Geometric modeling of knitted fabrics using helicoid scaffolds

Paras Wadekar¹, Prateek Goel¹, Chelsea Amanatides², Genevieve Dion², Randall D Kamien³ and David E Breen¹

Abstract
We present a bicontinuous, minimal surface (the helicoid) as a scaffold on which to define the topology and geometry of yarns in a weft-knitted fabric. Modeling with helicoids offers a geometric approach to simulating a physical manufacturing process, which should generate geometric models suitable for downstream mechanical and validity analyses. The centerline of a yarn in a knitted fabric is specified as a geodesic path, with constrained boundary conditions, running along a helicoid at a fixed distance. The shape of the yarn’s centerline is produced via an optimization process over a polyline. The distances between the vertices of the polyline are shortened and a repulsive potential keeps the vertices at a specified distance from the helicoid. These actions and constraints are formulated into a single “energy” function, which is then minimized. The yarn geometry is generated as a tube around the centerline. The optimized configuration, defined for a half loop, is duplicated, reflected, and shifted to produce the centerlines for the multiple stitches that make up a fabric. In addition, the parameters of the helicoid may be used to control the size and shape of the fabric’s stitches. We show that helicoid scaffolds may be used to define both knit and purl stitches, which are then combined to produce models of all-knit, rib, and garter fabrics.

Keywords
Geometric modeling, knitted fabrics, bicontinuous surfaces, helicoids, energy minimization, textile modeling

Date received: 17 January 2020; accepted: 27 February 2020

Introduction
The calculus of variations is the cornerstone of classical mechanics, elasticity theory, and modern economics. When problems are formulated as optimization problems, the equations governing motion, bending, or trade (respectively) describe critical points of the objective function.¹ When applied to geometric objects, this perspective is especially powerful. When the objective depends on geometric quantities, the minima, maxima, and other extrema are likewise geometric. Functionals of length lead to geodesic equations (shortest length), while functionals of area lead to minimal surfaces. Soap bubbles, for instance, minimize their area subject to a volume constraint leading to Plateau’s classic rules for foams.²³ Minimal surfaces also appear in biological and self-assembled systems where surfactants and lipids create area-minimizing, space-filling structures, so called triply periodic minimal surfaces (TPMS).⁴ Examples of these include the plumber’s nightmare, the gyroid, and the diamond surface. All three of
these surfaces divide space evenly into two regions and can be thought of as a pair of interwoven passages with sixfold, threefold, and fourfold connectivity, respectively.\(^5\)

Since minimal surfaces are the solutions to many extremal problems in physics,\(^6,7\) we posit that they may be used to define the topology and shape of yarns in a weft-knitted fabric. In previous works,\(^8,9\) we demonstrated, with physical prototypes, how yarns of a weft-knitted fabric may lie on a scaffolding of alternating left- and right-hand helicoids, a type of minimal surface, with the form of the helicoids producing the characteristic spatial relationships between the yarns. Here, we present the mathematics and algorithms that create geometric models of the yarns making up a weft-knitted fabric, which exploit the lattice-like structural features of bicontinuous helicoid surfaces. The centerline of the yarn is specified as a geodesic path, with constrained boundary conditions, running along a helicoid at a fixed distance. The yarn geometry is then generated as a tube around the centerline. The helicoid therefore acts as a scaffold on which to define the shape of the yarns and their intertwinings. In addition, the structure of the helicoids maintains the proper spatial and topological relationships between yarns in the fabric.

The shape of a yarn’s centerline is produced via an optimization process over a polyline. The polyline is initially placed over a helicoid in the approximate configuration that will define a half loop of a stitch. The distances between the vertices of the polyline are shortened and a repulsive potential keeps the vertices at a set distance from the helicoid. In addition, the locations of the polyline’s endpoints are constrained. This process effectively models the shrinking of the initial polyline, while performing collision detection/avoidance with the scaffold surface, producing a geodesic path along the helicoid. These actions and constraints are formulated into a single “energy” function, which is then minimized.\(^10\) The optimization process modifies the vertices to produce a minimum energy configuration that balances the inter-vertex stretching energy with the repulsive energy from the helicoid. This configuration, defined for a half loop, is then duplicated, reflected, and shifted to produce the centerlines for the multiple stitches that make up a fabric.

We see helicoid scaffolds providing a framework for simulating the weft-knitting process and for modeling/predicting the topological structure and geometric shape of the yarns in the resulting fabric. The work here lays the technical foundation for a planned effort to simulate weft-knitting with helicoid segments that represent needles in a knitting machine. The model will offer a geometric approach to simulating a physical manufacturing process, which should generate geometric models suitable for downstream mechanical and validity analyses.

**Related work**

The early work on modeling knitted textiles focused primarily on defining and analyzing the geometric structure of knit stitches.\(^11-13\) Remarkably, this work was done without the mathematical infrastructure of splines,\(^14\) which was not widely available until the 1970s. Much later work did utilize splines to describe the centerlines of yarn geometry in knitted materials.\(^15-19\) Follow-on research applied minimum energy analysis to determine the shape of relaxed yarn loops in individual stitches,\(^20-23\) as well as larger bulk properties of plain-knitted fabrics.\(^24-26\) This work was extended by Kyosev et al.\(^27\) to include the compression of the yarns in the loop. Sherburn\(^28\) developed a modeling approach/system that works on microscopic, mesoscopic, and macroscopic scales to predict the mechanical properties of textiles. It numerically models the behavior of yarns and the interactions between them to produce simulation results. Duhovic and Bhattacharyya\(^29\) simulated the knitting process in order to understand how each of a yarn’s deformation mechanisms contributes to the overall deformation energy/behavior of a yarn in a knitted fabric.

