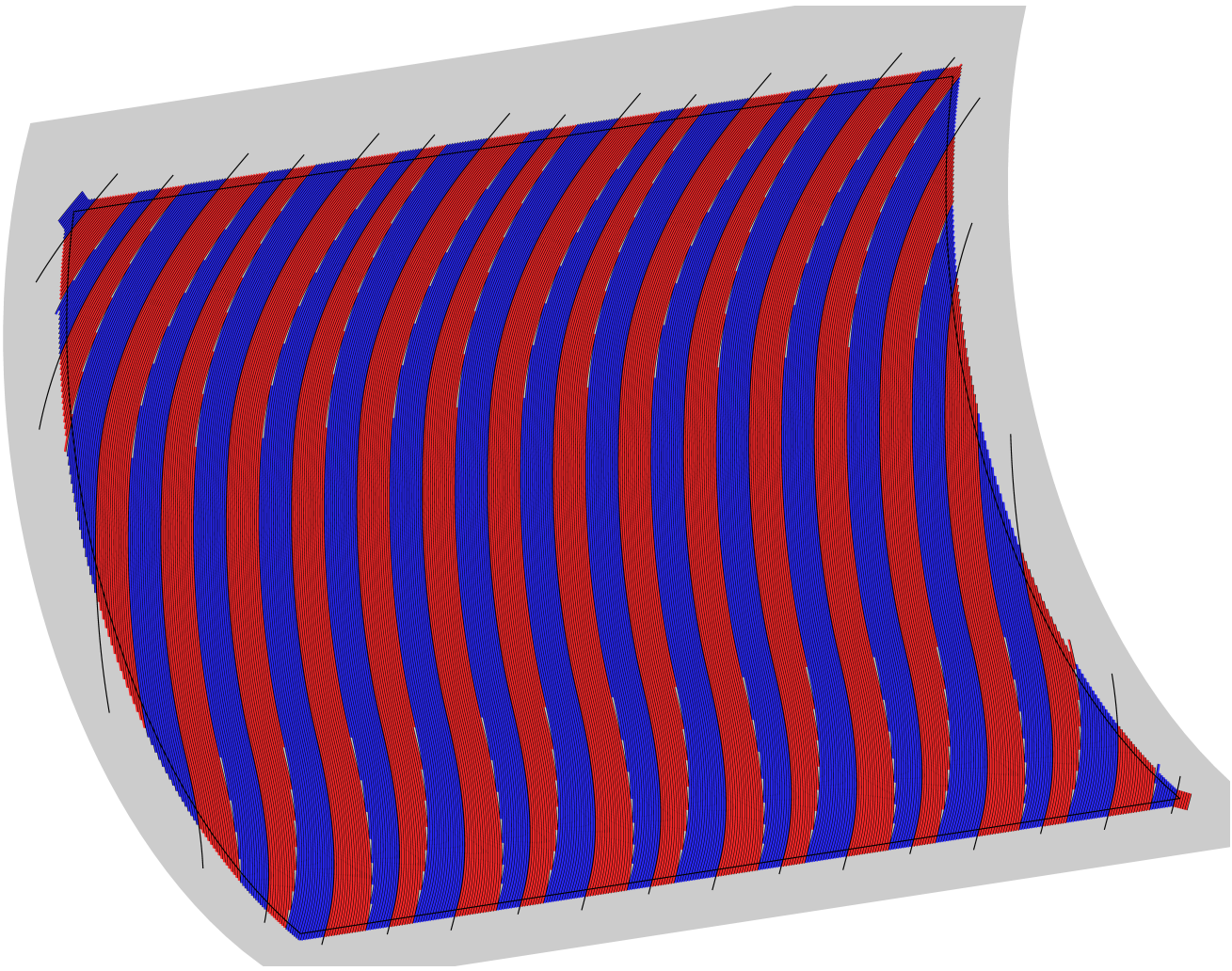
A Fibre-Path Generation Algorithm for Fibre-steered Variable Stiffness Laminates

Master of Science Thesis

Max Hijne



TU Delft Delft University of Technology

Aerospace Engineering

A Fibre-Path Generation Algorithm for Fibre-steered Variable Stiffness Laminates

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

Max Hijne

12-July-2022

Graduation Committee:

Chair Holder:	Prof. dr. ir. R. Benedictus
Committee Member / Supervisor:	Dr. ir. J.M.J.F. van Campen
Committee Member:	Dr. ir. D.M.J. Peeters
Committee Member:	Dr. ir. O.K. Bergsma

Acknowledgements

There are several people that I would like to thank for their aid in realizing this thesis. First of all, I want to express my thanks to my supervisor Julien, who has guided me through the thesis process and has been a constant source of feedback and support. Additionally, I would like to thank Daniël Peeters and Hugo Verhelst for their insights and help on various topics and also Regina Hoffmann for her guidance on structuring and writing my thesis.

And most importantly, I am immensely grateful to my parents for their unwavering support and patience during my studies, without whom I would not be where I am today.

Abstract

The development of novel materials has been of great interest to the aerospace industry in the pursuit of efficiency and cost reduction. Fibre-steered variable stiffness laminates have promising characteristics due to ability to better tailor designs for expected loading, as the fibre orientation can be varied within a layer. In particular, fibre-steered variable stiffness laminates are expected to provide significant improvements in buckling performance over conventional composite laminates. The TU Delft has been exploring the feasibility of developing a partially automated solution to generate such laminates based on high-level design drivers such as expected loading and weight, with the end result being a fibre-steered laminate design with path placement instructions for each layer. In line with this goal, this thesis was created to develop a fibre placement planning algorithm for fibre-steered layers on moderately curved open shells for a given fibre angle distribution.

The fibre-steered layers are to be produced by Automated Fibre Placement (AFP), which means that several manufacturing defects have to be considered. Gaps between courses are inevitable due to course shifting in combination with the prescribed fibre paths and tow cuts run the risk of tow straightening due to loss of steering control of the AFP roller head. Furthermore, steering limits have to be observed to prevent tow kinking.

The thesis objective was approached by dividing it in three parts. Firstly, the geometry of the mould and course paths were defined parametrically in the form of NURBS/B-splines surfaces and curves. NURBS/B-splines have been widely used in Computer Aided Modelling (CAM) and have extensive support in terms of existing algorithms and documented work. Additionally, performing most operations in the parametric space allows for greater accuracy compared to general finite element methods.

Secondly, the fibre angle distribution was converted into a fibre layer. This was initiated by generating initial course centre lines, which were then fit onto the parametric surface. Representations of tow courses were made by geodesically offsetting the course centre lines in the binormal direction. A curve-curve intersection algorithm was then used to accurately detect tow overlaps, which were handled according to the used cutting strategy. Two cutting strategies have been developed, one for general use and one to specifically prevent tow straightening. In addition, the curvature of the course centre line was registered to check for tow kinking/buckling, and the minimum cut length was enforced by evaluating the length of tow segments.

Lastly, fitness criteria for fibre-steered layers were defined. Layers were checked for coverage / gap percentage, fibre angle error and number of cuts. These can be used to objectively grade layers against each other, or to assess the effects of fibre steering against conventional laminates. A number of layers were generated and compared on these criteria. It was found that the layer coverage of fibre steered layers can be maximized when the course width and the rate of steering is maximized, whereas better fibre angle compliance is achieved when more, smaller width courses are used with a more gradual steering.

Contents

Nomenclature

Acronyms

AFP	Automated Fibre Placement
ATL	Automated Tape Laying
CAD	Computer-Aided Design
CAM	Computer-Aided Modelling
CTP	Continuous Tow Percentage
CTS	Continuous Tow Shearing
DLVFA	Discrete Linearly Varying Fibre Angle
LVFA	Linearly Varying Fibre Angle
MFAE	Maximum Fibre Angle Error
NURBS	Non-Uniform Rational B-Splines
TFP	Tailored Fibre Placement
VSL	Variable Stiffness Laminates

Greek Symbols

α	Interpolation Scalar
δ	Difference in u and v
ϵ	Error Tolerance
Γ	Control Polygon
κ_g	Geodesic Curvature
λ	Normalisation constant
μ	Knot Insertion Interval
σ	New Knot Location
τ	New Knot Location
θ	Fibre Angle

Roman Symbols

$\mathbf{B}$	Binormal Vector
$\mathbf{b}$	Unit Binormal Vector

$\mathbf{k}$	Curvature Vector
$\mathbf{n}$	Unit Normal Vector
$\mathbf{T}$	Tangent Vector
$\mathbf{t}$	Unit Tangent Vector
A	Surface Area
A	Weighted Coordinates
C	NURBS Curve
D	Connectivity Matrix
d	Christoffel Symbols
d	Distance between Centre of the Plate and T_1
e	Christoffel Symbols
G	First Fundamental Form of Differential Geometry
J	Jacobian Matrix of f and g with respect to u and v
k	Knot Multiplicity
L	Panel Length or Width
M	Gradient Matrix
m	NURBS Order
N	Basis Function
N	NURBS Basis Function
P	NURBS Control Point
p	NURBS Degree
Q	Point on the Surface
R	Control Points
S	NURBS Surface
s	Arc Length
s	Curve Length
t	NURBS Curve Parameter
T_0	Fibre Angle at the Centre of the Plate
T_1	Fibre Angle at a Characteristic Distance away from Centre
U	NURBS Knot Vector
u	NURBS Surface Parameter, Perpendicular to v
V	NURBS Knot Vector
v	NURBS Surface Parameter, Perpendicular to u
W	NURBS Weights Matrix
w	NURBS Weight

Chapter 1

Introduction

The demand for weight savings in the aerospace industry has led to the introduction of many novel materials being applied in structural components as a measure to reduce fuel consumption and increase their aircraft efficiency [1, 2]. In the late 1960s, the first composite parts were introduced in military as a measure to reduce weight [3]. Commercial aviation followed shortly after in the early 1970s. Initially only on smaller parts, but as the years progressed more and more of the (structural) components of civil aircraft became out of composites. Nowadays, the state of the art commercial aircraft, such as the A350-XWB and Boeing 787 are composed of between 40 and 50% of composites by weight, and around 80% by volume.

The main advantage of composites over conventional metals such as aluminium alloys is their superior strength-over-weight ratio, which allows for lighter parts. This is in part due to the ability to better tailor the properties of the composite laminate to the structural requirements compared to metals. The novel technique of producing fibre steered composite laminates, which have a variable stiffness over the panel dimensions, could be the next step in better tailoring structural components to loading to achieve further weight reductions [4].

The TU Delft has done significant work in developing strategies to design such variable stiffness composite laminates [5–8]. Currently, the need has arisen for a better algorithm to design fibre paths for these types of laminates. This would accelerate the design process and allow for testing and validation of the panels. Therefore, the development of a software solution for the design of machine paths for variable stiffness laminates is the goal of the thesis.

The latest research on the topic of fibre path generation for variable stiffness laminates by *Wurpel* (2015) illustrated the challenges of creating such an algorithm [9]. Currently, it can only be applied to designs of flat panels, whereas typical aerospace panels are doubly-curved. Furthermore, there is still room for improvement with regards to reducing the amount of voids per layer by utilizing better cutting strategies. To this end, the research question of this thesis is defined as follows:

"How can the TU Delft Three-Step Approach for VSL be applied to moderately doubly-curved shells for aerospace applications, whilst minimizing manufacturing defects by using cutting strategies?"

This thesis is divided as follows: In chapter 2, the state of the art on the topic of all the important aspects of variable stiffness laminates is explored. Furthermore, at the conclusion of chapter 2, the goals of the thesis are outlined. Then in chapter 3, a suitable framework for the new algorithm is selected. Afterwards, the construction of the new fibre path generation algorithm is discussed in chapter 4. Additionally, chapter 5 will supplement the new algorithm with fitness criteria to evaluate the generated layers. Finally, chapter 6 will demonstrate the new algorithm, after which conclusions and recommendations will be given in chapter 7.

Chapter 2

Current State of Variable Stiffness Laminate Design

The aim of this chapter is to provide background information on all important aspects of variable stiffness laminate design. This will provide the necessary context to specify goals for the thesis to answer the research question, which will then be examined in the further chapters.

This chapter has been divided in three parts. Section 2.1 covers the two main types of variable stiffness laminates. In section 2.2, manufacturing methods that have been used for the production of variable stiffness laminates are examined. Then, section 2.3 will be on the development and aspects of the Three-Step Approach for variable stiffness laminate design. Furthermore, the state of the art within the context of the research question is explored, after which the goals of the thesis can be properly defined.

2.1 Types of Variable Stiffness Laminates

As the focus of the thesis is on the development of production methods of so-called Variable Stiffness Laminates (VSL), this section will explain the two main methods to create this type of laminate. VSL are composites with a varying stiffness distribution along the length and width of the laminate [4, 9]. The reason to apply this technique is to better tailor laminates to loading. In particular, it has been shown that VSL could provide significant improvements in buckling performance over quasi-isotropic laminates [4, 5, 10, 11].

The term was first coined by *Gürdal and Olmedo* (1992), alongside an analysis of the effects of the spatially varying fibre orientations [12]. They studied the in-plane response of panels with steered fibres along a linearly varying path, and used a Rayleigh-Ritz method to determine the lowest buckling load when subjected to a uniform end displacement. Since then, the development of VSL at the TU Delft has diverged into two main topics, namely fibre steering and laminate blending.

2.1.1 Fibre Steering

Fibre steering is a production technique, where the fibres are no longer placed unidirectionally, but instead are steered and placed along a curvilinear path. This is typically achieved by using robotic arms with a capability to place the long strips of fibre reinforced material at the desired angle within a lamina. It was first used as an attempt to improve the performance composite panels in compression, by weaving fibres around a hole [13]. The resulting laminates performed better in compression than a conventional laminate with a drilled hole. This later expanded towards producing laminates exclusively through fibre steering. Fibre steering is done almost exclusively with the automated fibre placement manufacturing technique, which will be discussed in subsection 2.2.1. An early fibre steered panel made by *Tatting and Gürdal* (2002) can be seen in Figure 2.1. Note how the steering causes thickness variations over the panel, which is due to the variation of angles of the neighbouring bands.

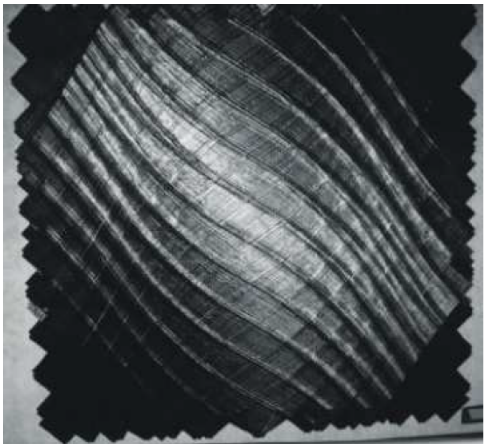


Figure 2.1: One of the early fibre steered panels by *Tatting and Gürdal* (2002) [14]

Early research on fibre steered laminates also introduced the concept of the linear fibre angle variation method by *Gürdal and Olmedo* (1993) [4]. This is a method to describe the path of the steered fibres with a small number of parameters. Here, the path is described by the fibre angle at the centre T_0, the fibre angle T_1 at a distance d from the panel centre and the direction angle θ which determines the direction of variation, as illustrated in Figure 2.2 and written formulaically in Equation (2.1). Nowadays, not every panel is designed by this concept anymore, and instead the use of lamination parameters has resulted in a broader spectrum of fibre steered laminates.

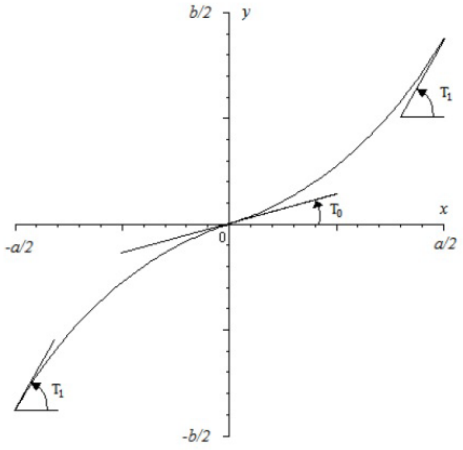


Figure 2.2: Description of a fibre path with the linearly varying fibre angle (LVFA) method [15]

$$\theta(y) = \begin{cases} (T_0 - T_1)\frac{2y}{L} + T_1, & \text{for} \quad 0 \leq y \leq \frac{1}{2}L \\ (T_0 - T_1)(2 - \frac{2y}{L}) + T_1, & \text{for} \quad \frac{1}{2}L \leq y \leq L \end{cases} \tag{2.1}$$

Fibre steering has been explored as a way to vary the stiffness distribution, and subsequently the load distribution along the panel. *Gürdal and Olmedo* (1992) and *Gürdal et al.* (2008) researched the effect of fibre steering by either aligning the fibres perpendicular to the load, or aligning the fibres alongside the load. One of these designs is shown in Figure 2.3. The best results were achieved when the loading from the centre was redistributed to the simply supported sides, which was achieved by steering the fibres in the centre perpendicular to the load. This resulted in improvements up to 80% when compared to the best constant stiffness laminates.

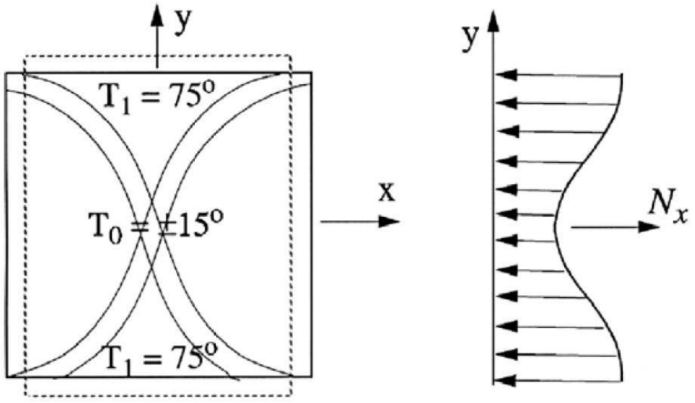


Figure 2.3: Redistribution of the load due to fibre steering perpendicular to the loading direction by *Gürdal et al.* (2008) [16]. Note that the fibre angle is defined with reference to the y-axis, whereas this thesis will define fibre angles with reference to the x-axis.

While most of the research on VSL has been on flat plates, there has been interest in the design of cylinders through the use of fibre steering as well. *Blom* (2010) and *Wu et al.* (2009) produced a number of cylinders to test for maximum fundamental frequency, maximum buckling load under bending and manufacturing constraints of the optimized designs [5, 10]. The designs were moderately more effective than their baseline counterparts, with improvements of around 20-30% per category.

The TU Delft has initialised a program to partially automate the design process of VSL produced by fibre steering, in order to speed up the production and testing [6, 7, 9]. The full approach will be reviewed in section 2.3.

2.1.2 Laminate Blending

Laminate blending is another technique utilized to create Variable Stiffness Laminates. Originally proposed by *Zabinsky* (1994), this technique describes designing a structure in segments, where some or all of the plies within that segment continue into other segments [6, 8, 17]. An example of such a multi-segment structure is illustrated in Figure 2.4, with the Roman numerals denoting the segments. For the most part, the optimisation of a segment is much more on a local scale than fibre steering. However, the segments cannot be designed independently from each other, or the structural integrity of laminate is negatively affected. Therefore, parts of neighbouring segments continue into other segments.

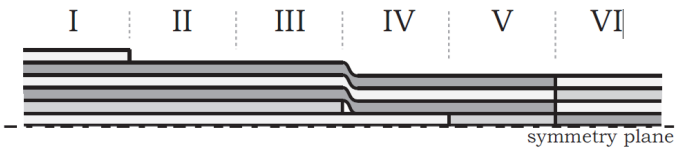


Figure 2.4: Example of a multi-segment structure [8]

Several blending definitions have been developed to ensure the segments are sufficiently compatible with each other. The most straightforward ones are the outward blending strategy and the inward blending strategy. Simply put, they only allow to drop plies on either the outer ply or the inner ply resp., as illustrated in Figure 2.5 [18]. Unfortunately, this severely limits the design space of blended laminates, as it is possible to have a compatible (manufacturable) blended laminate that does not abide by these restrictions. Therefore, most blended laminates follow a more relaxed blending definition, such as the one proposed by *van Campen et al.* (2008), which states that a laminate is fully blended "if there are no dropped edges in physical contact between its sections" [8, 19].

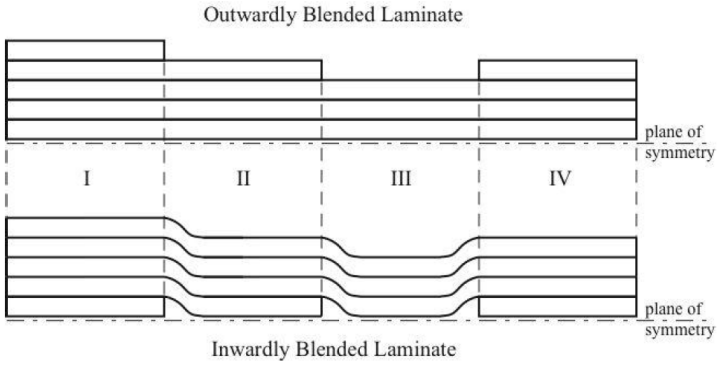


Figure 2.5: Outwardly and inwardly blended laminates [18]

While laminate blending may at some point in the future be combined with fibre steering, this thesis will focus solely on fibre steering to limit the scope of the project.

2.2 Compatible Manufacturing Techniques for the Production of VSL

This section aims to provide an overview of robot-assisted manufacturing techniques of composite structures, to evaluate them for their suitability for the production of VSL by using fibre steering. In subsection 2.2.1, automated fibre placement is discussed. This is the most commonly used technique for fibre steering. Then, in subsection 2.2.2 automated tape laying will be evaluated. This technique places larger widths of tow at the cost of reduced steering capabilities. Tailored fibre placement will be examined in subsection 2.2.3, which is a textile manufacturing technique that stitches fibre bundles onto a fabric base while steering the bundles in the desired orientation. Finally, in subsection 2.2.4 the continuous tow shearing technique is discussed. Continuous tow shearing is a more experimental technique, which trades increased steering capabilities for severely reduced production speed.

2.2.1 Automated Fibre Placement

Automated Fibre Placement (AFP) is a production technique able to produce advanced high-quality composite structures. It has widespread use in the aerospace industry, utilized to produce structural components for the Boeing 787 and 777X (fuselage, wings resp. i.a.) and the Airbus A350 (upper and lower wing skins, fuselage panels i.a.) [2, 20].

The technique places multiple strips of continuous fibre reinforced tape (tows) via a robotic head. The tows are typically consist of graphite fibres with an epoxy matrix, with dimensions of about 1/8th to 1/4th of an inch wide, and a thickness of 0.005 to 0.030 inch. Up to 32 tows can be placed simultaneously (together called a band), each of them fed by an arrangement of spools of tape. A schematic illustration of a single band is shown in Figure 2.6. The tows can be initialized and terminated individually. The tows are typically placed by locally heating them to make them tacky, which allows the tows to stick to the surface, and are then pressed onto the tool or previous ply with a consolidation device (e.g. a roller or a shoe). This process is repeated band for band until the laminate is completed. A schematic set-up of an AFP installation is shown in Figure 2.7 [20].

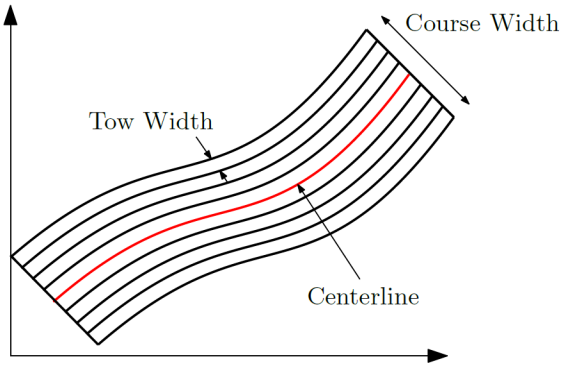


Figure 2.6: Schematic illustration of an AFP course [21]

After fibre placement, a final trimming step is typically needed to remove the excess material at the outer contour of the product to create a smooth finished surface. AFP provides a high consistency and repeatability due to the production being automated, the robot head allows to steer the tows for more design options and the individual initiation and termination of tows reduces scrap.

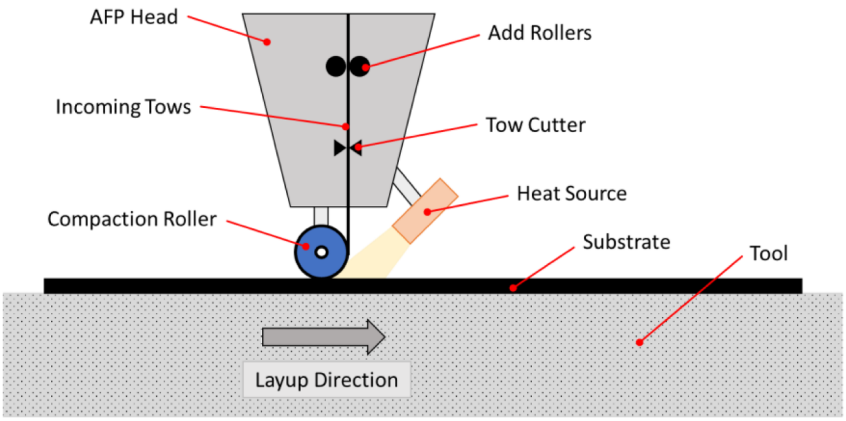


Figure 2.7: Example AFP Setup [20]

There are multiple manufacturing constraints that need to be taken into account when developing production methods for AFP. First of all, there is a curvature constraint to how much a tow can be steered. Depending on the width of the tow, if the curvature is too high the tow will buckle or kink [5], as can be seen in Figure 2.8. The wider the strip of material, the lesser the amount of steering can be applied before tow kinking occurs. Likewise, a higher number of tows per machine path will result in higher curvatures for tows on the outer edges of the course. Table 2.1 from *Blom* (2010) shows some values of the minimum turning radius (the inverse of the curvature) of the course centre line for a 102 mm wide course, but noted that the values are a function of the amount of compaction pressure, the layup speed and the material and therefore may vary [5].

