Cyclic Plain-Weaving on Polygonal Mesh Surfaces with Graph Rotation Systems

ERGUN AKLEMAN
Visualization Dept.
Texas A&M University

JIANER CHEN
Computer Science Dept.
Texas A&M University

QING XING
Architecture Dept.
Texas A&M University

JONATHAN L. GROSS
Computer Science Dept.
Columbia University

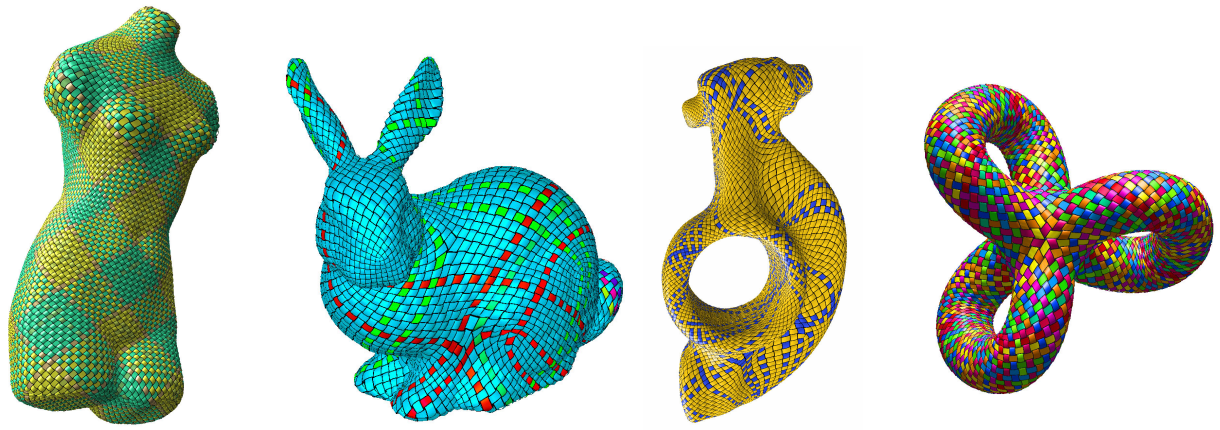


Figure 1: Examples of woven objects constructed with ribbons: Venus consists of five distinct cycles. The bunny has eight cycles, the rocker arm has only two cycles, and the genus-three object has 16 cycles. The first three models are created by the Quadcover method [Kalberer et al. 2007], courtesy of Wenping Wang and Li Yupei. The genus-three object is created using TopMod3D [Akleman et al. 2008].

Abstract

In this paper, we show how to create plain-weaving over an arbitrary surface. To create a plain-weaving on a surface, we need to create cycles that cross other cycles (or themselves) by alternatingly going over and under. We use the fact that it is possible to create such cycles, starting from any given manifold-mesh surface by simply twisting every edge of the manifold mesh. We have developed a new method that converts plain-weaving cycles to 3D thread structures. Using this method, it is possible to cover a surface without large gaps between threads by controlling the sizes of the gaps. We have developed a system that converts any manifold mesh to a plain-woven object, by interactively controlling the shapes of the threads with a set of parameters. We have demonstrated that by using this system, we can create a wide variety of plain-weaving patterns, some of which may not have been seen before.

CR Categories: G.2.2 [Graph Theory]: Topological Graph Theory—Graph Rotation Systems; I.3.5 [Computational Geometry and Object Modeling]: Geometric Algorithms—Links, Knots and Weaving.

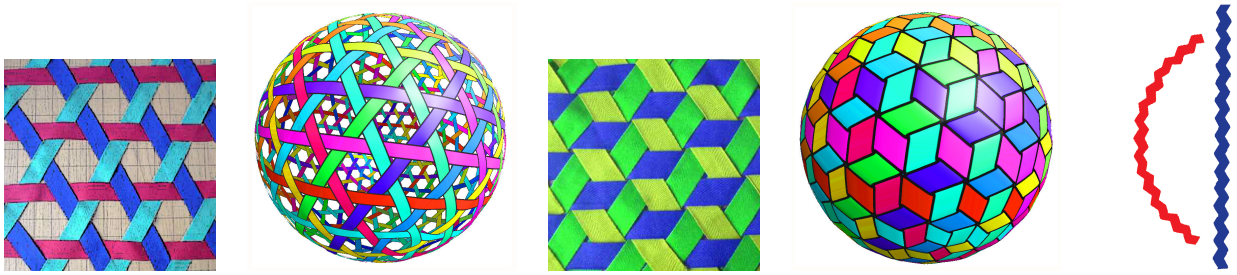
Keywords: Shape Modeling, Links and Knots, Weaving

ACM Reference Format
Akleman, E., Chen, J., Xing, Q., Gross, J. 2009. Cyclic Plain-Weaving on Polygonal Mesh Surfaces with Graph Rotation Systems. *ACM Trans. Graph. 28*, 3, Article 78 (August 2009), 8 pages.
DOI = 10.1145/1531326.1531384 http://doi.acm.org/10.1145/1531326.1531384.

1 Introduction

Knots and links are interesting structures that are widely used for tying objects together and for creating beautiful shapes such as woven baskets. To topologists, a knot is a 3D embedding of a circle and a link is a 3D embedding of more than one circle. We prefer to use the general term *link*, since each component of a link is also a knot. Mathematical links can be used to represent weaving structures such as a fabric, a cloth, or a basket. There exist a wide variety of weaving methods. Among them, the most popular is *plain-weaving*, which consists of threads that are interlaced so that a traversal of each thread alternately goes over and under the other threads (or itself) as it crosses them. To model a plain-weaving pattern on a surface, we construct an alternating projection of a link. We prove that it is possible to create such a plain-weaving pattern for any given manifold mesh by twisting all of the edges of a related orientable manifold mesh.

