Generation and optimization of fiber layup on curved composite structures

by

Nathan Scheirer

A thesis submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Aerospace Engineering

Program of Study Committee:
Stephen Holland, Major Professor
Adarsh Krishnamurthy
Vinay Dayal

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this thesis. The Graduate College will ensure this thesis is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University
Ames, Iowa
2020

Copyright © Nathan Scheirer, 2020. All rights reserved.
DEDICATION

I would like to dedicate this work to my wife Sarah, who has had to endure trying to explain my work on this project to her family and friends throughout the first six months of our marriage. Without her loving support this project would have been much more stressful and definitely not as enjoyable.
TABLE OF CONTENTS

LIST OF TABLES ......................................................................................... v

LIST OF FIGURES .................................................................................. vi

ACKNOWLEDGMENTS ........................................................................... vii

ABSTRACT .............................................................................................. viii

CHAPTER 1. INTRODUCTION ................................................................. 1
  1.1 Innovation ...................................................................................... 2
  1.2 Process .......................................................................................... 3
  1.3 Outline .......................................................................................... 4

CHAPTER 2. LITERATURE REVIEW .................................................. 5
  2.1 Related Composite Work ............................................................... 5
  2.2 Related Computer Graphics Work ................................................ 6

CHAPTER 3. METHODOLOGY .............................................................. 7
  3.1 Geodesic Parameterization ............................................................. 7
  3.2 Strain Energy Optimization ........................................................... 11
    3.2.1 Global Strain Energy ............................................................... 11
    3.2.2 Global Strain Energy Gradient ............................................... 13
    3.2.3 Optimization ......................................................................... 15

CHAPTER 4. EXPERIMENTS AND RESULTS .................................... 17
  4.1 Introduction ................................................................................... 17
  4.2 Arm-Chair ..................................................................................... 18
    4.2.1 Experiment ............................................................................. 18
    4.2.2 Results ................................................................................... 18
  4.3 Method Analysis ........................................................................... 22
  4.4 GPU-accelerated Geodesic Computation ....................................... 23

CHAPTER 5. APPLICATIONS ................................................................. 25
  5.1 Introduction ................................................................................... 25
  5.2 De-La-Mo Abaqus Integration ....................................................... 25
  5.3 Cardiovascular Fiber Modeling ..................................................... 26
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHAPTER 6. FUTURE WORK SUMMARY AND DISCUSSION</td>
<td>28</td>
</tr>
<tr>
<td>6.1 Future Work</td>
<td>28</td>
</tr>
<tr>
<td>6.2 Conclusions</td>
<td>28</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>30</td>
</tr>
<tr>
<td>APPENDIX. ADDITIONAL MATERIAL</td>
<td>33</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 4.1</td>
<td>Experimental model details</td>
<td>17</td>
</tr>
<tr>
<td>Table 4.2</td>
<td>Composite material properties</td>
<td>18</td>
</tr>
<tr>
<td>Table 4.3</td>
<td>RMSprop optimization parameters</td>
<td>22</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1.1</td>
<td>NASA’s ISAAC AFP machine</td>
<td>2</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>Single geodesic tracing step</td>
<td>8</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>Geodesic-based vertex assignment</td>
<td>9</td>
</tr>
<tr>
<td>Figure 4.1</td>
<td>Armchair unoptimized fiber orientations</td>
<td>19</td>
</tr>
<tr>
<td>Figure 4.2</td>
<td>Armchair optimized parameterization</td>
<td>20</td>
</tr>
<tr>
<td>Figure 4.3</td>
<td>Armchair strain energy during optimization</td>
<td>20</td>
</tr>
<tr>
<td>Figure 4.4</td>
<td>Ply angle vs compute time vs initial energy comparison</td>
<td>21</td>
</tr>
<tr>
<td>Figure 4.5</td>
<td>Armchair model</td>
<td>21</td>
</tr>
<tr>
<td>Figure 4.6</td>
<td>Model size vs compute time</td>
<td>22</td>
</tr>
<tr>
<td>Figure 4.7</td>
<td>CPU vs GPU comparison</td>
<td>24</td>
</tr>
<tr>
<td>Figure 5.1</td>
<td>Layered symmetric composite example</td>
<td>26</td>
</tr>
<tr>
<td>Figure 5.2</td>
<td>Biventricular cardiac geometry example</td>
<td>27</td>
</tr>
<tr>
<td>Figure 5.2</td>
<td>Various tested models with un/optimized results</td>
<td>35</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENTS

This work is supported by the National Aeronautics and Space Administration (NASA) under contract number NNL15AA12C through Center for Non-Destructive Evaluation (CNDE) at Iowa State University and, is a part of NASA’s Advanced Composites Project. I would like to thank Dr. Erik Frankfurter, Dr. Cara Leckey, and Dr. William P. Winfree from NASA for their valuable input and support for this project. I would also like to thank Dr. Adarsh Krishnamurthy and Dr. Stephen Holland for their consistent and enthusiastic support that they showed me throughout this project. Because of their invaluable discussions and insight this project was able to flourish and develop into what it is today.
ABSTRACT

Typically, in finite element model analysis and automated fiber placement, composite fiber orientations are computed using either simple coordinate axis rotations or a complicated full analysis model. The former performs well on simple surfaces with single axis curvature but degrades in performance when used on doubly-curved surfaces due to fiber crossing and buckling. A full analysis model provides the accuracy of a truly physically based solution, however this process can be quite extensive and does not lend itself well to industrial use.