Table 2.1: Typical turning radii for a 102 mm (4 in) course width [5]

Tow width		Typical minimum turning radius	
3.175 mm	(1/8 in)	635 mm	(25 in)
6.35 mm	(1/4 in)	1778 mm	(70 in)
12.7 mm	(1/2 in)	8890 mm	(350 in)

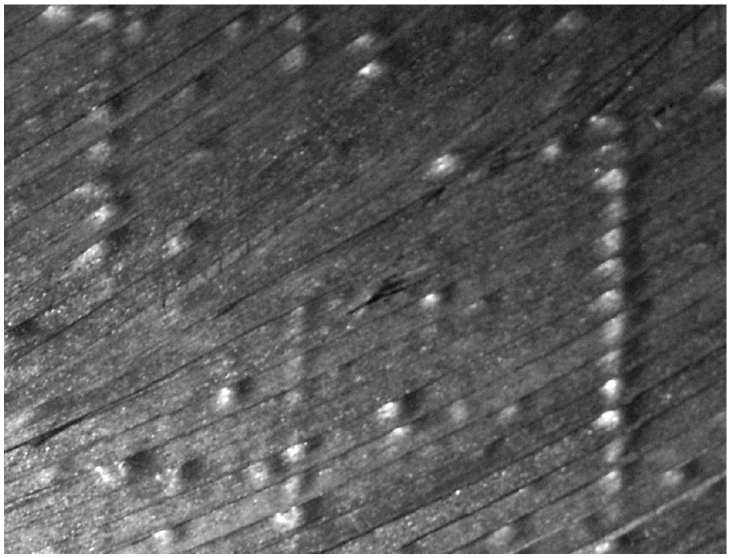


Figure 2.8: Kinking of tows due to excessive tow steering [5]

The second manufacturing constraint is that, due to the nature of how the tows are applied by the robot, which is parallel to the machine path, there are bound to be overlaps and/or gaps within a layer [4, 5]. These will result in thickness variations and/or voids respectively, as illustrated in Figure 2.10. The effect of these kinds of defects and others on the performance of the structure has been the subject of investigation, such as in [22], albeit on test coupons only. However, it can reasonably be assumed that both effects have a negative influence on the final laminate. Thickness variations could pose complications for any bonding of the panel to the rest of the structure, or could disrupt the airflow along the aircraft. Therefore, overlaps are not considered as a design choice for the thesis. Gaps or voids within the laminate are obviously also not considered to be ideal, but are deemed less disruptive than overlaps.

Furthermore, there might be problems with controlling the orientation of individual tows after terminating them. The steering capability comes from pulling the tow in the correct orientation, so once the tow is terminated the consolidation device can only press it onto the surface at whichever orientation the fibre is facing. This may lead to tow straightening, where the final bit of the tow deviates from the intended path to an uncurved state due to the intrinsic stiffness of the tow as seen in Figure 2.9. Tow straightening seems to occur more often on the outer radius of a curved path as compared to the inner radius, but tow straightening has been documented on both sides [5].

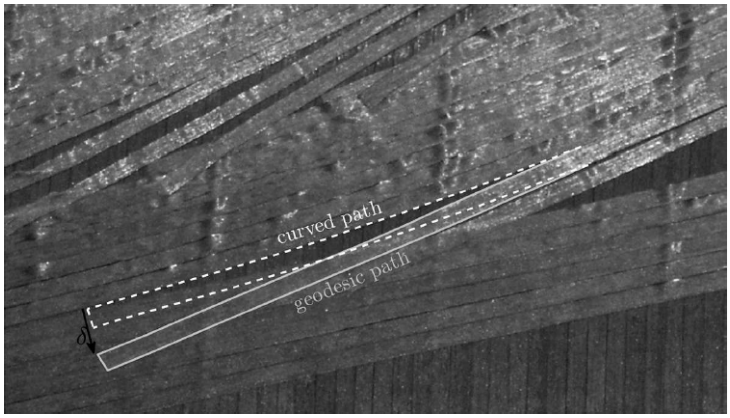


Figure 2.9: Tow Straightening [5]

Lastly, the final manufacturing constraint to be considered in this thesis is the minimum cut length. The minimum cut length is defined as the minimum tow length that needs to be placed before it can be cut again, and it varies depending on the machine configuration. *Blom* (2010) noted lengths between 2.5

and 6 inches, depending on machine configuration [5]. Tows shorter than this length cannot be properly controlled by the AFP machine and would induce defects if placed regardless.

2.2.2 Automated Tape Laying

The second, often-used robot-assisted manufacturing technique for composite structures is Automated tape laying (ATL). It is a similar technique to AFP in that it places composite tape by robots, but it utilizes a single, wider strip of composite tow rather than multiple smaller tows [23]. This makes it suitable for producing large composite parts by robotics with little variation in angle. It is for example utilized on the Boeing 787 Dreamliner, as well as the Airbus A350 XWB [2]. However, the large widths of the tows make the technique unsuitable for steering, as the steering limit is inversely proportional to the width of the tape. Furthermore, any gaps due to tow cuts are also proportional to the tow width, i.e. the gaps will be significantly larger compared to using multiple small tapes with AFP.

2.2.3 Tailored Fibre Placement

Tailored Fibre Placement (TFP) is a manufacturing method where the fibres are placed on a carrier material, which is then stitched in place [24–26]. The stitching allows the fibres to be steered in any direction, which can be used to create VSL. It has for example been used to construct the A350 window frames and the longerons of the NH-90 helicopter [25].

TFP can produce parts of a variety of sizes, but is best used on a smaller scale [24, 25]. Machine speeds have been quoted at around two to ten meters of roving per minute for a single-head machine, depending on the type of fibre used and the geometry of the product. Therefore, production of large panels such as fuselage frames or wind turbine blades would be impractical.

2.2.4 Continuous Tow Shearing

Continuous Tow Shearing (CTS) is a technique which was developed to address some of the shortcomings of fibre steered AFP designs, and is by far the most experimental. It has not been used yet for commercial production, but is still in development. It has mainly been spearheaded by the University of Bristol [27–29]. Instead of placing fixed-size composite tows that are steered parallel to the machine path, CTS shears the fibre bundle such that it covers the intended area. The main advantage of CTS over AFP is that CTS paths can be laid without overlaps and minimal gaps within a layer, as illustrated in Figure 2.10. However, the trade-off is severely reduced production speed compared to conventional AFP or ATL. Furthermore, there can be slight thickness variations between each path due to the amount of shearing required per layer, though it is much less significant than the inevitable gaps with AFP.

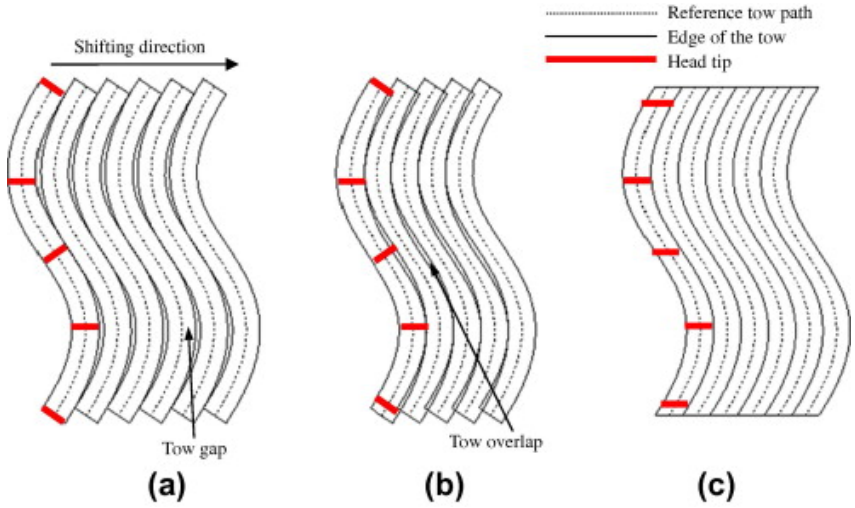


Figure 2.10: Illustration of shifting or shearing the fibre path. (a) and (b) illustrate the gaps and/or overlaps that occur when shifting with the AFP method. (c) displays the results for the same reference path with the CTS method, where the tows are sheared [28].

2.2.5 Comparison of the Various Manufacturing Techniques

Now that the different manufacturing techniques have been reviewed, they can be placed within the context of the thesis. This section will evaluate the manufacturing techniques for the design of VSL. First of all, ATL can be quickly disregarded due to the restrictive steering limit of the large width of the tape, as it allows for very little steering [5]. For the remaining three techniques, the comparison comes down to production speed versus defects. CTS has the capability of placing bands of tape with little to no voids or gaps, but has only a very limited production speed [28, 29]. On the other hand, AFP will inherently have gaps in within layers and has a limit to how much the tows can be steered (albeit less stringent compared to ATL [5]), but reaches a potential production speed of a meter per second [2, 20]. TFP exists somewhere in between the two, with slightly higher production speeds than CTS and fewer defects than AFP. Considering the large size of aircraft panels, alongside the number of layers per panel, the production speeds of CTS or TFP seem too restrictive for manufacturing large parts. Therefore, AFP has been chosen as the intended production technique for the thesis.

2.3 Three-Step Approach for VSL Design of the TU Delft

The aim of this section is provide an overview of the development of Three-Step Approach for VSL by the TU Delft, why it was needed and what still has to be done. It is the goal of the thesis to provide solutions to further the development of this approach. Furthermore, this section also reviews the latest advances on the topic of generating fibre paths for VSL designed by the Three-Step Approach and what can be added to this program. Finally, at the end of this section the objectives of the thesis with regards to the research question are outlined, which will conclude this background chapter.

2.3.1 Three-Step Approach

The design of VSL is by no means a trivial task. The introduction of conventional composite structures opened up a much larger design space compared to metals, such as the stacking sequence and layup of the material. VSL takes this a step further by attempting to optimize every point within a layer as well, through the use of the steering of fibres. Besides the manufacturing issues already illustrated in subsection 2.2.1, there is also the question of balancing the desire to adhere to the theoretical optimal design with its manufacturability, and the subsequent performance loss due to the compromises that have to be made.

To tackle these issues and due to the non-existence of necessary software tools, the TU Delft prompted the development of the so-called “three-step approach” for the design of VSL, schematically illustrated in

Figure 2.11 [6]. The intention is to provide a full, partially automated solution to generate manufacturable VSL for a given design load.

The first step consists of obtaining the optimum laminate stiffness distribution for a given design in terms of lamination parameters [6, 11]. Multiple design drivers can be included in this step, such as maximum in-plane stiffness, strength, natural frequency and buckling. The lamination parameters are converted to fibre angle distributions in the second step [7, 30, 31]. This step can take into account the minimum curvature and the thickness build-up. Finally, the discretized fibre angles need to be converted into manufacturable machine paths from which the fibre tows are laid [5, 9].

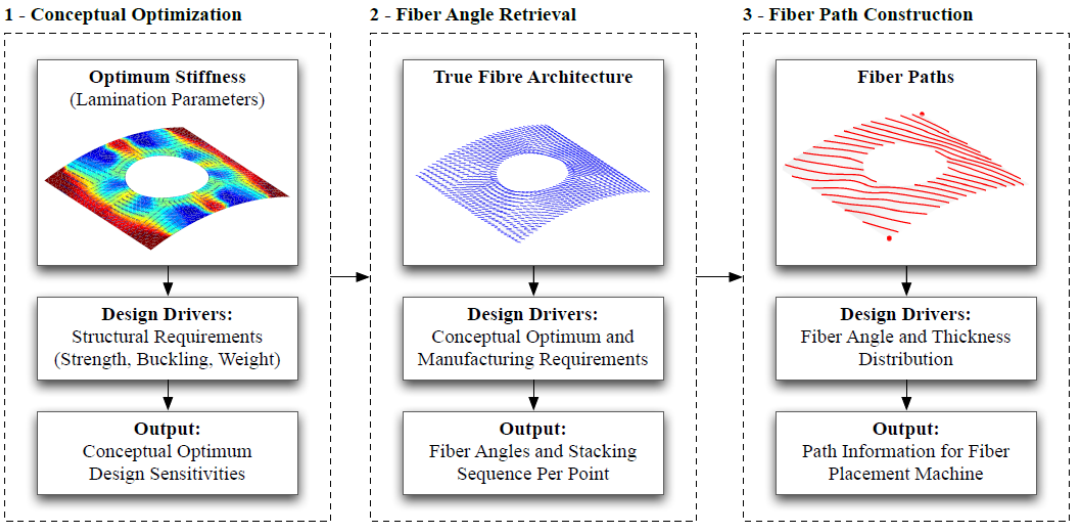


Figure 2.11: Schematic overview of the 3-step approach for VSL [6]

2.3.2 Streamline Algorithm

Within the Three-Step Approach, a special mention should be made of the streamline algorithm for its importance for the thesis. It converts the fibre angles from the second step into the centrelines of a course, which are then used as input for the third step, which the upcoming thesis seeks to expand upon and optimize. Furthermore, some understanding of algorithm is required to generate test cases for the various parts of the thesis with the algorithm.

For the construction of the course centrelines, a streamline analogy is used, as described by *Blom et al.* (2010), *Blom* (2010) or *IJsselmuiden* (2011) [5, 6, 32]. An optimization is performed to "minimize the maximum thickness", i.e. minimize the number of courses that overlap each other. This correlates to how much cutting of tows is required to prevent overlaps and how efficient the placed paths are in terms of area placed per machine pass.

One of the boundary conditions that is applied by the algorithm is the placement of the inflow and outflow nodes, as illustrated in Figure 2.12. This means that no paths are terminated prematurely by the algorithm. Consequently, this means that higher rates of fibre steering will lead to more inefficient paths, as the streamline algorithm will have to accommodate for the higher curvature. The effect is illustrated in Figure 2.13. Note how there are many more paths in Figure 2.13 (a) compared to (b), due to the higher steering. A possible solution would be to perhaps terminate some of the paths prematurely, or alternate "full" paths and "filler" paths, but this is mostly outside the scope of the thesis, as this would require changes to the streamline algorithm.

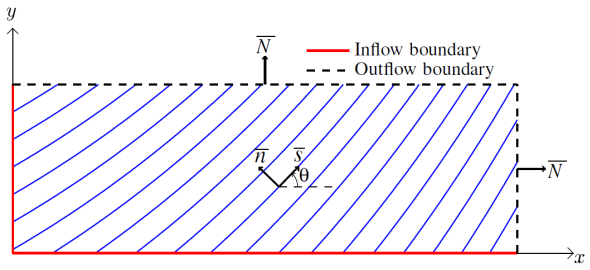


Figure 2.12: Inflow and outflow boundaries for the streamline algorithm [32]

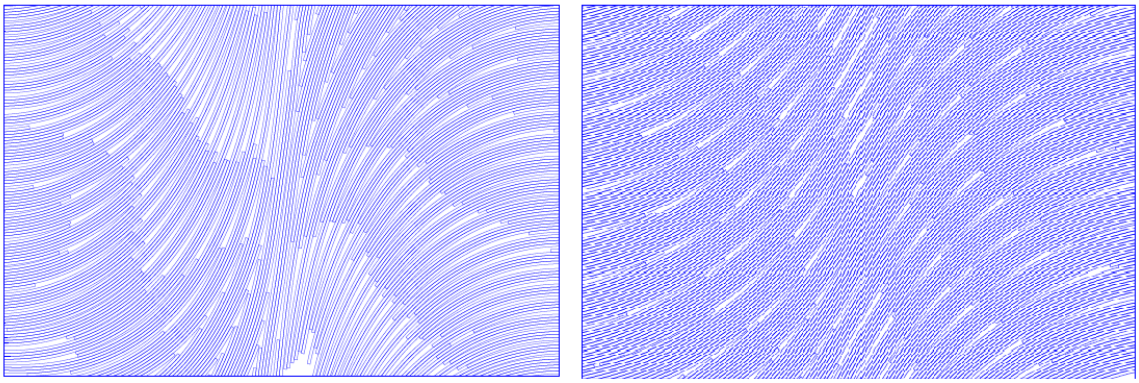


Figure 2.13: Comparison of steering effect on machine path efficiency in linearly-varying fibre angle (LVFA) layers. The notation prescribes the fibre angle at the longitudinal edges and the centre of the panel. (a) $< 0, 85.94, 0 >_x$ (b) $< 10, 62.36, 10 >_x$ [9]

2.3.3 State of the Art in the Design of Fibre Paths for VSL

The streamline algorithm in subsection 2.3.2 was the input for most recent MSc Thesis on the topic of developing machine paths for Automated Fibre Placement, made by K. Wurpel [9]. What was new about his work was that he developed a method to generate machine paths for flat 2D fibre steered composites by using the streamline algorithm. The streamlines were divided between alternating machine paths and boundary lines (tows were cut once they hit a boundary line) to minimize the angle mismatch. The thesis could be considered a first step in the right direction of the finalization of the program, as he explored some of the design options. However, there is still room to improve upon his work by making the approach compatible to a wider range of (curved) surfaces and by improving upon the cutting algorithm.

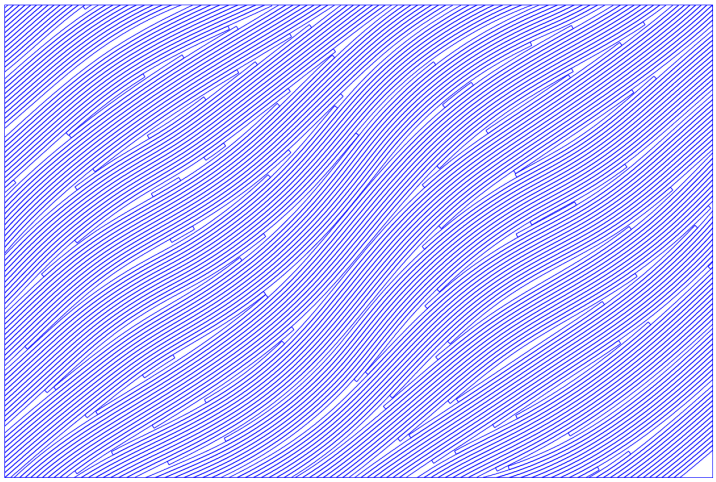


Figure 2.14: A steered layer from *Wurpel* (2015)[9]

2.3.4 Limitations of the Current Fibre Path Design for VSL

This section will name the possible improvements of the fibre path generation step of the Three-Step Approach. Each of these improvements will then be main topic of the following chapters.

First of all, the current method as mentioned in subsection 2.3.3 only supports the generation of fibre paths for 2D layers and some fringe cases in 3D. The design method could be sufficient for producing a flat test panel, but it will not be able to create designs for typical aerospace panels, as they are often doubly-curved. With the intention of the Three-Step Approach in mind, it would be much more preferable to have a robust algorithm that can handle any type of curved surface with the same approach. To this end, it would be better to use a parametrization of the surface, so that the subsequent operations such as finding intersections or offsetting curves can have the same approach, regardless of the geometry of the panel. This will be looked into further in chapter 3.

Furthermore, as can be seen in Figure 2.13 and Figure 2.14, there are still a lot of unnecessary gaps within a layer in the approach by *Wurpel* (2015). This is caused by the use of boundary lines, which are borders between two machines passes where the program automatically cuts the tow. Ideally, tows should only be cut when they would otherwise intersect with another tow, to minimize the number of gaps. This can only be achieved if the algorithm is improved upon. However, since the framework of the model will be different due to the parametrization, the fibre path generation algorithm has to be rebuilt from the ground up as well. The construction of the new algorithm has been documented in chapter 4.

Moreover, more thought can be put into cutting the tows in a more effective way. As illustrated by *Blom* (2010) and Figure 2.9, there could be benefits for adapting a certain cutting strategy, such as cutting on the inside of a curve to reduce the occurrences of tow straightening [5]. Some of the effects have already been researched by *Mishra* (2017) using finite element models from a structural standpoint [21], but it could be beneficial if the new program would have the option for different strategies for cutting. This could accelerate the research on physical test samples to evaluate the variable stiffness concept in general. The cutting strategies will be covered in section 4.8.

Finally, it would be beneficial if the program would be able to evaluate the generated fibre layers based on a number of variables. For example, it can be hard to determine from sight what the gap percentage of a layer would be, which would affect the performance of the panel. Likewise, there will always be a mismatch between the optimal fibre angle distribution from the Three-Step Approach, and what is physically possible to produce with the robots. As composite panels derive their properties in part from the fibre orientation, this would also have an effect on performance. An overview of these fitness criteria will be given in chapter 5.

Chapter 3

Computational Modelling of Variable Stiffness Laminates

This chapter will provide an overview of useful computer modelling techniques, as a large part of the thesis will revolve around generating machine paths, generating offset paths and cutting said paths to create as close of an representation of the fibre placed layer to the theoretical projections.

When selecting a computer modelling technique, it is important to keep a number of factors in mind. First of all, it is not the intention to develop new modelling techniques, but rather to utilize and adapt existing ones for the needs of this thesis. Therefore, it is paramount that the technique is well-documented by several sources. This ties into a second requirement, namely that the technique has widespread use in industry, as it will make the transition from computer model to physical design faster. Furthermore, the speed of the technique, the required storage space and the ease of use can all play a role in the selection.

To this end, there are three main techniques that have widespread use in the 3D computer aided modelling (CAM) industry today which could be used for the algorithm. By far the most popular method today is the aforementioned NURBS technique [33], due to its generality, properties and incorporation in international file formats IGES and STEP. However, B-splines (NURBS are a generalization of B-splines, or in more layman terms, B-splines are NURBS with all their weights set to one) and Bézier curves have their place as well. All three techniques have their origin in the second half the 20th century in either the automotive or the aerospace industry, as there was a need to compute curved 3D surfaces with the aid of computer numerically controlled manufacturing [34].

3.1 Bézier Curves

Bezíer curves are the first type of parametric curve to be considered, which are used in computer graphics. They have their origin in the late 1950s, when car manufacturers sought ways to incorporate Computer Aided Design (CAD) into their design process [34]. Two engineers at different French car companies (De Casteljau at Citroën and Bézier at Rénault respectively) utilized Bernstein polynomials to represent curves in a new way. The novelty lay in defining a curve by using control points which "pulled" the curve towards them, rather than using points that lay on the curve. The polygon created by the control points enclosed the curve (see Figure 3.1), and the alteration and design of such curves were quite intuitive. Bézier curves also have a few more characterizing properties which are of importance for the selection of a suitable technique. First of all, a Beziér curve always goes through first and final control point, i.e. the curve interpolates to the first and final control point, and approximates the control points in between. Furthermore, any change to the curve by altering any of the control points has a global effect on the shape of the curve.

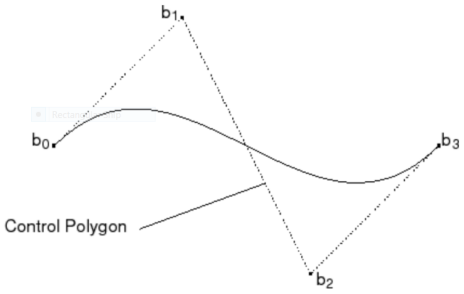


Figure 3.1: Example of a four control point Bézier curve [33]. b_i are the control points, the dashed lines encompass the control polygon and the curved line is the actual Bézier curve.

The Bézier curves have two main disadvantages when it comes to manipulation of the curve's properties [33]. First of all, the number of control points is directly related to the degree of the curve. To increase the complexity of the curve, the degree of the curve has to be increased as well. Higher degree curves are more inefficient to process and often numerically unstable [35]. The second disadvantage is related to the ability to make changes to a curve. For Bézier curves, every control point affects each point on the curve. Therefore, changing one control point changes the entire curve as well. This can be inconvenient or impractical for some of the applications the design algorithm has to be capable of, e.g. cutting/removing parts of the curve. There is little flexibility in positioning the control points while maintaining the desired continuity degree of the curve [35]. These issues are both addressed by the B-splines and NURBS techniques, which is why they are preferred for this thesis.

3.2 B-Splines and NURBS

As mentioned before, B-splines and NURBS are very closely related to each other, and it is therefore appropriate to discuss them in conjunction with one another. Naturally, B-splines pre-date NURBS as NURBS are a generalization of B-splines. The first proposal of the use of B-splines was made by Gordon and Riesenfeld in the early 1970s [35], but the fundamentals originate as early as the post-World War two era by Schoenberg, followed up by the development of essential algorithms by Cox and de Boor [33, 36].

They differ from Bézier curves in the sense that they are piecewise polynomials, i.e. the basis functions and control points only affect a part of the curve, as opposed to the whole curve. This allows for lower degree curves while maintaining a sufficient order of continuity, which are generally (more) numerically stable and not as computationally expensive as Bézier curves. Furthermore, it is more convenient to make alterations to a curve since only a part of the control polygon controls its shape. Therefore, B-splines or NURBS are the preferred method of choice for the parametric representation for this thesis. The properties of NURBS, with B-Splines as a special case of NURBS will be explored in the next sections.