Our approach uses a combinatorial structure called *graph rotation systems*, which facilitate representation of linked knots by manifold mesh structures. They also provide a formalism for the development of tools for interactive modeling of mathematical knots and links. In the representation by graph rotation systems, an orientable manifold mesh corresponds to an *unlink*, which is a link in which for each component, there is an embedding of a sphere that separates that component from all the others. If we twist any of the edges of the orientable manifold mesh, we create a non-orientable mesh and two cycles become linked. Section 2 describes a theorem that by twisting all edges of a manifold mesh, we obtain an alternating link on the surface, which represents a plain-woven pattern. Our theoretical work can also be viewed as a formalization of methods for drawing Celtic knots using planar graphs [Mercat 2001; Kaplan



(a) Sparse Triaxial Weaving (b) Our method for Sparse Weaving (c) Dense Triaxial Weaving (d) Our method for Dense Weaving (e) Unfolded Wavy Ribbons for (d)

Figure 2: *(a) A photograph of real "sparse" triaxial weaving that leaves gaps (see the large hexagonal-shaped gaps). (b) Our projection method (PR) can create the same type of "sparse" weaving, by leaving gaps as shown. (c) A real "dense" triaxial weaving. This type of weaving is not very common, since it is difficult to manufacture. (d) Using our projection method (PR), we can also create dense triaxial weaving on any regular triangular mesh, as shown. This particular mesh consists of 18 cycles of ribbons. (e) Unfolded versions of these ribbons show that the ribbons are wavy and that there are only two types used in (d). The construction requires 12 from the circular-type ribbon (left) and 6 from the straight-type ribbon (right). Photographs in (a) and (c) are courtesy of Tim Tyler.*

and Cohen 2003].

A mathematical link is 1-dimensional, without solid shape. For practical applications, the components of a link need to be converted to 3D thread structures, such as extruded lines (ribbons) or extruded surfaces (yarns). The resulting 3D thread structures must also be "smooth-looking" and "non-self-intersecting". Section 3 presents our conversion method, which we call *projection*. The projection method is significantly different from Celtic-knot drawing methods [Mercat 2001; Kaplan and Cohen 2003; Kaplan et al. 2004], which are based on the extrusion of a line or circle with smooth C^1 or C^2 continuous arcs that are defined by free-form parametric curves. Celtic-knot drawing methods can create results strongly resembling familiar woven-basket structures, which are created using bendable but straight structures. These structures can leave large gaps in some weaving patterns, such as sparse triaxial weaving [Scardino and Ko 1981], as shown in Figure 2. Our projection method is developed to control the size of the gaps, so that we can obtain both sparse and dense plain-weaving. Using this method, we can cover the original manifold shape with almost no gaps, with ribbons whose unfolded versions are wavy as shown in Figure 2e.

As a practical example, we have developed a system that converts any manifold mesh to a plain-woven object. Our system converts the mathematical knots to 3D thread structures, such that the shapes of the threads can be interactively controlled with a set of parameters. These 3D structures can cover the original orientable manifold meshes without having large gaps. In Section 3.2, we provide examples of some plain-woven objects. Most importantly, with this system we can create a wide variety of plain-weaving patterns. Section 4 presents our approach for creating and classifying plain-weaving patterns. Section 5 gives our conclusions and describes possibilities for future work.

2 Graph Rotation Systems

Formally, a *cyclic plain-weaving* on an orientable surface S is a projection of a link L on S, such that (1) there are no triple intersections at a single point on S, and (2) a traversal of the image on S of each component of L goes over and under alternatingly as it crosses the images of other components or of itself.

Our theoretical framework for cyclic plain-weaving is based on the theory of graph rotation systems, which have been extensively studied in topological graph theory [Gross and Tucker 1987]. It is well-known [Edmonds 1960] that a graph rotation system uniquely determines a graph embedding on an orientable or non-orientable surface, and thus uniquely determines the surface. Some of the concepts related to graph rotation systems have been implicitly [Mantyla 1988; Baumgart 1972; Guibas and Stolfi 1985] and explicitly [Akleman and Chen 1999] studied in computer graphics. An important concept in graph rotation systems is edge twisting. An edge has type 0 if it is untwisted and type 1 if it is twisted [Gross and Tucker 1987].

Face-Tracing Algorithm.

(A slight revision of the algorithm given by Gross and Tucker)

Subroutine FaceTrace(u_0, w_0 , t_0)
u_0, w_0 is an oriented edge, t_0 $0, 1$ is the "trace type".

1. trace u_0, w_0 ;
2. $t = t_0 + type([u_0, w_0])\ (mod\ 2)$;
3. u, w = Next(w_0, u_0 , t); $u = w_0$
4. **while** (u, w = u_0, w_0) and ($t = t_0$) **do**
   trace u, w ;
   $t = t + type([u, w])\ (mod\ 2)$;
   u, w = Next(w, u , t).

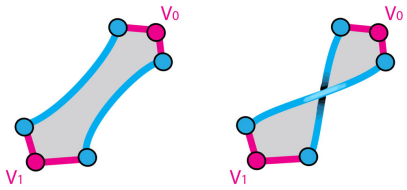
Algorithm Trace-All-Faces ($\rho(G)$)
$\rho(G)$ is a general graph rotation system.

while there is an untraced face corner (u, e, e') in $\rho(G)$ **do**
suppose that $e' =$ u, w ;
call FaceTrace(u, w , 0).

A fundamental algorithm on graph rotation systems, known as *face-tracing*, is given immediately above. This algorithm applied to a graph rotation system $\rho(G)$ on a surface S returns a collection of graph cycles that are the boundary-walks of the faces in $\rho(G)$, which induces a link projected on the surface S. For detailed explanation and discussion of the face-tracing algorithm see [Gross and Tucker 1987; Akleman et al. 2009].

In the *band decomposition* of a graph rotation system $\rho(G)$ on a surface S [Gross and Tucker 1987], each edge E in the graph G can be regarded as a flat paper strip, and the two sides of E are segments in the face boundary walks in $\rho(G)$. If we implement the operation "twisting E" by standing on one end of the paper strip

E and twisting the other end of E in clockwise direction by 180^0 (see Figure 3 for illustration), then the edge twisting operation corresponds to changing the face boundary walks in the graph rotation system, and inducing a new link projected on the surface S.