The first system to model and visualize complete knitted fabrics was developed by Meissner and colleagues.\(^30,31\) Their system (KnitSim) accepts Stoll knitting machine commands (knit a stitch, transfer loops between beds, and rack the beds) and simulates the knitting process to produce an explicit topological representation of a knitted fabric. By making assumptions about the length of yarns between yarn crossings, a relaxation process is run to produce a realistic two-dimensional (2D) geometric layout for the fabric.

In ground-breaking work, Kaldor et al.\(^32,33\) simulated complete swatches and articles of clothing consisting of knitted fabrics by modeling the geometry and physics of individual yarns in these items. This work was extended by Yuksel and colleagues\(^34-36\) to produce Stitch Meshes, an approach to generating Kaldor-style, yarn-level geometric models of knitted clothing from polygonal models that represent the clothing’s surface. Aspects of this work were utilized by Leaf et al.\(^37\) to produce an interactive design tool for simulating yarn-level patterns for knit and woven textiles.

Cirio et al.\(^38\) defined a topological representation of knits consisting of a limited set of stitches, in contrast to the yarn-geometry-based approach of Kaldor et al. They developed a mechanical model based on the representation for the simulation of knitted clothing. Liu et al.\(^39-42\) performed finite element modeling (FEM) simulations of knitted fabrics with solid yarn-level geometric models. Poincloux et al.\(^43\) utilized an empirical model to simulate the mechanical behavior of knitted fabrics.

Techniques have also been developed for generating production data directly from surface models for the manufacture of knitted artifacts. Igarashi et al.\(^44,45\) developed a technique for generating hand-knitting patterns from three-dimensional (3D) models, along with an interactive design and visualization system. McCann and colleagues\(^46-48\) have developed algorithms that can generate knitting machine commands based on a variety of polygonal...
models as input. Their algorithms allow for the interactive design of 3D shapes, which can then be manufactured on a commercial knitting machine.

Our work stands apart from previous work in that we show that the problem of generating geometric models of yarns in a knitted fabric may be solved as a geometric problem itself, without the need for complex physical simulations of the knitting process. The natural, parameterized shape of the yarns is produced automatically by laying them along minimal surfaces. The helicoid scaffolds directly enforce the correct topology of the yarns without recreating the knitting processes that generate the knitted fabric. Changes in the stitch pattern, such as rib or garter, also necessarily result in an appropriate change in the loop shapes and topological relationships to neighboring loops. These conditions are enforced by maintaining the distinct yarn paths on separate sides of the helicoids.

**Helicoids and the weft knit structure**

Knit fabrics consist of a series of intermeshing loops of yarn and can be produced by hand or by machine. Each individual loop in the fabric is known as a *stitch*, and these stitches can be varied by the direction in which the loops are formed. A *knit* stitch is created when the yarn is drawn through a previous loop from back to front, and a *purl* stitch is created when this process occurs instead from front to back. These stitch types are structurally symmetric; therefore, a fabric made with all knit stitches is topologically the same as a fabric made with all purl stitches. Looking at the yarn patterns visible on the surface of the fabric, knit stitches can be identified by a “v” shape, while purl stitches can be identified by a horizontal line of half circle shapes (see Figure 1(a) and (b)).

These two stitches can be combined in numerous patterns to form fabrics with wide variations in structure and behavior.

Two helicoids positioned side-by-side and twisting in opposite directions provide the scaffolding needed to define the loop of yarn in a single knit stitch (see Figure 2). These small sections of helicoids can be stacked on each other and replicated in columns to provide the scaffolding needed for the stitches in a knitted fabric, as seen in Figure 3.

When defining fabrics consisting of just one stitch type, either knit or purl, a single row of paired helicoids is sufficient for laying out the geometry (see Figure 4(a)). When creating fabrics consisting of a mix of knit and purl stitches, two rows of reflected helicoids are necessary. In order to represent a rib fabric on helicoids, the yarn must travel from a “front” helicoid pair, for a knit stitch, to a “back” helicoid pair for a purl stitch. The transition from the front helicoids to the back occurs over the “legs” of the loop (see Figure 4(b)). For garter fabrics, the yarn must also cross from one row of helicoids to another, but in this case the legs of each loop are attached to one row of helicoids and its head is attached to the other (see Figure 4(c)).

**Helicoid mathematics**

While a helicoid can be defined as the Riemann surface of the natural logarithm of a complex variable, for this work...
we equivalently define it as a biparametric surface in polar coordinates, \((r, \theta)\), \(r \in [-R, R]\), \(\theta \in [\theta_0, \theta_1]\). In this formulation, a point on helicoid \(H\) centered at the origin and extending along the Z-axis in Cartesian coordinates is defined by the following

\[
H(r, \theta) = [r \cos \theta, r \sin \theta, c] \tag{1}
\]

where \(c\) is a scaling factor that may be used to adjust the height of the helicoid and the distance between each cycle of the surface. Each span of \(\pi\) radians in \(\theta\) produces a single cycle of the surface, that is, a unique, non-repeating set of \([x, y]\) points. The surface’s tangents along its radial and angular directions are defined as the following

\[
\frac{\partial H}{\partial r} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} & \frac{\partial z}{\partial r} \end{bmatrix} = [\cos \theta, \sin \theta, 0] \tag{2}
\]

\[
\frac{\partial H}{\partial \theta} = \begin{bmatrix} \frac{\partial x}{\partial \theta} & \frac{\partial y}{\partial \theta} & \frac{\partial z}{\partial \theta} \end{bmatrix} = [-r \sin \theta, r \cos \theta, c] \tag{3}
\]

Figure 5 presents a helicoid with parameter values \(r \in [-2, 2]\), \(\theta \in [0, 2\pi]\), and \(c = 1.11 \left(\frac{7}{2\pi}\right)\). It contains two complete cycles. Note that a given value of \(\theta\) defines a straight line on the surface parallel to the X-Y plane and rotated \(\theta\) degrees from the Y-axis. As \(\theta\) increases, this straight line sweeps out the surface as it moves in the Z direction.