3.2.1 NURBS Terminology

As there are many features that make up a NURBS curve, their attributes will be explained on a surface level as a way to acquaint the reader in this section. More thorough explanations on the topic of NURBS can be found in [33, 35–37].

A NURBS curve C, as defined in [35], is given by Equation (3.1):

$$C(t) = \frac{\sum_{i=0}^{n} N_{i,p} w_i P_i}{\sum_{i=0}^{n} N_{i,p} w_i} \tag{3.1}$$

Where $\{P_i\}$ are the control points, $\{w_i\}$ are the weights and $\{N_{i,p}(t)\}$ are the B-spline basis functions, with degree p on the knot vector U. The weighted control points, the degree and the knot vector are the basic building blocks for any NURBS curve.

Control Points

The *control points* and the *weights* are the main defining factors for the shape of the curve. Depending on the weights (and to a lesser extent, the knot multiplicity and knot vector spacing, which will be discussed later), a curve will move more or less towards a given control point, as can be seen in Figure 3.2. Moreover, control points only affect an interval where the points are active, based on the order of the curve. A NURBS curve is always contained within the convex hull of the control points. Also, unlike Bézier curves, a NURBS curve does not have to go through the first and last control point.

Weights

The weights of a NURBS curve, or maybe more correctly, the weights distribution of a NURBS curve, give the designer one extra layer of options to tweak the design. A higher relative weight value will make a curve move more towards a specific control point, as shown in Figure 3.2. If all the weights are set to be one, then the NURBS curve is effectively a *B-spline*, i.e. a simpler form of parametric curve. It should be noted that altering weights is not an absolute necessity, and one should always consider creating their curves/surfaces without weights unless there is a reason not to. For example, some geometrical shapes such as circles can only be defined exactly if a weights distribution are used.

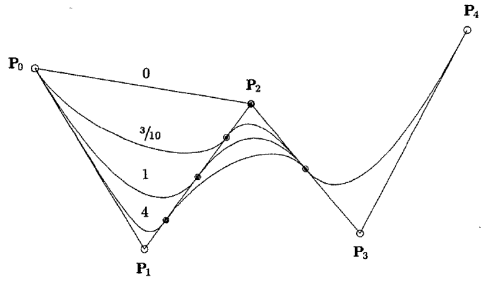


Figure 3.2: Increasing the weights for the control point P_1 moves the curve more towards P_1 [35]. It can be noted how the curves intersect the control polygon between P_1 and P_2 at different locations due to the varying weights for P_1, but coincide with each other again after they crossed the polygon between P_2 and P_3 due to the piecewise nature of NURBS.

Degree

The *degree* p is positive whole number which defines the type of polynomial used to interpolate between points. A degree of one creates linear curves, a degree of two has quadratic curves and so on. Higher degree polynomials result in more continuous functions. For most applications, cubic curves are used for their curvature continuity. The *order* m of a curve is equal to the degree plus one, which defines how many of the control points affect any given point on the curve. Therefore, the order is also the minimum number of control points required for any curve.

Basis Functions

The *basis functions* are the "weighted, piecewise polynomial functions which form a basis for the vector space of all the piecewise polynomial functions of the desired degree and continuity" [35–37]. In practice, most of the properties of NURBS act so through the basis functions. However, to properly explain the basis functions, it is more convenient to first cover the parts that make up the basis functions, as they have many intricacies.

Knot Vector

The *knot vector* is an ordered, non-decreasing list of parameter values (called knots), which determines where and by how much the control points affect the curve. The length of the knot vector is always equal to the sum of the number of control points and the order of the curve. The knot vector divides the parametric space into intervals/lengths called *knot spans*, which determine which control points have

an influence on a region of the curve. As one travels from one knot span to another, new control points become active and the oldest control points get discarded, hence why a knot vector has to be non-decreasing. On any knot span, the maximum number of control points that can affect the curve is equal to the order of the curve. Similarly, the number of knot spans then has to be equal to the number of control points minus the order plus one. A few examples of knot vectors are shown in Figure 3.3.

$$U_1 = [0\ 0\ 0\ 0\ 1\ 2\ 3\ 4\ 5\ 5\ 5\ 5] \quad (a)$$
$$U_2 = [0\ 0\ 0\ 0\ 3\ 2\ 1\ 4\ 5\ 5\ 5\ 5] \quad (b)$$
$$U_3 = [0\ 0\ 0\ 0\ 2\ 4\ 6\ 8\ 10\ 10\ 10\ 10] \quad (c)$$

Figure 3.3: (a) An example (clamped) knot vector. (b) An incorrect knot vector, since it is not non-decreasing. (c) An equivalent knot vector to (a), as only the ratio between the individual knots matter for the shape of the curve.

Knot Multiplicity

It is possible for consecutive knots to have the same value. The number of repetitions of the same knot value is referred to by the *knot multiplicity*. Increasing the knot multiplicity decreases the continuity of the curve at that point. At a knot of multiplicity k, the curve will be C^{p-k} continuous. If a location on the curve is C^{-1} continuous, then that location is considered "clamped", as the knot span then has a width of zero and the curve coincides with the respective control point. Therefore, the knot multiplicity can never be higher than the order of the curve. The knot vectors shown in Figure 3.3 all have a multiplicity of four at their endpoints, which would clamp the knots at the first and final control point if the order of the curve would be four as well.

Increasing the knot multiplicity through knot insertions is a common way to split curves into segments by effectively creating a new clamping point within the curve [33, 35]. This does not alter the shape of the curve itself, but only the knots and control points. Because NURBS are piecewise polynomials, the two segments are independent of each other and can be removed or split if desired.

Knot Vector Types

There are various denominations for knot vectors. An *open knot vector* is a knot vector where the initial and final point on the NURBS curve do not coincide with the initial and final control points. A *closed/clamped knot vector* on the other hand has the first and final point on the curve coincide with the first and final control point respectively. This is achieved by having a knot with a multiplicity of the order at the beginning and end of the vector. Subsequently, the knot span will have zero length there. A *closed knot vector* can be achieved by repeating some of the control points and knots. This creates closed contours. The various types of curves due to the knot vector can be seen in Figure 3.4.

Furthermore, a *uniform knot vector* is defined by *Piegl and Tiller* (1997) as a vector where the internal knots are equally spaced, such as the knot vectors (a) and (c) in Figure 3.3, whereas a *non-uniform knot vector* can have different spacings or more than one repetition.

For the purposes of the upcoming thesis, clamped knot vectors are deemed the most appropriate. The ability to select the endpoints of a curve through the control points is practical and clamped knot vectors are the most common for CAD. Also, since the parametric range of the knot vector is only defined in the centre part of the knot vector (with a length of the number of control points minus the order), effectively the outer knots are ignored for the evaluation, whether the knot vector is clamped or not.

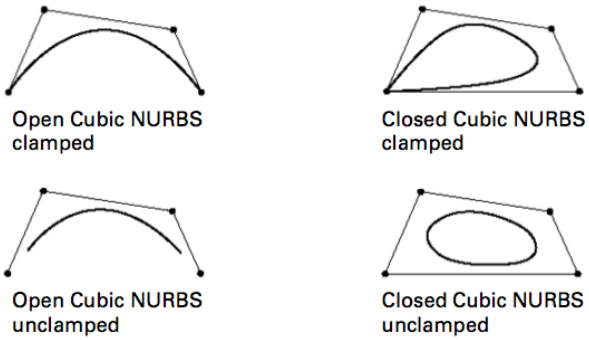


Figure 3.4: Different knot vectors can lead to distinctly different curves for the same set of control points [38].

Knot Vector Spacing

The selection of a knot vector can be an unintuitive process. Particularly, the selection of the spread of the knots has not yet been discussed. Unfortunately, there is no unified answer to which spread is ideal, although the "spacing" between knots does have a noticeable effect on the resulting curve, as can be seen in Figure 3.5. Here, the degree, weights and control points are kept constant, but the spread of the knots is varied, which leads to three distinctly different curves. In *Piegl and Tiller* (1997), several methods for the selection of a suitable knot vector are explained. Generally, it is desirable that the knots are evenly spaced, to minimize the length of each span and each segment that a single control point influences.

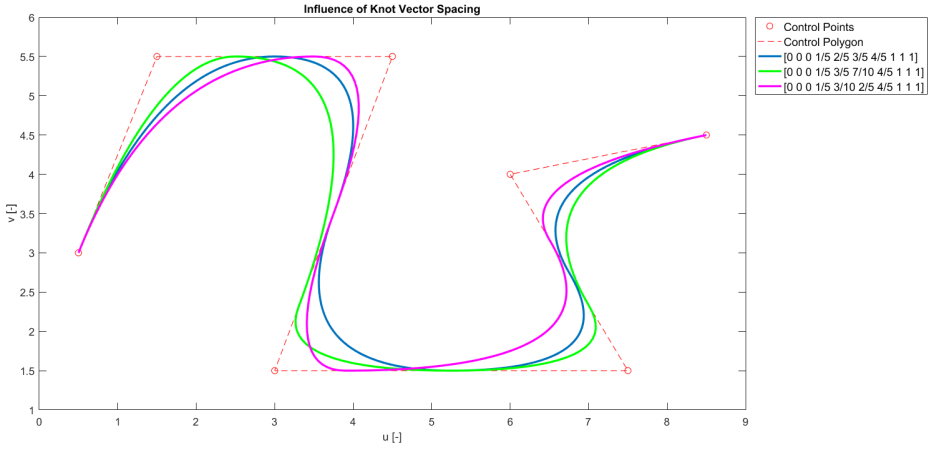


Figure 3.5: Effect of knot vector spacing on the resulting NURBS curve. By changing the spacing, the curve was altered while the control points remained the same.

Computation of Basis Functions

Basis functions are typically computed recursively with the "Cox-de Boor recursion formula" [36, 39, 40]. The i^{th} basis function $N_{i,p}(t)$ with degree p and knots t is defined as:

$$N_{i,0}(t) = \begin{cases} 1, & \text{if } t_i \leq t < t_{i+1} \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

$$N_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t) \tag{3.3}$$

The Equations (3.2) and (3.3) show two things. Firstly, a 0^{th} degree basis function is equal to the step function on the corresponding knot span. Secondly, to compute a basis function of degree p, two basis functions of degree $p-1$ at that interval are required. When working with higher degree curves and many control points, this can be quite an arduous process to do by hand. Fortunately, these formulas lend themselves well enough for computer algorithms. In *Piegl and Tiller* (1997), many fundamental algorithms are already written out for ease of use [35]. Moreover, it also has several examples of Bézier/B-spline/NURBS curves to be used to gain some experience with them. An illustration of the shape of basis functions of varying degrees is shown in Figure 3.6.

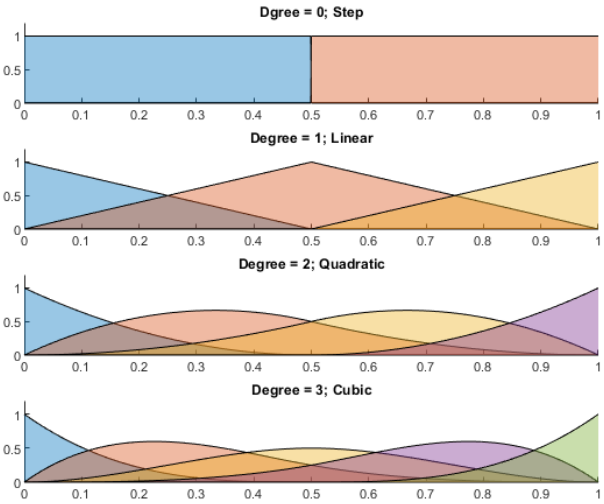


Figure 3.6: (Clamped) basis functions for various degrees with a single knot at the halfway point.

Homogeneous Coordinates

A common way to simplify rational curves such as NURBS is to use *homogeneous coordinates* for more efficient processing [35]. A rational NURBS curve in n-dimensional space is represented as a polynomial curve in $(n+1)$-dimensional space by including the weights in the control points. A given control point $P_i = (x_i, y_i, z_i)$ in three-dimensional space becomes $P_i^w = (w_i x_i, w_i y_i, w_i z_i, w_i)$ in four-dimensional space. Equation 3.1 can then be rewritten into Equation 3.4.

$$C(t) = \frac{C^w(t)}{W(t)} = \frac{\sum_{i=0}^{n} N_{i,p} P_i^w}{W(t)}; \quad W(t) = \sum_{i=0}^{n} N_{i,p} w_i \tag{3.4}$$

NURBS Derivatives

NURBS derivatives are be an essential tool for performing many algorithms that will be used in this thesis, such as offsetting and intersecting. There are several ways to evaluate NURBS derivatives, but this thesis will use the method which continues from the homogeneous coordinate system, where the curve derivatives of $C(t)$ will be expressed in terms of the derivatives of $C^{(w)}(t)$.

Consider the numerator of Equation 3.4. The k^{th} order derivative of $C^w(t)$ is given by Equation 3.5, where the k^{th} order derivative of the basis function $N_{i,p}^{(k)}(t)$ is given by Equation 3.6 [35].

$$C^{(k)w}(t) = \sum_{i=0}^{n} N_{i,p}^{(k)} P_i^w \tag{3.5}$$

$$N_{i,p}^{(k)}(t) = \frac{p}{p-k}\left(\frac{t-t_i}{t_{i+p}-t_i} N_{i,p-1}^{(k)}(t) + \frac{t_{i+p+1}-t}{t_{i+p+1}-t_{i+1}} N_{i+1,p-1}^{(k)}(t)\right) \tag{3.6}$$

Furthermore, note that the coordinate components of $C^w(t)$ can be related to $C(t)$ by way of Equation 3.7. Here, $A(t)$ has the coordinates from $C^w(t)$, while excluding $W(t)$.

$$C(t) = \frac{W(t)C(t)}{W(t)} = \frac{A(t)}{W(t)} \tag{3.7}$$

Then, the general Leibniz rule can be used to write the k^{th} order derivative of $A(t)$ as Equation 3.8, where the values for $A^{(k)}(t)$ are obtained from Equation 3.5[41]. Rearranging the terms gives the final expression for the k^{th} order derivative of $C(t)$ in Equation 3.9.

$$\begin{aligned} A^{(k)}(t) &= (W(t)C(t))^{(k)} \\ &= \sum_{i=0}^{k} \binom{k}{i} W^{(i)}(t)C^{(k-i)}(t) \\ &= W(t)C^{(k)}(t) + \sum_{i=1}^{k} \binom{k}{i} W^{(i)}(t)C^{(k-i)}(t) \end{aligned} \tag{3.8}$$

$$C^{(k)}(t) = \frac{A^{(k)}(t) - \sum_{i=1}^{k} \binom{k}{i} W^{(i)}(t)C^{(k-i)}(t)}{W(t)} \tag{3.9}$$

Knot Insertion

Knot insertion is the process of adding a knot to a knot vector without altering the shape of the curve [35]. While it has many uses, the main reason of interest for this thesis is the ability to subdivide or cut curves through multiple knot insertions. Whenever a knot is inserted, a number of control points are altered as well. Due to the local support property of NURBS, only $p-1$ control points are affected, which are replaced by p new control points. Suppose a knot $t \in [t_i, t_{i+1})$ is inserted, then the new set of $(n+1)$ control points Q is given by the old n control points P in the form of Equation 3.10.

$$Q_a = \begin{cases} P_a, & a \leq i-p \\ (1 - \frac{t-t_a}{t_{a+p}-t_a})P_{a-1} + \frac{t-t_a}{t_{a+p}-t_a}P_a, & i-p \leq a \leq i \\ P_{a-1}, & \text{otherwise} \end{cases} \tag{3.10}$$

3.2.2 NURBS Surfaces

NURBS surfaces are computed in much of the same way as NURBS curves are. It is obtained as a tensor product between two NURBS curves. Two independent knot vectors, typically denoted by U and V are used. To compute a point on the curve, one must find the knot spans for each variable, then compute the basis functions in the same way as before, and then simply multiply them with the control points. The full equation to compute a point on the surface then becomes as shown in Equation (3.11). Note that the basis functions only depend on their respective independent variables, and therefore can be manipulated independently as well.

$$S(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)w_{i,j}P_{i,j}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)w_{i,j}} \tag{3.11}$$

A NURBS surface, as illustrated in Figure 3.7, holds the same properties as a NURBS curve would with the same knot vector with regards to continuity and local modification. Furthermore, knot vectors U and V do not have to have the same degree. A prime example would be a NURBS curve profile with an arbitrary degree, which is extruded (degree in the direction of the extrusion is equal to one). Again, *Piegl and Tiller* (1997) has a number of examples available for study [35].

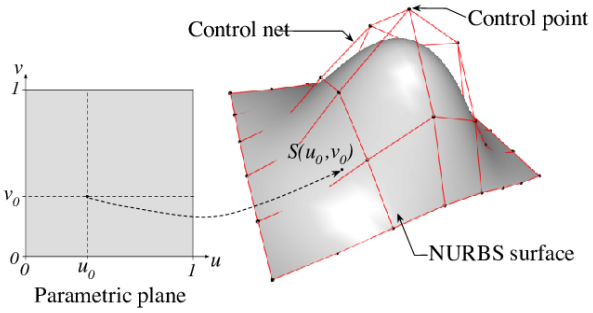


Figure 3.7: A NURBS surface in parametric and Cartesian coordinates [42].

NURBS Surface Derivatives

NURBS surface derivatives can be derived in the same fashion as NURBS curve derivatives, but the process is a lot more laborious due to the extra variable. The first step is expressing the NURBS surface in homogeneous coordinates as shown in Equation 3.12, with $P^w_{i,j} = (w_{i,j}x_{i,j}, w_{i,j}y_{i,j}, w_{i,j}z_{i,j}, w_{i,j})$.

$$S^w(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)P^w_{i,j} \tag{3.12}$$

Derivatives in terms of $S^w(u,v)$ can be expressed with the same basis function derivatives from Equation 3.6:

$$\frac{\partial^{k+l}}{\partial^k u \partial^l v} S^w(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} N^{(k)}_{i,p}(u)N^{(l)}_{j,q}(v)P^w_{i,j} \tag{3.13}$$

Similar to Equation 3.7, the coordinate values of $S^w(u,v)$ are grouped in a variable $A(u,v)$ in Equation 3.14, which is then used to apply the general Leibniz rule to in Equation 3.15.

$$S(u,v) = \frac{W(u,v)S(u,v)}{W(u,v)} = \frac{A(u,v)}{W(u,v)} \tag{3.14}$$

$$A^{(k,l)} = \left((W(u,v)S(u,v))^k\right)^l = \sum_{i=0}^{k}\binom{k}{i}\sum_{j=0}^{l}\binom{l}{j}W^{(i,j)}S^{(k-i,l-j)} \tag{3.15}$$

Finally, the general expression for a $(k,l)^{th}$ surface derivative is given in Equation 3.16. If desired, a more complete derivation can be found in *Piegl and Tiller* (1997).

$$S^{(k,l)} = \frac{1}{W}\left(A^{(k,l)} - \sum_{i=1}^{k}\binom{k}{i}W^{(i,0)}S^{(k-i,l)} - \sum_{j=1}^{l}\binom{l}{j}W^{(0,j)}S^{(k,l-j)} - \sum_{i=1}^{k}\binom{k}{i}\sum_{j=1}^{l}\binom{l}{j}W^{(i,j)}S^{(k-i,l-j)}\right) \tag{3.16}$$

3.2.3 Selection of Computer Modelling Technique

For this thesis, the choice between B-splines and NURBS is one of the right tool for the job. NURBS are able to handle more types of shapes such as cylinders, circular arcs and other bodies of revolution and has more design options due to the inclusion of weights. However, the selection of these weights outside of known cases (such as cylinders) can be arbitrary and difficult. Therefore, the surfaces and curves will be in a B-spline format, unless stated otherwise due to the choice of curve or surface.

Chapter 4

Construction of the New Fibre Path Generation Algorithm

In the previous three chapters, the outline of the new fibre path algorithm has already been somewhat formulated by a set of requirements from the research question, and a set of design choices in terms of manufacturing technique and computer modelling framework. At this point, it is useful to reiterate and specify what the goals will be of the algorithm, which are as follows:

- be able to design for moderately doubly-curved open shells.
- be able to create steered fibre paths.
- be designed for Automated Fibre Placement (AFP), with a variable input for the number of tows and the tow width
- be designed in a B-spline or NURBS framework
- be able to address or identify (some of the) potential manufacturing issues

Similarly, the limitations of the algorithm should be mentioned. In particular, it should be noted that some causes of manufacturing defects will not be considered for this thesis. First of all, the algorithm will work under the assumption that the mould and AFP roller are compatible for a given path and number of tows and the interaction between the mould surface and the AFP roller will not be considered. Secondly, it is assumed that mould surface does not change due to the thickness build-up of placed tows. For thin-walled structures, this could still be a conservative assumption, but it would not hold up for thicker structural components.

The chapter is divided in several sections, some of which have an overarching theme. A schematic overview of the flow of the chapter is provided in Figure 4.1. First of all, the conventions used during the thesis are provided in section 4.1, such as the description of surfaces and curves and concepts of differential geometry.

Secondly, course centre line (re)construction is covered by sections 4.2 to 4.4. As the path from the streamline code is in Cartesian coordinates, the points first need to be projected onto the parametric surface. Then, the path is reconstructed in B-spline format through a curve fitting algorithm. Also, some observations are made on the streamline code and its fibre path design.

Tow generation is discussed in sections 4.5 and 4.6. The course centre lines are elongated beyond the panel edges to ensure the full course reaches every border. Then, offsets in the binormal direction of the curve represent the outer edges of individual fibres.

Tow cutting is covered by sections 4.7 and 4.8. Intersections are found between paths with a curve-curve intersection algorithm, then tows are cut according to a cutting strategy and panel edges. A small subset of designs is presented that limits or eliminates tow straightening, although it remains a difficult problem in the automation of fibre-steered designs.

Finally, two sections are dedicated to addressing some of the manufacturing issues that may arise with AFP. In section 4.9, the minimum cut length requirement of AFP tows is addressed, and in section 4.10, the method for checking the curvature of the tows is described.

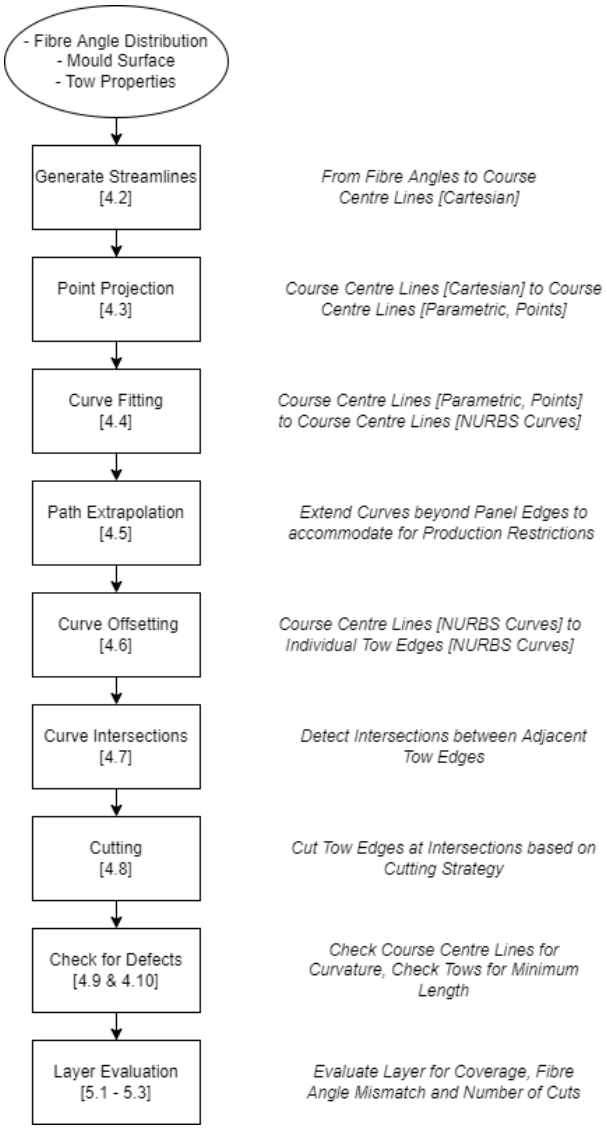


Figure 4.1: Schematic overview of the steps in the creation and evaluation of a layer.

4.1 B-Spline Description of the Mould and Paths

In this section, the notations and conventions of the surface and curves are described as they will be used throughout the thesis. Furthermore, this section also introduces two concepts of differential geometry which will be referred to in this chapter, as they play a vital role in bridging the gap between the parametric B-spline surface and the 3D Euclidean surface.

Figure 4.2 shows a schematic representation of the mould surface. The mould surface $S(u, v)$ is defined by clamped knot vectors U and V and the control points matrix P, and is a function of surface parameters u and v. The outer edges of the mould surface coincide with the outer control points due to the clamped knot vectors.

In a similar fashion, curves on the surface are denoted by $C(t)$, with clamped knot vector T and control point vector P and is a function of surface parameter t. Curve control points are defined in surface parameters u and v, such that the control points and the curve by definition lie on the mould

surface. Points on the curve can then be returned to Cartesian coordinates as any other point on the surface.