Therefore, this work presents a new two-step process for defining optimal composite fiber paths for use in automated fiber placement machines and finite element analysis models. Building upon work done previously in the area of discrete geodesic path generation, a method for optimizing fast approximate geodesic paths on triangular meshes using a strain energy minimization technique is presented. A comparison of the effectiveness of this process with those already discussed in the literature show how this process is fast, accurate, and ready to use in commercial applications. This study also shows how this process effectively finds optimal fiber paths on complex, non-developable surfaces, which will improve finite element analysis models and provide composite manufacturers with the ability to create components with complex geometry. Improvements to this process are discussed and have been implemented such as utilizing the parallel computing capabilities of a GPU, which speeds up the process by computing geodesics in parallel for large surfaces. The algorithms presented in this study are available freely online and have been successfully integrated into an automated finite-element model-building software called De-La-Mo (https://idealab-isu.github.io/autofiber/).
CHAPTER 1. INTRODUCTION

Nathan Scheirer, Stephen Holland, and Adarsh Krishnamurthy

Modified from a manuscript under review in *Computer-Aided Design*

The popularity of Automated Fiber Placement (AFP) manufacturing has steadily increased over the last two decades. From the overview of robotic fiber placement machines by Olsen and Craig (1993) to various studies of the defects introduced by the AFP process (Fayazbakhsh et al., 2013), there have been many investigations into this technique throughout literature. NASA has procured an AFP machine named the Integrated Structural Assembly of Advanced Composites, or ISAAC, (Figure 1.1) to aid in their composite manufacturing research and development (Wu et al., 2014). AFP machines are multi-degree of freedom robots with specifically designed attachments that allow for the laying of composite fibers or tape. These systems are similar to computer numeric controlled (CNC) milling machines in the sense that they can be programmed to lay fibers along a predetermined path, which supplies composite material designers with more flexibility and control over material properties. However, path programming for AFP machines is a complicated process and requires manual user intervention to create high-quality paths that produce composites without layup defects.

In many finite element analysis programs, the method for computing composite fiber orientations is as simple as rotating a given orientation into the plane with each surface element. Other commonly used path planning strategies for composite fibers involve computing geodesic lines. A geodesic is the mathematical representation of a straight line in curved, three-dimensional space. By definition, a geodesic represents the locally shortest path from one point on a surface to another. Geodesics have found usefulness in a variety of other industries such as computer graphics, structure design, and even general relativity and space-time. In addition to geodesic fiber paths, many researchers have investigated the simulation and optimization of composite fiber draping
Figure 1.1: NASA’s ISAAC AFP machine (McDonald, 2015)

Van Der Ween, 1991) to compute optimal fiber path trajectories. Within this work, a process for optimizing approximate geodesic fiber orientations on a triangular mesh by minimizing fiber strain energy is presented. The results of this optimization, which is presented later in this study, aim to exhibit composite fiber orientations that accurately represent the relaxation of composite fibers after placement, but before the composite matrix has completely solidified. By computing these fiber paths before the composite is manufactured, one can use AFP technology to place the fibers in near minimum strain configuration and avoid inaccuracies attributed to geodesic path generation techniques, especially on complex, doubly curved surfaces.

1.1 Innovation

Minimal strain or minimum energy composite draping models have been proposed and studied in various areas of the literature. However, there is not much research on combining the intrinsic details of the geometry of the model with the physics that governs the layup of composite fibers. Geodesic paths are primarily employed to compute fiber orientations; however, in cases where geodesics can cross paths or diverge due to complicated geometry, once cannot rely on the geometry alone. Full physical modeling of the composite fiber layup can provide an alternate, highly accurate, and well-formulated approach. Unfortunately, physical modeling approaches are computationally expensive
and can be challenging to integrate with practical applications. In this paper, it is shown that by using the geometry to produce a seed for an optimization algorithm based on a known physical model, composite fiber ply angles that are free from crossing and do not diverge can be produced, while still leveraging the speed of discrete geodesic computation. What is presented here is also a process that is highly applicable to industrial use. The implementation that has been created has been successfully used to transfer fiber orientations to Abaqus (Dassault Systèmes Abaqus SIMULIA, 2017) for finite element analysis (please see Section 5). In addition, this process has been tested on a wide variety of surface models, and it has shown excellent results.

1.2 Process

The method proposed in this paper consists of three major components. The first is a module which handles model loading and management. Currently, our implementation can handle STL and X3D file formats and can output fiber orientations in a format readily capable of input into Abaqus, or for further computation. This module also handles computing the 2D or flattened representation of the model surface, which is used as a reference state for the strain energy minimization. The second component (Section 3.1) is a geodesic based parameterization method which uses geodesic paths to create a UV parameterization of the surface. This is done by computing discrete geodesics over the tessellated surface and then each vertex is assigned a UV coordinate based on its location relative to the closest geodesic path. The final component of the process (Section 3.2) is a global strain energy minimization technique that utilizes finite strain theory to compute the lowest strain energy state of the parameterized surface. The resulting optimized parameterization can then be used to find composite ply angles at any point on the surface. Considering that these ply angles are representative of the minimum strain state, they can be used in finite element programs to better model the final orientations of composite fibers after layup, or in AFP, to compute composite fiber paths which result in a near relaxed state.
1.3 Outline