**Distance from helicoid**

In order to find an optimal path for the yarn on the helicoid, the minimum distance from each point on the yarn to the helicoid needs to be computed. The yarn path is initially defined as a Catmull-Rom spline.\(^{14}\) The spline is discretized into a finite number of points, thus approximating the yarn path with a polyline.

**Approximate distance**

In order to speed up the optimization calculation, we developed an algorithm for computing an approximate
distance to the helicoid. In Figure 6, \( P \) is any point within distance \( R_h \) from the Z-axis. Let \( Q \) be the point on the helicoid that is closest to \( P \), and \( Q' \) be the closest point on the helicoid with the same X and Y coordinates as \( P \). The length of line \( PQ' \) can be computed as

\[
\theta' = \frac{1}{\tan^{-1}\left(\frac{P_y}{P_x}\right)}
\]

(4)

\[
d' = (P_z - c\theta') \mod \frac{c}{2}
\]

(5)

\[
dist(P, Q') = \begin{cases} 
\frac{c}{2} - d', & \text{if } d' \geq \frac{c}{4} \\
\frac{c}{2}, & \text{otherwise}
\end{cases}
\]

(6)

where \( \theta' \) is the angular parameter for \( P \) and \( Q' \), and \( c\theta' \) is the \( z \) value for \( \theta' \) in the helicoid’s first cycle. Applying the modulus ensures that the distance \( d' \) is computed in the correct cycle, and equation (6) handles the case when \( Q' \) is above \( P \) rather than below it.

The normal to the helicoid surface can be computed as the cross product of the two tangents, given in equations (2) and (3)

\[
n = \frac{\vec{H}}{\partial r} \times \frac{\vec{H}}{\partial \theta}
\]

(7)

\[
n = c \sin \theta \hat{i} - c \cos \theta \hat{j} + r \hat{k}
\]

(8)

where \((r, \theta)\) are the polar coordinates of the point \( Q \). Since we do not have the exact location of \( Q \), we will use the X and Y coordinates of point \( P \) to compute \( r \)

\[
r \approx \sqrt{P_x^2 + P_y^2}
\]

(9)

The line \( PQ' \) is parallel to the Z-axis. Hence, the angle between the normal and the Z-axis is given by the inverse cosine of the dot product of the normalized vectors

\[
\phi = \cos^{-1}\left(\frac{n \cdot \hat{k}}{|n| |\hat{k}|}\right)
\]

(10)

\[
\phi = \cos^{-1}\left(\frac{r}{\sqrt{c^2 + r^2}}\right)
\]

(11)

As seen in Figure 6, line \( QQ' \) lies approximately along the surface of helicoid, while line \( PQ \) lies along the normal to the surface. This means that \( \Delta PQQ' \) is close to being a right triangle. Now, the distance between points \( P \) and \( Q \) can be approximated using equations (6) and (11) as follows

\[
dist(P, Q) \approx dist(P, Q') \cos \phi
\]

(12)

This only approximates the distance between the point \( P \) and the surface above or below it, but not the distance to the helicoid’s central axis. If the distance computed with equation (12) is more than the distance to the central axis, the minimum distance is set to the distance to the central axis, that is

\[
d = \min\left(dist(P, Q), \sqrt{P_x^2 + P_y^2}\right)
\]

(13)

**Accurate distance**

Since the previous calculation is approximate and introduces some error, because it is based on the invalid assumptions that the \( r \) values for \( P \) and \( Q \) are the same and that \( \Delta PQQ' \) is a right triangle, an exact distance calculation is required to produce a final, correct result with the yarn lying on the helicoid surface.

For each point \( P \), we find a point \( Q \) on the helicoid that is closest to \( P \). If \( P \) lies within the vertical projection of the helicoid, that is, if the distance of \( P \) from the central axis of the helicoid is less than radius \( R_h \), we know that the line \( PQ \) lies along the normal to the helicoid at point \( Q \). That normal is orthogonal to two tangent vectors lying in the surface (see Figure 7). For a given helicoid surface \( \vec{H} \), its tangents along the radial and angular directions are given by \( \frac{\partial \vec{H}}{\partial r} \) and \( \frac{\partial \vec{H}}{\partial \theta} \) (equations (2) and (3)), respectively.