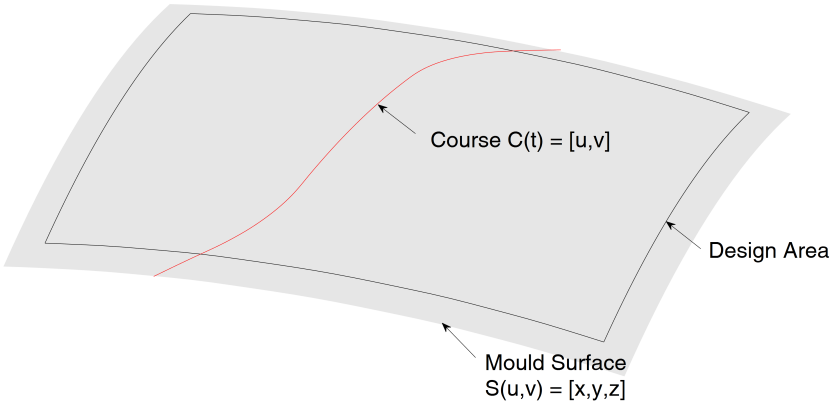


Figure 4.2: Schematic of the mould, the design area and a curve

Moulds can either be constructed manually (recommended only for flat plates or known geometrical shapes such as cylinders) by creating a mesh of control points and two knot vectors, or imported from a CAD data exchange format such as STEP if it is curved. It should be noted though that the CAD software may choose not to use B-splines / NURBS to describe a surface. During the thesis, the most consistent results were achieved by using the Generative Shape Design toolbox from CATIA, with the surface being defined by a number of multi-section surfaces [43].

Regarding the mould, a distinction must be made between the mould surface and the "design surface", as illustrated in Figure 4.2. The design surface is defined as the outer dimensions of the fibre steered panel, which should always be smaller than the mould surface. This is because the AFP machine requires a part of the surface to be free for several production reasons, such as the minimum cut length of fibres and the contact between the AFP rollers and the mould.

Although the surface is defined in parametric form, it will still need to refer back to properties and relations in Euclidean space, such as lengths, areas, angles and curvatures, e.g. "What's the length and angle of this tow on the surface?". Fortunately, such relationships between the two spaces have already been well established, such as in the first fundamental form of differential geometry [44].

The first fundamental form is defined as the inner product between two tangent vectors of a surface in $\mathbb{R}^3$, and can be used to calculate metric properties of a surface consistent with the two spaces. In particular, this will be useful for the calculations of lengths of a curve along the surface, such as in sections 4.4 to 4.6. It is written in many different ways, but Equation (4.1) shows the metric tensor notation.

$$g = \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix} = \begin{bmatrix} S_u^2 & S_u S_v \\ S_u S_v & S_v^2 \end{bmatrix} \tag{4.1}$$

In a similar light, to help define the properties of a point on one of the parametrized (B-spline) curves, irrespective of its motion, the Frenet-Serret frame is used [45]. The Frenet-Serret frame is an orthonormal basis in $\mathbb{R}^3$ which consists of three unit vectors: unit tangent vector $\mathbf{t}$, unit normal vector $\mathbf{n}$ and binormal unit vector $\mathbf{b}$. These will be used to determine the direction of offsets and extrapolations in sections 4.5 and 4.6. An illustration of the frame is shown in Figure 4.3, which demonstrates how the axis system moves with the direction of the curve.

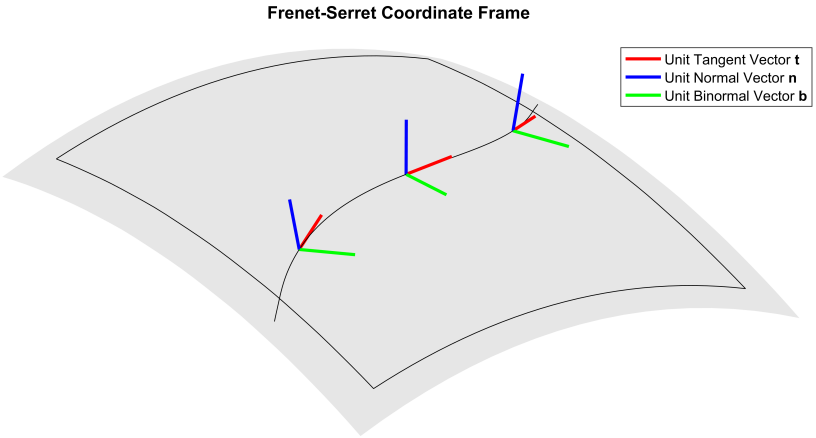


Figure 4.3: Frenet-Serret or TNB frame.

4.2 Progenitor Course Generation

This section covers the generation of the progenitor curves (centre lines of the machine passes) of a fibre steered layer. The conversion of a fibre angle distribution to fibre paths is one of the biggest challenges for the feasibility of fibre steered panels, as the difference between the theoretical lamination parameters of the design and the actual performance of a steered panel can make or break the projected performance gains over conventional composite panels. The streamline algorithm is discussed, as well as possible improvements that will be used throughout the thesis.

4.2.1 Streamline Algorithm Course Generation

As mentioned in subsection 2.3.2, the streamline algorithm generates the paths while attempting to minimize the maximum thickness through a finite element method optimization. To generate the progenitor paths, it requires a finite element mesh, a fibre angle distribution and the width of one machine pass (i.e. the tow width times the number of tows per pass) as inputs.

The finite element mesh consists of a simple triangle mesh with a connectivity matrix, and a desired fibre angle for each data point. The points of the mesh are generated by a modified form of the point projection algorithm as will be explained in section 4.3, where a point on the surface is found for a combination of x and y coordinates, i.e. the z-coordinate is found for a given combination of x and y. This is done to have control over the spacing of the mesh, because generating points on the surface by using the surface parameters alone does not guarantee that the points are e.g. evenly spaced.

The fibre angle distribution is given either by the type of layer, such as unidirectional or linearly varying as illustrated in Figure 2.2, or generated through an lamination parameter optimisation procedure, such as described in *Peeters et al.* (2014) [30]. The streamline algorithm takes these angles and the other inputs and outputs a number of fibre paths.

Unfortunately, streamline code has been mostly a "black box", and the results have often been subpar. Initial tests with a flat plate and constant angle or Linearly Varying Fibre Angle (LVFA) distributions have shown that it cannot estimate the offset distance between progenitor curves correctly, which results in overlapping courses. This problem is exacerbated when the algorithm is applied to curved plates, which is the main application goal for this thesis, or when there is a large variation between the minimum and maximum fibre angle, as *Wurpel* (2015) had already shown, such as in Figure 4.4. While this example is obviously an extreme case, the effect is visible for all fibre angle distributions. However, the streamline algorithm is the only tool available to convert the fibre angles into paths, especially for more complex fibre angle distributions. Therefore, it was decided to continue with the streamline algorithm for the optimised angle distributions, and to selectively use parts of it for simpler constant angle, LVFA and other "prototype" designs, as described in sections 4.2.2 and 4.2.3.

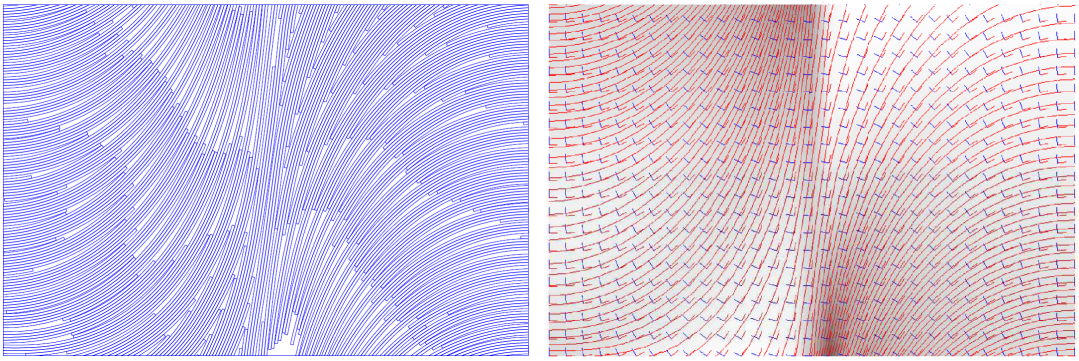


Figure 4.4: A <0,85.94,0>$_x$ LVFA layer by *Wurpel* (2015) on the left and its progenitor curves (in red) on the right [9].

4.2.2 Altered Progenitor Curve Design

To address the issues with the offset distance with the streamline algorithm, a work-around for generating progenitor curves for LVFA layers is proposed. Instead of using the full results from the streamline code, only a single progenitor curve is taken from the centre of the plate. It is then "copied" by generating offset courses in the transverse direction of the progenitor, using the methods from sections 4.4 and 4.6. The offset distance can be accurately approximated by using the surface derivatives, and is set to minimum width the full course will have in the transverse direction. This ensures that at some point in the layer, the full course (with all tows) can be placed, and that the gaps between courses are minimized. Additionally, this prevents uncovered areas from forming near the corners of a layer, as described by *Wurpel* (2015) in several of his designs due to lack of data points. An comparison of the two methods is shown in Figure 4.5. The difference in offset distance between the two methods is small, but any course spacing inaccuracies would result in gaps of at most one full tow width between courses when no overlaps are allowed. Additionally, the improved spacing results in the LVFA courses being slightly more efficient by placing more tows per course.

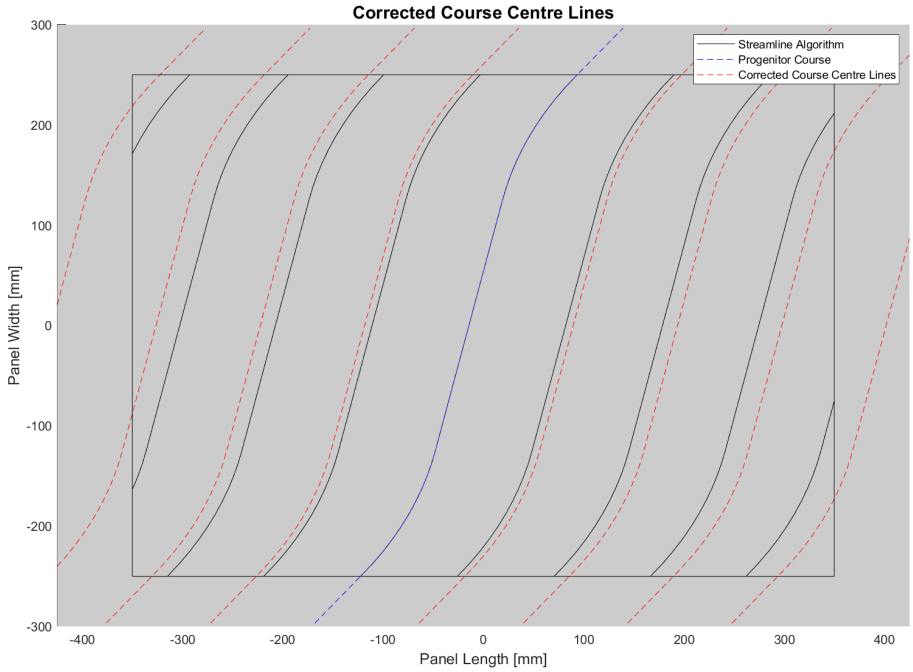
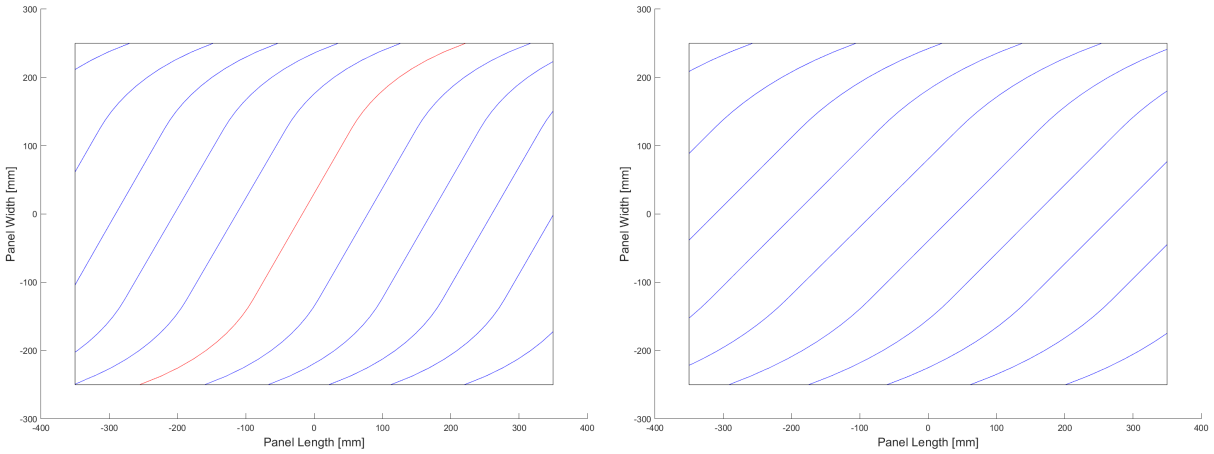


Figure 4.5: Output from the streamline algorithm and the corrected course centre lines. The progenitor course, indicated in blue, was copied from the streamline algorithm, then elongated and offset using the methods described in this chapter to generate the new course centre lines.

Note that the 'single-progenitor' work-around as described is only applicable if the progenitor exists along the full length or width of the layer, depending on the direction in which the fibre angle variation is applied. This is illustrated in Figure 4.6. However, the improvement in course generation is deemed significant enough to include the work-around for suitable fibre angle distributions.



(a) $<20,60,20>_y$ (b) $<20,45,20>_y$

Figure 4.6: Feasibility of the 'single-progenitor' work-around on two fibre angle distributions. Whereas Figure 4.6a has a suitable candidate for the work-around (indicated in red), Figure 4.6b does not, as there is no course that exists along the full width of the panel. Copying any course in Figure 4.6b would therefore alter the fibre angle distribution.

4.2.3 New Fibre Path Design Type

LVFA designs have been used for proof of concept or prototype purposes, as they are relatively easy to describe and reproduce despite often not being optimised for actual design cases. However, most of (older) the LVFA designs allowed fibres within a layer to overlap instead of cutting them, such as in Figure 2.1. During the design process of generating fibre steered layers, it was noted that the continuous variation of the fibre angle in LVFA designs had a detrimental effect to the coverage of the layer when fibres had to be cut. Every angle variation causes towpaths to overlap with their neighbours, which therefore occurs constantly for LVFA designs. From a manufacturing point of view, it would be more beneficial if the paths were constant, but then it would no longer be a steered layer. However, if only a section of the layer of a LVFA design was designated to have a constant fibre angle, then this would have multiple beneficial effects while still having a varying stiffness distribution (i.e., Discrete Linearly Varying Fibre Angle or DLVFA instead of LVFA). The constant region improves the coverage of the layer and may be advantageous when a smooth surface patch is required, such as for joining. The formulaic description of these paths is shown in Equation (4.2), where L is the width of the panel, L_1 is the length of one of the zones where the fibre is steered, measured along the panel width, and L_2 is the length of the constant region, also measured along the panel width.

$$\theta(y) = \begin{cases} (T_0 - T_1)\frac{y}{L_1} + T_1, & \text{for} \quad 0 \leq y \leq L_1 \\ T_0 & \text{for} \quad L_1 \leq y \leq L_1 + L_2 \\ (T_0 - T_1)\frac{L-y}{L_1} + T_1, & \text{for} \quad L_1 + L_2 \leq y \leq L \end{cases} \tag{4.2}$$

The main benefit comes in the form of fewer overlaps due to the constant fibre angle. Fewer overlaps means that fewer tows have to be cut, which in turn reduces the percentage of gaps or voids in a layer. Secondly, fewer cuts mean that there are fewer chances for manufacturing defects such as tow straightening to occur. Finally, the region can be used as a reference for offsetting courses, since the width of the course with respect to the panel dimensions can be approximated quite well, which is something the streamline algorithm struggles with. The main downside of this approach is the more stringent requirements on the steering limit, as the tow steering has to occur in a much more limited

space. Quantitative comparisons between the two types of designs will be made in chapter 6, but a visual comparison can already be made with Figure 4.7.

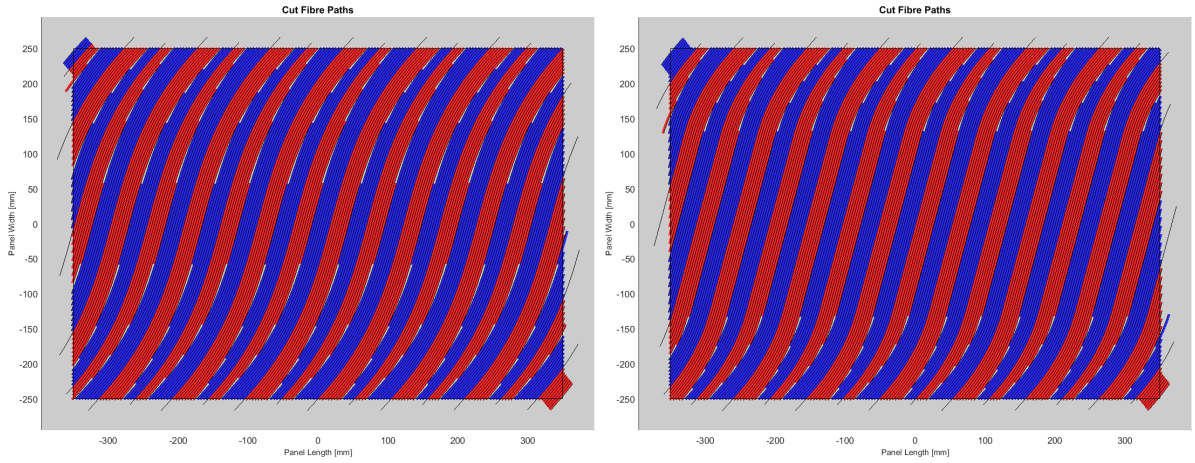


Figure 4.7: A $<47,75,47>_y$ LVFA layer on the left and a suggested DLVFA layer on the right. The DLVFA layer has identical fibre angles to the LVFA design at the centre and the outer ends, but a zone (in this case between a quarter and three quarters of the width of the plate) is designated to have a constant fibre angle instead.

4.3 Point Projection on the Surface

As the results from the (altered) streamline algorithm are in Cartesian coordinates, they need to be adapted to be compatible with the B-spline framework. Due to the discrete nature of the finite element method, the generated points may not exactly lie on the mould surface, and therefore cannot be converted outright. Instead, a point projection algorithm must be used to find a nearest solution such that $S(\bar{u}, \bar{v}) \approx Q$ within set tolerances ϵ. *Piegl and Tiller* (1997) and *Patrikalakis et al.* (2009) suggest using Newton's method to approximate $[\bar{u}, \bar{v}]$ [33, 35].

For a two variable Newton's method, two functions f and g are required for iteration on u and v. Each iteration, the gradients of f and g are computed at a point $[u_i, v_i]$ and set to zero, as shown in Equation (4.3). This will give a new initial point $[u_{i+1}, v_{i+1}]$. When the new value for $S(u_{i+1}, v_{i+1})$ is computed, often there is still an error, which is denoted by r as in Equation (4.4). As more iterations are performed, r should decrease and u and v should converge to the nearest point on the surface to point Q.

$$\begin{aligned} f(u_i, v_i) &= r(u_i, v_i) S_u(u_i, v_i) = 0 \\ g(u_i, v_i) &= r(u_i, v_i) S_v(u_i, v_i) = 0 \end{aligned} \tag{4.3}$$

$$r(u_i, v_i) = S(u_i, v_i) - Q \tag{4.4}$$

The convergence may depend on the selection of the initial guess $[u_0, v_0]$, which therefore should be selected carefully. The strategy employed in this thesis is to generate a number of points in a grid on the surface for known values of u and v. The point nearest to point Q is selected as the initial guess.

Equation (4.3) must be solved, and for that a system of two equations is set up as shown in Equations (4.5) to (4.8). J_i is the Jacobian matrix of f and g with respect to u and v, δ_i is the difference in u and v between two iterations and κ_i contains f and g. By rewriting Equation (4.5) into Equation (4.9), one can then solve for $[u_{i+1}, v_{i+1}]$.

$$J_i \delta_i = \kappa_i \tag{4.5}$$

$$\delta_i = \begin{bmatrix} du \\ dv \end{bmatrix} = \begin{bmatrix} u_{i+1} - u_i \\ v_{i+1} - v_i \end{bmatrix} \tag{4.6}$$

$$J_i = \begin{bmatrix} f_u & f_v \\ g_u & g_v \end{bmatrix} = \begin{bmatrix} |S_u|^2 + r(u,v)S_{uu} & S_uS_v + r(u,v)S_{uv} \\ S_uS_v + r(u,v)S_{vu} & |S_v|^2 + r(u,v)S_{vv} \end{bmatrix} \tag{4.7}$$

$$\kappa_i = -\begin{bmatrix} f(u_i, v_i) \\ g(u_i, v_i) \end{bmatrix} \tag{4.8}$$

$$\begin{bmatrix} u_{i+1} \\ v_{i+1} \end{bmatrix} = \begin{bmatrix} u_i \\ v_i \end{bmatrix} - \begin{bmatrix} |S_u|^2 + r(u_i,v_i)S_{uu} & S_uS_v + r(u_i,v_i)S_{uv} \\ S_uS_v + r(u_i,v_i)S_{vu} & |S_v|^2 + r(u_i,v_i)S_{vv} \end{bmatrix}^{-1} \begin{bmatrix} f(u_i, v_i) \\ g(u_i, v_i) \end{bmatrix} \tag{4.9}$$

The convergence criteria are then as follows:

1. Absolute distance: the computed point is within error tolerance ϵ_1 of point Q as shown in Equation (4.10).
$$|S(u_i, v_i) - Q| \leq \epsilon_1 \tag{4.10}$$
2. Zero cosine: shown in Equation (4.11).
$$\frac{|S_u(u_i,v_i)(S(u_i,v_i)-Q_i)|}{|S_u(u_i,v_i)||(S(u_i,v_i)-Q_i)|} \leq \epsilon_2 \quad \frac{|S_v(u_i,v_i)(S(u_i,v_i)-Q_i)|}{|S_v(u_i,v_i)||(S(u_i,v_i)-Q_i)|} \leq \epsilon_2 \tag{4.11}$$
3. Parameters converge: if Equation (4.12) is satisfied, then the u and v do not noticeably change any more.
$$|(u_{i+1} - u_i)S_u(u_i, v_i) + (v_{i+1} - v_i)S_v(u_i, v_i)| \leq \epsilon_1 \tag{4.12}$$

If any of the conditions is met, then the algorithm is terminated and a projection $S(\bar{u}, \bar{v}) \approx Q$ is found. Otherwise, Equation (4.9) is computed again. This is repeated for all data points Q of a curve, which can now be reconstructed into a curve onto the B-spline surface with the curve fitting algorithm in section 4.4.

This method was also used to generate the finite element method mesh for the streamline code in section 4.2, but in a slightly altered form. Instead of comparing all three components of $S(u_i, v_i)$ with the point Q, only the x and y components are considered. The result is a point on the surface with a z component for a given x and y, which is then used as a coordinate of the finite element mesh. It should be noted that some geometries could have multiple valid solutions for this altered algorithm, which would makes the selection of the initial guess even more critical.

4.4 B-spline Curve Fitting

As mentioned in section 4.3, a curve fitting algorithm is required to reconstruct the course centre lines onto the B-spline surface. This is a standard type of operation as described in *Piegl and Tiller* (1997) [35]. There are two types of curve fitting to select from: curve interpolation and curve approximation. Whereas interpolation follows the data points exactly, approximation naturally only does it approximately. There are several reasons to elect curve approximation over curve interpolation for this application.

First of all, the data points in this case are from a finite element method. The finite element mesh geometry is a discretization of the mould surface, which introduces errors in the data set. Furthermore, the points are then also projected onto another surface. This will most likely introduce further noise to the data set. These two sources of noise would affect the smoothness and accuracy of the curve if interpolation was used. However, curve approximation would be able to capture the general shape of the curve without having to fit through every point, therefore minor errors would not affect it as much. Lastly, the varying number of the data points per curve also advocates for an approximation method.

The curve approximation will be a least squares solution to minimize the deviation from the curve to the data points. Also, the number of control points can be varied accordingly to prevent undesirable oscillation of the curve. Since the points are not distributed evenly along the length of the curve due to the finite element mesh, the least squares solution will be weighted proportionally to the arc length of the segments. This way, smaller segments will have a lower impact on the overall shape of the curve. Furthermore, a suitable knot vector must be chosen such that each knot span contains at least one data point from the projection, before the control points may be computed.

