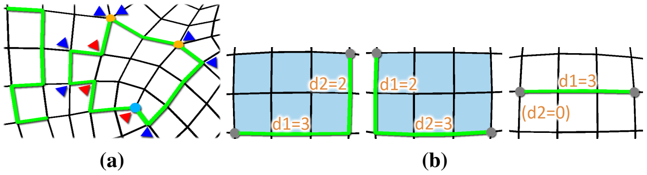


Figure 13: *(a) An open path (left) and a loop (right). Blue triangles indicate normal convex turn points and red triangles indicate concave turn points. (b) All possible cases of the smallest convex region containing a vertex pair connected by separatrices with length d_1 and d_2 in the counter-clockwise order. (left) and (middle) Two non-degenerate cases. (right) Degenerate case where d_2 is* 0.

$M, 1 \leq n \leq N\}$ (Figure 12). Any one-step requadrangulation from the case (m_0,n_0) must be one of the following: (1) (m_0+1,n_0), (2) (m_0-1,n_0), (3) (m_0,n_0+1), and (4) (m_0,n_0-1). Each of these moves corresponds to the irregular vertex pair moving toward one of the four sides of R without mutual spinning or change in distance between the irregular vertex pair.

To prove these theorems we need the following definitions adapted from [Li et al. 2010].

Definition 7.5 *A* path *γ (Figure 13a) on the mesh M consists of a sequence of edges $e_i = (v_i, v_{i+1})$ for $0 \leq i < N$. N is the* length *of γ. A path is a* loop *(Figure 13a right) if $v_0 = v_N$. Otherwise, γ is an* open path *(left). A loop γ is* degenerate *if there exists a vertex in γ that is incident to at least three edges in γ.*

A *degenerate open path* can be defined in a similar fashion. An open path γ consisting of only regular vertices is *regular*. For a regular open path γ, the edges in γ divide the 1-ring neighborhood of any interior vertex v on γ into two subsets of quadrilaterals. v is a *turn point* if there are three quadrilaterals on one side and one quadrilateral on the other side. Otherwise, it is a *non-turn point*. For non-regular open paths turn points and non-turn points are undefined. For any vertex v on a loop γ, v is a *non-turn point* if there are two quadrilaterals in the exterior of the loop, otherwise it is a *turn point*. A path γ is *straight* if every interior vertex in γ is regular and non-turning.

A *region* R is a subset of the quadrilaterals in M whose dual graph is connected. That is, for any two quadrilaterals s and t in R, there is a sequence of quadrilaterals such that $t_0 = s$, $t_N = t$, and t_i and t_{i+1} share an edge in R for all $0 \leq i < N$. The boundary of R, denoted by ∂R, is a loop. R is *degenerate* if ∂R is a degenerate loop, otherwise it is non-degenerate. In this paper we assume a region is non-degenerate and has exactly one boundary loop unless otherwise specified. A turn point on ∂R is *convex* if it has more than two adjacent quadrilaterals in $\bar{R}$, the exterior of R. Otherwise the turn point is *concave* (Figure 13a). The *angle* of a turn v, denoted by $k(v)$, is $(m(v)-2)\frac{\pi}{2}$ where $m(v)$ is the number of incident quadrilaterals of v in $\bar{R}$. Because we only consider boundary vertices with a valence up to six, a convex turn point can have a turning angle up to π ($m(v) = 4$). We refer to such points as a *sharp* convex turn point. A sharp convex turn point is considered as two *normal* convex turn points ($m(v) = 3$) connected by an edge of length zero. Finally, a region R is *convex* if there are no concave turn points on ∂R. Otherwise, it is *concave*.

Definition 7.6 *Two regions are* compatible *if their boundaries are indistinguishable from the exterior. That is, there exists a way to walk each boundary and produce the same sequence of turning angles of the vertices.* Remeshing *a region is defined as finding a different but compatible region.*

A *side* of a region R is a sequence of edges of ∂R between two normal convex turn points. Alternatively we can say two regions are compatible if there exists a way to walk each boundary and produce the same sequence of length of sides.

Definition 7.7 *An* open separatrix *is an open and straight path whose interior vertices are regular and at least one of the end vertices is irregular. A* closed separatrix *is a loop that contains exactly one irregular vertex. In the remainder of the paper we are only interested in open separatrices. There are $l(v_0)$ separatrices emanating from v_0.*

The discrete Gauss-Bonnet theorem relates the total turning angle $\sum_{v \in \partial R} k(v)$ along the boundary ∂R to the total discrete Gauss curvature of the vertices of R as follows:

$$\sum_{v \in \partial R} (m(v)-2) + \sum_{v \in intR} (4 - l(v)) = 4\chi(M) \tag{1}$$

where $\chi(M)$ is the *Euler characteristic* of M.

To prove Theorems 7.2, 7.3 and 7.4, first we need the following lemmas.