This work is structured as follows: Chapter 1 consists of an introduction as well as a brief look at the novelty and overall methodology of this process, Chapter 2 comprises a literature review and explanation of related work in the relevant fields, Chapter 3 contains an in depth look at the methodology and process described in this work, Chapter 4 presents the experiments and results performed to demonstrate the effectiveness of this approach, Chapter 5 describes some applications which could benefit from this process in industry and research, and, finally, Chapter 6 describes future applicable work and the conclusions made following the completion of this work. Example images of this process on a variety of geometries can be viewed in Figure 2.
CHAPTER 2. LITERATURE REVIEW

2.1 Related Composite Work

Researchers have investigated the computation of optimal fiber path trajectories in previous studies (Nagendra et al., 1995; Setoodeh et al., 2006). These studies are primarily interested in “variable-stiffness” panels, which are composite structures with curvilinear fiber paths that optimize the components' ability to withstand certain load requirements. Similar fiber path generation techniques, such as fixed angle path planning and variable angle path planning, are discussed by Li et al. (2015). They conclude that fixed angle path planning is difficult to perform on complex geometry due to the inability to find an adequate reference path that is optimal over the entire surface. In their work they continue to discuss the advantages of variable angle path planning because of its flexibility and ability to adhere to various design constraints. Reference paths are commonly found using a geodesic path, which is then offset over the entire surface to determining composite fiber paths for AFP manufactured composites. Kumar et al. (2003) investigated three different methods for computing geodesics on both non-uniform rational B-spline (NURBS) surfaces (continuous, exact) and tessellated surfaces (discrete, approximate). They show that the most accurate method of computing geodesics is directly on the NURBS surface; however, most finite element analysis programs and computer-aided design models utilize tessellated file formats that do not provide a parameterized surface. NURBS also has the disadvantage that for complex geometry, multiple ”patches” may need to be constructed to represent the entire surface accurately, therefore ensuring connected geodesics between patches may be difficult. In comparison, they concluded that, especially when working with non-developable tessellated surfaces, an error is introduced into the discrete geodesic paths due to the approximation of the tessellation. However, since most CAD models utilized a tessellated format and not a NURBS representation, regardless of path generation technique, there will be an error present due to mesh format. Additionally,
Kam and Snyman (1991) present their method for computing optimal composite fiber layups by minimizing the potential of moving particles on curved composite plates under different loading conditions. Multiple researchers such as Cherouat and Bourouchaki (2013) and Van Der Ween (1991) utilize various draping techniques to determine fiber orientations. Van Der Ween (1991) presents a draping method based on minimum elastic energy, which, as their research suggests, concludes that this process works quite effectively on doubly-curved surfaces. Various commercial products contain packages which compute composite fiber orientations for AFP applications such as CGTech’s Vericut or the Ingersoll company’s CPS (Composite Programming Software). Li et al. (2015) contains further details of these AFP composite fiber path planning techniques as well as other off-the-shell applications and academic research endeavours.

2.2 Related Computer Graphics Work

Concurrently, geodesics have been studied in the computer graphics industry quite extensively. Geodesics have been used in shortest path algorithms (Kapoor, 1999), texture synthesis (Ma and Gagalowicz, 1986), woven structure generation (Vekhter et al., 2019), and many other graphical computational applications. Among these studies, a variety of geodesic computational methods have been developed. Within the work of Pottmann et al. (2010), three geodesic computational methods are investigated, such as creating parallel curves from a single geodesic along a Jacobi field, a solution to the geodesic Eikonal equation, and projection of an estimated field onto approximate geodesic fields. Furthermore, Vekhter et al. (2019) discusses geodesic foilations which can be used to generate efficient woven structures that mimic similar structures found in nature. The woven structures in these materials follow geodesic paths to reduce buckling and twisting. These materials exhibit extraordinary structural properties and can benefit a wide variety of engineering industries such as civil engineering, nano-scale material engineering, and medicine.
CHAPTER 3. METHODOLOGY

3.1 Geodesic Parameterization

In continuum mechanics and finite strain theory, deformation can be characterized by three components: a reference state, a deformed state, and a deformation gradient which maps the change between the reference and deformed states. In this study, it is proposed that instead of starting from a known reference state, the reference state (of minimum strain energy) is determined from the deformed state. In doing so, one can realize the relaxed or minimally strained fiber orientations over the entire surface, therefore this accounts for the complexity of non-developable surfaces and error introduced during tessellation. The minimum strain energy state corresponds to a nearly relaxed surface which exhibits important engineering properties such as reduced buckling, twisting, and internal energy (Vekhter et al., 2019). An important note to make about this technique is that this analysis, while it reduces the overall strain in the material, is most accurate for the composite material immediately before curing due to its elastic nature. Once curing has begun and the composite matrix has begun to melt the composite fibers will move and no longer follow the predefined composite orientations exactly. This issue could be resolved by performing a post-cure orientation analysis after preforming the calculations described here, however this issue will be present regardless of the before-cure fiber path planning strategy. Furthermore, by minimizing strain energy the potential for wrinkling to occur within the material during curing is reduced. This is due to the reduced internal strain energy, fiber stretching, and fiber compression resulting from the optimization of the fiber positions.