Consider n points Q_i of the same curve C at coordinates $[\bar{u}_i, \bar{v}_i]$ for $i = 1 : n$, that have been projected on the parametric surface in section 4.3. The arc length of a segment of the curve can then be approximated per Equation (4.13):

$$ds = \sqrt{g_{11}du^2 + 2g_{12}dudv + g_{22}dv^2} \tag{4.13}$$

Where ds is the line segment, g_{ij} are the coefficients of the first fundamental form as shown in Equation (4.1) and du and dv are the differences in u and v respectively between two data points. Since the g_{ij} coefficients depend on the slope of the curve, it is best to evaluate them at midpoints between two data points, defined as $\bar{u}_{avg}$ and $\bar{v}_{avg}$.

$$\bar{u}_{avg} = \frac{\bar{u}_{i+1} + \bar{u}_i}{2} \quad \text{for } i = 1{:}n\text{-}1 \tag{4.14}$$

$$\bar{v}_{avg} = \frac{\bar{v}_{i+1} + \bar{v}_i}{2} \quad \text{for } i = 1{:}n\text{-}1 \tag{4.15}$$

Equations (4.1) and (4.13) to (4.15) combine to create $n-1$ line segments with lengths ds, which are summed together for total length s. These are then used to create a diagonal matrix in Equation (4.16), that will function as weights for the least squares approximation, based on the length of the segments. The weights are determined according to the trapezoid rule.

$$W = \frac{1}{2s}\begin{bmatrix} ds(1) & & & \\ & ds(1)+ds(2) & & \\ & & \ddots & \\ & & & ds(n-1) \end{bmatrix} \tag{4.16}$$

To generate the knot vector, first the curve parameter $\bar{t}$ is approximated by mapping the length of the curve s on [0,1] according to $\bar{t} = s(i)/s$ [35]. Then, for a given degree p and number of control points c, a knot distribution scheme can be used from *Piegl and Tiller* (1997) to compute the knots t [35]. First, the average of the number of points per knot span, denoted by d is computed in Equation (4.17). An interpolation scalar α is chosen such that it reflects the distribution of parameters $\bar{t}$ in Equation (4.18). Then, the internal knots t are computed in Equation (4.19). This procedure ensures that each knot span contains at least one $\bar{t}$, which is a condition from [36] for a well-conditioned matrix $(N_{i,p}^T(t)N_{i,p}(t))$ for the least squares approximation. Finally, the control points P_k can be computed from Equation (4.20).

$$d = \frac{n}{c-p} \tag{4.17}$$

$$l = \text{int}(jd) \qquad \alpha = jd - l \qquad \text{for } j = 1, ..., c-p-1 \tag{4.18}$$

$$t_{p+j+1} = (1-\alpha)\bar{t}(l+1) + \alpha\bar{t}(l+2) \qquad \text{for } j = 1, ..., c-p-1 \tag{4.19}$$

$$P_k = [u_k, v_k] = (N_{i,p}^T(\bar{t})WN_{i,p}(\bar{t}))^{-1}\left[N_{i,p}^T(\bar{t})W\bar{u}_i, N_{i,p}^T(\bar{t})W\bar{v}_i\right] \quad \text{for } i = 1:n, \;\; k = 1:c \tag{4.20}$$

This completes the process of reconstructing the data points into B-spline curves on the B-spline surface. These course centrelines will serve as progenitor paths for the offsetting procedure in section 4.6, to simulate the tow edges from a single AFP machine pass. Before this can be done however, the paths need to be extended beyond the panel edges first, for reasons outlined in section 4.5.

4.5 Path Extrapolation

After curve fitting, the course centre lines should accurately represent the intended paths from the finite element method optimization. Unfortunately, the optimization procedure only produces lamination parameters and fibre angles within the panel edges. It does not account for what happens (partially) outside the panel dimensions. However, the paths need to be extended beyond the panel boundaries to avoid gaps and for other manufacturing reasons.

It is important that the course centre lines are not affected by the extension, therefore this step can only take place after fitting the curves. Having said that, the requirements on the extrapolated paths are a lot less stringent than on paths within the panel. Also, the common concerns regarding extrapolation in data sets as a form of prediction beyond the original observation range shouldn't apply here, as it is merely a way to fill voids and prevent manufacturing issues that would otherwise occur. Rather, the main concern should be that the paths do not vary widely in direction, but are somewhat equidistantly spaced along the edges without interfering too much with neighbouring paths. To this end, the most straightforward approach is to do linear extrapolations from the endpoints of each course centreline.

The approach in this thesis is based on an offsetting procedure from *Patrikalakis et al.* (2009) and *Patrikalakis and Bardis* (1989), which is similar in implementation, but along a different direction of the Frenet-Serret frame from Figure 4.3.

Consider two points A and B on a B-spline surface. There exists a (geodesic) path such that the distance travelled between A and B is at a minimum. Equation (4.13) gives an expression for a segment of the arc length in terms of surface parameters u and v, which are functions of a curve parameter t. The total length between points A and B is then given by integration in Equation (4.21) [46]:

$$L = \int_{t_A}^{t_B} \sqrt{g_{11}\dot{u}^2 + 2g_{12}\dot{u}\dot{v} + g_{22}\dot{v}^2}dt \tag{4.21}$$

where the dots denote derivatives with respect to t. In order to minimize the total length from Equation (4.21), standard methods of variational calculus can be used.

$$\frac{d}{dt}\frac{\delta F}{\delta \dot{u}} - \frac{\delta F}{\delta u} = 0 \quad ; \quad \frac{d}{dt}\frac{\delta F}{\delta \dot{v}} - \frac{\delta F}{\delta v} = 0 \tag{4.22}$$

where F is the integrand of Equation (4.21). As shown in *Patrikalakis and Bardis* (1989), the curve parameter t can be substituted with arc length s if a normalization is applied [46]. When used on Equation (4.22), the following system of two non-linear equations is obtained in Equations (4.23) and (4.24):

$$\frac{d^2u}{ds^2} + d_{11}(\frac{du}{ds})^2 + 2d_{12}\frac{du}{ds}\frac{dv}{ds} + d_{22}(\frac{dv}{ds})^2 = 0 \tag{4.23}$$

$$\frac{d^2v}{ds^2} + e_{11}(\frac{du}{ds})^2 + 2e_{12}\frac{du}{ds}\frac{dv}{ds} + e_{22}(\frac{dv}{ds})^2 = 0 \tag{4.24}$$

where d_{ij} and e_{ij} are the Christoffel symbols, as shown in Equations (4.25) to (4.30). The unit normal vector $\mathbf{n}$ is given in Equation (4.31).

$$d_{11} = \frac{\mathbf{n}\cdot(S_{uu}\times S_v)}{|S_u \times S_v|} \tag{4.25}$$

$$d_{12} = \frac{\mathbf{n}\cdot(S_{uv}\times S_v)}{|S_u \times S_v|} \tag{4.26}$$

$$d_{22} = \frac{\mathbf{n}\cdot(S_{vv}\times S_v)}{|S_u \times S_v|} \tag{4.27}$$

$$e_{11} = \frac{\mathbf{n}\cdot(S_u\times S_{uu})}{|S_u \times S_v|} \tag{4.28}$$

$$e_{12} = \frac{\mathbf{n}\cdot(S_u\times S_{uv})}{|S_u \times S_v|} \tag{4.29}$$

$$e_{22} = \frac{\mathbf{n}\cdot(S_u\times S_{vv})}{|S_u \times S_v|} \tag{4.30}$$

$$\mathbf{n} = \frac{S_u \times S_v}{|S_u \times S_v|} \tag{4.31}$$

To solve Equations (4.23) and (4.24), the two second order differential equations can be rewritten to a system of four first order differential equations in Equations (4.32) to (4.35).

$$\frac{du}{ds} = u' \tag{4.32}$$

$$\frac{dv}{ds} = v' \tag{4.33}$$

$$\frac{du'}{ds} = -d_{11}u'^2 - 2d_{12}u'v' - d_{22}v'^2 \tag{4.34}$$

$$\frac{dv'}{ds} = -e_{11}u'^2 - 2e_{12}u'v' - e_{22}v'^2 \tag{4.35}$$

Equations (4.32) to (4.35) require four boundary conditions to be solved. The most straightforward approach is to use the initial location and initial derivatives of the extrapolation points, as both are known at any location due to the properties of the B-spline curve. These are shown in Equation (4.36). The initial derivatives are normalized with λ from Equation (4.37) in Equation (4.39), such that the normalisation condition from Equation (4.38) holds.

$$[u_0, v_0] = C(t_0) \quad ; \quad [\dot{u}_0, \dot{v}_0] = C_t(t_0) \tag{4.36}$$

$$\lambda = \frac{1}{\sqrt{g_{11}\dot{u}_0^2 + 2g_{12}\dot{u}_0\dot{v}_0 + g_{22}\dot{v}_0^2}} \tag{4.37}$$

$$g_{11}\dot{u}_0^2 + 2g_{12}\dot{u}_0\dot{v}_0 + g_{22}\dot{v}_0^2 = 1 \tag{4.38}$$

$$[u_0', v_0'] = \frac{C_t(t_0)}{\sqrt{g_{11}\dot{u}_0^2 + 2g_{12}\dot{u}_0\dot{v}_0 + g_{22}\dot{v}_0^2}} \tag{4.39}$$

Obtaining the initial conditions concludes the development of the path extrapolation function. The set of differential equations and initial conditions can now be solved by an ODE solver. For the purposes of this thesis, the ode45.m solver of MATLAB was used [47]. The extrapolation distance can be set as an input, which allows the user to elongate the path to account for voids as well as manufacturing constraints.

4.6 Geodesic Offsets to Simulate Tow Paths

This section will provide an overview on how tow paths are simulated in this thesis. In section 4.4 and section 4.5, the progenitor curve has been reconstructed and elongated such that it represents the machine pass of an AFP robot. To simulate a band of tows, offsets from the progenitor curves are generated, according to the width and number of the tows.

As mentioned in section 4.5, the extrapolation procedure has a lot of similarities with the offset procedure. It uses the same approach, same set of differential equations and is based on the same references [33, 46]. But whereas the extrapolation travels in a direction tangent to the curve endpoints, offsets are taken in the binormal direction, denoted by binormal vector $\mathbf{B}$, or the unit binormal vector $\mathbf{b}$ in the Frenet-Serret frame. Consequently, the initial derivatives need to be altered to travel in the direction orthogonal to the curve along the surface, obtained from Equation (4.40) and Equation (4.41).

$$\mathbf{T} = \dot{u}_0 S_u + \dot{v}_0 S_v \tag{4.40}$$

$$\mathbf{B} = (S_u \times S_v) \times \mathbf{T} = u'(0)S_u + v'(0)S_v \tag{4.41}$$

The left part of Equation (4.41) can be rewritten from a double cross product to a sum of two dot products in Equation (4.42). Then, by rearrangement the terms for the initial derivatives and applying the same normalization as in Equation (4.39), the offset direction can be obtained from Equation (4.43)

and Equation (4.44). Note how the plus-minus sign correspond with the two possible directions of geodesic offsets relative to the progenitor curve.

$$\mathbf{B} = -(S_v \cdot \mathbf{T})S_u + (S_u \cdot \mathbf{T})S_v = U_1 S_u + V_1 S_v \tag{4.42}$$

$$u'(0) = \pm \frac{U_1}{\sqrt{g_{11}U_1^2 + 2g_{12}U_1V_1 + g_{22}V_1^2}} \tag{4.43}$$

$$v'(0) = \pm \frac{V_1}{\sqrt{g_{11}U_1^2 + 2g_{12}U_1V_1 + g_{22}V_1^2}} \tag{4.44}$$

As in the previous section, the four initial conditions are used by the MATLAB solver to generate points offset from the progenitor curve. Then, a curve-fitting procedure from section 4.4 is required to construct final curves. These lines represent the boundaries of a single tow, which will be compared against neighbouring tows in section 4.7 to check for intersections.

4.7 Curve-Curve Intersection Algorithm

As mentioned before in subsection 2.2.1 on automated fibre placement, the occurrence of gaps and/or overlaps are inevitable due to the nature of VSL and the AFP technique. Since this thesis will only consider designs without overlaps, an algorithm has to be developed to find all the intersections between the neighbouring tows and then decide on where the cutting occurs. This section will cover the first part of this problem, namely of a curve-curve intersection algorithm.

The offset lines from section 4.6 represent the outer borders of the tows of a single machine pass. An intersection is found whenever two offset lines cross, which has to be handled by cutting one of the two tows. Finding intersections is also imperative to determine the crossings of tows with the panel boundary. This is needed from a manufacturing standpoint to reduce scrap waste while conforming to the minimum cut length for placing AFP tows.

Since finding intersections is another fundamental technique for many Computer Aided Design (CAD)/ Computer Aided Modelling (CAM) applications, there are a variety of options available. *Sederberg and Parry* (1986) and *Patrikalakis et al.* (2009) were used to obtain an overview of the current methods [33, 48]. For curve-curve intersections, three different kinds of methods have been identified.

1. Bézier clipping, which is a curve-curve intersection algorithm that utilizes the convex hull property of Bézier curves to iteratively clip away regions where no intersections can occur until all intersections are found [49].

2. Subdivision methods, as described by *Lane and Riesenfeld* (1980), *Koparkar and Mudur* (1983) and *Wang* (1984) [50–52]. This method subdivides the curves into smaller parts based on characteristic points as a form of preprocessing. These subdivisions are then guaranteed to lie within the bounding box defined by the end points of the curve segment, which makes finding the intersections of these curves very convenient.

3. Modified Newton-Raphson methods, which all (loosely) utilize the root-finding technique in some way to find the intersections. *Mørken et al.* (2009) uses the control polygons of the control points of two B-spline curves as a preprocessing step to determine quickly where intersections may occur, before applying Newton's Method [53]. *Rajab and Piegl* (2014) proposed an algorithm to find curve-curve intersections by using a knowledge-guided NURBS system with a tight bi-arc decomposition along with pre- and post-processing the results for accuracy [54].

The decision was made to use the method from *Mørken et al.* (2009), since it would require the least amount of preprocessing and the examples from the paper were similar to the type of intersections expected in the thesis [53].

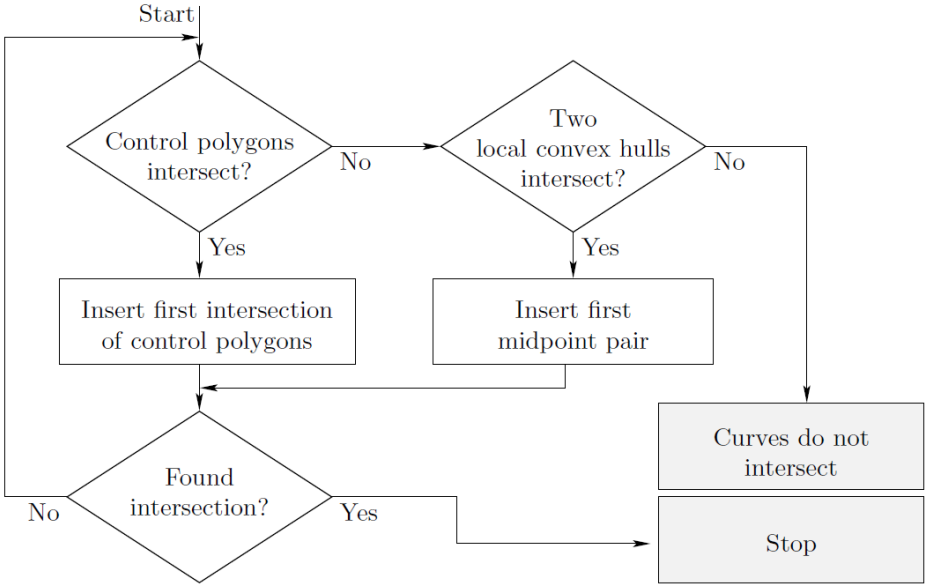


Figure 4.8: Schematic outline of the intersection finding algorithm [53].

The full procedure is best explained (globally) with the help of Figure 4.8. First the control polygons, consisting of the control points of the two curves, are generated. If the control polygons intersect, then the curves should also intersect at some point. To find this point, knot insertions are iteratively performed until the new knots of both curves have converged to the intersection, which can then be stored. A demonstration of this procedure is shown in Figure 4.9. As illustrated, knot insertions do not change the shape of the curve itself, but add new knots and control points to alter the control polygon as a form of refinement.

However, it is possible for the curves to intersect without intersections between the control polygons. To account for this, if no intersection is detected, then also the two local convex hulls are compared. If there is any overlap, then a midpoint pair of knots is inserted to refine the control polygons. This is repeated until either a control polygon intersection is found, or the convex hulls no longer overlap. If there is no control polygon intersection or convex hull overlap, then the curves do not intersect on that interval.

The algorithm will now be explained in more detail in two parts: first the setup of the two curves is described, then the method for selecting the values for the knot insertions to attain the intersections.

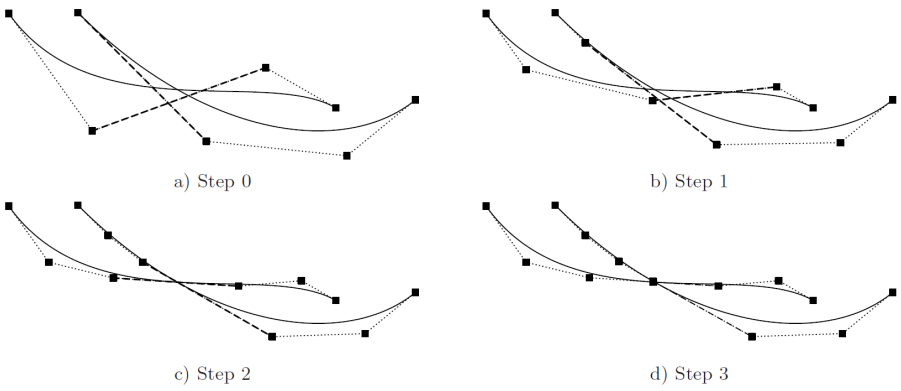


Figure 4.9: Through repeated knot insertions of the intersections between the control polygons, a sequence is obtained that converges to the intersection of the two curves. [53].

Consider a B-spline curve f. The curve has a set of control points P that together form the interpolants of the linear piecewise function of the control polygon Γ_f. A one-to-one mapping between the knots s and the control polygon Γ_f is required to iteratively perform knot insertions to find the intersections. To achieve this, the knot interval $[s_{d+1}, s_{n+1}]$ is used, where d is the degree of the curve and n the number of control points. A relation can then be established between the control polygon and the knots in Equation 4.45, with the knot average $\bar{s}$ (or Greville abscissa [55]) shown in Equation 4.46. A similar procedure is performed for the second curve g, with control polygon Γ_g, control points Q and knots t.

$$\Gamma_f(s) = \frac{\bar{s}_{i+1} - s}{\bar{s}_{i+1} - s_i} P_i + \frac{s - \bar{s}_i}{\bar{s}_{i+1} - s_i} P_{i+1} \tag{4.45}$$

$$\bar{s}_i = (s_{i+1} + ... + s_{i+d})/d \tag{4.46}$$

Now onto the knot insertion. The process of knot insertion does not change the shape of the curve, but it does add new knots and control points, as illustrated in Figure 4.9. Furthermore, some of the existing control points are replaced.

Consider σ to be a new knot location, inserted in the interval $\sigma \in [s_\mu, s_{\mu+1})$. This results in a new set of control points $(R_i)_{i=1}^{n+1}$. Since B-splines are piece-wise continuous, only d control points are affected. This means that $R_i = P_i$ for $i = 1, ..., \mu - d$ and $R_i = P_{i-1}$ for $i = \mu + 1, ..., n + 1$. The affected control points are generated according to Equation 4.47. For the second curve, the new knot location will be referred to with τ.

$$R_i = \frac{\sigma - s_i}{s_{i+d} - s_i} P_i + \frac{s_{i+d} - \sigma}{s_{i+d} - s_i} P_{i-1} \quad \text{for } i = \mu - d + 1, ..., \mu \tag{4.47}$$

The actual locations of the knots are determined with a two-variable Newton's method. Suppose the two control polygon segments $[P_{i-1}, P_i]$ and $[Q_{j-1}, Q_j]$ intersect. The two segments can be equated by rewriting Equation 4.45 for the two knot insert values σ and τ in Equation 4.48.

$$\frac{\bar{s}_i - \sigma}{\bar{s}_i - \bar{s}_{i-1}} P_{i-1} + \frac{\sigma - \bar{s}_{i-1}}{\bar{s}_i - \bar{s}_{i-1}} P_i = \frac{\bar{t}_j - \tau}{\bar{t}_j - \bar{t}_{j-1}} Q_{j-1} + \frac{\tau - \bar{t}_{j-1}}{\bar{t}_j - \bar{t}_{j-1}} Q_j \tag{4.48}$$

Then, each side of equation Equation 4.48 is rewritten according to Equation 4.49, which then leads to Equation 4.50 when rearranging the terms. Grouping the "derivatives" together in two-by-two matrix M in Equation 4.51, the modified Newton's method can be recognized, as shown in Equation 4.52.

$$P_i + (\bar{s}_i - \sigma)\frac{P_i - P_{i-1}}{\bar{s}_i - \bar{s}_{i-1}} = P_i + (\bar{s}_i - \sigma)\Delta P_i \quad ; \quad \Delta P_i = \frac{P_i - P_{i-1}}{\bar{s}_i - \bar{s}_{i-1}} \tag{4.49}$$

$$P_i - Q_j = -((\sigma - \bar{s}_i)\Delta P_i - (\tau - \bar{t}_j)\Delta Q_j) \tag{4.50}$$

$$M = \begin{bmatrix} \Delta P_{i1} & -\Delta Q_{j1} \\ \Delta P_{i2} & -\Delta Q_{j2} \end{bmatrix} \tag{4.51}$$

$$\begin{bmatrix} \sigma \\ \tau \end{bmatrix} = \begin{bmatrix} \bar{s}_i \\ \bar{t}_j \end{bmatrix} - M^{-1}(P_i - Q_j) \tag{4.52}$$

As this is an iterative process, the knot insertions continue until a desired accuracy is reached. This is repeated for all polygon segments until every intersection is found. The results are stored in a table, which functions as a database for the cutting algorithm. It registers the location of the intersection in terms of the curve parameter t, the opposing course and tow that the curve intersected with (important for the cutting logic), the number of times a curve intersects with an opposing tow (important for tow adds/restarts) as well as the "type" of intersection. For example, an intersection between a course and the panel edges is registered differently than an intersection with a neighbouring tow, as the panel edges indicate the boundaries wherein the course should exist, i.e. the start and end location. The types of intersection and the subsequent decision-making is further expanded upon in section 4.8.

4.8 Tow Cutting

This section will cover the work to cut tows according to their intersections. The general guideline for the cutting algorithm is to allow the user to decide where to cut tows with the same basis, i.e. to register all possible cutting locations and to let the user decide which parts to keep. Secondly, the types of intersections that were encountered are described, and how they were handled. Finally, the chosen cutting 'strategy' is described. Although there are many possible ways to cut the tows, for this thesis only one type is considered.

4.8.1 Cutting Algorithm

The first step is to prepare the curves such that parts can be "cut" with the intersections known from section 4.7. With B-spline curves, this is typically done by applying multiple knot insertions at the desired location until the continuity of the curve reaches C^{-1} [35], as increasing the knot multiplicity decreases the curve continuity at that location. The curve has then effectively been split in two, as the two segments are practically independent of each other, only sharing the knots and control point at the cut. This allows for the removal of parts of the curve without affecting the other segments. Then, according to the cutting strategy, it can be decided per segment whether it should be cut. Similarly to the intersection tables, a table for each segment of each curve is made, alongside a vector and matrix for the control points and knots of the 'segmented' curve, i.e. with the knots inserted. The segment table refers to the indices of the first and last knot for the segment, as well as the first and last control point. Furthermore, each segment also has a boolean which prescribes if it exists in the design, or whether it is cut. This allows for different cutting strategies for the same design without altering previous work. Examples of an intersection table and segments table are shown in Table 4.1 and Table 4.2, and how they are used to cut the tows in this thesis is explained in detail in Appendix A.

Table 4.1: Example intersection table.

#	Curve Parameter	Course	Tow	Intersection Type
1	0.05	4	15	1
2	0.12	4	16	1
3	0.17	4	16	2
⋮	⋮	⋮	⋮	⋮
n	1	4	14	0

Table 4.2: Example segments table.

#	First Knot	Last Knot	First CP	Last CP	Active
1	1	9	1	5	0
2	6	18	6	14	0
3	15	23	15	19	1
⋮	⋮	⋮	⋮	⋮	⋮
m	46	56	46	52	0

Regarding the cutting logic, if a tow segment does not overlap with any other segments, then it should always exist. Unfortunately, this cannot be deduced by only looking at the intersection locations. For example, a tow could start within the area of another course, leading to the overlapping parts to be removed, but added/restarted later when it is no longer overlapping. This is where the intersection table comes in, as it checks the opposing tow that it intersected with and registers the number of times it intersected with this tow before. This provides the algorithm with the direction of the tow with respect to the neighbouring tows, i.e. does it go further towards the opposing tows, or does it move away. Then, if a unplaced/cut tow intersects again with the outer tow of its neighbour, it can be deduced that the tow can be restarted. An illustration of tow adds/restarts is shown in Figure 4.10.

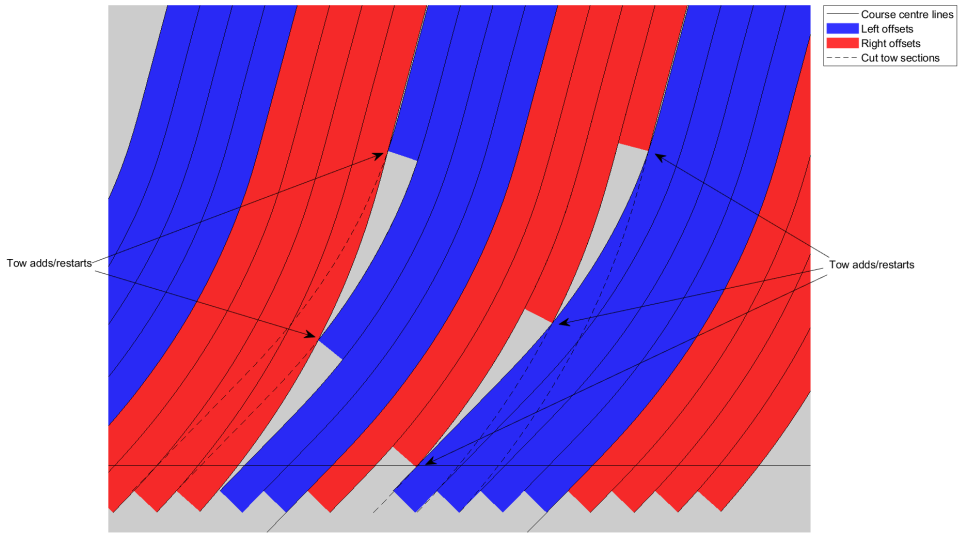


Figure 4.10: Tow adds/restarts within a layer. The middle course is cut on both sides at the start of the course due to overlap with its neighbours, with the unplaced/cut parts illustrated with the dashed lines. When a 'cut' tow intersects with the outer edge of a neighbouring course, it can be added/restarted.