Lemma 7.8 *Under the assumption of Theorems 7.2, 7.3 and 7.4, the smallest region between v_1 and v_2, R_α, is unique and must be a rectangular grid of size $d_1 \times d_2$ in the counter-clockwise order. If d_1 or d_2 is zero we have the degenerate case of the two irregular vertices connected by a single separatrix.*

The proof of Lemma 7.8 is based on [Li et al. 2010] and is given as additional material. All possible cases of the smallest convex region are shown in Figure 13b.

We note that the smallest *enclosing* convex region for an irregular vertex pair is the 1-ring neighborhood of R_α.

Lemma 7.9 *Given a convex region R that contains no irregular vertices and no sharp convex turn points on its boundary ∂R, the number of sides (equal to the number of normal convex turn points) in ∂R is given by $4 - G(R)$ where $G(R) =$ is total discrete Gaussian curvature of R.*

Proof From Equation 1 we have $\sum_{v \in \partial R}(m(v)-2) = 4\chi(R) - \sum_{v \in intR}(4-l(v))$. Since R has only convex turn points, for any normal convex turn point on ∂R we have $m(v) = 3$ and non-turn point $m(v) = 2$. Consequently, the left side of this equation, i.e., $\sum_{v \in \partial R}(m(v)-2)$, is equal to the number of normal convex turn points in ∂R. On the right side of this equation, $\chi(R) = 1$ since R is simply-connected. Furthermore, $G(R) = \sum_{v \in intR}(4-l(v))$ is the total discrete Gaussian curvature of R. ▮

A consequence of this lemma states that the number of sides of a convex region R contains a $3-3$, $5-5$, and $3-5$ irregular vertex pair has 2, 6, and 4 sides, respectively (see Figure 14 for examples).

Lemma 7.10 *The atomic step to expand a convex region R while keeping R convex without introducing irregular vertices is to add a strip of quadrilaterals adjacent to one side of R.*

Proof To expand a region at least one quadrilateral belonging to $\bar{R}$ adjacent to an edge in ∂R has to be added. Consequently, for the two ending vertices of the edge, denoted as v_a and v_b, both gain an adjacent quadrilateral and $m(v_a)$ and $m(v_b)$ decrease by 1. Thus v_a and v_b will become concave turn points if it is not a convex turn point ($m(v_a), m(v_b) <= 2$). To avoid creating concave turn points adding more quadrilaterals adjacent to edges in ∂R right next the first edge is necessary until both ending vertices are convex turn points. ▮

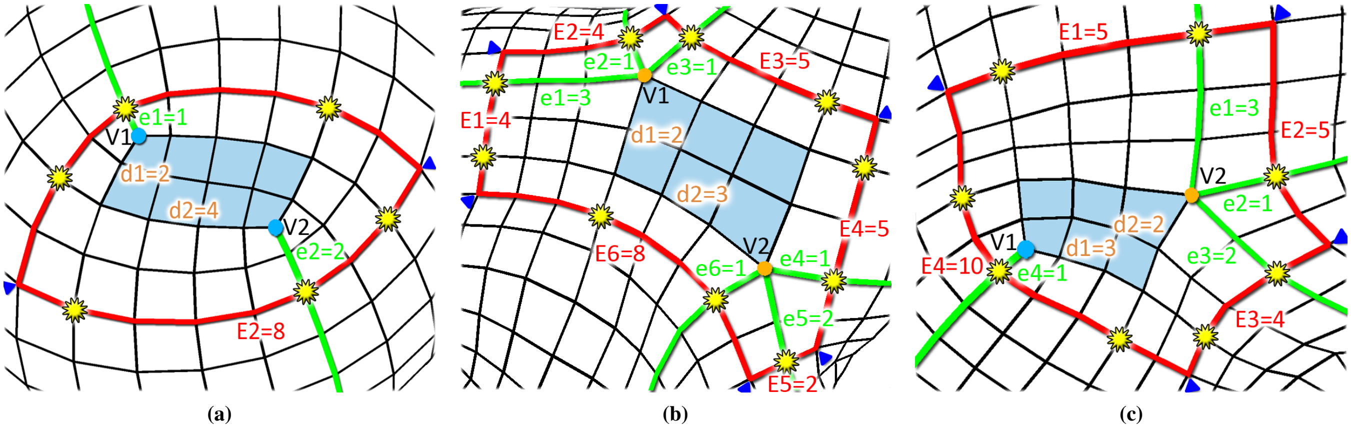


Figure 14: *Examples of convex regions R enclosing exactly one (a) $3-3$, (b) $5-5$ and (c) $3-5$ pair v_1 and v_2 connected by separatrices of length d_1 and d_2. The smallest region between the pair R_α is shown in blue. ∂R is shown in red. Blue triangles indicate normal convex turn points. Separatrices of v_1 and v_2 that go outward from R_α are shown in green. E_1 to E_6 denote the length of sides of R. e_1 to e_6 denote the number of quad strips extended from R_α along the outward separatrices. The intersection points between separatrices and the region boundary are marked with yellow stars.*