Since the vector \( \vec{PQ} = (P - Q) \) is orthogonal to both of these tangents, its dot product with each is zero

\[
\frac{\partial \vec{H}}{\partial r} \cdot (P - Q) = 0
\]

(14)

\[
\frac{\partial \vec{H}}{\partial \theta} \cdot (P - Q) = 0
\]

(15)
Given a point \( \mathbf{P} = [P_x, P_y, P_z] \) near the helicoid, we need to compute the values of \( r \) and \( \theta \) in order to find the point \( \mathbf{Q} = [r \cos \theta, r \sin \theta, c \theta] \). Substituting the radial tangent vector from equation (2) into equation (14) produces

\[
(P_z - r \cos \theta) \cos \theta + (P_y - r \sin \theta) \sin \theta = 0
\]

This can be simplified to produce an equation for \( r \)

\[
r = P_z \cos \theta + P_y \sin \theta
\]

Similarly, substituting the angular tangent vector from equation (3) into equation (15) produces

\[
-r \sin \theta (P_z - r \cos \theta) + r \cos \theta (P_y - r \sin \theta) + c (P_z - c \theta) = 0
\]

Simplifying this equation and substituting \( r \) from equation (17) produces

\[
\frac{P_z^2 - P_y^2}{2} \sin 2\theta + P_z P_y \cos 2\theta - c^2 + cP_z = 0
\]

Equation (19) is a non-linear equation in \( \theta \) that requires a separate optimization to compute a value for \( \theta \). Once \( \theta \) is obtained, \( r \) is computed from equation (17) and the point \( \mathbf{Q} \) can be obtained as \( \mathbf{Q} = [r \cos \theta, r \sin \theta, c \theta] \). Then the minimum distance \( d \) from \( \mathbf{P} \) to the helicoid is \( ||\mathbf{PQ}|| \).

There are however infinite solutions to equation (19), since there is one point \( \mathbf{Q} \) in each cycle of the helicoid for which the line \( \mathbf{PQ} \) lies along the normal to the helicoid. Hence, the solution is restricted to the same cycle of the helicoid as \( \mathbf{P} \)

\[
\theta \in \left[ \frac{P_z}{c} \frac{\pi}{2}, \frac{P_z}{c} + \frac{\pi}{2} \right]
\]

**Distance to a left-handed helicoid**

A left-handed helicoid (where \( z = -c\theta \)) can be obtained by negating any one of the coordinates of the right-handed helicoid. These two helicoids twist in opposite directions. In order to define a left-handed helicoid next to a right-handed one, we place another helicoid along the X-axis, offset by \( 2R_h \), and reflect the helicoid equations across the plane \( x = R_h \).

In the previous sections, the distance between a given point and a right-handed helicoid was computed. To find the distance between a point and a left-handed helicoid centered at \( x = 2R_h \), the point is first reflected across the \( x = R_h \) plane, similar to the equation of the helicoid itself. So instead of finding the distance from \( \mathbf{P} = [P_x, P_y, P_z] \) to the left-handed helicoid, we compute the distance from \( \mathbf{P}' = [2R_h - P_x, P_y, P_z] \) to the right-handed helicoid centered at the origin.

**Distance from a point lying outside the helicoid’s circular projection**

In the previous sections, the computed distance to the helicoid from a point \( \mathbf{P} \) assumed that \( \mathbf{P} \) lies within the vertical projection of the helicoid. Simply put

\[
\sqrt{P_x^2 + P_y^2} \leq R_h
\]

If this is not true, the distance must be computed from \( \mathbf{P} \) to the edge of the helicoid. Let \( \mathbf{Q} \) be the point on the helicoid closest to \( \mathbf{P} \) assuming that the helicoid extends infinitely in the radial direction. Let \( \mathbf{Q}' \) be the point on the (radially) finite helicoid that is closest to \( \mathbf{P} \), as shown in Figure 8. It can be seen that the point \( \mathbf{Q}' \) lies on the same radial line as that of \( \mathbf{Q} \) and also that it lies on edge of the helicoid. Thus, if \( \mathbf{Q} = [r \cos \theta, r \sin \theta, c \theta] \), then \( \mathbf{Q}' = [R_h \cos \theta, R_h \sin \theta, c \theta] \). If the condition in equation (22) is not satisfied, we first find \( \theta \) to get the location of
Q′ and then compute the distance ‖PQ′‖ instead of ‖PQ‖.

**Yarn path generation**

The shape of the yarn path is governed by different forces acting on the yarn, each of which is defined by a separate energy term. The yarn is repelled from the helicoid, while its length is shortened, which effectively stretches it across the surface. Thus, the yarn path stabilizes when the energies are balanced and it becomes as short as possible without penetrating the helicoid.

**Computing yarn energy**

The total energy of the yarn is the sum of the energies computed at \( N−1 \) of the \( N \) vertices of the polyline that approximate it

\[
E_{\text{total}} = \sum_{i=1}^{N-1} E_i
\]

(23)

The reason for not computing an energy at the last vertex will be apparent given the definition of the length energy. The total energy associated with vertex \( i \) is given by

\[
E_i = \alpha E_{\text{len}} + \beta E_{\text{dist}}
\]

(24)

where \( \alpha \) and \( \beta \) (which are both set to 1 in our examples) are scaling factors that allow us to control the influence of the individual energy terms. The energy term used to shrink the yarn is

\[
E_{\text{len}} = (\text{Length}_i − \text{Targetlength})^2
\]

(25)

where \( \text{Length}_i \) is the distance between vertex \( i \) and vertex \( i+1 \) and \( \text{Targetlength} \) is parameter that is adjusted in order to shorten the polyline. The vertex look-ahead in this calculation is the reason for only looping to vertex \( N−1 \) in equation (23). Note that the total length of the yarn is

\[
\text{Yarnlength} = \sum_{i=1}^{N-1} \text{Length}_i
\]

(26)

we normally set \( \text{Targetlength} \) to be one half of the initial length of the yarn polyline divided by the number of polylines.

The center of the yarn should never be less than \( R_y \) (the yarn radius) distance from the helicoid, otherwise the yarn will penetrate the helicoid. We therefore define a distance energy to maintain this constraint based on the distance \( d \) from vertex \( i \) to the helicoid computed with the algorithms in the “Distance from helicoids” section, as follows

\[
E_{\text{dist}} = (d − R_y) \log (d / R_y)
\]

(27)

The distance energy is only computed for \( d \) values less than \( R_y \). The equation is defined in this form in order to go to infinity at \( d = 0 \) and to have a value and derivative of 0 at \( d = R_y \).