To begin, a reasonable approximation of the relaxed surface mesh must be determined to reduce optimization time. Geodesics have been utilized frequently in the remeshing domain (Sifri et al., 2003) and shortest path planning (Remeikova et al., 2019). A geodesic can be defined as a locally shortest path between two points in curved space where the relative angle of the path does not
change (i.e. constant velocity). Therefore, a forward marching algorithm inspired partially by the work performed by Surazhsky et al. (2005), Kumar et al. (2003), and Kimmel and Sethian (1998) on calculating geodesics for triangular meshes was created. Kumar et al. (2003) describes two methods, method 2a and method 2b, which are most similar to the approach developed in this work. The approach created here is a simple geodesic propagation algorithm and is expressed in Algorithm 1. A graphical representation of one step of this algorithm is shown in Figure 3.1. Utilizing this geodesic algorithm, a new parameterization of the surface is created with basis vectors orientated in the chosen direction of the geodesic paths. The entire parameterization procedure can be observed in Algorithm 2, and a simple representation of the assignment procedure can be seen in Figure 2.

The initial basis vector directions need to be chosen such that it allows all vertices of the surface to be parameterized. For developable models, this is not usually an issue; however, with doubly-curved surfaces obtaining a full parameterization of the surface can be difficult. A few methods to remedy this issue were explored, such as increasing the number of geodesic lines in areas with low geodesic path density, interpolation between geodesic lines, and finally, averaging the directions of the surrounding parameterized vertices. All of these methods were performed in a manner which takes care not to flip the normal of any faces in the parameterized UV space. A face normal flip is indicative of crossing geodesics and results in an invalid parameterization if a vertex is assigned
Algorithm 1: Geodesic tracing algorithm

**input**: element, point, direction, length

**output**: intpnt, nextvector, nextelement, record

1. neighbors ← FindElementNeighbors(element)
2. point_uv ← ConvertElementVerticesBarycentric(element)
3. duv ← ConvertDirectionBarycentric(direction)
4. edge ← FindIntersectedEdge(point_uv, duv)
5. intpnt ← FindIntersectionPoint(edge, duv)
6. if CheckIntersection(record, intpnt) then
   7. return intpnt, None, None, record
7. record ← AddToRecord(intpnt)
8. nextelement ← FindNextElement(duv, element)
9. if nextelement is None then
   10. return intpnt, None, nextelement, record
11. nextdirection ← RodriguesRotation(normal[element], normal[nextelement], direction)
12. return intpnt, nextvector, nextelement, record
Algorithm 2: Geodesic parameterization algorithm

**input**: parameterization, vertices, normals

**output**: geodesic based parameterization

1. geoparameterization ← zeros(parameterization.shape[0], 2)
2. for i in range(0, len(vertices)) do
3.   neighbors ← FindNeighbors(vertices[i])
4.   for neighbor in neighbors do
5.     direction, length, uvstart, element, distance ← FindCloseGeodesic(neighbor, vertices[i])
6.     abovebelow ← dot(cross(direction, distance.u), normals[element])
7.     ppoint.u ← uvstart.u + distance.u + length
8.     ppoint.v ← abovebelow*distance.v + uvstart.v)
9.   if not CheckNegativeArea() then
10.      geoparameterization[i] ← ppoint
11. return geoparameterization

Based on a "crossed" geodesic. Various initial basis vector direction methods were investigated and chosen to provide the most optimal method for vertex assignment. The optimal method allows the algorithm to assign a basis vector for each vertex of the mesh. The basis vector angles used to create the parameterization are not necessarily crucial for the final fiber orientation directions because the optimization step will remove any discrepancies. Once optimized, the parameterization can be queried for any ply angle, and the corresponding optimal fiber orientations can be returned.

A geodesic path is terminated in a few different cases: intersection with a previously traced geodesic or an edge of the surface. In our algorithm, an edge is defined as a difference in element normals that is close to ninety degrees. Each geodesic path is stored in a record which maps the geodesic to a surface element and is used later to assign each vertex based on the closest geodesic path. The parameterization resulting from this procedure is then passed into the optimization algorithm described in the next section as the deformed state of the finite strain model.
3.2 Strain Energy Optimization

Discrete geodesics on tessellated surfaces will not always follow an optimal trajectory due to the inaccuracy of the tessellation compared to the real surface shape (Kumar et al., 2003). This error stems not only from numerical calculations but physical inconsistencies due to the tessellation procedure, especially in doubly-curved surfaces, which do not represent the difference in element normals accurately. Using a discrete geodesic path created in this manner for path generation may create undesired fiber orientations that can increase internal stresses or produce unwanted material properties. Geodesic paths on certain doubly-curved surfaces can also lead to path crossing (see Figure 4.1), which is undesirable for composite manufacturing because fiber paths would need to be trimmed and overall material integrity could be compromised. Therefore, it is proposed that by minimizing the strain energy between a discrete geodesic based parameterization and the tessellated surface, fiber orientations can be computed which reduce crossing paths and internal energy. For developable models, the geodesic approximation is a very good estimate of the lowest strain energy state; however, once this process is attempted on most non-developable models, the geodesic approximation is no longer minimal and the geodesic paths exhibit undesirable behavior, i.e., crossing or deviation.