Theorem 7.11 *Given a convex region R enclosing a $3-3$, $5-5$, or $3-5$ pair, the configuration is* uniquely *determined by the following vector of couples (S_{ij}, u_{ij}) where i is the index of the irregular vertex, j is the index of a separatrix emanating from v_i, S_{ij} is the side on ∂R that intersects with this separatrix, and u_{ij} is the graph distance of this intersection point from the starting point of this side (counterclockwise). In Figure 14 each of these locations is highlighted with a yellow star-shaped symbol.*

The proof for Theorem 7.11 is similar to that of Theorem 7.1. Basically the location of the intersection points on the boundary and the side lengths of the region boundary are related to the distance of the irregular vertex pair to each edge of ∂R. This relation can be characterized by a linear system that has a unique solution. Consequently, should the solution have only positive entries (all distances are positive), there is a unique configuration corresponding to this set of parameters. ∎

The following lemmas show the underlying structure of R containing exactly a $3-3$, $5-5$ or $3-5$ pair:

Lemma 7.12 *Under the assumption of Theorem 7.2, arbitrary convex regions R can be constructed as follows: from the smallest convex region between the $3-3$ pair R_α, there are e_1 strips of quadrilaterals extending from one v3 vertex parallel to its outward separatrix and e_2 strips of quadrilaterals extending from another v3 vertex parallel to its outward separatrix. An example is shown in Figure 14a.*

Lemma 7.13 *Under the assumption of Theorem 7.3, arbitrary convex regions R can be constructed as follows: from the smallest convex region between the $5-5$ pair R_α, there are e_1, e_2 and e_3 strips of quadrilaterals extended from one v5 vertex parallel to its three outward separatrices, and e_4, e_5 and e_6 strips of quadrilaterals extended from another v5 vertex parallel to its three outward separatrices. An example is shown in Figure 14b.*

Lemma 7.14 *Under the assumption of Theorem 7.4, arbitrary convex regions R can be constructed as follows: from the smallest convex region between the $3-5$ pair R_α, there are e_1, e_2 and e_3 strips of quadrilaterals extended from the v5 vertex parallel to its three outward separatrices, and e_4 strips of quadrilaterals extended from the v3 vertex parallel to its outward separatrix. An example is shown in Figure 14c.*

Proof For R_α, which is unique according to Lemma 7.8, Lemma 7.12 holds with e_1 and e_2 equals 0, Lemma 7.13 holds with e_1 to e_6 equals 0, and Lemma 7.14 holds with e_1 to e_4 equals 0. Consider R_α as the unique first step in constructing R. By Lemma 7.9 we know that R has exactly 2, 6, and 4 sides when containing exactly a $3-3$, $5-5$, and $3-5$ pair. By Lemma 7.10 we know that the atomic step to expand a convex region is to add one strip of quadrilaterals adjacent to one side. It can be easily verified that each side is perpendicular to one outward separatrix of one $v3$ or $v5$ vertex. Since the strip of quadrilaterals added is parallel to one side, it is also perpendicular to one outward separatrix of one $v3$ or $v5$ vertex, thus Lemmas 7.12, 7.13 and 7.14 hold after the expansion.

Since these lemmas hold for the unique first step of constructing R and hold after all possible atomic steps to expand R, by mathematical induction we know that they hold for arbitrary convex regions R, which must be constructed from the unique R_α by several atomic expansion steps. ∎

We now prove Theorem 7.2.

Proof By Lemma 7.12 we denote the two sides of R adjacent to the e_1 and e_2 strips of quadrilateral as E_1 and E_2, as shown in Figure 14a. It can be easily verified that the following equations hold: $E_1 = d_1 + d_2 + 2e_2$ and $E_2 = d_1 + d_2 + 2e_1$.

Consequently, e_1 and e_2 are uniquely determined given E_1, E_2, d_1, and d_2. We now seek to enumerate (and parameterize) all possible such requadrangulations when $E_1 = E_2$.

After remeshing we denote the changes of each variable as Δd_1, Δd_2, Δe_1 and Δe_2. Since $E_1 = E_2$ are constant we have the following set of equations: $\Delta d_1 + \Delta d_2 + 2\Delta e_2 = 0$ and $\Delta d_1 + \Delta d_2 + 2\Delta e_1 = 0$.

From these equations we know that $\Delta d_1 + \Delta d_2$ must be even otherwise Δe_2 and Δe_1 will not be an integer. In other words Δd_1 and Δd_2 have to be either both even or both odd. Furthermore, if $\Delta d_1 = \Delta d_2 = 0$ then $\Delta e_2 = 0$ and $\Delta e_1 = 0$. All variables do not change thus R remains identical.