**Initialization and energy minimization**

In order to produce an optimized yarn path on a helicoid, an initial path is specified with a Catmull-Rom spline around the helicoid (see Figure 9(a)). In our examples, the spline is discretized into a polyline of approximately 150 points. An optimization is run that minimizes equation (23) using the approximate distance calculation in the “Approximate distance” section. The resulting yarn polyline is close to the true solution, but with some regions that penetrate the helicoid. This approximate configuration then becomes the initial conditions for another optimization process, but using the accurate distance calculation instead, which produces the final, correct configuration (see Figure 9(b)). This two-step process provides about a 4x speed-up in the yarn path calculation over one that only uses the accurate distance calculation.

The resulting yarn geometry specifies a half-loop of a knit stitch, which can be reflected to produce a full loop, as in Figure 2. In order to define purl stitches that connect to knit stitches, a second plane of reflected helicoids is needed. The initialization and optimization of the yarn used in a garter fabric (alternating rows of knit and purl stitches) is slightly different, since a garter half loop spans two helicoids, one behind the other. This specific loop is needed in order to model how a knit stitch transitions into a purl stitch going in the upward (wale) direction. With the right-handed
Figure 10. Initial and optimized yarn for garter stitch: (a) back view of initial yarn, (b) back view of optimized yarn, (c) side view of initial yarn, and (d) side view of optimized yarn.

Figure 11. Complete initial and optimized garter loop: (a) initial yarn and (b) optimized yarn.

helicoid centered at the origin, the left-handed one is centered at $(2R_h, 0, 0)$. Figure 10 shows the initial and optimized yarns for a garter half loop, with the full loop presented in Figure 11. Since the yarn lies partially on two different helicoids, the distance computation has to be modified slightly. The method described in the “Distance to a left-handed helicoid” section is used to compute the distance to the left-handed helicoid when $P_x > R_h$.

Constraints and replication

A number of constraints are imposed on the optimized yarn in order to enforce the boundary conditions that then allow for the half loop to be continuously replicated. They are as follows:

- The Y-coordinates of the endpoints are fixed to be equal to the helicoid radius. This ensures that the endpoints of the yarn are held at $y = -R_h$ and $y = R_h$.
- The X-coordinate of the lower endpoint is fixed to be equal to the helicoid radius. This allows for the loop to connect with a yarn coming from a helicoid in a different x-plane, a boundary condition needed to produce rib structures.
- At both endpoints, the second point from the end has the same X and Z coordinates as the endpoint. This ensures that the tangents at both half-loop ends are parallel to the Y-axis, a boundary condition that produces $C^1$ structures.

With these constraints, the optimized half loop of the yarn can be easily replicated to obtain a complete loop, by reflecting it across the $y = R_h$ plane. This process is shown in Figure 2. The second helicoid is centered at $(0, 2R_h, 0)$. This full loop may then be duplicated and shifted in increments of $4R_h$ in the y direction and $c/2$ (the equivalent to $\pi$ radians in $\theta$) in the z direction to produce a model of an all-knit fabric, like the one in Figure 15.

In order to represent a rib fabric (alternating columns of knit and purl stitches) on helicoids, the yarn must travel from a “front” helicoid pair, for a knit stitch, to a “back” helicoid pair for a purl stitch (see Figure 4(b)). A model of a rib fabric can be generated from a model of an all-knit fabric by reflecting alternating columns of stitches across the $x = R_h$ plane (see Figure 16).

A model of garter fabric can be generated by first duplicating and shifting the necessary number of garter full loops (Figure 11). This loop represents a knit stitch in a garter fabric. The loop is then simply reflected across the $x = R_h$ plane at the loop locations with purl stitches. A model of a garter fabric is presented in Figure 17.

Given the two optimized loops, a geometric model of a knitted fabric consisting of an arbitrary combination of knit and purl stitches may be generated. If a given stitch (knit or purl) has a different stitch above it, the garter loop of Figure 10(b) is used to represent the stitch. Otherwise, the plain loop of Figure 2 is used. The stitch type will determine if the geometric loop is positioned on the front helicoid (knit stitches) or the back helicoid (purl stitches). See Figure 18 for an example.
We conducted an experiment to investigate the physical accuracy of the models generated by our approach. The experiment involved minimizing the bending energy of the polyline after removing some of the constraints. The yarn’s length, endpoints, and the tangents at those endpoints are kept fixed while the distance from the helicoid energy is removed. Removing this energy leaves the polyline free-floating without any interaction with the helicoid. The minimization is done on a half stitch and the result is then mirrored and joined to obtain the complete stitch.

Curvature at each polyline vertex is computed using the following equation

\[
K = \begin{cases} 
\frac{2}{\sigma} \cos \left( \frac{\theta}{2} \right) & \frac{\pi}{4} < \theta \leq \pi \\
-\left( \frac{\pi}{4} \right)^2 \dot{\rho} + \tau + \left( \frac{\pi}{4} \right) \rho & 0 \leq \theta \leq \frac{\pi}{4}
\end{cases}
\]  

(28)

\(K\) approximates the curvature around a point along the polyline. The function fits a circle to a point and both of its neighbors and assumes the curvature at the central point is equal to the circle’s. \(\rho\) and \(\tau\) are constants that enforce \(C^1\) continuity between the two components of this piece-wise function. The second component makes \(K\) go to infinity as \(\theta\) approaches zero. Further details about this computation are given in Breen et al.\textsuperscript{50}

\(K\) is computed for all the points except the endpoints and then used to compute the polyline’s total bending energy

\[
E_{bend} = \sum_{i=1}^{K-2} K_i^2
\]  

(29)

This energy is then minimized by moving the vertices of the polyline while keeping the endpoints of the half stitch and their tangents fixed. The results of the optimization obtained by mirroring and concatenating the optimized half stitches are shown in Figures 12 and 13. Figure 12 shows the knit and garter loops before and after minimizing bending energy. It can be seen that the major changes after optimization occur at the regions where the models were outside the helicoid projections. Figure 13 shows the overlaid models for both stitches and highlights the minimal differences between the models.