3.2.1 Global Strain Energy

The global strain energy of the geodesic parameterization can be computed following the definition provided by finite strain theory and continuum mechanics. To begin, each element of the surface model is converted to a 2D representation, $p_{3D}$, using the in-plane basis vectors with the highest singular value. This 2D parameterization is the reference state of the surface model. The resulting parameterization based on the geodesic paths, $p_{UV}$, will be considered as the deformed state of the surface model. Since this study is primarily dealing with triangular elements, both of these parameterizations will contain a $2 \times 3$ matrix for each element; therefore, a row of ones must be appended to each element in the parameterization to create convenient square matrices. According to continuum mechanics, the deformation gradient is a gradient mapping between a
deformed and reference configuration (Lubliner, 2008) and is defined as

$$F = \frac{\partial x}{\partial X}. \quad (3.1)$$

With this in mind a formula can be conceived for computing the deformation gradient given both reference and deformed configurations as follows

$$F = p_{3D} * p_{UV}^{-1}. \quad (3.2)$$

This calculation will result in a $3 \times 3$ matrix for each face in the surface model, however since the third row of each element matrix was padded, the components of the deformation gradient related to this direction can be ignored, namely the third column and third row of the deformation gradient, reducing it down to a $2 \times 2$ matrix with the following components:

$$F = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}. \quad (3.3)$$

It is important that only rotation independent deformation is considered, therefore the right Cauchy-Green deformation tensor (Kaye et al., 1998) will be used. Additionally, this definition of the right Cauchy-Green deformation tensor follows infinitesimal strain theory, which applies to this analysis nicely because it is assumed that movement of the parameterization nodes during optimization is very small. The right Cauchy-Green deformation tensor, $C = F^T F$, can be proven to be rotation independent because the deformation gradient consists of two components, $R$ and $U$, where $R$ represents the rotation and $U$ is the right stretch tensor. Therefore, the right Cauchy-Green deformation tensor can be decomposed into the following components

$$C = RR^T UU^T. \quad (3.4)$$

Since a rotation multiplied by the transpose of itself is the identity what is left is only the square of the deformation due to stretch (Kaye et al., 1998). Belytschko et al. (2013) defines strain using the right Cauchy-Green deformation tensor as

$$\epsilon = \frac{1}{2}(C - I). \quad (3.5)$$
However, strain in this case is defined as $[\epsilon_{11}, \epsilon_{22}, \gamma_{12}]$, therefore the final term must be divided by 2 to obtain $\gamma_{12}$. Strain energy density as defined by Kelly (2013) in equation 8.2.15 is

$$u = \frac{1}{2} \sigma_{xx} \epsilon_{xx}. \quad (3.6)$$

Modifying this equation slightly to obtain the total strain energy over the entire surface yields

$$\text{Energy} = \sum_{k=0}^{N} \frac{1}{2} \epsilon_{kk}^2 S A_{uv_k}, \quad (3.7)$$

where $N$ is the total number of elements, $S$ is the stiffness tensor related to the material, and $A$ is the area of each element in the UV parameterization. The area of each element of the UV parameterization is calculated using the following,

$$A = \frac{1}{2} |p_{uv}|. \quad (3.8)$$

The resulting energy exhibits the difference in potential between the surface model and the geodesic based parameterization. In the next section of this study it will be discussed how this value can be minimized to determine the lowest energy configuration for any surface geometry.

### 3.2.2 Global Strain Energy Gradient

Ideally, to minimize a value, the optimization algorithm must be supplied with a gradient of the function in order to know in what direction will minimize or maximize the loss function. Here, the proof for computing the gradient of the strain energy equation derived in the previous section will be discussed. The resulting gradient is then used in the optimization algorithm to determine the final vertex configuration, which will minimize the global strain energy. Before the derivative of the strain energy can be computed, the gradient of the element area and the strain of each element with respect to nodal displacement must be derived. This process begins with the definition of the area of each element,

$$A_{uv} = \frac{1}{2} |p_{uv}|. \quad (3.9)$$
Then, the minor matrix, $M$, of $A_{uv}$ where $M_{ij} = \left| \text{minor}(p_{uvij}) \right|$ must be obtained and in which each minor is computed by removing the $i$\textsuperscript{th} row and $j$\textsuperscript{th} column of $p_{uv}$ and taking its determinant. The cofactor matrix can then be computed with

$$Cof = ((-1)^{i+j} M_{ij})_{1 \leq i,j \leq n^*},$$

which, when transposed, is the adjugate matrix of $A_{uv}$. Bringing all of these components together and by using Magnus & Neudecker’s (Magnus and Neudecker, 2019) version of Jacobi’s formula, which expresses the derivative of the determinant of a matrix in terms of the adjugate matrix, the gradient of the area with respect to nodal displacement can be derived,

$$\frac{\partial A_{uv}}{\partial p_{uv}} = \frac{1}{2} \text{tr}(\text{adj}(A_{uv}) \frac{\partial p_{uv}}{\partial p_{uvij}}),$$

where $\frac{\partial p_{uv}}{\partial p_{uvij}}$ is the derivative of each nodal displacement with respect to moving all the other nodes of each element. Then the derivative of the deformation gradient (Eq. 3.2) with respect to nodal displacement can be computed as

$$\frac{\partial F}{\partial p_{uv}} = p_{uv}^{-1} \frac{\partial p_{uv}}{\partial p_{uvij}} p_{uv}^{-1} p_{3D}.$$  

(3.12)

The last component which must derived is the gradient of strain with respect to nodal displacement. By taking the derivative of Eq. 3.5 the following can be concluded:

$$\frac{\partial \epsilon}{\partial p_{uv}} = \frac{1}{2} \left( \frac{\partial F^T}{\partial p_{uv}} F + F^T \frac{\partial F}{\partial p_{uv}} \right).$$

(3.13)

As with the strain calculation, this result must also account for the $\frac{\gamma_{12}}{2}$ term by dividing the third element of the strain gradient by two. Now, it is known from Eq. 3.7 that the formula for the energy of a specific element is

$$E = \frac{1}{2} \epsilon^2 S A_{uv}.$$  

(3.14)

Therefore, with the application of the chain rule, the derivative of Eq. 3.14 with respect to nodal displacement is found to be

$$\frac{\partial E}{\partial p_{uv}} = \tau S \frac{\partial \tau}{\partial p_{uv}} A_{uv} + \frac{1}{2} S \epsilon^2 \frac{\partial A_{uv}}{\partial p_{uv}},$$

(3.15)
where \( S \) is the stiffness tensor of the material. With Eq. 3.15, the gradient of the strain energy for every element in the surface mesh with respect to nodal displacement of all vertices in the U or V direction can be calculated.