An untwisted edge A twisted edge

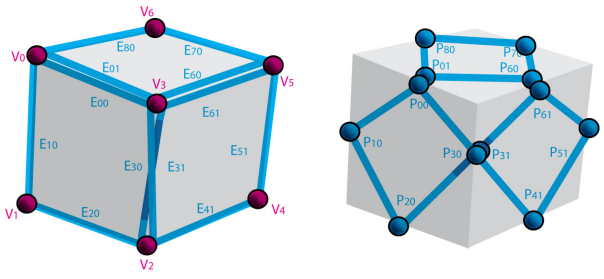
Figure 3: *Twisting an edge of a mesh on a surface S.*

To construct a cyclic plain-weaving on an orientable surface S_h, we start with a graph rotation system $\rho_0(G)$ with no twisted edges that determines a graph embedding on S_h. The face boundary walks of $\rho_0(G)$ form a collection of disjoint cycles on S_h, which we regard a projection of a link onto that surface. This is the initial weaving on S_h, in which each edge of G lies between two parallel strands on S_h. When we apply the edge-twisting operations on $\rho_0(G)$ as described above, we obtain a new graph rotation system $\rho(G)$. Under the face tracing algorithm, this new rotation system $\rho(G)$ induces a new collection of cycles, which is a new link projected on the surface S_h.

The following theorem is a foundation for our development of cyclic plain-weaving (see [Akleman et al. 2009] for a proof of the theorem):

Theorem 2.1 *Let $\rho_0(G)$ be a graph rotation system with no twisted edges, which corresponds to an embedding of the graph G on an orientable surface S_h. Let A be an arbitrary subset of edges of G. If we twist all edges in A, then the resulting graph rotation system induces a cyclic plain-weaving on S_h.*

The plain-weaving cycles that are created by twisting edges are mathematical links and do not have a solid shape. In order to create geometric forms, these cycles need to be converted to 3D thread structures, such as ribbons (extruded lines) or yarns (extruded polygons). The resulting 3D thread structures must look smooth and must not self-intersect.



(a) One edge-twisted Base Mesh (b) PR Control Polygon of (a)

Figure 4: *An illustration of the smooth curve creation process. (a) An initial mesh, with one twisted edge. (b) The corresponding PR control polygon.*

3 Creating 3D Thread Structures

In practice, we twist all the edges of graph G, instead of an arbitrary subset of edges, and for our rotation system $\rho_0(G)$, we only consider the most commonly used polygonal mesh surfaces in computer graphics. In our polygonal mesh surfaces, every vertex has valence at least 3, and every face has at least three sides (i.e. triangles). Moreover, every edge has positive length, and every vertex has position information.

Let E_i denote an edge of a manifold mesh, and let $E_{i,0}$ and $E_{i,1}$ denote the over and under half-edges (in the sense of [Mantyla 1988]) that lie close beside the twisted edge E_i. We assume that we can assign and compute a unit normal vector $\vec{n}_i$ for every edge E_i. The faces do not have to be flat, but we assume that for each face we have an approximating planar polygon that is given by a normal vector and a center point. The edge normal vector $\vec{n}_i$ can be computed as the average of the normals of the two approximating planes for the two respective sides of edge E_i. The faces do not have to be convex but if we project a face to its approximate plane from the center point of the face, then all projected edges must be visible. All these conditions eliminate degenerate faces, and they guarantee that we can have a normal vector defined for all the edges of the manifold mesh.

Our goal is to create dense weaves, such as the dense triaxial weaving shown in Figure 2. Extrusion methods are appropriate for drawing Celtic knots in a planar surface [Mercat 2001; Kaplan and Cohen 2003], but they are not suitable for covering an arbitrary surface since they leave large gaps, and they cannot create dense weaving for all possible weaving patterns. In order to be able to create a dense weaving on an arbitrary polygonal surface, we have developed the *projection method (PR)* that provides control of the size of the gaps in the weave. Moreover, by using our method, the unusual structure of some weaving patterns becomes more perceptible (this is demonstrated by the images in Figures such as 11(a) and 11(b) at the end of the paper).

3.1 Projection Method

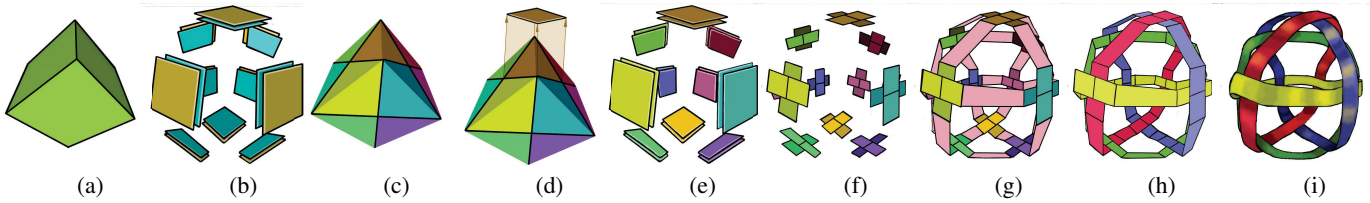
With projection method (PR), we create two types of 3D thread structures, which we call ribbons and yarns. To simplify the presentation, we also differentiate between control and smooth versions of these structures. Smooth ribbons are simply smoothly curved versions of control ribbons, which are cyclic chains of quadrilaterals. Similarly, smooth yarns are created from control yarns, which are toroidal meshes. The method is presented only for surfaces with all edges twisted. The algorithm of the projection method consists of four main steps. The first step is to construct a control polygon which we call the *PR control polygon*. Without loss of generality, we will explain the process using two examples on a cube that include mostly untwisted edges (see Figure 4). The second step is to construct a PR control ribbon from the PR control polygon. The third step is to create PR control yarns from PR control ribbon. The last step converts PR control ribbons and PR control yarns to smooth ribbons and smooth yarns. We explain the steps of the process using the example shown in Figure 5, using a cube-shaped manifold with all edges twisted.