In the case of a regular digon containing a $3-3$ pair that is not connected by a single separatrix, each edge of the digon will intersect one separatrix from one vertex, say v_1, and two separatrices from the other, say v_2. Suppose that the first edge in the digon intersects separatrix 1 of v_1 and separatrices 0 and 2 of v_2, i.e., $S_{11} = S_{20} = S_{22} = 1$. Recall that u_{11}, u_{20}, and u_{22} denotes the location of the intersection points. Consequently, we have $u_{20} < u_{11} < u_{22}$. Notice that $d_1 = u_{22} - u_{11}$ and $d_2 = u_{11} - u_{20}$. When the irregular vertex pair is connected by a separatrix, each edge in the digon will only intersect one separatrix from each vertex. In this case, $d_1 = u_{22} - u_{11}$ and $d_2 = 0$.

When $d_1 + d_2 = E_1$, the irregular vertex pair must appear on the boundary of R. Since we discuss only convex regions free of irregular vertices on its boundary, this is not allowed. Furthermore, once d_1 and d_2 are given, the set of intersection points between any separatrix from v_1 or v_2 with any edge will be determined. According to Theorem 7.11, there is at most one valid configuration (requadrangulation) that satisfies the requirements on positions of intersection points. Consequently, all possible requadrangulations of R with a $3-3$ pair can be parameterized by m and n as follows: $\{(m,n) \mid m \geq 0, n \geq 0, 0 < m+n < E_1 - 2\}$. Here $m = d_1$ and $n = d_2$. For degenerate cases, i.e., $d_2 = 0$, both $(d_1, 0)$ and $(0, d_1)$ are allowed. This gives rise to a pyramid-shaped grid such as the one shown in Figure 10. When E_1 is even, there are $N(N-1)$ mutually distinguishable configurations. When E_1 is odd, there are N^2 mutually distinguishable configurations. Here $N = \lfloor \frac{L}{2} \rfloor$ and $L = E_1 = E_2$. Every grid point corresponds to a valid requadrangulation, and $(0,0)$ corresponds to the case in which the irregular vertex pair is merged into a single $v2$ vertex.

Given a configuration, it is possible to move the irregular vertex pair in at most four directions, corresponding to the four neighboring grid points. Each of these four atomic changes can be realized using a pair-wise movement operation. Given this, we now have an explicit algorithm of realizing any valid configuration inside a regular digon. For configurations where $m+n$ is even, we start with the unique configuration corresponding to the $(0,0)$ case. For configurations where $m+n$ is odd, we start with the unique configuration corresponding to the $(0,1)$ case. The uniqueness is a result of Theorem 7.1. Next, we find a shortest path in the grid between the starting case and the target configuration and perform necessary steps to move from the former towards the latter through appropriate graph-level operations. This demonstrates that each valid configuration inside a regular digon can be realized, i.e. there is one and only one remeshing corresponding to every grid point. ∎

The proof for Theorem 7.3 is similar, except that the relationships governing these variables are given by:

$$
\begin{aligned}
E_1 &= d_1 + e_2 + e_6 \\
E_2 &= e_1 + e_3 \\
E_3 &= d_2 + e_2 + e_4 \\
E_4 &= d_1 + e_3 + e_5 \\
E_5 &= e_4 + e_6 \\
E_6 &= d_2 + e_1 + e_5
\end{aligned}
\tag{2}
$$

When R is a regular hexagon containing a $5-5$ pair, i.e., $E_1 = E_2 = E_3 = E_4 = E_5 = E_6$, there is again a two-fold rotational symmetry which maps two different configurations to the same remeshing. These two cases corresponding to the positions of the irregular vertex pair are swapped. In the remainder of the discussion we will treat the two configurations as one, which is equivalent to modulating out the two-fold symmetry. When the irregular vertices are not connected by a single separatrix, it is straightforward to verify that there are exactly four edges in R that intersect two separatrices, one from v_1 and v_2 each. Due to the two-fold symmetry in the hexagon. These four edges form two pairs of opposing edges. Consequently, it is sufficient to consider only two consecutive edges from the four.

After the initial modulation, there is still a three-fold rotational symmetry, which corresponds to three pairs of consecutive sides. For example, in Figure 14b the sides with length E_3 and E_4 are one such pair. Every valid configuration for one pair can be used to generate a solution for the other two pairs through an appropriate rotation, and vice versa. Consequently, it is sufficient to consider only the case where the pair of sides each intersect with one separatrix from v_1 and v_2. Notice that the distance between the two intersection points on the first side is equal to d_1, and the distance between the intersection points on the second side is d_2.

For the special case when there is a connecting separatrix between v_1 and v_2, we consider that d_2=0. Consequently, the set of solutions can be parameterized by the same set of m and n as for the $3-3$ pair. However, due to the aforementioned three-fold symmetry, we add the third index p which ranges from 1 to 3 to distinguish them. This leads to three squares being glued together to form the surface of a half cube (Figure 11). Like the $3-3$ case, in a regular hexagon there is one and only one valid configuration corresponding to each point in the grid. Any valid configuration can be explicitly realized. ∎

We now prove Theorem 7.4.