### 3.2.3 Optimization

In this work, a variation of the RMSprop optimization algorithm proposed by Hinton et al. (2012) was used. This algorithm is commonly used in the machine learning community to train neural networks. The modified algorithm that was developed (Algorithm 3) was inspired by the article written by Bouda (2017) and his code examples. RMSprop is a gradient-based optimization algorithm which normalizes the gradient with a moving average of the square of the gradients. Additionally, the algorithm used in this study utilizes the concept of momentum, which decreases convergence time while not sacrificing accuracy by adjusting the velocity of the descent rather than the position. This algorithm was chosen for this study after comparing some off-the-shelf optimization algorithms, such as the conjugate gradient algorithms supplied in the Scipy (Virtanen et al., 2020) optimize package. This RMSprop algorithm also allows greater flexibility in adjusting the optimization parameters.
**Algorithm 3:** RMSprop optimization algorithm

**Input**: \( F, dF, x_0, \text{precision}=1e^{-4}, \text{maxsteps}=10000, \text{lr}=1e^{-3}, \text{decay}=0.7, \text{eps}=1e^{-8}, \mu=0.8 \)

**Output**: optimized vertex locations in UV space, \( x \)

1. Initialize variables
2. if \( F(x) < \text{precision} \) then
   - return \( x \)
3. for 0 to maxsteps do
4.   \( b_0 \leftarrow F(x) \)
5.   \( dx \leftarrow dF(x) \)
6.   \( dx_{\text{mean}_sqr} \leftarrow \text{decay} \times dx_{\text{mean}_sqr} + (1 - \text{decay}) \times dx^2 \)
7.   \( \text{momentum} \leftarrow \frac{\mu \times \text{momentum} + \text{lr} \times dx}{(\sqrt{dx_{\text{mean}_sqr}} + \text{eps})} \)
8.   \( x \leftarrow \text{momentum} \)
9.   if \( \text{abs}(F(x) - b_0) < \text{precision} \) then
10.  break
11. return \( x \)
CHAPTER 4. EXPERIMENTS AND RESULTS

4.1 Introduction

A variety of models have been tested with the strain energy optimization algorithm derived in the previous section. Therefore, in order to exhibit the most useful results, this discussion will focus primarily on non-developable surfaces. The results of this algorithm on different surfaces are shown in Figure 2 for reference and to show how well it can perform on different classes of surfaces. The majority of the results shown in this section will focus on a model that has been named the armchair model. This model was chosen because the critical point on the spherical region of the model clearly explains how geodesics would be a poor option for determining fiber paths due to path crossing. The details of the armchair model can be seen in a summary of all the tested models in Table 4.1.

The optimization algorithm can take the parameterization created from generated geodesic paths and determine the lowest strain energy parameterization, which can be queried for fiber orientations at any point on the model. All of the fiber orientation results shown in this study were performed on a computer with a 4.1 GHz CPU Intel 6700K and 16 GB RAM. The experiments

<table>
<thead>
<tr>
<th>Model</th>
<th># of Vertices</th>
<th># of Faces</th>
<th># of Geodesics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arm Chair</td>
<td>161</td>
<td>280</td>
<td>62</td>
</tr>
<tr>
<td>Cylinder</td>
<td>204</td>
<td>352</td>
<td>128</td>
</tr>
<tr>
<td>Torus</td>
<td>290</td>
<td>504</td>
<td>630</td>
</tr>
<tr>
<td>Saddle</td>
<td>1024</td>
<td>1922</td>
<td>560</td>
</tr>
<tr>
<td>Hat Stiffener</td>
<td>1485</td>
<td>2816</td>
<td>202</td>
</tr>
<tr>
<td>Curved plate</td>
<td>6559</td>
<td>12796</td>
<td>406</td>
</tr>
</tbody>
</table>
described here use an anisotropic material with properties based on a simple composite fiber-
reinforced material (CFRP), which is summarized in Table 4.2.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_1$ (GPa)</td>
<td>1.415e11</td>
</tr>
<tr>
<td>$E_2$ (GPa)</td>
<td>8.5e9</td>
</tr>
<tr>
<td>$E_3$ (GPa)</td>
<td>8.5e9</td>
</tr>
<tr>
<td>$\nu_{12}$</td>
<td>0.33</td>
</tr>
<tr>
<td>$\nu_{13}$</td>
<td>0.33</td>
</tr>
<tr>
<td>$\nu_{23}$</td>
<td>0.33</td>
</tr>
<tr>
<td>$G_{12}$ (GPa)</td>
<td>5.02e9</td>
</tr>
<tr>
<td>$G_{13}$ (GPa)</td>
<td>5.02e9</td>
</tr>
<tr>
<td>$G_{23}$ (GPa)</td>
<td>2.35e9</td>
</tr>
</tbody>
</table>

### 4.2 Arm-Chair

#### 4.2.1 Experiment

For the armchair model experiment, the initial parameterization was created with a geodesic
angle of zero degrees and a spacing of 0.1, which resulted in 62 geodesic paths. A no-displacement
constraint was applied on the point of the parameterization closest to the origin in order to guarantee
a full definition of the optimization problem. The geodesic based layup on the armchair model
can be seen in Figure 4.1.