Step 1. Construct PR Control Polygons:

1.1. *Trace all face boundary walks using the Face-Tracing Algorithm*: Figure 4a shows the face boundary walks. For instance, for the face with a twisted edge in Figure 4a, the computed face boundary walk is the cyclically ordered set

$$K_1 = \; E_{0,0}, E_{1,0}, E_{2,0}, E_{3,0}, E_{6,1}, E_{5,1}, E_{4,1}, E_{3,1}$$

1.2. *Assign a position $p_{i,j}$ to each half-edge $E_{i,j}$*: We compute $p_{i,j}$ by adding a displacement vector $\vec{v}$ to the average of the positions of two endpoints of $E_{i,j}$ as



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 5: *The steps of the projection method (PR). (a) The initial mesh, a cube, with all its edges twisted, to create a cyclic plain-weaving. (b) The projection planes. (c) Quadrilateral edge regions that are obtained from the two endpoints of the edge and centers of the two faces on the two sides of the edge. (d) A projection of an edge region to one of the its corresponding projection planes. (e) All the projected edge-regions. (f) Planar pieces that consist of two quadrilaterals. (g) Pink-colored quadrilateral connecters that connect two corresponding planar pieces. (h) The resulting control meshes with one consistent color for each cycle. (i) The final smooth ribbon that is created from the control mesh in (h).*

$$p_{i,j} = (p^1_{i,j} + p^2_{i,j})/2 + (-1)^j h e_i \vec{n}_i$$

where $p^1_{i,j}$ and $p^2_{i,j}$ denote the positions of two endpoints of $E_{i,j}$. The user controlled parameter h is a small positive real number that controls relative displacement. The quantity e_i is the length of the edge E_i, the parameter $\vec{n}_i$ is the normal vector to edge E_i, and $(-1)^j$ is $+1$ if the cycle segment $E_{i,0}$ is over and -1 otherwise.

1.3. *For each face boundary walk, construct a PR control polygon by replacing $E_{i,j}$ with $p_{i,j}$*: With this operation, a boundary walk such as K_1 turns into a polygon

$$P_1 = \; p_{0,0}, p_{1,0}, p_{2,0}, p_{3,0}, p_{6,1}, p_{5,1}, p_{4,1}, p_{3,1}$$

that we call the *PR control polygon* (see Figure 4b). Note that since $p_{3,0}$ is slightly above the cube surface and $p_{3,1}$ is slightly below, the polygon P_1 is not self-intersecting.

Step 2. Construct PR Control Ribbons:

2.1. *Assign a plane to each half-edge $E_{i,j}$*: The plane is given as $\vec{n}_i.(p - p_{i,j}) = 0$, where $\vec{n}_i$ is the normal vector to edge E_i. So, each edge has two corresponding planes, one slightly below the surface, and the other one slightly above. Figure 5b provides an exploded view to show both planes.

2.2. *Assign an edge-region to every edge E_i*: An edge region is defined as the non-planar quadrilateral that consists of the two end-vertices of the edge and the two centers of the two faces on the two sides of the edge. Figure 5c shows all of the edge regions for a cube.

2.3. *Project each edge-region to the two planes of its half-edges*: Figure 5d shows one projection; the projection creates two planar quadrilaterals for each edge as shown in Figure 5e.

2.4. *Compute a fractional quadrilateral from the projected quadrilateral, and subdivide the fractional quadrilateral into two quadrilaterals*: The fractional quadrilateral is computed as a fraction of the projected quadrilateral with user controlled fractional values c and w, using bilinear interpolation. Then this fractional quadrilateral is subdivided by creating two quadrilaterals along the thread, in the same direction as c. The whole process is illustrated in Figures 6(b). If the values of c and w are not the same, these shapes form crosses in space, as shown in Figure 5f.

2.5 *Connect these two-quadrilaterals with quadrilateral connectors using face trace order*: Figure 5g shows pink-colored quadrilateral connecters that connect their two corresponding planar pieces. The result is the PR Control Ribbon as shown in Figure 5h.

Step 3. Construct the PR Control Yarn:

A PR control yarn is constructed from a PR control ribbon in three steps, as shown in Figure 6(c):

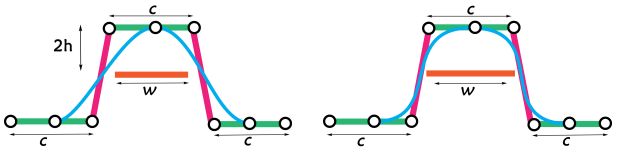
3.1 Define an ellipsoid for each internal edge of the quadrilaterals. The center of the ellipsoid is chosen to be the center of the internal edge. The semi-major and semi-minor axes are chosen to be $h/2\ e_i\ \vec{n}_i$ and half of the width of the internal edge.

3.2 Create an n-sided convex polygon by sampling the ellipsoid.

3.3 Construct a generalized toroid by connecting the n-sided convex polygons.

Step 4. Construct the PR Smooth Ribbons and Yarns:

PR smooth ribbons are constructed by smoothing PR control ribbons by cubic Beziér surfaces that use one connector and two side quadrilaterals as a control mesh. This approach guarantees that resulting piecewise smooth surfaces have G^1 continuity in boundaries. PR smooth yarns are constructed by smoothing PR control yarns using off-line Catmull-Clark Subdivision.



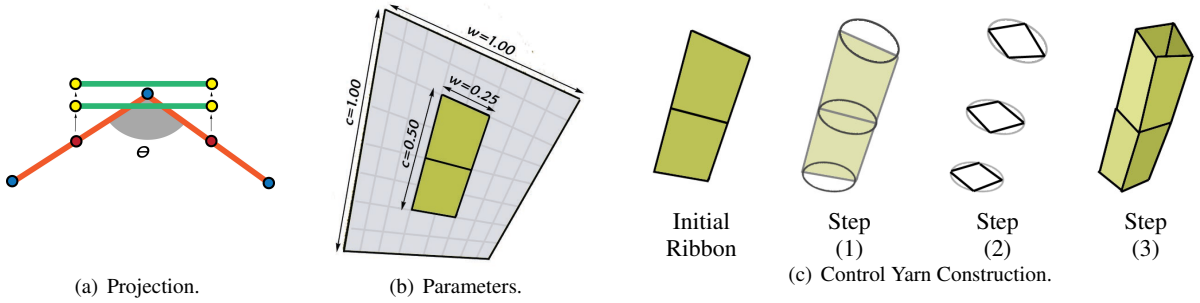
(a) Cubic Beziér (b) Quadric B-Spline

Figure 7: *A cross-sectional illustration of both upper and lower ribbons, showing the effect of w and c on collision avoidance. The green line represents the planar control part of the upper ribbon, and the orange line represents the lower ribbon. The pink line is the connector, and the blue curve is the cross-section of the upper ribbon. (a) A piecewise cubic Beziér. (b) A quadric B-Spline.*

3.2 Examples and Results

We have developed a system that converts polygonal meshes to cyclic plain-woven objects. A user can interactively change the parameters c, w and h to achieve different results. In the projection method (PR), a dense weaving is obtained with $c \approx 1$, with $w \approx 1$, and with relatively very small h values. Small values of c and w provide sparse weaving. All the woven-object images in this paper, except for Figures 8 and 9, are direct screen captures from the system; they were created in real-time. The colors of the thread cycles are randomly chosen. For the images in the paper we used saturated colors. Our PR algorithm guarantees that the sizes are relative to the underlying polygons. Therefore, the actual widths of ribbons are different in different parts of the mesh.