Secondly, regardless of cutting strategy, intersections with the edges of the panel indicate the start- and endpoints of the tow (unless the minimum cut length is not reached) and need to be trimmed accordingly. However, due to the approach of using the outer edge of the fibre as the curve, this can lead to incorrect results. This is best illustrated with the help of Figure 4.11. Notice how all the tow edges have been trimmed to the panel edge, but the blue tows do not cover the edge of the panel. This will always occur for either the left or the right offsets of the tows, depending on the orientation of the course and the panel edge. It is solved by instead looking at the inner edge of the tow as a trimming point, which is the panel edge intersection of the curve to the right of the tow. Projecting this point onto the curve (arrow and dashed line), the blue tows can be trimmed correctly. In this case, the neighbouring intersection is offset one tow width towards the blue tow edge (simulating a tow width), and then a point projection algorithm is applied to find the nearest point on the curve.

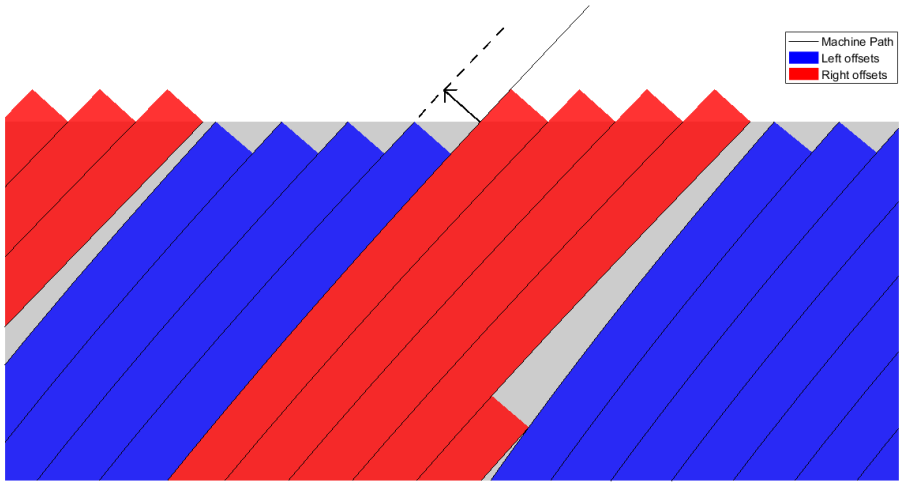


Figure 4.11: Tow trimming to panel dimensions. In this example, the red and blue tows are all trimmed by comparing their outer edge, as seen from the course centre line, to the panel edge. This results in the blue tows being too short, due to their flipped orientation compared to the red tows. Point projection from the inner point (arrow and dashed line) allows for correct trimming.

Another issue that arose from using the outer edge of the fibre, was that looking solely at intersections between the offset curves (or tow edges) fails to account for tow overlaps without intersections. For example, imagine a case of two parallel machine paths with identical fibre angles, but spaced too close together. It is then possible for two fibres to overlap, without their edges ever intersecting. This can be solved by including the endpoints of the tow to the outer edge of the tow, with an extra intersection check. This too should be registered in the intersections, but with a different flag such that it can be recognized as an 'endpoint intersection'. It is registered differently because a endpoint intersection should not be taken into account when checking for tow restarts.

4.8.2 Cutting Strategy

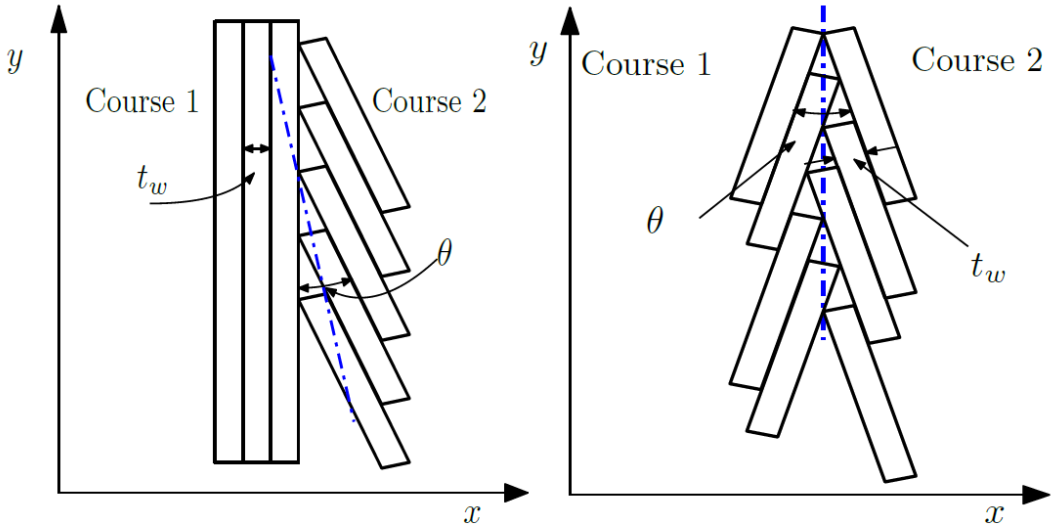
Finally, a decision has to be made on the cutting strategy/strategies for the designs. Throughout the years, several options have been suggested. Factors to consider are the number of cuts, panel coverage and manufacturing issues such as tow straightening.

Single-Sided and Alternating Course Cutting

The simplest option from a programming point of view would be to use single-sided cuts, irrespective of any other parameters other than course number. An example of single-sided cutting is shown in Figure 4.12. Double single-sided cuts (or alternating course cutting in terms of tow courses) have the added benefit that half the courses would effectively be untouched. Alternating course cutting is a robust form of cutting that can be applied almost anywhere, regardless of fibre angle distribution. The main downside is that tow straightening is not taken into consideration, which could cause significant manufacturing defects. Alternating course cutting was included as a cutting strategy due to its broad applicability, and a flow diagram explaining the cutting process is shared in Appendix B.

'Zipper' cuts

'Zipper' cuts have been suggested as a way to minimise the fibre angle mismatch by evenly cutting both sides of a course. However, initial research by *Mishra* (2017) did not show any benefit for this type [21]. Additionally, fibre straightening could occur even more frequently than alternating course cutting, as even more tows would be affected. As part of the emphasis of this thesis is on the minimization of manufacturing defects, this concept was discarded. An example of 'zipper' cuts can also be found in Figure 4.12.



(a) Single sided cuts **(b)** Zipper cuts

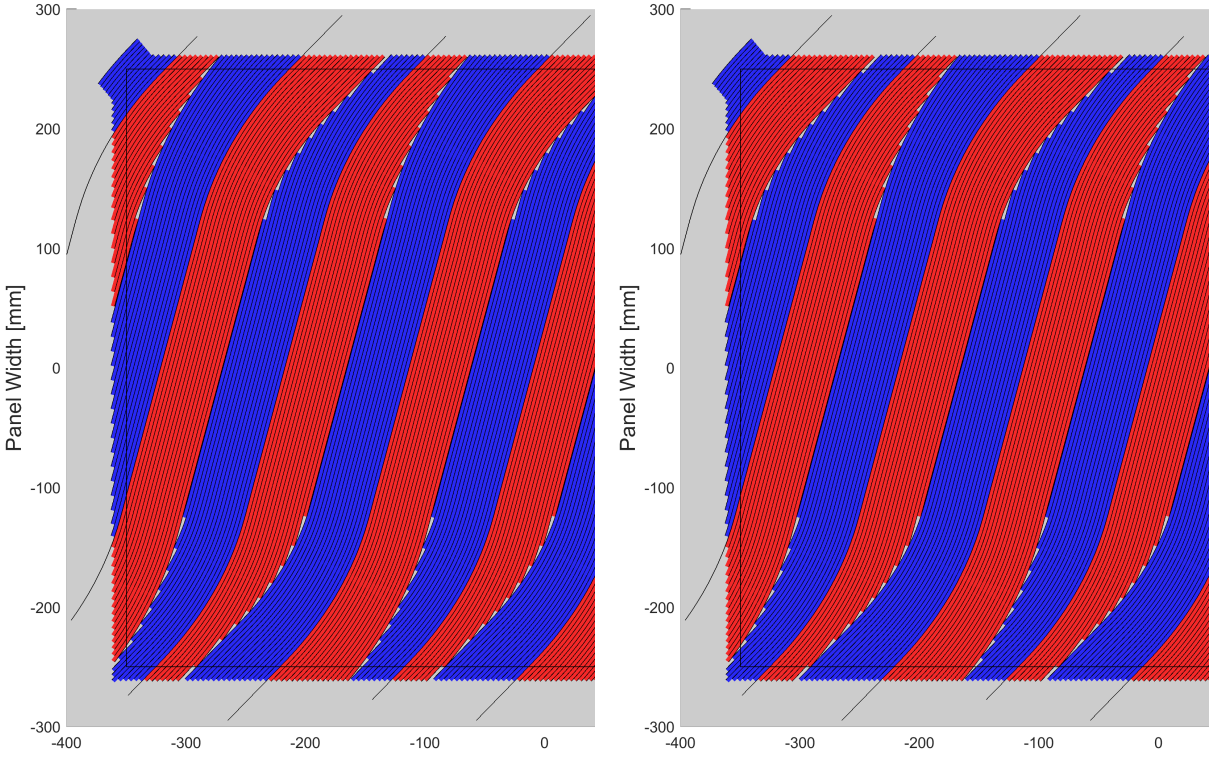
Figure 4.12: Single-sided cuts and Zipper cuts were compared by *Mishra* (2017) from a performance standpoint [21].

Curvature-Based Cutting

Curvature-based cutting has been suggested based on the experiences from *Blom* (2010) with tow straightening of cut tows [5]. She noted that tow straightening occurred less on the inner side of a curve compared to the outer side when a tow was cut. This is equivalent to using the curvature of the tow courses as a guideline for which side of the courses to cut. Unfortunately, curvature-based cutting could not be implemented correctly in this thesis. The usage of the B-spline curves with its control points and the curve-fitting procedure from section 4.4 led to regions with minute changes in curvature where they would not be expected, which led to incorrect results. Therefore, this method was not pursued further in this thesis. It should also be noted that curvature-based cutting would not fully eliminate tow straightening, as it is only less likely to occur on inner side of a curve, instead of fully eliminating it.

Anti-Straightening Cutting

Anti-straightening 'cutting' is a strategy that severely limits the design space at the benefit of fewer potential manufacturing issues, mostly notably tow straightening. Tow straightening only occurs when the AFP robot loses the capability to steer the tow in the desired direction, i.e. when it is cut. However, tow adds/tow restarts do not have this problem, as noted by *Blom* (2010), since the machine retains control [5]. In other words, if a layer can be designed such that tows in overlapping regions between two courses are only added instead of being cut, this should eliminate the risk of tow straightening. Unfortunately, this is easier said than done and should only be possible for a specific subset of (D)LVFA panels. For example, high amounts of steering and/or multiple angle shifts would all but negate the possibility of only doing tow adds without incurring extra gaps and/or other manufacturing defects. Nevertheless, it is included in this thesis as an example of a set of steered fibre layers without overlaps or tow straightening, which could be used for validation or testing purposes. A comparison between alternating course cuts and "anti-straightening" cuts for a compatible fibre angle distribution is shown in Figure 4.13 and a flow diagram explaining the cutting process is provided in Appendix B.

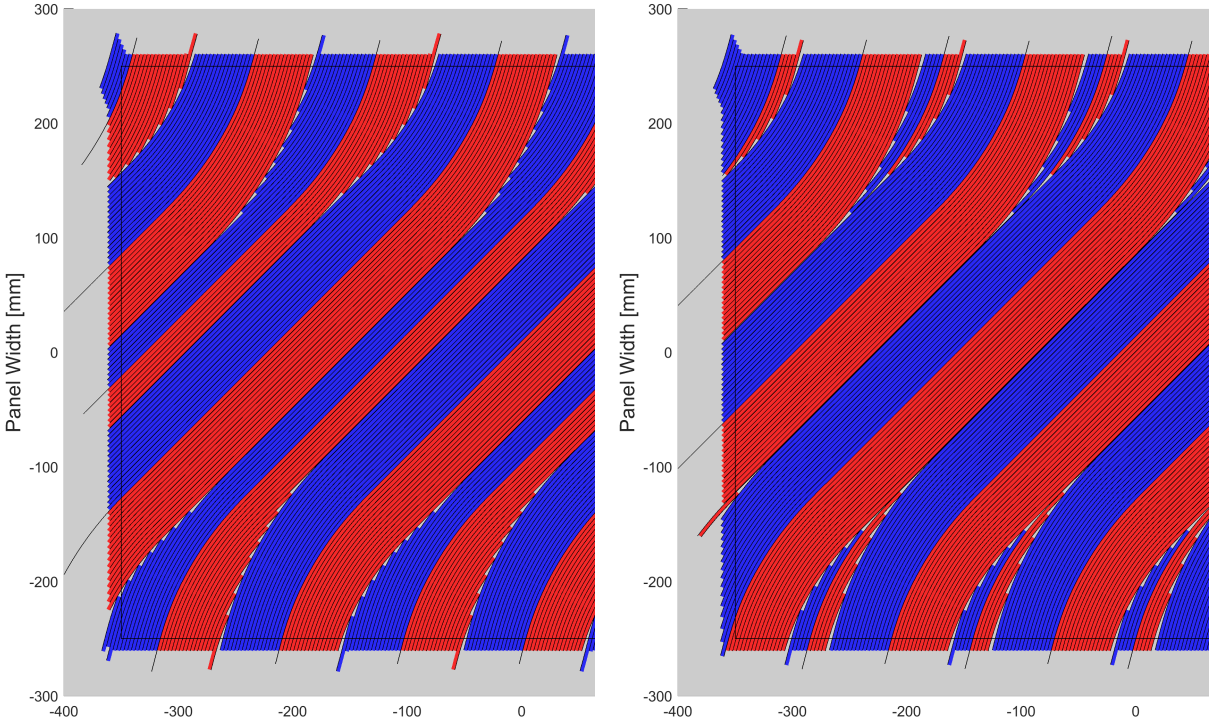


(a) Alternating course cutting

(b) Anti-straightening cutting

Figure 4.13: Comparison between two $<45,75,45>_y$ DLVFA layers with differing cutting strategies. With alternating course cutting, only the odd courses are 'cut', and the even courses are untouched. In contrast, the courses in the anti-straightening layer are all 'cut' at the start of the course, but become full-width at the end.

The anti-straightening method as just described can only be used for fibre angle distributions where the overlap areas are at the ends of the course. If the fibre angles in Figure 4.13b were reversed, then the overlaps would occur in the middle instead, negating the ability to only add tows within the panel dimensions. A variation on anti-straightening cutting can be achieved by using 'filler courses' instead, meant to fill gaps between two regular, full-size courses. This is illustrated in Figure 4.14. An extra filler course is placed at the half-way point between each two full courses. The filler courses are then cut with alternating course cuts, resulting in small patches that fill some of the gaps between the two full courses.



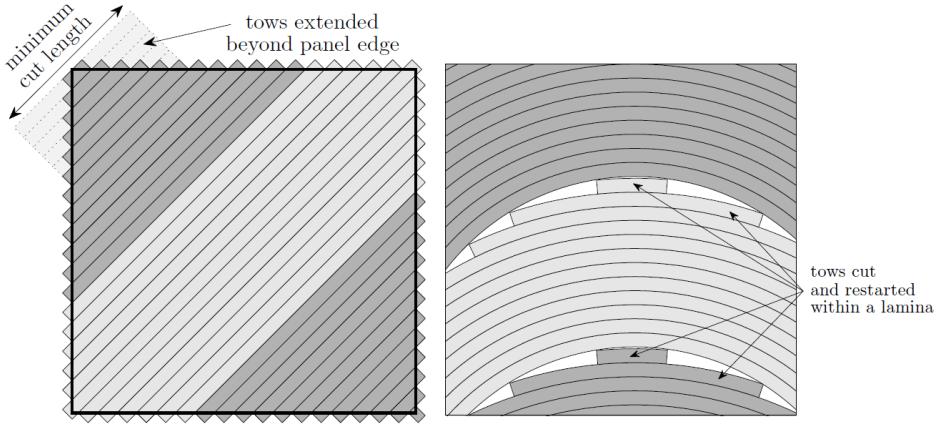
(a) Alternating course cutting.

(b) Anti-straightening cutting by using filler courses.

Figure 4.14: Comparison between two $<75,45,75>_y$ DLVFA layers with differing cutting strategies. Note how the overlaps occur in middle of the course for Figure 4.14a. 'Regular' anti-straightening cuts are therefore not possible for this fibre angle distribution, but with filler courses a setup can still be achieved where tows are only added within the panel dimensions.

4.9 Minimum Cut Length

As mentioned in subsection 2.2.1, AFP machines have a limit to the minimum length of the tow that can be placed. Therefore, a check must be created to account for this occurrence, by comparing the length of the tow to the minimum cut length. Then, the possibilities to extend the tow are examined, and otherwise it is removed from the design. The cases are illustrated in Figure 4.15. In addition, if a tow segment can be extended, but is shorter than a chosen length threshold, it may be removed as well to prevent single protruding tows.



(a) Short tows at part or ply boundaries (b) Short tows within a ply

Figure 4.15: Potential cases where the minimum cut length may play a role [5]. Whereas the tows in (a) can be extended, the tows in (b) cannot, and may have to be removed if below the threshold.

The intersections at the ends of the tow are examined to determine if it can be elongated. Tow elongation can only be performed if it borders at least one panel edge. The number of bordering panel edges is taken into consideration as well, since the elongations are preferably divided evenly on each side. If a tow is too short and is enclosed by two other tows, then it has to be removed.

The tow length is calculated by using an arc length numerical integration. B-spline curves can be infinitely sampled within their domain, so the approximation can be very accurate depending on the interval of the curve parameter. It should be noted however that the lengths of the outer tows are used, which may vary slightly from the actual length of the fibre. However, as the tows widths are small compared to their length, this will only incur a minor error and can be accounted for by simply adding a small percentage to the desired cut length in the algorithm. For m points $P_i = [x_i, y_i, z_i]$ on the curve C, the total length of a tow edge is given by Equation (4.53).

$$L = \sum_{i=1}^{m-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \tag{4.53}$$

Then, the new tow ends are determined. First, an approximation is made by calculating the parametric speed $\frac{ds}{dt}$ (change in arc length s for a change in curve parameter t) at the endpoints of the tow and projecting outwards. Although the parametric speed is not constant in general, it is often close to constant if the control points and knots are spaced evenly. With the parametric speed at each end determined, the approximated curve parameter corresponding with the required elongation of the tow can be calculated. Finally, the length of the new curve is re-examined to ensure it is within the margin. This is repeated until the desired length is reached.

4.10 Curvature

The curvature of the course centre lines is the final manufacturing constraint from subsection 2.2.1 that will be considered by the new algorithm. As mentioned there, tows may kink out of plane if too much steering is applied. Although the values of maximum steering from Table 2.1 are not set in stone, it is important to register the curvature to give some measure to predict possible tow kinking. Specifically, the geodesic curvature κ_g will be used for the evaluation, because it is a representation of planar curvature as if you were on the (curved) surface.

The approach in this thesis closely mimics the one by *Patrikalakis et al.* (2009), with one important exception. Whereas *Patrikalakis et al.* (2009) uses an arc length parametrized curve, this thesis uses the regular curve parameter t and applies a normalization as needed, similar to what was used in sections 4.5 and 4.6 with Equation (4.37). The following equations will show the approach from *Patrikalakis et al.*

(2009) and the equivalent curve parameter expressions.

For a given B-spline surface $S(u, v)$ and an arc parametrized curve $C(s)$ on S, let there be a point on the surface such that:

$$S(s) = S(C(s)) = S(u(s), v(s)) \tag{4.54}$$

The unit tangent vector $\mathbf{t}$ of C is then obtained by differentiating with respect to the arc length s and applying the chain rule in Equation (4.55), which can then be related to curve parameter derivatives through the normalization λ.

$$\begin{aligned} \mathbf{t} = \frac{dS(u(s), v(s))}{ds} &= S_u \frac{du}{ds} + S_v \frac{dv}{ds} \\ &= \lambda (S_u \frac{du}{dt} + S_v \frac{dv}{dt}) \end{aligned} \tag{4.55}$$

Likewise, by differentiating the unit tangent vector $\mathbf{t}$ with respect to the arc length and applying the chain rule, the curvature $\frac{d\mathbf{t}}{ds}$ or $\mathbf{k}$ is obtained in Equation (4.56).

$$\begin{aligned} \frac{d\mathbf{t}}{ds} &= S_{uu}(\frac{du}{ds})^2 + 2S_{uv}\frac{du}{ds}\frac{dv}{ds} + S_{vv}(\frac{dv}{ds})^2 + S_u\frac{d^2u}{ds^2} + S_v\frac{d^2v}{ds^2} \\ &= \lambda^2(S_{uu}(\frac{du}{dt})^2 + 2S_{uv}\frac{du}{dt}\frac{dv}{dt} + S_{vv}(\frac{dv}{dt})^2 + S_u\frac{d^2u}{dt^2} + S_v\frac{d^2v}{dt^2}) \end{aligned} \tag{4.56}$$

Then, the scalar quantity of the geodesic curvature κ_g can be written as the product of the curvature vector $\mathbf{k}$ and the binormal vector $\mathbf{b}$ in Equation (4.57). Effectively, the geodesic curvature is obtained by taking the components of the curvature in the direction of the binormal vector. The right-hand side of Equation (4.57) is rewritten to use the expressions from this section and the unit normal vector $\mathbf{n}$ from Equation (4.31).

$$\kappa_g = \mathbf{k} \cdot \mathbf{b} = \frac{d\mathbf{t}}{ds} \cdot (\mathbf{n} \times \mathbf{t}) \tag{4.57}$$

Equation (4.57) provides a single value for any single point of the curve C, which can then be sampled as desired and the absolute value of the curvature compared to some maximum value.

During the verification of the curvature code, an issue was identified regarding the smoothness of the fit from the curve fitting algorithm, which gave unexpected results for the curvature of a curve. This is discussed in Appendix C.

Chapter 5

Fitness Criteria for Variable Stiffness Laminates

This chapter is dedicated to defining measures of objectively grading fibre steered designs from the fibre path algorithm. As illustrated in Figure 2.10, fibre steering with Automated Fibre Placement (AFP) typically causes defects or errors in the form of gaps or overlaps, as identical courses cannot be shifted without either of these cases occurring. Additionally, as will be shown in this chapter, fibre steering can be a source of fibre angle errors, due to the way that the fibre angle remains constant along the width of the course. Lastly, the number of cuts within a panel layer, and what fraction of the tows are affected by cuts are registered as well. Each cut tow within the panel could be considered to create a load discontinuity, and each cut reduces the production speed.

It should be noted that none of the aforementioned phenomena occur in 'conventional' constant stiffness composite layers. Therefore, the fitness criteria are predominantly parameters to compare different fibre steered designs to each other or to assess the consequences of using fibre steering designs over constant stiffness laminates.

5.1 Layer Coverage

The layer coverage is a measure of how much of the design area of a layer is covered by tows, given as the ratio of the tow area over the total area of the layer. Inevitably, gaps will occur between courses when they are shifted with AFP, but gaps should be kept to a minimum and therefore registering the uncovered area has merit in rating designs quantitatively. For Linearly Varying Fibre Angle (LVFA) designs, the percentage of gaps is proportionate to the tow width, the number of courses and the fraction of the length of the course that is steered, as it is the steering of courses that induces the overlap of neighbouring tows.

The coverage is calculated by taking the total area of the design and comparing it to the area of all the fibres within the design area. The design area can be approximated by converting the NURBS/B-spline surface into a triangulated mesh and then calculating the area of each triangle with the aid of a connectivity matrix. The connectivity matrix D relates a set of points P of the triangulated surface, shown schematically in Figure 5.1, and has the size of $2(m-1)(n-1)\times 3$, where m and n are the number of points in the length and width of the mesh, with each row containing the indices of the three corners of a triangle. The area of a single triangle is then given as the cross product of the three corners as shown in Equation (5.1)[56].

$$A = 1/2 \sum_{i=1}^{2(m-1)(n-1)} |P(D(i,2)) - P(D(i,1)) \times (P(D(i,3)) - P(D(i,1))| \tag{5.1}$$

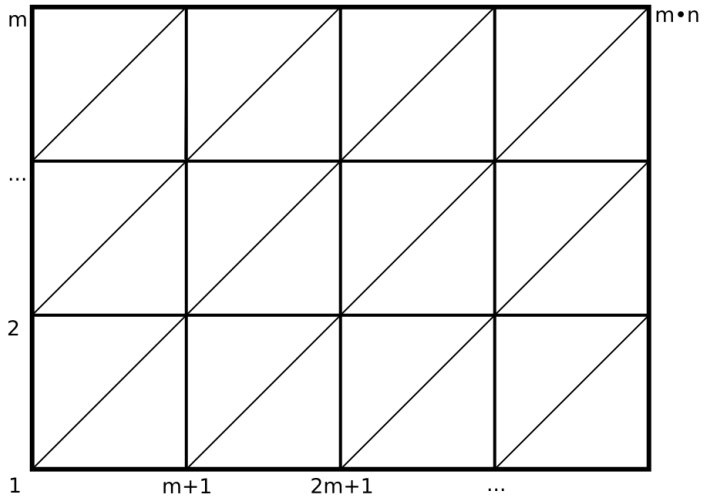


Figure 5.1: Numbering of the triangulated mesh as used in the connectivity matrix D

The tow areas are calculated similarly with the triangle area approximation, but within the bounds of the mould and the two edges of the fibre. The intersection and segment tables of Appendix A are used here to determine which segments are active, and what type of intersection occurs at their endpoints. If the intersection is with a panel edge, then the panel edge intersection of the $N-1$th offset is used to trim the tow area to only include fibre area within the design area, as shown in Figure 5.2. If the intersection is with a tow, then the intersection is projected onto the $N-1$th offset instead, by using a point projection algorithm. Then, the two borders are sampled by m points each for a $m \times 2$ mesh and the area is calculated with Equation (5.1).