#### 4.2.2 Results

The desired ply angle for this example was chosen to be 45 degrees, therefore the geodesic based
parameterization was rotated by the desired ply angle prior to optimization. Then, the geodesic
based parameterization’s strain energy was minimized relative to a flattened representation of the
armchair model. Figure 4.2 shows a comparison of the original, geodesic based parameterization
Figure 4.1: Unoptimized \textit{armchair} model layup

and the optimized, minimum strain energy parameterization. Initially, the geodesic based parameterization has a strain energy of 7.70 J/m and within 112 seconds the optimization has reduced the strain energy to 0.38 J/m, as seen in Figure 4.3, at which point the optimizer termination tolerance had been reached. The units reported for strain energy are in Joules per meter which is the strain energy density over the 2D space, therefore the material thickness remains in the unit definition.

Once the geodesic based parameterization has been created any ply angle can be calculated by rotating the geodesic parameterization by the desired angle and re-optimizing. A comparison of desired ply angle, compute time, and initial strain energy is presented in Figure 4.4 and the corresponding ply angle optimizations can be seen in Figure 4.5. It can be concluded from this experiment that compute time must, primarily, be related to geometrical complexity and, secondarily, optimization initial conditions. At desired ply angles of 30 and 70 degrees the most amount of time was taken to optimize, however at 30 degrees the initial starting energy was much lower than at 70 degrees.
Figure 4.2: *Armchair* - parameterization optimized to 45 degree fiber directions

Figure 4.3: *Armchair* - energy over 10000 optimization iterations
Figure 4.4: Varying desired fiber ply angle vs compute time and initial energy

Figure 4.5: Armchair - composite fiber layups 0, 45, and 90 degrees
4.3 Method Analysis

A variety of other models were tested and analyzed, and in this section insights derived from those tests will be discussed. Firstly, the effect of model size on optimization time was briefly studied, and the corresponding results can be seen in Figure 4.6.

![Figure 4.6: Compute time for various sizes of models. Please note the use of a logarithmic scale time.](image)

For all runs in this experiment the desired ply angle was 45 degrees and the optimization termination residual was set to 1e-6. The RMSprop parameters were tuned for each model in order to get the quickest convergence, which can be seen in Table 4.3.

<table>
<thead>
<tr>
<th>Model</th>
<th>Step Size (lr)</th>
<th>Momentum (μ)</th>
<th>Decay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arm Chair</td>
<td>1e-3</td>
<td>0.8</td>
<td>0.7</td>
</tr>
<tr>
<td>Cylinder</td>
<td>1e-3</td>
<td>0.8</td>
<td>0.7</td>
</tr>
<tr>
<td>Torus</td>
<td>1e-3</td>
<td>0.8</td>
<td>0.7</td>
</tr>
<tr>
<td>Saddle</td>
<td>1e-4</td>
<td>0.8</td>
<td>0.1</td>
</tr>
<tr>
<td>Hat Stiffener</td>
<td>1e-3</td>
<td>0.8</td>
<td>0.7</td>
</tr>
<tr>
<td>Curved plate</td>
<td>1e-4</td>
<td>0.8</td>
<td>0.3</td>
</tr>
</tbody>
</table>
From these results, the time to parameterize and optimize each model can be compared. Therefore, it can be deduced that model size appears to have little effect on the optimization training time, which further reinforces our conclusion earlier that model geometry in certain ply angles must significantly contribute to optimization complexity.

### 4.4 GPU-accelerated Geodesic Computation

In addition, as model size increases, geodesic parameterization increases; however, it is possible to speed up the geodesic generation by moving those calculations to a GPU. Special care would have to be taken in order to ensure that geodesics who cross, either themselves or other geodesics, are terminated before being used to assign a vertex, however, a large speedup in terms of computation time can be observed. A simple experiment showing the results of calculating the geodesics on a GTX 970 GPU can be seen in Figure 4.7. Each thread of the GPU is responsible for computing one geodesic path, therefore, depending on the number of possible threads in the GPU, most of the geodesic paths will be computed simultaneously, rather than sequentially. One method for dealing with crossing geodesics would be to store all paths in shared memory and check for intersections against that geodesic record; however, this introduces a race case and may encounter errors due to timing. Another solution would be to compute all geodesic paths disregarding all other currently generating paths and trim each geodesic after computation, therefore at each intersection, one of the two geodesics would be terminated and the other allowed to remain.
Figure 4.7: CPU vs GPU computation time for geodesic implementation. Please note the use of a logarithmic scale time.
CHAPTER 5. APPLICATIONS

5.1 Introduction

This discussion of the geodesic generation and strain energy optimization will conclude with a short overview of a couple of applications which this method can be applied to in research. As was discussed earlier in this study, this process can be used to determine optimal paths for automated tape layup machines such as NASA’s ISACC AFP machine. In addition to this industry use, this process can also be used to improving Finite Element Analysis (FEA) models, which will be exhibited in our integration with the De-La-Mo software (Bingol et al., 2019) developed at Iowa State University (ISU). Finally, an example will be shown where this process is applied to a highly complex cardiovascular heart model to determine accurate heart tissue fiber orientations.