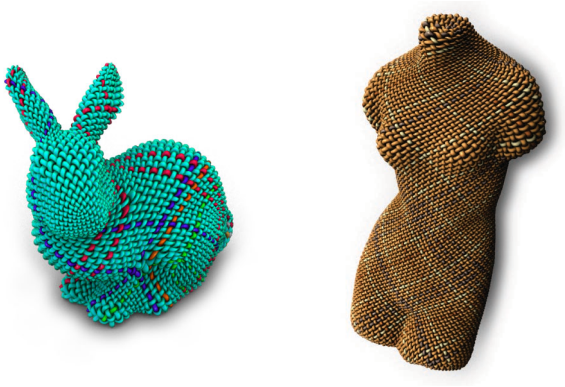
The projection method closes the gaps better if the angle θ between two faces on the two sides of the edge, as shown in Figure 6(a), is



(a) Projection. (b) Parameters. (c) Control Yarn Construction.

Figure 6: *(a) A cross-sectional view of a projection, where blue circles represent edges, orange lines represent the two faces on the two sides of an edge, and red points are the centers of these faces. Green lines represent two planes that correspond to the two half-edges of the given edge, and yellow points are projections of the red center points. (b) An illustration that shows the effect of parameters c and w. The parameter w controls the relative width of the ribbon and c controls the length of the planar region in the direction of the ribbon. (c) The construction of a PR control yarn.*

between 120^0 and 180^0. The closer that θ approaches 180^0, the better it is for closing gaps. This can be achieved with a few applications of a suitable subdivision scheme. For example, using the Doo-Sabin scheme, after each subdivision all θ values become closer to 180^0, while the faces become more nearly regular, more nearly convex, and more nearly planar. We do not provide direct collision avoidance; however, choosing values of c and w between 0 and 1 is sufficient to avoid collisions. As can be seen in Figure 7, the value of h is not particularly important for collision avoidance in smooth ribbons. For producing smooth ribbons, the quadric B-spline provides slightly better collision avoidance, however, visually we prefer cubic Beziér surfaces.



bunny Venus

Figure 8: *PR smooth yarns for the bunny and Venus models in Figure 1. These are obtained by smoothed PR control yarns with Catmull-Clark subdivision. The images are off-line rendered.*

4 Creating, Analyzing and Categorizing Plain-Weaving Patterns

Creating interesting weaving patterns is a problem similar to creating interesting periodic tilings on surfaces [Kaplan et al. 2004; Kaplan 2007]. Our approach is to create mesh patterns, by applying a variety of subdivision schemes one after another. In this way, it is possible to populate a polygonal mesh with all possible regular and semi-regular tilings [Akleman et al. 2005], which can then be converted to interesting weaving patterns [1]. One advantage of constructing cyclic plain-weaving patterns in this way is that we can also analyze and categorize weaving patterns based on the regular and semi-regular tiling patterns of the initial meshes.

The cyclic plain-weaving obtained by twisting of all the edges of an orientable manifold mesh surface consists of visibly quadrilateral ribbon pieces, which correspond to the "upper" pieces at the crosses of the two sides of the twisted edges. Each quadrilateral ribbon piece is surrounded by four gaps, two of which correspond to the two ends of the edge and the other two to the faces on the two sides of the edge. The number of ribbons around a gap defines the shape of the gap. We call this *the valence of the gap*. Since we create cyclic weaving on polygonal surfaces in which every face has three or more sides, and in which every vertex has valence at least three, in our examples gap valences are always at least three. If in the initial mesh, the two endpoints of an edge e have valences d_0 and d_2, respectively, and the two faces on the two sides of the edge e have numbers of sides d_1 and d_3, respectively, then the four gaps around the ribbon piece have valences d_0, d_1, d_2, and d_3. The valences of the four gaps around a ribbon piece can be given using a Schlafli-like notation with a four-tuple (d_0, d_1, d_2, d_3), which can be used to categorize the structure of the weaving pattern. Since all faces and vertices in the initial mesh correspond to gaps in the weaving, the dual of the mesh will result in the same cyclic plain-weaving. Based on this background, we can create and analyze the weaving patterns that are constructed using polygonal meshes, i.e., we ignore cases that include 2-valent gaps.

4.1 Regular Plain-Weaving Patterns

We call weaving pattern *regular* if all gaps have the same valence. By the above discussion, if the initial mesh consists entirely of faces of valence d and vertices of valence d, then the gaps in the resulting cyclic plain-weaving will all be of valence d. If the initial mesh consists entirely of d-sided faces and d-valent vertices, then the resulting gaps will all be d-sided. Therefore, the problem becomes

[1] In the Euclidean plane, there are only two distinct regular tiling patterns and seven distinct semi-regular tiling patterns and their duals. These tiling patterns can be described by the Schlafli notation [Akleman et al. 2005]. In Schlafli notation, $(3, 3, 4, 3, 4)$ is a semi-regular tiling that consists entirely of pentagons, and the valences of the vertices of every pentagon follow the cyclic pattern $3, 3, 4, 3, 4$. In the dual of this tiling, all the vertices are 5-valent, and the number of sides of the cycle of polygons around each vertex follows that same cyclic pattern: triangle, triangle, quadrilateral, triangle, quadrilateral. The same notation can represent regular tiling patterns, e.g., $(6, 6, 6)$ is a triangular tiling in which each vertex has valence 6.