Proof By Lemma 7.14 we denote the four sides of R adjacent to the e_1 and e_4 strips of quadrilateral as E_1 and E_4, as shown in Figure 14c. It can be easily verified that the following equations hold:

$$
\begin{aligned}
E_1 &= d_1 + e_2 + e_4 \\
E_2 &= e_1 + e_3 \\
E_3 &= d_2 + e_2 + e_4 \\
E_4 &= d_1 + d_2 + e_1 + e_3
\end{aligned}
\tag{3}
$$

After remeshing we have $\Delta E_i = 0$ for $1 \leq i \leq 4$. This implies that $\Delta d_1 = \Delta d_2 = 0$, i.e., any remeshing of R cannot change the relative position of the two irregular vertices. However, these configurations can differ in terms of e_1 and e_2. Given the values of E_1, E_2, e_1, and e_2 such that $1 \leq e_1 < E_2$ and $1 \leq e_2 < E_1 - d_1$, there exists a unique legal requadrangulation of R. Consequently, the set of possible requadrangulations can be parameterized by e_1 and e_2, and the range of $m = e_1$ and $n = e_2$ are given by $[1, E_2 - 1]$ and $[1, E_1 - d_1 - 1]$, respectively. This is illustrated in Figure 12. Any atomic requadrangulation will translate R_α towards one of the sides of ∂R, corresponding to the four neighboring grid points. ∎

8 Complementary Operations

Irregular Vertex Cancellation: We can move a $v3$ vertex to collide with a $v5$ vertex, or vice versa, by applying multiple pair-wise movement operations. When one $v3$ and one $v5$ vertex collide they cancel each other and both become regular. At least one other irregular vertex needs to be involved in this cancellation. In this fashion we develop a $3-5$ **pair cancellation** operation. It is possible that the last step of a $3-5$ pair cancellation is equivalent to one $3-3-5-5$ removal operation and two pairs of irregular vertices are canceled at once. Examples can be found in Figures 16 and 17.

Irregular Vertex Merging: A $3-3$ pair can be merged to a $v2$ vertex and a $5-5$ pair can be merged to a $v6$ vertex when their graph distance is even. Theorems 7.2 and 7.3 provide the theoretical analysis that is related to such a merge.

Irregular Vertex Alignment: Under the assumptions of Theorems 7.2 and 7.3, arbitrary $3-3$ and $5-5$ pairs can be aligned by applying multiple movement operations until $d_1 = 0$ or $d_2 = 0$.

Smoothing: We use iterative Laplacian mesh smoothing to improve the geometry if the connectivity edits degrade the shape of the mesh above a user-defined tolerance. The user can select uniform weights or cord-length weights, and elect to preserve sharp features by constraining the positions of vertices on sharp edges. The smoothing scheme can improve the aspect ratios of modified faces. After each iteration all vertices are projected back onto the original mesh. We have also experimented with a scheme in which newly generated vertices are pulled towards vertices in the original mesh if the distance between the new and original vertices is above a threshold. The projection and pulling scheme can narrow the difference to the original mesh.

9 Comparisons to Triangle Mesh Editing

We show the analogy between the editing operations for triangular meshes proposed in [Li et al. 2010] and our operations for quadrilateral meshes. First, it is impossible to generate, move, or delete a single irregular vertex within a convex region for both triangular and quadrilateral cases. Thus the simplest possible operations that do not increase the number of irregular vertices must involve an irregular vertex pair. For the triangular case they are $5-7$, $5-5$, and $7-7$ pair movements, and for the quadrilateral case they are $3-5$. $3-3$, and $5-5$ pair movements.

The $5-7$ pair (triangular case) and $3-5$ pair (quadrilateral case) movements both will not change the relative distance between the pair, measured in the length of their connecting separatrices.

Similarly, the $5-5$ / $7-7$ pair (triangular case) and $3-3$ / $5-5$ pair (quadrilateral case) movements both will change the relative distance under specific constraints. In general in both cases the pair will move in a symmetric, rotating fashion.

We note that despite the similarity in the suites of editing operations for both triangular and quadrilateral meshes, i.e., both comprise a hierarchy of basic, atomic semantic, and composite operations, our paper is the first to be able to enumerate all possible configurations given a regular convex region. Such analysis was not present in [Li et al. 2010]. Consequently, while they demonstrated that it is possible to move a $5-5$, $7-7$, or $5-7$ pair in practice, no guarantee was provided, neither was an explicit algorithm given to transform from one valid configuration to a different valid configuration within the regular convex region.

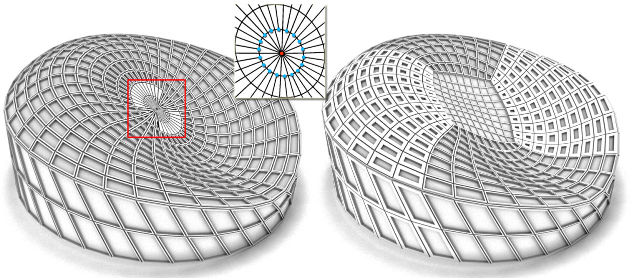


Figure 15: *A simple disc shaped mesh with higher order (36) irregular vertices is converted to a semi-regular mesh. The faces and edges of the mesh define the panels of the designed structure.*

10 Applications

Irregular Vertex Cancellation and Alignment: In Figure 1 we improve a highly irregular structure designed by an architect that consists of quadrilateral glass panels. We reduce the numbers of irregular vertices from 168 (92 $v3$, 62 $v5$, and 14 $v6$) to 24 (12 $v3$ and 12 $v5$) by multiple $3-5$ pair cancellation operations. Then we align the irregular vertices to improve the flow of the panel structure. The editing process takes approximately 12 minutes (7 minutes for irregular vertices cancellation and 5 minutes for alignment).