This experiment shows that generating a geodesic path on a minimal surface produces yarn geometry that is in a stable mechanical configuration. Minimizing the curvature-squared would force an unconstrained polyline to become straight, but given the constraints on the endpoints, a much more complex 3D shape is produced. Since the yarn geometry only slightly moves after minimizing the total bending energy of the polyline, it can be deduced that our helicoid scaffold approach, which is based on a purely geometric calculation, produces a result that is in a mechanical, that is, physical, energy minimum state, thus supporting the argument that

**Figure 12.** Knit loop (a) before and (b) after optimizing for minimum curvature. Garter loop (c) before and (d) after optimizing for minimum curvature.

**Figure 13.** Overlaid models before and after optimizing for curvature: (a) knit loop and (b) garter loop.
modeling with minimal surfaces produces physically accurate geometric configurations of yarn geometry.

**Results**

**Computation time**

All geometric models included in the article were generated on an Apple iMac with a 4.2 GHz Intel Core i7 processor and 32 GB of RAM. The computations were done in MATLAB using the unconstrained optimization function *fminunc*, with parallel computing enabled which allows for simultaneous processing on all four cores of our system’s CPU. The generation of the optimized half-loop yarn model dominates the computation time of our approach. While the time needed to produce an inaccurate result, as seen in Figure 14(a), is only 30 s, generating the accurate result of Figures 2, 9(b), and 14(b) requires 22 min. This increased time comes from the separate optimization needed for every distance-to-helicoid computation that is

![Figure 14. Yarn path obtained using the (a) fast approximate distance calculation method and (b) the compute-intensive accurate distance calculation method.](image)

![Figure 15. Geometric model of an all-knit fabric: (a) front view with helicoid scaffolding and (b) top, front, and side views without the helicoids.](image)

![Figure 16. Geometric model of a rib fabric: (a) front view with helicoid scaffolding and (b) top, front, and side views without the helicoids.](image)
performed for every point in the polyline during every iteration of the global optimization process. These computation times did not change significantly with changes in the model’s parameter values.

We use a hybrid approach that first employs the fast inaccurate distance method to bring the initial polyline (Figure 9(a)) close to the surface, which is followed by a polyline refinement using the accurate distance calculation to bring the yarn model onto the surface. The hybrid approach provides significant speed-ups over computations that exclusively use the accurate distance calculation for the whole process. Employing only the accurate distance calculation to produce the final results in Figures 2, 9, and 14 requires 84 min of computation time. Since the hybrid approach requires only 22 min, it provides about a $4\times$ speed-up in the calculation.

**Approximate versus accurate results**

Figure 14 presents the different results produced by the approximate distance computation method and the accurate method. When using the approximate method, it can be seen that the yarn penetrates into the helicoid to a depth of around 10% of the yarn radius. When using the accurate method, no visible penetration is observed.

**Different fabric examples**

Figures 15(a), 16(a), and 17(a) present the yarn geometric models lying on helicoids generated by our approach for all-knit, rib, and garter fabrics, respectively. The yarn models with the helicoid scaffolds removed are presented in the (b) subfigures of these same figures. Figure 18 presents a yarn geometric model for a swatch consisting of a random pattern of knit and purl stitches, with the helicoid scaffolds removed. The model parameters are set as follows: yarn radius $R_y = 0.5$, helicoid radius $R_h = 2.0$, and helicoid height $c = 10/(2\pi)$.

**Changing model parameters**

There are three parameters in our approach that can be used to control the shape and size of the loops in the modeled fabrics. They are the yarn radius $R_y$, the helicoid
radius $R_h$, which controls the width of the helicoid, and the helicoid height scale factor $c$, which controls the spacing between each cycle of the helicoid. Since the yarn lies on the helicoid, changing the shape of the helicoid directly changes the shape of the yarn model. We have generated numerous models with a variety of parameter values. The effect of changing the yarn radius is seen in Figure 19. Figure 20 shows the effect of changing the helicoid radius, with other parameters set to default values. Figure 21 shows the effect of changing the height scale factor of the helicoid.

Changing the yarn radius changes the distance of the yarn’s centerline from the helicoid surface. Changing the helicoid’s radius affects the width of the resulting stitches. Changing the helicoid’s height scale factor affects the height of the resulting stitches.

Conclusion and future work

In this article, we have presented the mathematics and algorithms needed to utilize the helicoid, a bicontinuous, minimal surface, as a scaffold for defining the topology and geometry of yarns in a weft-knitted fabric. The geometry of a half-loop of yarn is specified as a geodesic path along the surface with fixed boundary conditions. This optimized path may be duplicated, reflected, and shifted to produce the centerlines for the multiple stitches that make up a fabric. The geodesic path is produced via an optimization...
process which shortens the distance between the vertices that make up the yarn model, while keeping them a set distance from the helicoid surface. The yarn geometry is generated as a tube around the centerline. A second plane of reflected helicoids is utilized to define purl stitches, which then allows us to create models of not only all-knit fabrics but also rib and garter fabrics. Since the yarn geometry lies on the helicoid surfaces, changing the parameters of the helicoid (radius and height scaling) controls the size and shape of the stitches in the modeled fabrics.