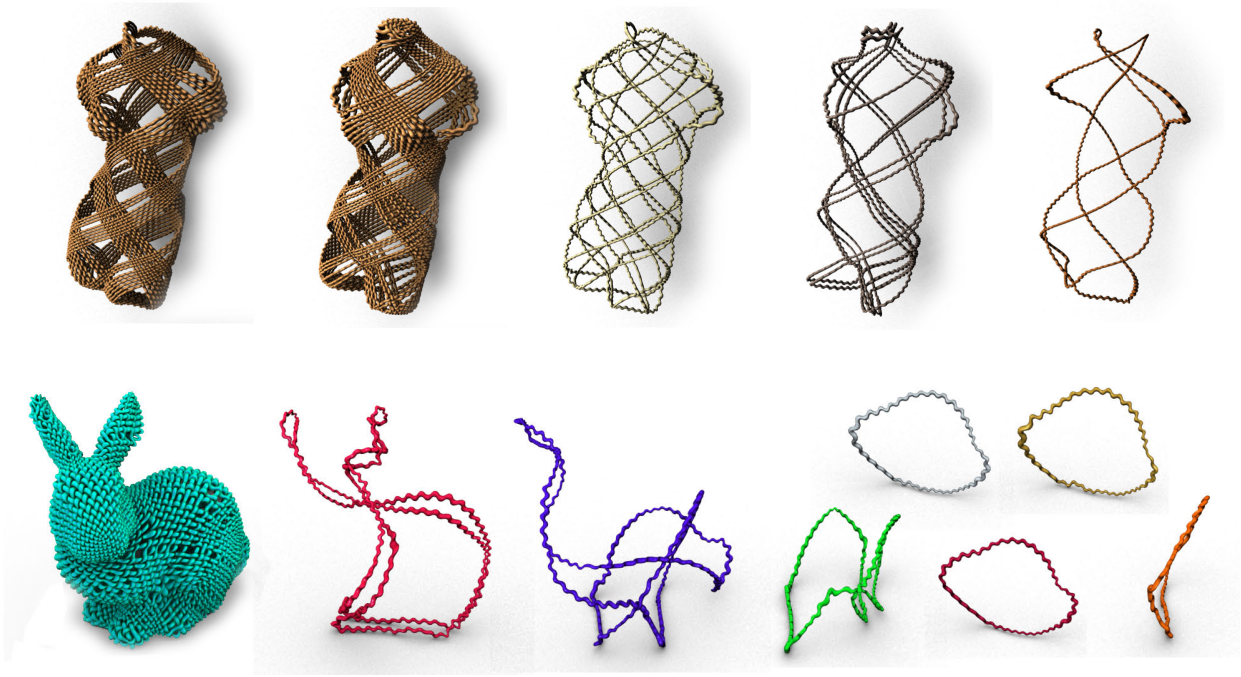


Figure 9: *These off-line rendered images show all five cycles of the Venus model and and all eight cycles of the bunny model in Figure 8.*

that of designing initial meshes on surfaces that have both face valence d and vertex valence d.

Let g denote the genus of surface underlying the initial mesh. For $g = 0$, there is only one mesh in which all faces are d-sided and all vertices d-valent, which is the tetrahedron, with $d = 3$. For the tetrahedron, the resulting plain-woven object is the Borromean rings, which is the simplest regular case. There are no other regular weaving pattern for $g = 0$. The case of genus $g = 1$ corresponds both to a toroid and to an infinite plane. For $g = 1$, valence $d = 4$ corresponds to a regular quadrilateral tiling of the infinite plane (See Figure 10a). This particular pattern is very useful since we can populate any manifold mesh, regardless of its genus, with mostly quadrilaterals and 4-valent vertices, simply by applying several iterations of vertex insertion schemes such as Catmull-Clark [Catmull and Clark 1978], corner-cutting schemes such as Doo-Sabin [Doo and Sabin 1978], and Simplest [Peters and Reif 1997] or its dual stellation with edge removal [Zorin and Schröder 2002]. This tiling pattern converts to a weaving pattern $(4, 4, 4, 4)$, which again can be considered as quadrilaterals with 4-valent vertices. It is therefore possible to create an object with mostly a $(4, 4, 4, 4)$ weaving pattern for any shape and any genus, by introducing just a few extraordinary gaps, i.e., gaps that do not have valence 4. Many of the examples in this paper are mostly $(4, 4, 4, 4)$ non-genus-1 meshes converted to a plain-woven object. Two genus-1 examples are shown in Figure 10. For every genus higher than 1, cases with $d = 5, 6, 8$, and 12 exist; however, it is not possible to populate a mesh with $d = 5, 6, 8$, and 12 without increasing the genus [Akleman and Chen 2006]. Hence, the case $d = 4$ is the only regular weaving pattern that can be used for any genus with only a few extraordinary gaps. Moreover, the dense $(4, 4, 4, 4)$ weaving pattern can physically be obtained using plant branches, such as wicker or rattan. These two facts together may help explain the overwhelming popularity of the $(4, 4, 4, 4)$ weaving pattern in basket-making.

(a) (b) (c)

Figure 10: *Weaving patterns obtained from (a-b) only (4,4,4,4) and (c) mostly (4,4,4,4) meshes.*

4.2 Semi-Regular Plain-Weaving Patterns

We call a weaving pattern *semi-regular* if the cycle of valences (d_0, d_1, d_2, d_3) is the same for every visible ribbon piece. The platonic solids other than the tetrahedron result in semi-regular weaving patterns. Since a mesh and its dual produce the same type of woven object, the octahedron and the cube can be converted into the same type of the woven object, which we will classify as $(4, 3, 4, 3)$; and similarly, the dodecahedron and the icosahedron are converted into an object with a semi-regular $(5, 3, 5, 3)$ weaving pattern. The more interesting cases are the ones that correspond to planar semi-regular tilings, since it is again possible to populate any-genus mesh with such tilings. We have identified two such cases that correspond to weaving patterns, which are known as triaxial and ring weaving.