In Figure 15 we transform a mesh with higher order irregular vertices (36 $v3$, 2 $v36$) to a semi-regular (4 aligned $v3$ vertices) mesh by several valence reduction operations and several irregular vertex alignment operations.

Connectivity Improvement: Figure 16, 17, and 18 show our editing framework is capable of repairing various kinds of defects found in results of state-of-art quadrangulation approaches. Even though these existing quad remeshing algorithms produce excellent results, they are essentially heuristics to tackle an NP-complete problem resulting from the fact that the selection and positioning of irregular vertices is a discrete optimization problem. Therefore, we believe that it is unlikely that a general solution can be found. In our analysis a user can identify several meaningful edits that can improve a mesh for many automatically generated results. We therefore argue that manual editing tools are an essential component of a complete mesh processing pipeline.

Future Work: We believe that several automatic mesh processing algorithms can be built using the proposed editing operations. As a preliminary result we show how irregular vertices can be reduced by iteratively performing a $3-5$ pair cancellation operation in Figure 19. This result is achieved by a greedy optimization scheme, where at each step the operation with lowest cost is selected from a set of candidate operations. The questions that we plan to address are how to select candidate operations, how to evaluate the cost of a candidate operation, and how to preserve the shape of the input model. For our initial result we simply considered all possible operations as candidates and we used the sum of the affected vertices discrete mean curvature as cost function. As future work, it might also be possible to improve upon existing work in irregular vertex aware quad mesh simplification and to find new applications in pattern design on surfaces.

11 Conclusion

In this paper we propose editing operations for quad meshes to explicitly control the location, orientation, type, and number of irregular vertices. We analyzed what edits are possible and what edits are impossible in a quad mesh with sharp feature edges. Our conclusion of this theoretical analysis is that the movement of a pair of irregular vertices is the best possible compound editing operation. These operations and their effects on the mesh are also discussed in this paper. We believe that these editing operations are essential in applications such as quadrangular mesh optimization and pattern design. We underline our argument by showing how our editing operations can improve the output of important state-of-the-art remeshing algorithms. We hope that our research can make a contribution to the communities of 3D artists, designers, and architects. Here, quad meshes are often created manually, even in a quad-by-quad fashion, instead of automatically generated by remeshing techniques.

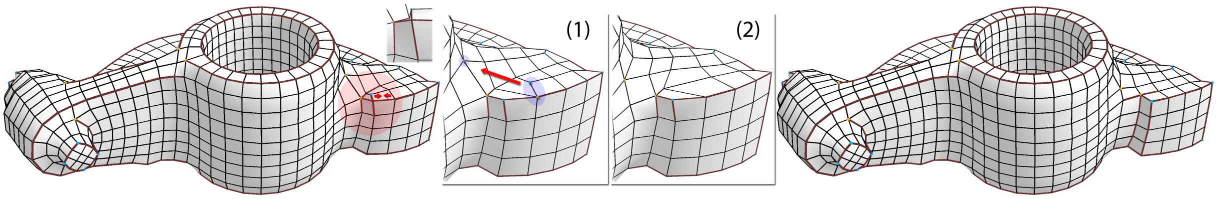


Figure 16: *Using our editing framework to improve a remeshed rockarm model from Mixed-Integer Quadrangulation [Bommes et al. 2009]. (left) Original mesh with an ill-shaped corner part. The diagonal $3-5$ pair leads to a highly non-planar face in between and distorts the nearby faces on the upper-left side. (1) The misplaced v3 vertex is made regular by a quad collapse. (2) The generated $3-5$ pair is moved upper-left to cancel with the v5 vertex by a $3-5$ pair cancellation. Movement of vertices on sharp features are constrained. (right) Mesh improved by our editing framework. Now the corner part has a nice structure.*