5.2 De-La-Mo Abaqus Integration

De-La-Mo is an automated framework for creating 3D CAD models and pre-processing scripts for use in FEA software. Traditionally, FEA software packages compute fiber orientations based on simple coordinate frame rotations for each facet in the model. This works effectively for surfaces that are simply curved in one axis. However, for models that are curved in more than one axis, this process will prove ineffective. Therefore, this new geodesic/strain energy-based approach has been integrated with De-La-Mo to provide accurate fiber orientations even for surfaces that are curved in more than a single axis. Specifically, our process is injected into the De-La-Mo pre-processing step, which creates the composite fiber layup and sets the fiber orientation for each FEA element to be equivalent to the orientation computed by the strain energy optimization. Figure 5.1 shows the results of an experiment utilizing the strain energy optimization module on the De-La-Mo example labeled 08_Curved_No_Delam which creates a curved 8-layer composite plate with a layup of \([0, -45, 45, 90]_s\) and applies a static load to one end. This example shows the capability of this
process to perform on a doubly curved surface within De-La-Mo and an FEA modeling environment. It exhibits the ability of our De-La-Mo integration and is one of a few examples on common composite surfaces, which are presented as part of the online documentation for De-La-Mo. Additional information on downloading and using the open-source De-La-Mo code including our fiber orientation module can be found at http://thermal.cnnde.iastate.edu/de-la-mo/index.html

5.3 Cardiovascular Fiber Modeling

Computing accurate fiber orientations is essential for use in biomechanics simulations of cardiac models (Jafari et al., 2019). Specifically, performing standard coordinate system transformation to determine fiber orientations have been shown to be inaccurate, especially on these complicated heart geometries. Therefore, another example application is shown where this process is applied to a complex geometry curved in multiple axes. Computing the fiber orientation of cardiac models is challenging both due to the complexity of the geometry as well as the change in fiber orientation through the thickness of the heart wall. Using our approach, fiber orientations at both the external surface (Epicardium) and the internal surface (Endocardium) can be computed and make use of coordinate frame interpolation approaches (Krishnamurthy et al., 2016) to interpolate the fiber directions accurately inside the cardiac walls. The results of this process for a cardiac biventricular model at the standard $+60^\circ$ and $-60^\circ$ are shown in Figure 5.2.
Figure 5.2: Biventricular cardiac geometry (left) with +60° fiber orientations (middle) and −60° fiber orientations (right).
CHAPTER 6. FUTURE WORK SUMMARY AND DISCUSSION

6.1 Future Work

Following the completion of this work a few areas of improvement were identified as potential opportunities for future work. Although a variety of different models and the effectiveness of this process on those models have been discussed in this study, the method for geodesic parameterization could be improved to increase robustness and simplify usage. Currently, the process for obtaining a valid geodesic parameterization can be quite difficult, especially if the geometry has many voids or extreme topology. Validation of the resulting composite fiber orientations against simulation or AFP experimental composite components would solidify the accuracy of this new method for predicting relaxed composite fiber paths. Additionally, the minimum strain energy optimization could also be improved with a combination of other draping methods to further optimize the fiber paths, such as the effect of interlocked fibers or other draping and wrinkling forces.

6.2 Conclusions

In this study, a method for computing composite fiber orientations on doubly-curved surfaces, which reduce geodesic path crossing and minimize global strain energy was presented. This process shows that despite tessellation error or geodesic parameterization inconsistencies, a global strain energy minimum can be reached for complex surface models in reasonable computing time. The computed fiber orientations represent the composite fiber orientations natural inclination to relax during matrix solidification in the composite manufacturing process. As a consequence, this approach enables developing accurate finite element models and fiber paths for AFP manufacturing, which will already be in a near relaxed state during layup. This nearly relaxed state reduces undesirable material properties such as potential for wrinkling, fiber crossing, gaps, buckling. This process was analyzed by comparing various experimental variables between different models showing the
effectiveness of this technique on developable and non-developable models of varying complexity and size. It has been concluded that the running time of the optimization is significantly influenced by the complexity of the surface, rather than model size or initial conditions. Finally, the entirety of this process has been demonstrated on a few industrial use cases such as integration with Abaqus, a finite element analysis program. The Python implementation of this study is available open-source online at https://idealab-isu.github.io/autofiber/.
REFERENCES


APPENDIX. ADDITIONAL MATERIAL

Unoptimized Cylinder Paths

Strain Energy:
2.5e-28 J/m
5.8e-28 J/m
7.2e-28 J/m
2.48e-28 J/m
5.8e-28 J/m
7.2e-28 J/m

Optimized Cylinder Paths

(a) Cylinder
Strain Energy:
83.4 J/m 17.64 J/m 68.36 J/m
0.102 J/m 0.095 J/m 0.109 J/m

Optimized Torus Paths

Unoptimized Torus Paths

(b) Torus

Strain Energy:
111.23 J/m 24.57 J/m 137.98 J/m

Unoptimized Saddle Paths

Optimized Saddle Paths

(c) Saddle
Figure 2: Various models tested which show the un/optimized composite fiber layups at 0, 45, and 90 degrees.