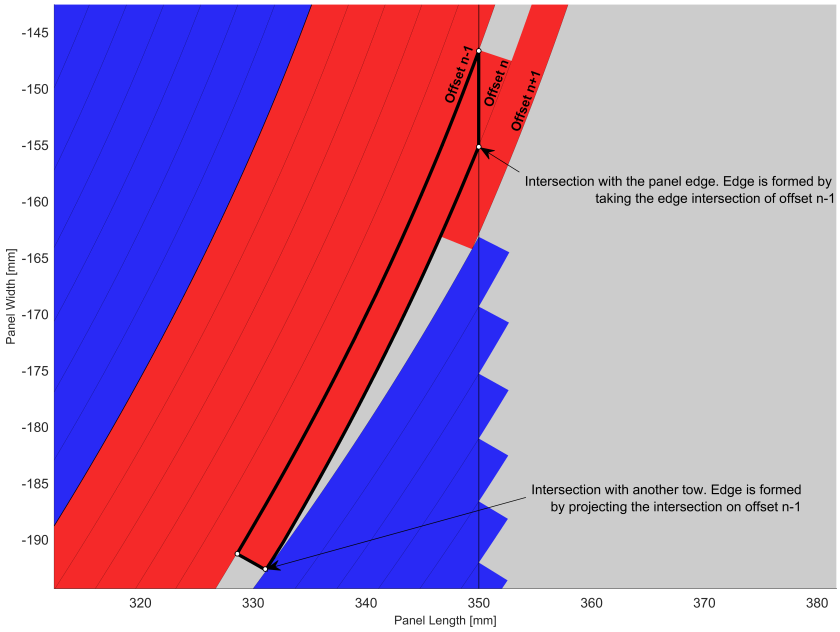


Figure 5.2: Determining the bounds of the area of a fibre.

This method also allows the calculation of the total fibre area placed on the mould, including the parts of the tows outside the design area. This can be used as an estimate of the required fibre material to produce the layer.

5.2 Fibre Angle Error

The Fibre Angle Error is the difference in fibre angle between a placed tow and the design value. Although the design angle is followed as closely as possible by course centre line during the progenitor course generation process, the actual fibres will most likely have some mismatch, as a point on a fibre shares the same orientation of a point on the centre line perpendicular to the course direction. This is illustrated in Figure 5.3. The effect is magnified the further the fibre is located from the course centre line. This effect has to be registered such that the designer can make adjustments to the design if a part of the design becomes non-compliant.

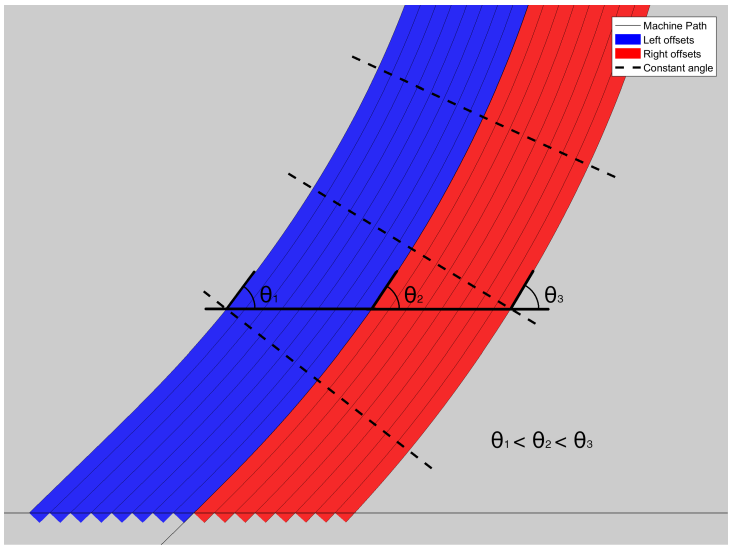
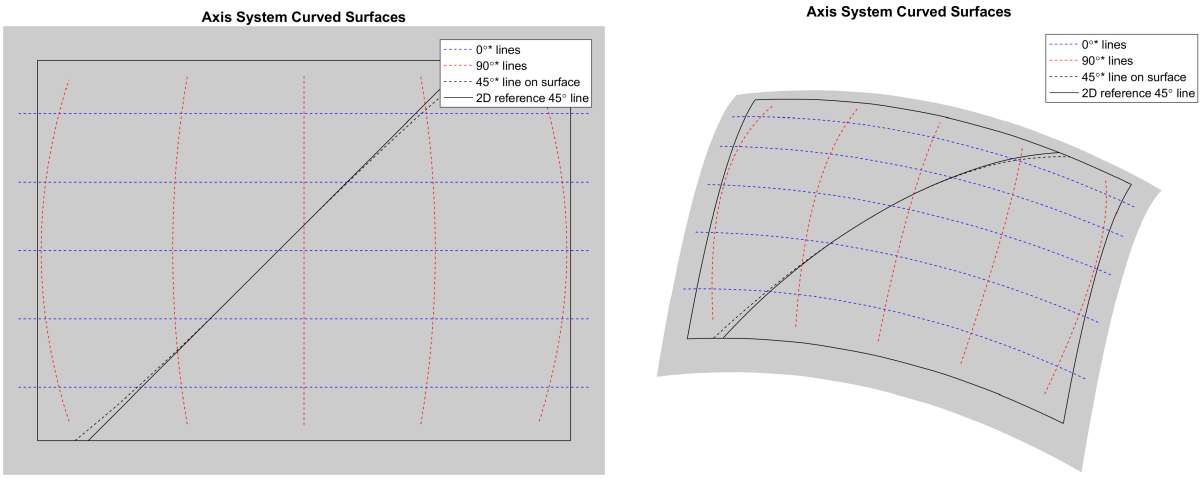


Figure 5.3: An illustration of fibre angle mismatch within a single course. The fibre angle is equal along a dashed line. Assuming that the course centre line exactly follows the design angle, there will still be fibre angle mismatches due to the geometry of the course.

Before the direction of a fibre can be assessed, an axis system needs to be defined for each point on the surface such that the fibre angle can be evaluated in the tangent plane of the surface (for 2D structures, normal trigonometry can be used). The axis system consists of a modified 0°* line, which is defined as the slope of the surface in x, the unit normal vector $\mathbf{n}$ as shown in Equation (4.31) and completed by a modified 90°* line, defined as the cross product between $\mathbf{n}$ and 0°*. The local fibre angle orientations are illustrated in Figure 5.4. Notice how the 90°* lines bend due to the doubly-curved shape of the mould.



(a) Top View

(b) Perspective View

Figure 5.4: Local fibre angle orientations for the curved surface, as well as a comparison between a 2D 45° line and a 45° line for the surface.

The direction of the fibre is obtained from the unit tangent vector $\mathbf{t}$, as shown in Equation (5.2). It is then projected onto unit vectors in 0°* and 90°* to get the components in-plane from which the fibre angle can be deduced.

$$\mathbf{t} = \frac{\mathbf{T}}{|\mathbf{T}|} = \frac{\dot{u}S_u + \dot{v}S_v}{|\dot{u}S_u + \dot{v}S_v|} \tag{5.2}$$

Then, the obtained fibre angle can be compared against the design value. In the case of (D)LVFA designs, the position of the evaluation point is sufficient to obtain the exact design value, from Equation (2.1) or Equation (4.2). For generic designs, where the input is a grid of fibre angles as seen in Figure 2.11, an interpolation step is required. The interpn.m function of MATLAB can be used to interpolate an N-D grid with various interpolation methods [57]. For this application, a linear interpolation method is used.

An example of an fibre angle error figure for a $<40,65,40>_y$ layer with a constant region is shown in Figure 5.5. The fibre angle mismatch along the width of the course is clearly visible, with the lowest errors occurring at the course centre lines and at the constant regions. However, the mismatch is not exactly symmetric, instead showing some oscillations near the inflection points of the course. This is a result of the fitting procedure as described in Appendix C. A closer examination of the fibre angle in Figure 5.6 reveals the same oscillation issue at the endpoints of the curve, where the fibre angle is set to continue with the last prescribed value (in this case 40°). Recommendations are made in section 7.2 to tackle this effect.

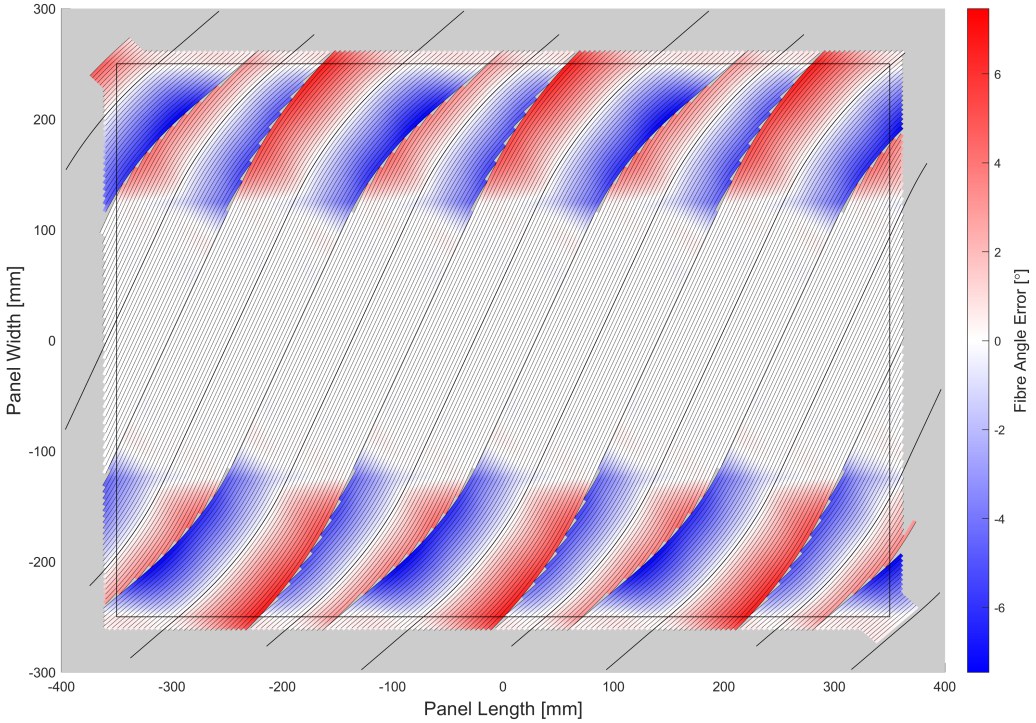


Figure 5.5: Fibre angle error illustration of a $<40,65,40>_y$ with a constant region.

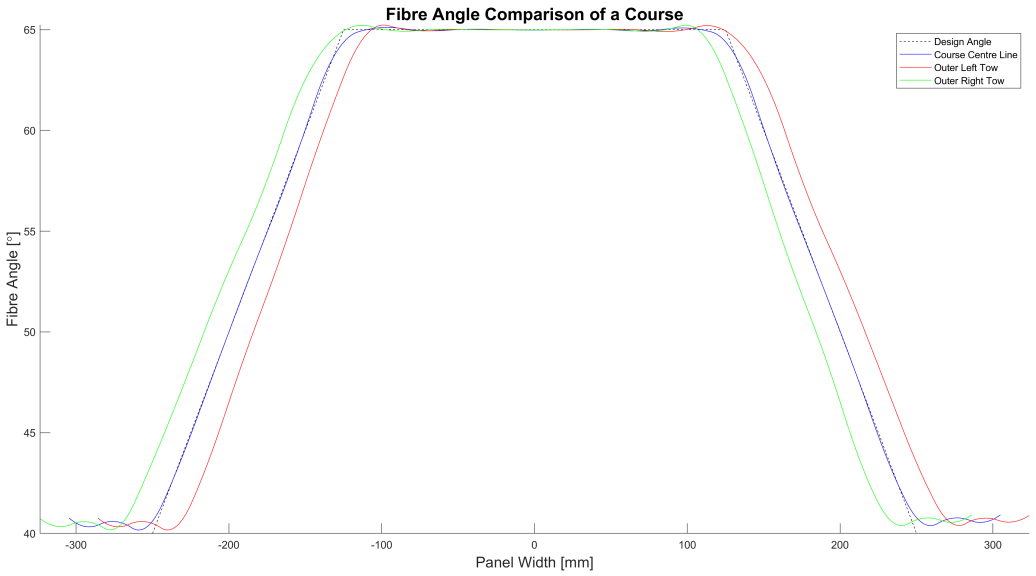


Figure 5.6: Fibre angle comparison for the fifth course of the $<40,65,40>_y$ layer from Figure 5.5 and the design angle. While the course centre line generally shows good adherence to the design angle, it struggles with the inflection points and the ends of the curve.

5.3 Continuous Tow Percentage and Number of Cuts

The final fitness criteria is for registering the occurrence of cuts and non-continuous tows within the panel. Cuts and tow restarts/adds affect the production speed, as the AFP roller head has to slow down while trimming or initiating the tow. Non-continuous tows are also a cause of load discontinuities and stress concentrations within the panel [58].

The intersection and segment tables, as explained in Appendix A, are used to determine cuts and non-continuities. For each active segment / tow within the panel dimensions, the two outer intersections are examined. If any of these are not an panel edge intersection, then it is counted as a cut and the tow is non-continuous. Every non-continuous tow is considered 'cut' for the purposes of this section, even if the tow was added mid-course. An illustration is shown in Figure 5.7. It should be noted there is a significant difference in the tow continuity per course near the corners, compared to the longer courses in the centre, which should be taken into account when viewing this number.

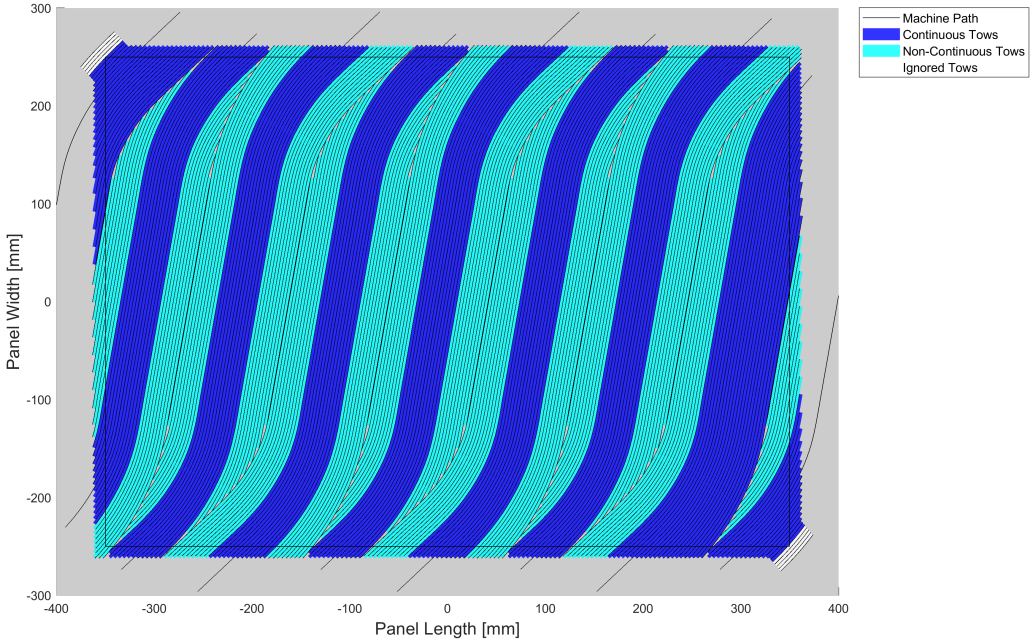


Figure 5.7: Continuous tow counting for a given layer. Tows that are cut anywhere within the layer dimensions are counted as 'non-continuous'. Tows that are placed for the production reasons, but are otherwise outside the dimensions of the layer are ignored.

Chapter 6

Results

In this chapter, the results from the developed algorithm are displayed. In section 6.1, the algorithm is demonstrated on different types of surfaces and fibre angle distributions, and a quantitative comparison of several parameters on the fitness criteria of chapter 5 is made in section 6.2.

The parameters of the layers are not based on any actual designs, but are merely chosen for demonstrative purposes. Specifically, the curvature constraint has been ignored for all but one of the layers, as a compromise was made for the generation of images in the results section. In order to not exceed the curvature constraint, the layers would have either needed to be steered a lot less or the size of the layer would have needed to be significantly increased. Both cases were deemed undesirable. Limiting the steering would have lead to fewer course overlaps and fewer points of interest on the layer and a larger mould in combination with the small tows would obscure some of the results without additional close-ups.

6.1 Algorithm Demonstration for Typical Use Cases

The main goal of thesis has been to extend the fibre steering method to include doubly-curved shells, and to accept a range of fibre angle distributions. To this end, three layers were chosen to demonstrate the algorithm's capabilities. A Discrete Linearly Varying Fibre Angle (DLVFA) layer on a doubly-curved plate is shown in subsection 6.1.1 and a generic fibre angle distribution layer is given in subsection 6.1.2. Lastly, a curvature-constrained layer is shown in subsection 6.1.3 to give a better idea of the required scale of fibre steering.

6.1.1 Doubly-Curved Plate

This example is a $<$-45,-75,-45$>_y$ layer with a constant region on a doubly-curved plate and demonstrates the ability of the algorithm to generate layers on curved surfaces, shown isometrically in Figure 6.1.

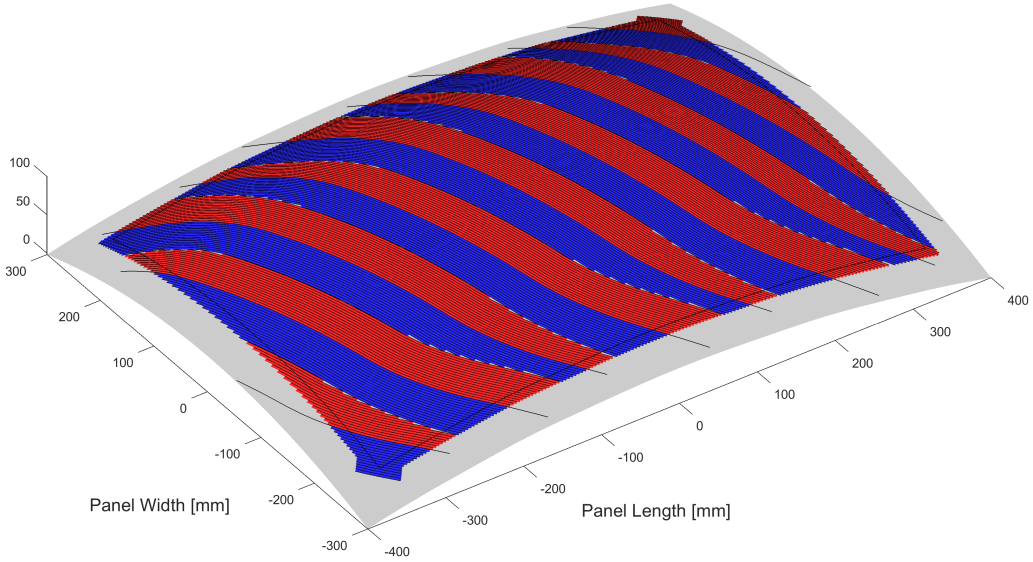


Figure 6.1: Isometric view curved plate.

The top view in Figure 6.2 shows that all tows are placed in the constant region of the layer. This means that the offsetting procedure works for curved plates. Likewise, the intersection detection and cutting algorithm also show no errors, with gaps only occurring in the overlap areas between courses. This illustrates the value of using a parametric approach when modelling generic curved shells.

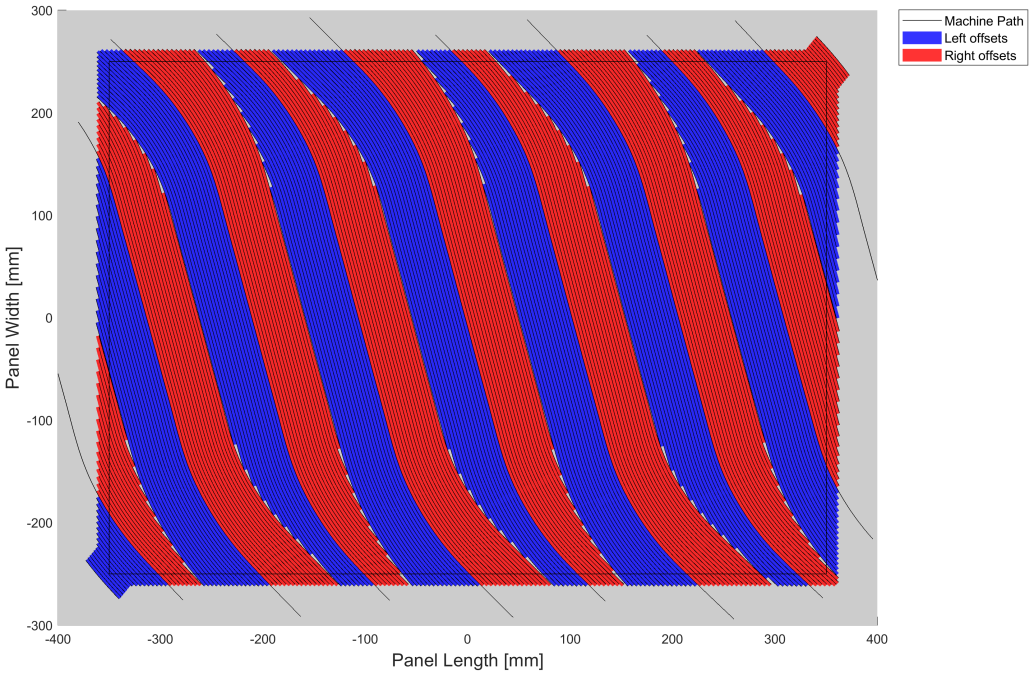


Figure 6.2: Top view curved plate.

The fibre angle error results in Figure 6.3 are noteworthy as well. It can be seen that the error is larger at the outer edges of the plate, and that the lowest errors no longer occur at the centre line of a progenitor course, in contrast to Figure 5.5. This is a result of the single-progenitor approach from subsection 4.2.2 in combination with the curved surface. Only the progenitor curve shows reasonable compliance with what is expected, with the lowest errors at the course centre line and a mostly symmetrical fibre angle error distribution. This is a downside of using the single-progenitor approach and somewhat limits its use for doubly-curved moulds.

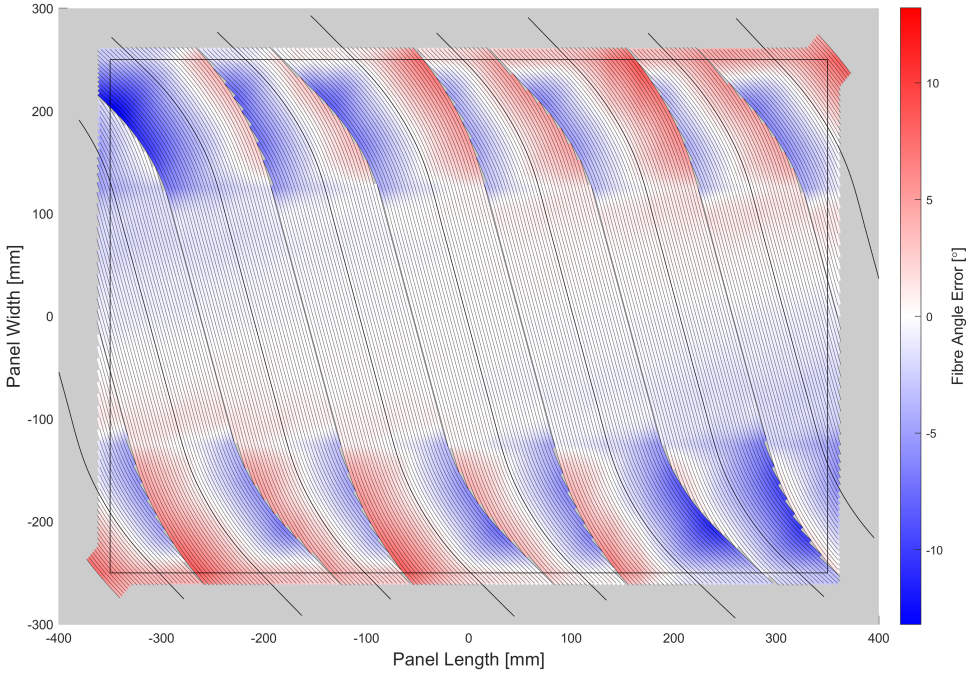


Figure 6.3: Fibre angle error of the curved plate.

6.1.2 Generic Fibre Path

This example demonstrates the algorithm on a generic fibre angle distribution on a flat plate. The distribution was taken from an example of *Wurpel* (2015), although the chosen panel dimensions and course properties are different. Paths were fully generated by the streamline code instead of the single-progenitor approach for obvious reasons. The fibre angle distribution and the resulting course centre lines from the streamline code are illustrated in Figure 6.4 and the ensuing layer is shown in Figure 6.5.