The work presented here is a proof-of-concept that demonstrates the effectiveness of helicoid scaffolds for modeling weft-knitted fabrics. Our intent is to incorporate these models into a computer-aided design (CAD) system and to use them as input to physical simulations of knitted fabrics. We recognize that the current compute times needed to generate the models are not conducive for interactive design. Future work will focus on methods that speed up the calculations, for example, creating an implementation in a compiled language and investigating ways to increase parallelism.

Currently, we only model knit and purl stitches. In the future, we will incorporate other stitches into our modeling approach, for example, miss, tuck, and transfer stitches. Once these stitches are implemented, plans are being formulated for using helicoid scaffolds as a framework for simulating the weft knitting process and the knitted structures that it produces. In this future work, needles and the way they define the topological/geometric relationships between yarns will be represented with helicoid segments. Simulating the knitting process will involve transferring yarn loops between helicoid segments, similar to the actions of needles. The helicoid framework allows us to reproduce this mechanical process with a geometric

Figure 21. Effect of changing the helicoid’s height scaling factor while keeping the helicoid radius and yarn radius constant: (a) $c = 5/2\pi$, (b) $c = 10/2\pi$, (c) $c = 15/2\pi$, and (d) $c = 20/2\pi$. 
computation, alleviating the need for a physical simulation in order to produce correct topological structures and plausible yarn geometries.

Finally, we note that our yarn models exhibit a curvature discontinuity when outside of the helicoid’s radius. We aim to address this issue in our future models.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This material is based upon work supported by a National Science Foundation Graduate Research Fellowship under Grant No. DGE-10028090/DGE-1104459 and NSF grant CMMI #1537720. This work was supported in part by a Simons Investigator Grant from the Simons Foundation to R.D.K. This material is based upon work supported by a National Science Foundation Grant (under Dr. J. L. Lagrange's direction) awarded to R.D.K. and in part by a Simons Investigator Grant to R.D.K. from the Simons Foundation to R.D.K. This work was supported in part by a Simons Investigator Grant to R.D.K. This work was supported in part by a National Science Foundation Grant (under Dr. J. L. Lagrange's direction) awarded to R.D.K.

ORCID iDs

Paras Wadekar https://orcid.org/0000-0002-0949-652X
Randall D Kamien https://orcid.org/0000-0003-1598-5626