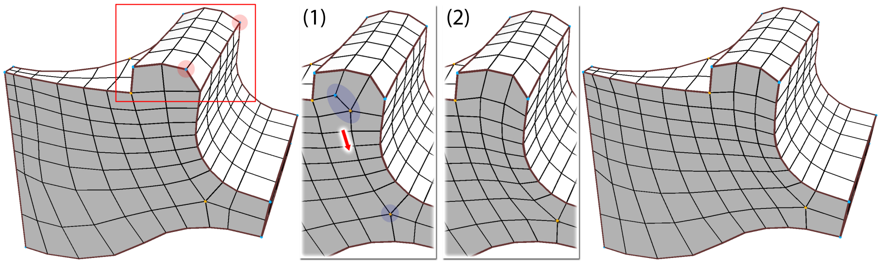


Figure 17: *Using our editing framework to improve a remeshed fandisk model from Wave-Based Anisotropic Quadrangulation [Zhang et al. 2010]. (left) Original mesh with an ill-shaped upper part caused by a mis-aligned $3-3$ pair. (1) The misplaced v3 vertex is moved to proper location by an edge flip. (2) The created $3-5$ pair is moved downward to cancel with the v5 vertex by a $3-5$ pair cancellation. (right) Mesh improved by our editing framework. Now the upper part has a nice structure and the face strips flow consistently through the front surface.*

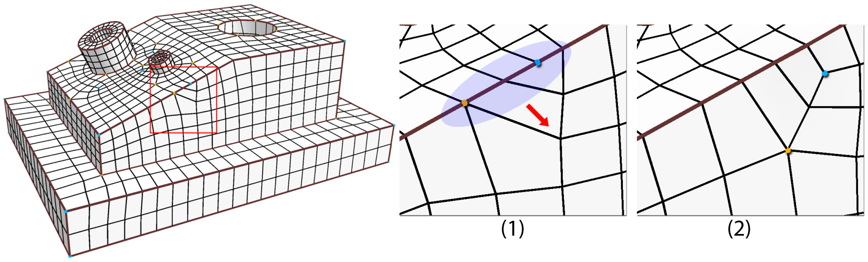


Figure 18: *Using our editing framework to fix a defect on a remeshed accessory model from Wave-Based Anisotropic Quadrangulation [Zhang et al. 2010]. (left) Original mesh with a v3 irregular vertex positioned on the feature edge. The v3 vertex's deficiency of valence leads to a degenerated, triangular shaped quad face on its lower side. (1) to (2) The v3 vertex is pulled off the feature edge by a $3-5$ pair movement. The shape of nearby faces are improved because of the better mesh topology.*

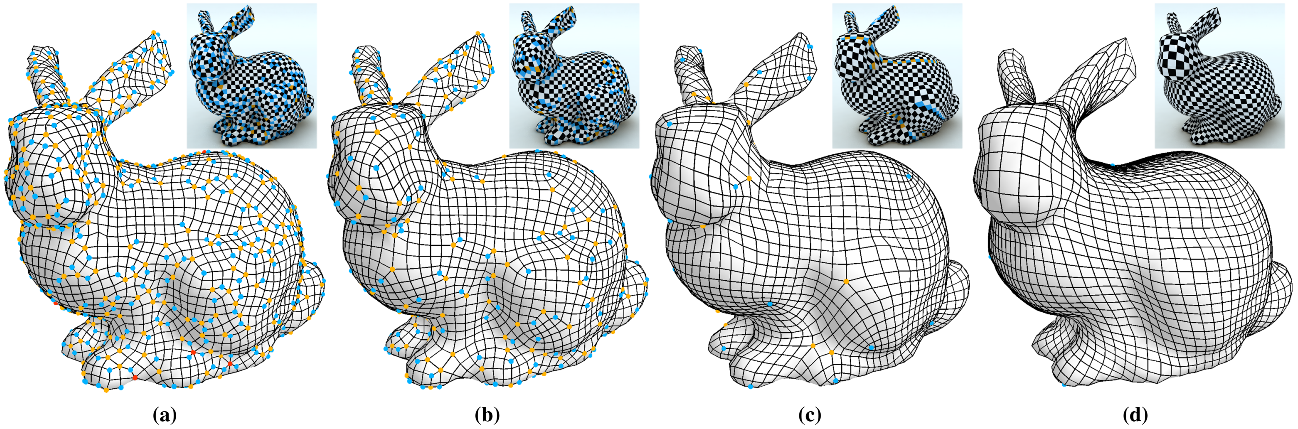


Figure 19: *Progressive irregular vertex cancellation. (a) Bunny model produced by a mesh simplification algorithm [Tarini et al. 2010], in which 956 of the total 3006 vertices are irregular. (b) and (c) Intermediate results with 400 and 48 irregular vertices. (d) Maximally reduced form with only eight v3 irregular vertices. The smaller figures show the same models with a checkerboard pattern by greedy coloring (so that adjecent quad faces have different colors).*

Acknowledgments

We would like to thank Alla Sheffer, David Bommes, and the anonymous reviewers for insightful comments. We thank David Bommes [Bommes et al. 2011], Muyang Zhang [Zhang et al. 2010], and Marco Tarini [Tarini et al. 2010] for providing datasets. This research has been funded by NSF (IIS-0915990, CCF-0643822, CCF-0830808, and IIS-0917308).

References

AKLEMAN, E., AND CHEN, J. 2006. Practical polygonal mesh modeling with discrete gaussian-bonnet theorem. In *GMP*, 287–298.