Triaxial: semi-regular $(6, 3, 6, 3)$ *weaving patterns obtained from mostly* $(6, 6, 6)$ *regular structures.* Meshes with mostly $(6, 6, 6)$ regular structures are meshes that consist mostly of triangles with 6-valent vertices or of hexagons with 3-valent vertices. The weaving pattern that is obtained by such meshes is called *triaxial*. Mostly $(6, 6, 6)$ meshes can easily be obtained by reiteratively applying subdivisions such as Loop [Loop 1987] or $\sqrt{3}$ subdivision [Kobbelt

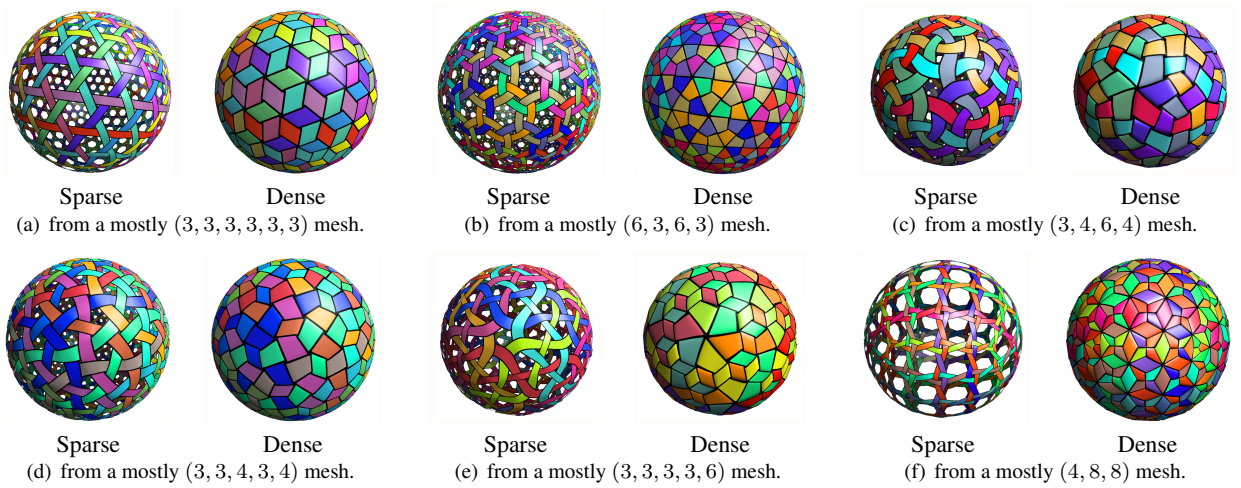


Figure 11: *Examples of weaving patters obtained from mostly regular and semi-regular meshes. The Figures 11(a) and 11(b) show two semi-regular weaving patterns. The rest of the patterns are not semi-regular.*

2000] and their dual schemes [Oswald and Schröder 2003]. When we convert such meshes to woven objects, the resulting weaving patterns shown in Figure 11(a) visually correspond to $(6,3,6,3)$ type meshes [Akleman et al. 2005].

Ring Weaving: semi-regular $(6,4,3,4)$ *weaving patterns obtained from mostly* $(6,3,6,3)$ *semi-regular meshes.* Ring weaving is just a derivative of the $(6,3,6,3)$ case. If we have a mostly $(6,3,6,3)$ semi-regular mesh, it is converted to a plain-weaving pattern that corresponds to $(6,4,3,4)$-type tilings. A mesh with a $(6,3,6,3)$ semi-regularity can easily be obtained by applying a stellation with edge-removal operation only once to a mesh that consists of mostly hexagons with 3-valent vertices or of mostly triangles with 6-valent vertices. Figure 11(b) shows an example of mostly $(6,4,3,4)$ that is obtained from a mesh that is mostly $(6,3,6,3)$.

4.3 Other Interesting Patterns

Regular and semi-regular weaving patterns come from two regular tiling patterns and one semi-regular tiling pattern. The other six semi-regular tiling patterns can also provide some interesting weaving patterns. If we have a mesh that consists mostly of one of these six planar semi-regular tiling patterns, we can still perceive some interesting weaving patterns which are easy to classify, based on the semi-regular tiling patterns of the initial mesh. Here we provide a few examples of such interesting weaving patterns.

Weaving Patterns obtained from meshes that contain mostly $(3,4,6,4)$ *semi-regular mesh structures.* A mesh that contains mostly $(3,4,6,4)$ structures can be obtained by applying a vertex insertion scheme such as Catmull-Clark [Catmull and Clark 1978] to a mesh that contains mostly $(6,6,6)$. The resulting weaving patterns are not semi-regular but they can easily be recognized as having a character as shown in Figure 11(c).

Weaving Patterns obtained from pentagonal meshes that contain mostly $(3,3,4,3,4)$ *or* $(3,3,3,3,6)$ *semi-regular structures.* The semi-regular pattern $(3,3,4,3,4)$ can be obtained by applying pentagonal subdivision [Akleman et al. 2004] to a mostly $(4,4,4,4)$ mesh and $(3,3,3,3,6)$ can be obtained by applying pentagonal subdivision to a mostly $(6,6,6)$ mesh. (See Figures 11(d), and 11(e)).

Weaving Patterns obtained from mostly $(4,8,8)$ *and* $(3,12,12)$ *meshes.* If the vertex valence exceeds 6, then the resulting pattern starts to look like it consists of flowers. $(4,8,8)$ and $(3,12,12)$ meshes, which can be obtained using vertex truncation, can be converted into nice flower patterns, as shown in Figure 11(f).

The procedures to create other semi-regular tiling patterns are presented in [Akleman et al. 2005]. All examples in this section are created by using TopMod3D [Akleman et al. 2008], a topological modeling software system that is freely available at www.topmod3d.org.

5 Conclusion and Future Work

This paper introduces a new method of creating plain-weaving structures based on graph rotation systems. With the graph rotation system structures, we have formally demonstrated that by twisting a subset of edges of an orientable manifold mesh, we can obtain an alternating link, which is the mathematical model for a plain-weaving. Based on this result, we have developed the projection method (PR) to convert a link projection on a polygonal surface to a plain-woven object. The projection method is described here only for surfaces with all edges twisted. This method can be further extended for other implementations of edge-twisting, and it can be used to create a much wider variety of 3D links. Not only can our system provide traditional woven objects, but it can also create unusual woven patterns that may not have been used before, since some of these patterns cannot be assembled by hand. Thus, another advantage is that we are not bound by the limitations of hand-weaving. We can use wavy ribbons of varying width that can almost completely cover a surface, without leaving large gaps.

Although a woven-look can be achieved by using texture maps, having 3D geometry allows us to achieve more realism in interactive rendering, such as real-time shadows and the capacity to change the width of the ribbons in real-time. Moreover, our smooth-yarn models can be printed using a 3D printer, and our ribbon models can be cut using laser cutting and physically constructed. We are planning to create large sculptures in collaboration with architects and sculptors, such as the sculptor James Mallos, who has recently constructed a large triaxial woven sculpture of a fingertip [Mallos 2009], using a Mercat type algorithm on a manifold mesh surface with a boundary.