References

1. Lagrange JL. Essai sur une nouvelle methode pour determine les maxima et minima des formules integrales indefinies. In: Miscellanea Taurinensia II. 1762, pp. 173–195.
2. Plateau J. Statique Experimentale et Theorique des Liquides Soumis aux Seules Forces Molculaires. Paris: Gauthier-Villiard, 1873.
3. Taylor J. The structure of singularities in soap-bubble-like and soap-film-like minimal surfaces. Ann Math 1976; 103(3): 489–539.
4. Hajduk D, Harper P, Gruner S, et al. The gyroid: a new equilibrium morphology in weakly segregated diblock copolymers. Macromolecules 1994; 27(15): 4063–4075.
5. Torquato S and Donev A. Minimal surfaces and multifunctionality. Proc R Soc Lon Ser A 2004; 460: 1849–1856.
6. Santangelo C and Kamien R. Elliptic phases: a study of the nonlinear elasticity of twist-grain boundaries. Phys Rev Lett 2006; 96: 137801.
7. Matsumoto E, Kamien R and Santangelo C. Smeectic pores and defect cores. Interface Focus 2012; 2(5): 617–622.
8. Knittel C. Tools to predict and understand the self-folding behavior of knitted textiles. PhD Thesis, Drexel University, Philadelphia, PA, 2019.
9. Knittel CE, Tanis M, Stoltzfus AL, et al. Modelling textile structures using bicontinuous surfaces. J Math Arts (under review).
10. Witkin A, Fleischer K and Barr A. Energy constraints on parameterized models. In: SIGGRAPH ’87: proceedings of the 14th annual conference on computer graphics and interactive techniques. Anaheim, CA; 27–31 July 1987. pp. 225–232. New York: ACM.
11. Peirce F. Geometrical principles applicable to the design of functional fabrics. Text Res J 1947; 17(3): 123–147.
12. Leaf G and Glaskin A. The geometry of a plain knitted loop. J Text I 1955; 46(9): T587–T605.
13. Munden D. The geometry and dimensional properties of plain-knit fabrics. J Text I 1959; 50(7): T448–T471.
14. Cohen E, Riesenfeld R and Elber G. Geometric modeling with splines. Natick, MA: AK Peters, 2001.
15. Demiroz A and Dias T. A study of the graphical representation of plain-knitted structures part I: model for the graphical representation of plain-knitted structures. J Text I 2000; 91(4): 463–480.
16. Lin H, Zeng X, Sherburn M, et al. Automated geometric modelling of textile structures. Text Res J 2012; 82(16): 1689–1702.
17. Kurbak A and Ekmen O. Basic studies for modeling complex weft knitted fabric structures part I: a geometrical model for widthwise curlings of plain knitted fabrics. Text Res J 2008; 78(3): 198–208.
18. Kurbak A and Kayacan O. Basic studies for modeling complex weft knitted fabric structures part II: a geometrical model for plain knitted fabric spirality. Text Res J 2008; 78(4): 279–288.
19. Kurbak A. Geometrical models for balanced rib knitted fabrics part I: conventionally knitted 1×1 rib fabrics. Text Res J 2009; 79(5): 418–435.
20. Shanahan W and Postle R. A theoretical analysis of the plain-knitted structure. Text Res J 1970; 40(7): 656–665.
21. Hepworth B and Leaf G. The mechanics of an idealized weft-knitted structure. J Text I 1976; 67(7–8): 241–248.
22. De Jong S and Postle R. Energy analysis of mechanics of weft-knitted fabrics by means of optimal-control theory. Part I: nature of loop-interlocking in plain-knitted structure. J Text I 1977; 68(10): 307–315.
23. Semnani D, Latifi M, Hamzeh S, et al. A new aspect of geometrical and physical principles applicable to the estimation of textile structures: an ideal model for the plain-knitted loop. J Text I 2003; 94(3–4): 202–211.
24. De Jong S and Postle R. An energy analysis of the mechanics of weft-knitted fabrics by means of optimal-control theory. Part II: relaxed-fabric dimensions and tensile properties of the plain-knitted structure. J Text I 1977; 68(10): 316–323.
25. Choi K and Lo T. An energy model of plain knitted fabric. Text Res J 2003; 73(8): 739–748.
26. Choi KF and Lo TY. The shape and dimensions of plain knitted fabric: a fabric mechanical model. Text Res J 2006; 76(10): 777–786.
27. Kyosev Y, Angelova Y and Kovar R. 3D modeling of plain weft knitted structures of compressible yarn. Res J Text Appar 2005; 9(1): 88–97.
28. Sherburn M. Geometric and mechanical modelling of textiles. PhD Thesis, University of Nottingham, Nottingham, 2007.
29. Duhovic M and Bhattacharyya D. Simulating the deformation mechanisms of knitted fabric composites. Compos Part A: Appl S 2006; 37(11): 1897–1915.
30. Meissner M and Eberhardt B. The art of knitted fabrics, realistic & physically based modeling of knitted fabrics. *Comput Graph Forum* 1998; 17(3): 355–362.
31. Eberhardt B, Meissner M and Strasser W. Knit fabrics. In: House D and Breen D (eds) *Cloth modeling and animation*. Natick, MA: AK Peters, 2000, pp. 123–144.
32. Kaldor JM, James DL and Marschner S. Simulating knitted cloth at the yarn level. *ACM T Graphic* 2008; 27(3): 65.
33. Kaldor JM, James DL and Marschner S. Efficient yarn-based cloth with adaptive contact linearization. *ACM T Graphic* 2010; 29(4): 105.
34. Yuksel C, Kaldor JM, James DL, et al. Stitch meshes for modeling knitted clothing with yarn-level detail. *ACM T Graphic* 2012; 31(3): 37.
35. Wu K, Gao X, Ferguson Z, et al. Stitch meshing. *ACM T Graphic* 2018; 37(4): 130.
36. Wu K, Swan H and Yuksel C. Knittable stitch meshes. *ACM T Graphic* 2019; 38(1): 10.
37. Leaf J, Wu R, Schweickart E, et al. Interactive design of periodic yarn-level cloth patterns. *ACM T Graphic* 2018; 37(6): 202.
38. Cirio G, Lopez-Moreno J and Otaduy MA. Yarn-level cloth simulation with sliding persistent contacts. *IEEE T Vis Comput Gr* 2017; 23(2): 1152–1162.
39. Liu D, Christie D, Shakibajahromi B, et al. On the role of material architecture in the mechanical behavior of knitted textiles. *Int J Solids Struct* 2017; 109: 101–111.
40. Liu D, Shakibajahromi B, Dion G, et al. A computational approach to model interfacial effects on the mechanical behavior of knitted textiles. *J Appl Mech* 2018; 85(4): 041007.
41. Liu D, Koric S and Kontsos A. A multiscale homogenization approach for architectured knitted textiles. *J Appl Mech* 2019; 86(11): 111006.
42. Liu D, Koric S and Kontsos A. Parallelized finite element analysis of knitted textile mechanical behavior. *J Eng Mater: T ASME* 2019; 141(2): 021008.
43. Poincloux S, Mokhtar AB and Lechenault F. Geometry and elasticity of a knitted fabric. *Phys Rev X* 2018; 8(2): 021075.
44. Igarashi Y, Igarashi T and Suzuki H. Knitting a 3D model. *Comput Graph Forum* 2008; 27(7): 1737–1743.
45. Igarashi Y, Igarashi T and Suzuki H. Knitty: 3D modeling of knitted animals with a production assistant interface. In: *Annex to Eurographics 2008 Proceedings*, pp. 187–190, http://www.kisc.meiji.ac.jp/~yuki_i/papers/knitty_euro08.pdf
46. McCann J, Albough L, Narayanan V, et al. A compiler for 3D machine knitting. *ACM T Graphic* 2016; 35(4): 49.
47. Narayanan V, Albough L, Hodgins J, et al. Automatic machine knitting of 3D meshes. *ACM T Graphic* 2018; 37(3): 35.
48. Lin J, Narayanan V and McCann J. Efficient transfer planning for flat knitting. In: *Proceedings of the 2nd ACM symposium on computational fabrication*, Cambridge, MA, June 2018, Article 1, pp. 1–7. New York: ACM.
49. Knittel C, Nicholas D, Street R, et al. Self-folding textiles through manipulation of knit stitch architecture. *Fibers* 2015; 3(4): 575–587.
50. Breen D, House D and Wozny M. A particle-based model for simulating the draping behavior of woven cloth. *Text Res J* 1994; 64(11): 663–685.