ALLIEZ, P., MEYER, M., AND DESBRUN, M. 2002. Interactive geometry remeshing. *ACM Trans. Graph. 21*, 3, 347–354.

ALLIEZ, P., COHEN-STEINER, D., DEVILLERS, O., LÉVY, B., AND DESBRUN, M. 2003. Anisotropic polygonal remeshing. *ACM Trans. Graph.*, 485–493.

BOMMES, D., ZIMMER, H., AND KOBBELT, L. 2009. Mixed-integer quadrangulation. *ACM Trans. Graph. 28*, 3, 77.

BOMMES, D., LEMPFER, T., AND KOBBELT, L. 2011. Global structure optimization of quadrilateral meshes. In *Computer Graphics Forum (Eurographics 2011)*, 375–384.

BOTSCH, M., KOBBELT, L., PAULY, M., ALLIEZ, P., AND LEVÝ, B. 2010. *Polygon Mesh Processing*. A K Peters, Natick, Massachusetts.

DANIELS, J., SILVA, C. T., SHEPHERD, J., AND COHEN, E. 2008. Quadrilateral mesh simplification. *ACM Trans. Graph.*, 148:1–148:9.

DESBRUN, M., MEYER, M., SCHRÖDER, P., AND BARR, A. H. 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH 1999*, 317–324.

DONG, S., KIRCHER, S., AND GARLAND, M. 2005. Harmonic functions for quadrilateral remeshing of arbitrary manifolds. *Comput. Aided Geom. Des. 22*, 392–423.

DONG, S., BREMER, P.-T., GARLAND, M., PASCUCCI, V., AND HART, J. C. 2006. Spectral surface quadrangulation. *ACM Trans. Graph.*, 1057–1066.

HOPPE, H. 1996. Progressive meshes. In *SIGGRAPH 1996*, 99–108.

HULL, D., AND BACON, D. 2001. *Introduction to dislocations*. Butterworth-Heinemann.

KÄLBERER, F., NIESER, M., AND POLTHIER, K. 2007. Quadcover - surface parameterization using branched coverings. *Comput. Graph. Forum 26*, 3, 375–384.

LI, Y., ZHANG, E., KOBAYASHI, Y., AND WONKA, P. 2010. Editing operations for irregular vertices in triangle meshes. *ACM Trans. Graph.*, 153:1–153:12.

MARINOV, M., AND KOBBELT, L. 2004. Direct anisotropic quad-dominant remeshing. In *PG*, 207–216.

MYLES, A., PIETRONI, N., KOVACS, D., AND ZORIN, D. 2010. Feature-aligned t-meshes. *ACM Trans. Graph.*, 117:1–117:11.

NASRI, A., S. M., AND YASSEEN, Z. 2009. Filling n-sided regions by quad meshes for subdivision surfaces. vol. 28, 1644–1658.

NIESER, M., PALACIOS, J., POLTHIER, K., AND ZHANG, E. 2010. Hexagonal global parameterization of arbitrary surfaces. In *ACM SIGGRAPH ASIA 2010 Sketches*, ACM, New York, NY, USA, SA '10, 5:1–5:2.

PALACIOS, J., AND ZHANG, E. 2007. Rotational symmetry field design on surfaces. *ACM Trans. Graph.*, 55.

RAY, N., LI, W. C., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. Periodic global parameterization. *ACM Trans. Graph. 25*, 4, 1460–1485.

RAY, N., VALLET, B., LI, W. C., AND LÉVY, B. 2008. N-symmetry direction field design. *ACM Trans. Graph. 27*, 2, 10:1–10:13.

RAY, N., VALLET, B., ALONSO, L., AND LEVY, B. 2009. Geometry aware direction field design. *ACM Trans. Graph.*.

SCHAEFER, S., WARREN, J., AND ZORIN, D., 2004. Lofting curve networks using subdivision surfaces.

SURAZHSKY, V., AND GOTSMAN, C. 2003. Explicit surface remeshing. In *2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, 30.

TARINI, M., PIETRONI, N., CIGNONI, P., PANOZZO, D., AND PUPPO, E. 2010. Practical quad mesh simplification. vol. 29, 407–418.

TAUBIN, G. 2002. Detecting and reconstructing subdivision connectivity. *The Visual Computer 18*, 5–6.

TONG, Y., ALLIEZ, P., COHEN-STEINER, D., AND DESBRUN, M. 2006. Designing quadrangulations with discrete harmonic forms. In *SGP '06*, 201–210.

ZHANG, E., MISCHAIKOW, K., AND TURK, G. 2006. Vector field design on surfaces. *ACM Trans. Graph. 25* (October), 1294–1326.

ZHANG, E., HAYS, J., AND TURK, G. 2007. Interactive tensor field design and visualization on surfaces. *IEEE Transactions on Visualization and Computer Graphics 13*, 1, 94–107.

ZHANG, M., HUANG, J., LIU, X., AND BAO, H. 2010. A wave-based anisotropic quadrangulation method. *ACM Trans. Graph.*, 118:1–118:8.