The projection method (PR) can be used to create 3D thread structures for a much wider variety of linked knots than polygonal meshes with edges twisted in the way described in the current paper. For example, the edge twisting can also be implemented by twisting an edge end in counterclockwise direction instead of in clockwise direction, or twisting some edges in clockwise direction and some edges in counterclockwise direction. Even more general, the edge twisting can be classified as k^+ or k^- twistings (i.e., edges are twisted in clockwise or counterclockwise directions by $k \times 180^0$). These generalized twistings can be represented by subdividing an edge k times (thereby creating $k-1$ new 2-valent vertices) and twisting each one of the resulting edges in clockwise or counterclockwise direction by 180^0 . In this paper, we restricted to vertex valences larger than 2, but with a minor modification, the underlying projection method (PR) can accept any 2-valent vertex. Since the projection method (PR) as presented here represents the traces of the sides of an edge as two planar pieces with one atop the other, not side-by-side as in the untwisted case, untwisted edges cannot be used. However, untwisted edges could be included with a relatively minor extension, which we have omitted here to simplify the description.

The power of the projection method is essential, since our theoretical framework already allows the construction of a much wider variety of weavings than only those that can be created from a manifold mesh with all edges twisted. For instance, by twisting a proper subset of the edges instead of all of them, it is possible to control the number of cycles. There exist several approaches to creating single-cycle plain-woven objects. Moreover, we observe that twisting an arbitrary subset of edges creates visual results that are more similar to Celtic knots than to weaving. By applying negative and positive twists on different subsets of edges, it is possible to create weaving structures such as *twill* or *satin*. Extended graph rotation systems provide theoretical tools for exploring, discovering and proving such ideas. It will be exciting to investigate further applications and extensions of graph rotation systems.

We are grateful to Don House and Scott Schaefer and to several anonymous reviewers for their corrections and suggestions, to Wenping Wang and Li Yupei for providing us with the bunny and Venus models, to Christine Liu for her creation of off-line rendered images, to Robert Graf for his help in the creation of video supplement, and to Tim Tyler for allowing us to use his photographs of sparse and dense triaxial weavings. We are also thankful to John Keyser, Gary Greenfield, Fred Parke, and Lou Tassinary for their insightful comments.

References

AKLEMAN, E., AND CHEN, J. 1999. Guaranteeing the 2-manifold property for meshes with doubly linked face list. *International Journal of Shape Modeling 5*, 2, 149–177.

AKLEMAN, E., AND CHEN, J. 2006. Regular mesh construction algorithms using regular handles. In *Proc. of SMI 2006*, 171–181.

AKLEMAN, E., SRINIVASAN, V., MELEK, Z., AND EDMUNDSON, P. 2004. Semi-regular pentagonal subdivision. In *Proc. of SMI 2004*, 110–118.

AKLEMAN, E., SRINIVASAN, V., AND MANDAL, E. 2005. Remeshing schemes for semi-regular tilings. In *Proc. of SMI 2005*, 44–50.

AKLEMAN, E., SRINIVASAN, V., CHEN, J., MORRIS, D., AND TETT, S. 2008. Topmod3d: An interactive topological mesh modeler. In *Proc. of CGI 2008*, 10–18.

AKLEMAN, E., CHEN, J., GROSS, J., AND XING, Q. 2009. Graph rotation systems as a model for cyclic weaving on orientable surfaces. Technical Report TR 2009-4-4, Computer Science Department, Texas A&M University.

BAUMGART, B. J. 1972. *Winged-edge polyhedron representation*. Master s thesis, Technical Report CS-320, Stanford University.

CATMULL, E., AND CLARK, J. 1978. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design*, 10, 350–355.

DOO, D., AND SABIN, M. 1978. Behavior of recursive subdivision surfaces near extraordinary points. *Computer Aided Design*, 10, 356–360.

EDMONDS, J. 1960. A combinatorial representation for polyhedral surfaces. *Notices American Mathematics Society*, 7, 646.

GROSS, J. L., AND TUCKER, T. W. 1987. *Topological Graph Theory*. Wiley Interscience, New York.

GRUNBAUM, B., AND SHEPHARD, G. 1987. *Tilings and Patterns*. W. H. Freeman and Co, NY.

GUIBAS, L., AND STOLFI, J. 1985. Primitives for the manipulation of general subdivisions and computation of voronoi diagrams. *ACM Transaction on Graphics*, 4, 74–123.

KALBERER, F., NIESER, M., AND POLTHIER, K. 2007. Quadcover - surface parameterization using branched coverings. 10–19.

KAPLAN, M., AND COHEN, E. 2003. Computer generated celtic design. In *Proc. of 14th Eurographics Workshop on Rendering*, 9–16.

KAPLAN, M., PRAUN, E., AND COHEN, E. 2004. Pattern oriented remeshing for celtic decoration. In *Proc. of Pacific Graphics 2004*, 199–206.

KAPLAN, C. S. 2007. Semiregular patterns on surfaces. In *Visual Proc. of ACM SIGGRAPH 2007: sketches*, 78.

KOBBELT, L. 2000. $\sqrt{3}$-subdivision. In *Proc. of SIGGRAPH 2000*, 103–112.

LOOP, C. 1987. *Smooth Subdivision Surfaces Based on Triangles*. Master s thesis, University of Utah.

MALLOS, J., 2009. How to weave a basket of arbitrary shape. Accepted to ISAMA 2009; also see http://jamesmallos.blogspot.com/.

MANTYLA, M. 1988. *An Introduction to Solid Modeling*. Computer Science Press, Rockville, Ma.

MERCAT, C. 2001. Les entrelacs des enluminure celtes. *Dossier Pour La Science 15* (January).

OSWALD, P., AND SCHRÖDER, P., 2003. Composite primal/dual $\sqrt{3}$-subdivision schemes. Computer Aided Geometric Design, CAGD.

PETERS, J., AND REIF, U. 1997. The simplest subdivision scheme for smoothing polyhedra. *ACM Transactions on Graphics 16*, 4, 420–431.

SCARDINO, F. L., AND KO, F. K. 1981. Triaxial woven fabrics. *Textile Research Journal 51*, 2, 80–89.

ZORIN, D., AND SCHRÖDER, P., 2002. A unified framework for primal/dual quadrilateral subdivision schemes. Computer Aided Geometric Design, CAGD.