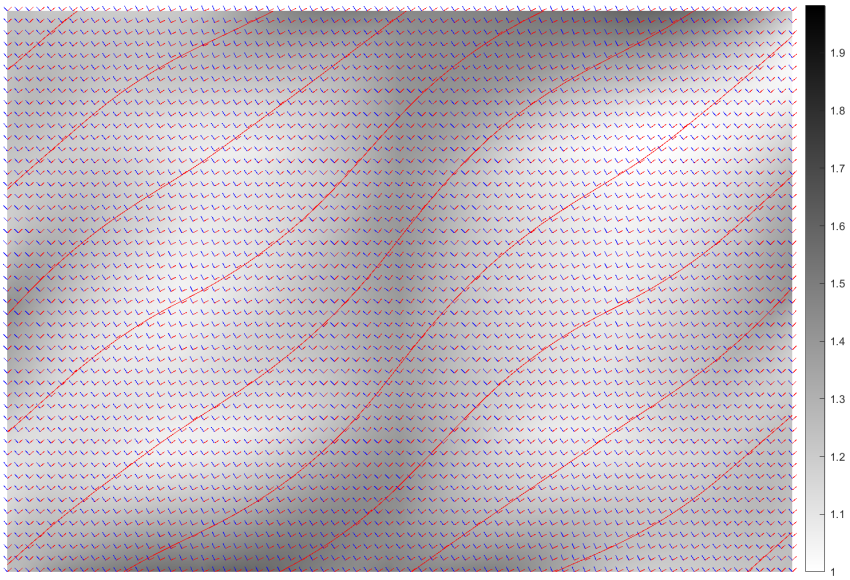


Figure 6.4: Generic fibre angle distribution and resulting course centre lines, as given by the streamline code. The tangent and normal directions of the fibre angle inputs are given by the red and blue local axis. The bar graph is an indication of how 'crowded' an area on the layer is.

Gaps between courses are much more prevalent in this layer compared to the doubly-curved plate of

subsection 6.1.1. This is partially because of the streamline code, which seems to have difficulties with correctly spacing the courses. A peculiar detail was that the streamline code sometimes achieved better results in terms of coverage were when the offset distance between courses was artificially increased above the course width, effectively pretending that the courses were wider than they actually are. It is possible that the streamline code views the offset distance as the upper limit of the distance between two courses. However, the nature of generic fibre angle distributions make it difficult to prevent these gaps completely, regardless of the method used.

As it is not possible to do anti-straightening cuts in this layer, the alternating cutting method was used instead. In this case, it means that the odd-numbered courses are cut and the even-numbered courses are ignored. As a result, there are areas where only a few tows remain active for an odd-numbered course. This is undesirable, as the fibre angle of tows nearest to the course centre lines are the closest to the desired fibre angle distribution. In contrast, as can be seen in the fibre angle error illustration of Figure 6.6, the largest errors occur almost exclusively on the outer edges of the even-numbered courses. This begs the question if the current cutting techniques are appropriate enough.

In section 4.8 on tow cutting, the concept of 'zipper-cuts' was discarded because of the limited benefits reported by *Mishra* (2017) compared to single-sided cutting and the high risk of fibre straightening. Although fibre straightening cuts are practically unavoidable for this fibre angle distribution, the result from Figure 6.6 may yet prompt an implementation of a form of 'zipper' cutting for generic fibre angle distributions, as the fibre angle compliance could be considered more critical for such layers. On the other hand, if fibre straightening is deemed a critical manufacturing flaw, then this should also be reflected in the fibre angle distributions themselves, to allow for some form of anti-straightening cuts.

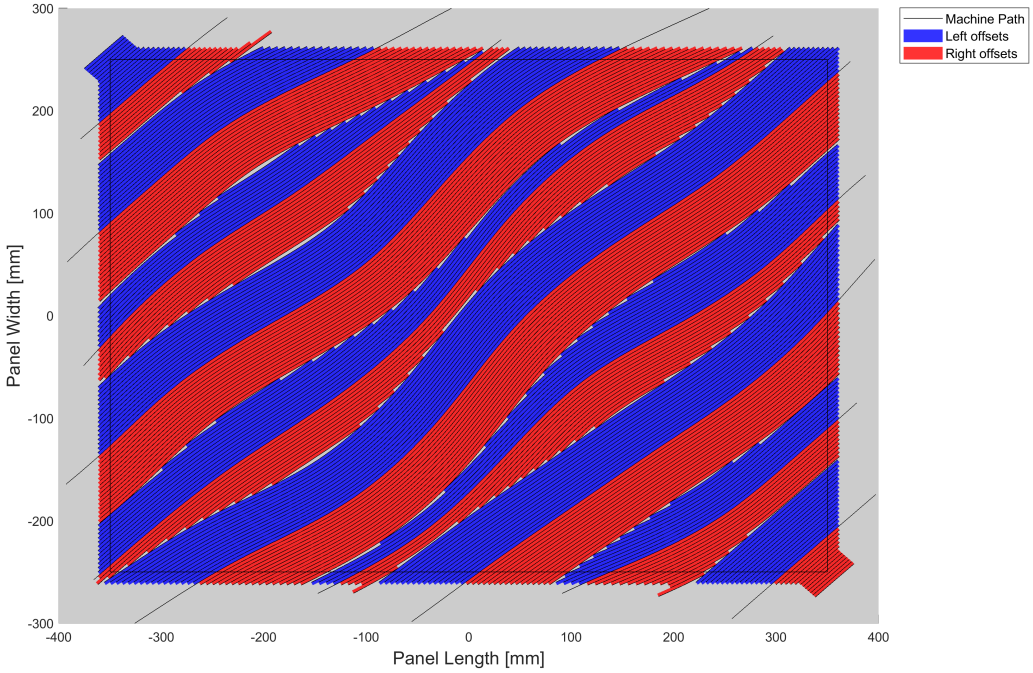


Figure 6.5: Generic fibre angle distribution from *Wurpel* (2015).

Figure 6.6 also displays more frequent fibre angle error oscillations compared to an example image such as Figure 5.5. The inflection points of the courses make it difficult to obtain the lowest fibre angle errors at the course centre line due to the current fitting algorithm, with courses with fewer inflection points performing better. Large errors also occur where the local fibre angle is no longer a function of the fibre angle distribution, but of the extrapolated endpoints of a course. This predominantly occurs around the corners of the layer.

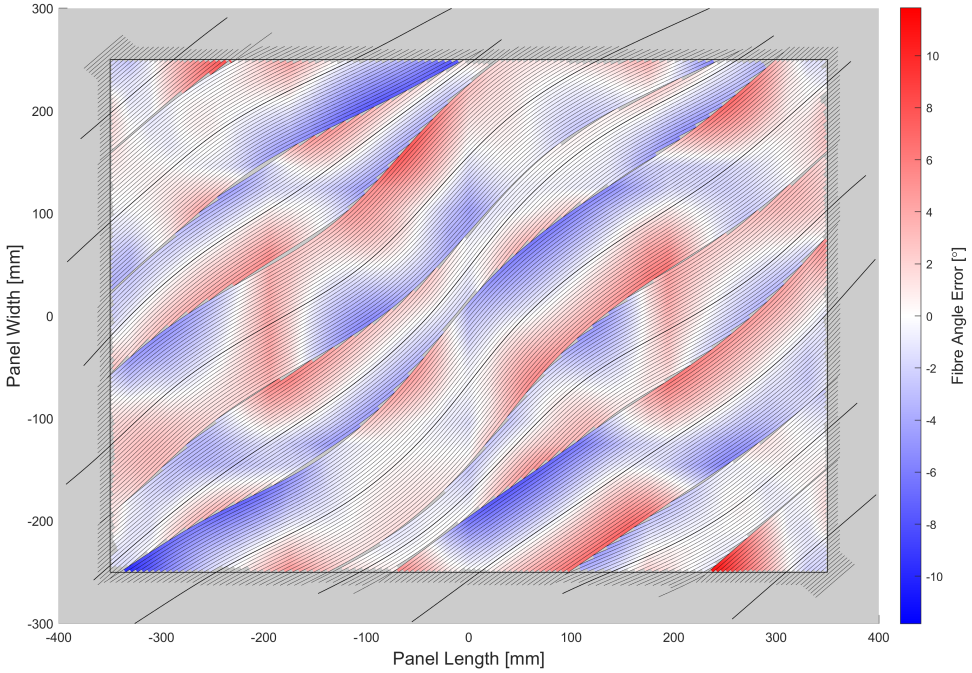


Figure 6.6: Fibre angle error of the generic fibre layer.

6.1.3 Curvature-Constrained Layer

The final layer in this demonstration section is one that is constructed while adhering to the curvature constraint. Due to the scale of the layer, individual tows are more difficult to discern, which is why this setup was not chosen for every layer.

The mould is created from a half-cylinder with a radius of $0.75m$, and was constructed with NURBS by extruding a semi-circle. For drawing purposes, the drawn mould was then trimmed to be slightly larger than the size of the layer. The layer consists of steered $<48,75,48>_y$ courses from a single-progenitor approach, with a constant region between $3/10^{th}$ and $7/10^{th}$ of the width of the mould. The top view of the layer is shown in Figure 6.7, and a rotated isometric view is shown in Figure 6.8.

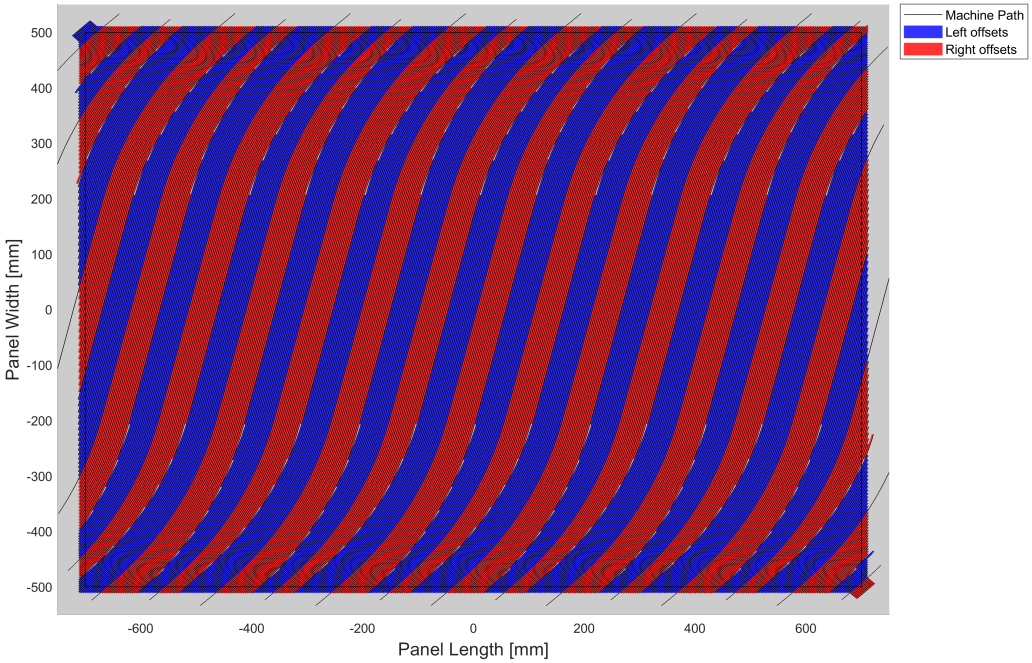


Figure 6.7: Top view of the curvature-constrained layer.

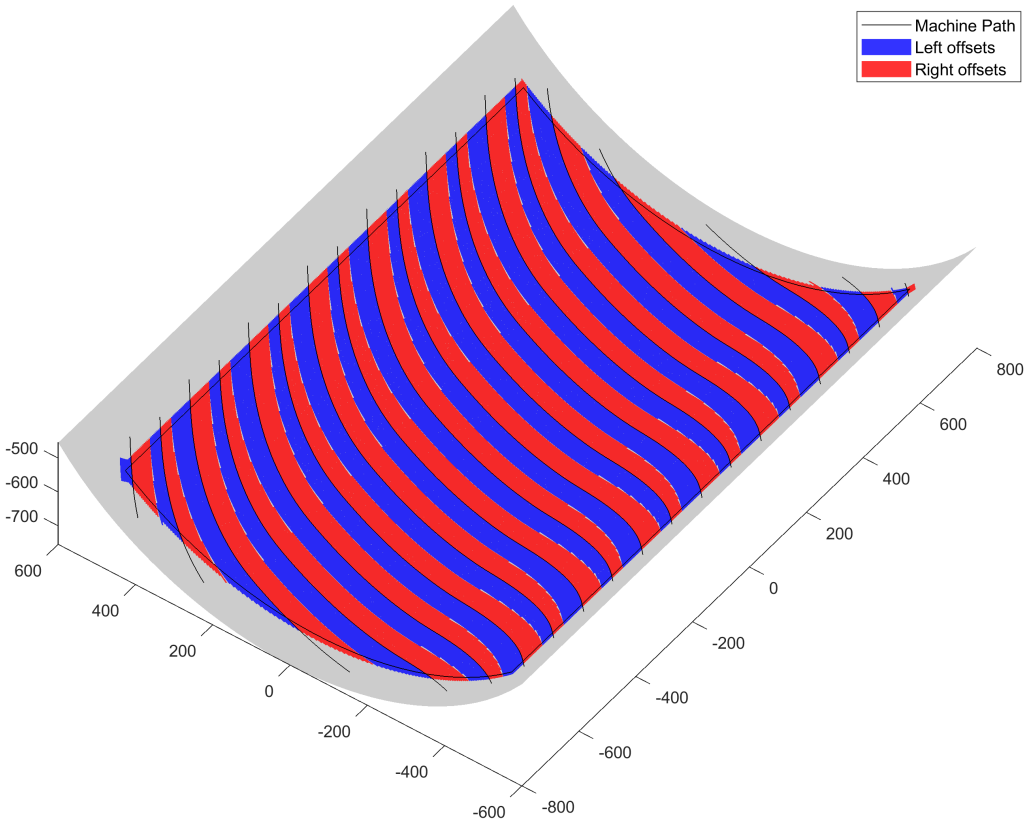


Figure 6.8: Isometric view of the curvature-constrained layer.

The curvature constraint was taken to be at most $\pm 1/635mm$ or $\pm 1.57 \times 10^{-3} mm^{-1}$, based on the figures from Table 2.1, and is shown for all courses in Figure 6.9. It should be noted that there are slight variations in the curvature profiles of the courses, despite all courses having the same fibre angles. This

is due to some of the courses having different lengths, which meant that during fitting, fewer data points were available. This led to small differences in the curvature.

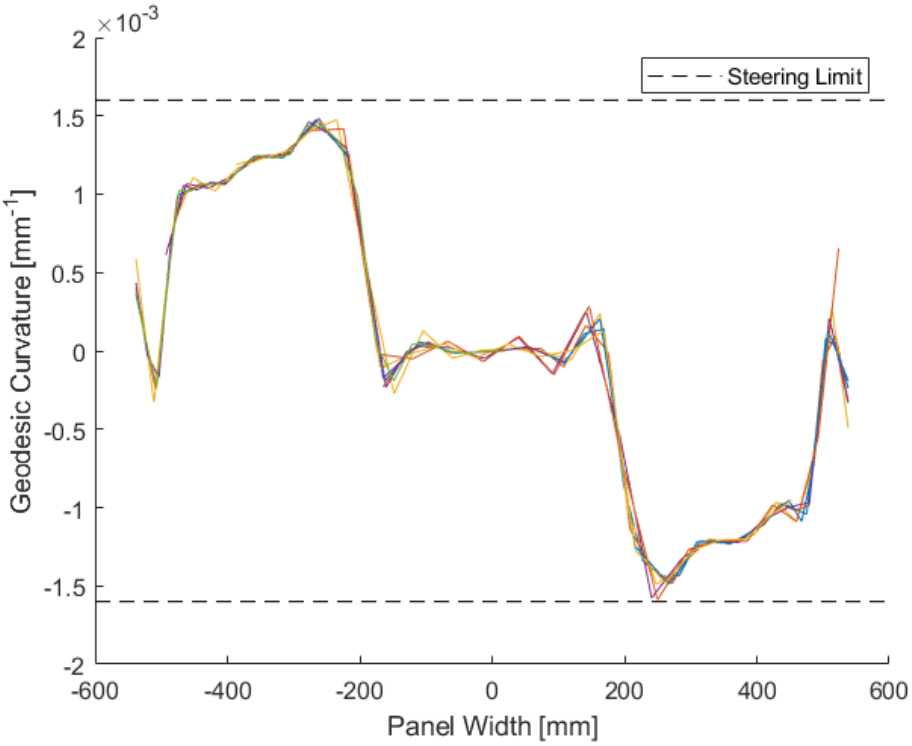


Figure 6.9: Curvature values of the course centre lines of the layer.

Finally, the fibre angle error of this layer should be mentioned. As Figure 6.10 shows, the curvature-constrained layer has a maximum error of only around 2.5°. This is significantly lower than the previous two layers in this section, and it raises the question if the two 'demonstration' layers in this section provide an distorted view of the expected magnitude of the fibre angle error. It must therefore be reiterated that all but this layer are solely made for the demonstration of the new algorithm.

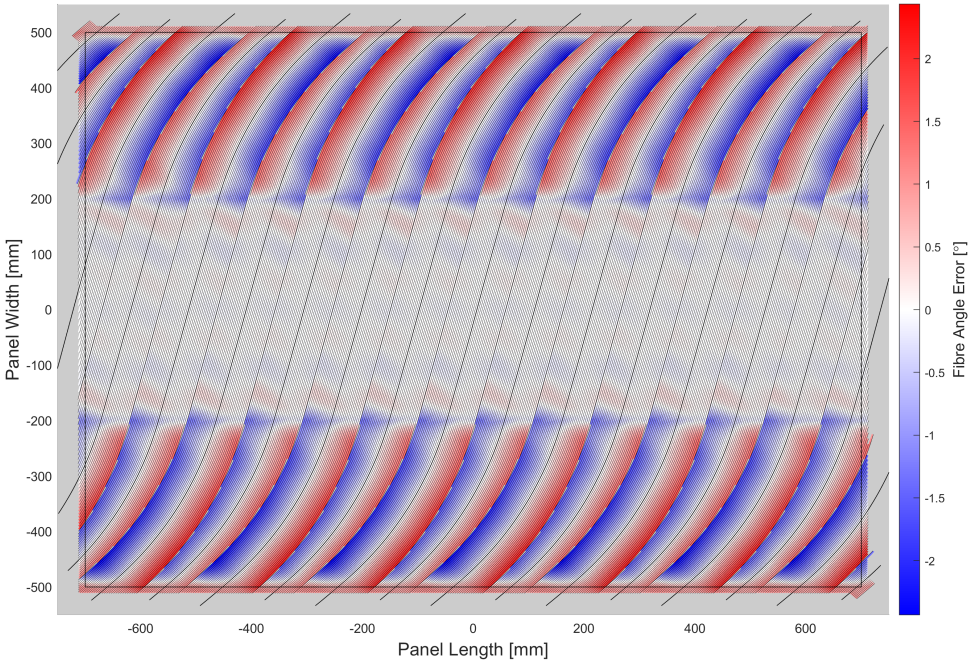


Figure 6.10: Fibre angle error of the curvature-constrained layer.

6.2 Results Comparison

In this section, the effects of several parameters on the fitness criteria from chapter 5 are investigated by comparing the results of a single fibre layer. A 'baseline' design is presented in subsection 6.2.1 against which the other designs are compared. The results are summarized in a table in subsection 6.2.9. Unless otherwise stated in their respective sections, each design consists of the following specifications:

- Flat plate with a 700 x 500 mm design area.
- Courses consist of 32 x 1/8 inch tows.
- A DLVFA fibre angle distribution of $<45,75,45>_y$ with a constant region between $1/4^{th}$ and $3/4^{th}$ of the width of the plate.
- 50 mm minimum cut length for tows.
- 10 mm margin around the design area.
- Anti-straightening cuts as described in subsection 4.8.2.

It should be noted that the values for the minimum cut length and the margin were chosen solely for demonstration purposes, and are not based on any manufacturing specifications.

6.2.1 Baseline Design

The baseline design follows all the specifications mentioned at the start of this section. They were selected because it seemed the most conventional and manufacturable. Gaps are limited to the overlap areas between courses, with the number of courses themselves also being kept to a minimum for the same reason. As a result, the fibre angle error is quite high at the outer ends of each course. It will be up to testing and other future work to determine the feasibility of this kind of design.

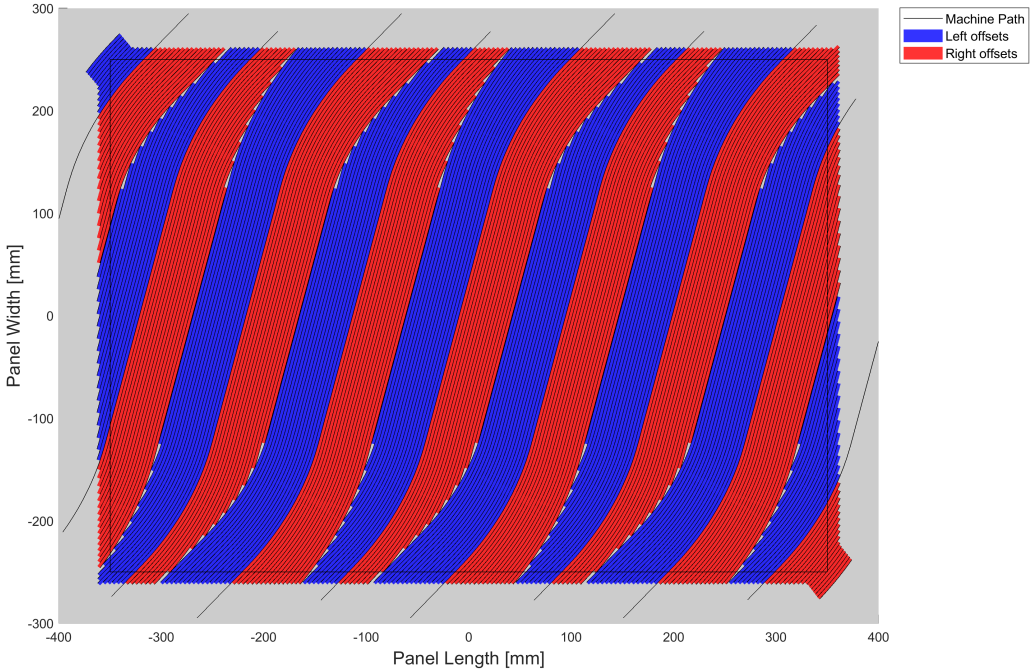


Figure 6.11: Baseline design.

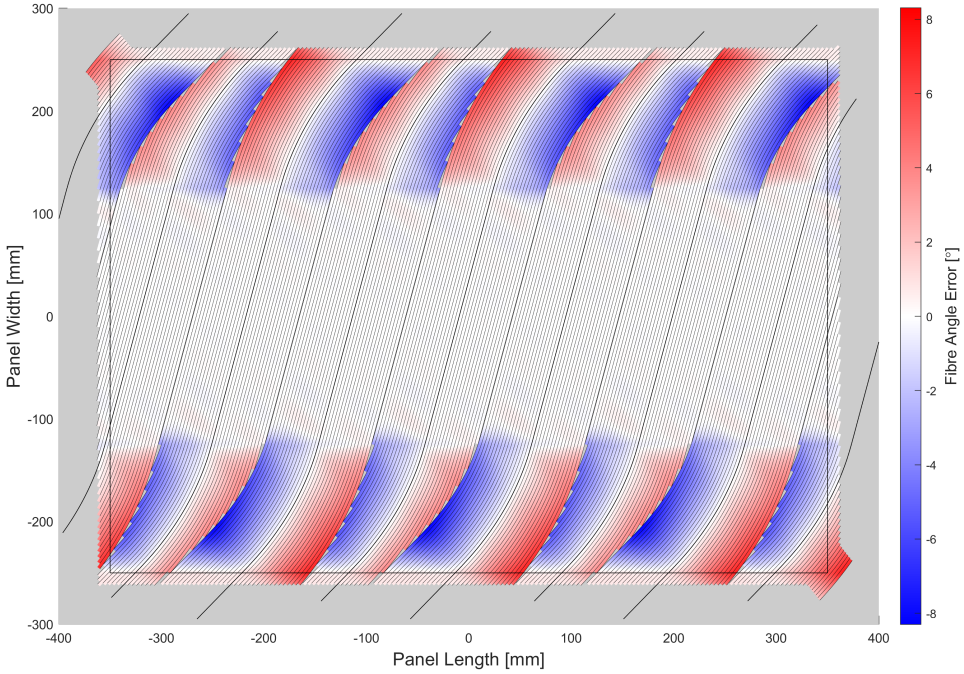


Figure 6.12: Fibre angle error of the baseline design.

6.2.2 Effect of Number of Tows

This design has half the number of tows per course and therefore (roughly) double the number of courses compared to the baseline. This has two distinct effects on the results. Firstly, by halving the course width, the maximum fibre angle error has been significantly decreased. Additionally, the smaller course width would also allow for higher amounts of steering, as the minimum turning radius from Table 2.1 is a function of both tow width and course width. Secondly, due to the extra courses, there are also more

areas where courses overlap, resulting in more cuts and therefore gaps. Effectively, halving the course width trades a lower coverage for a lower fibre angle error and more steering capability.

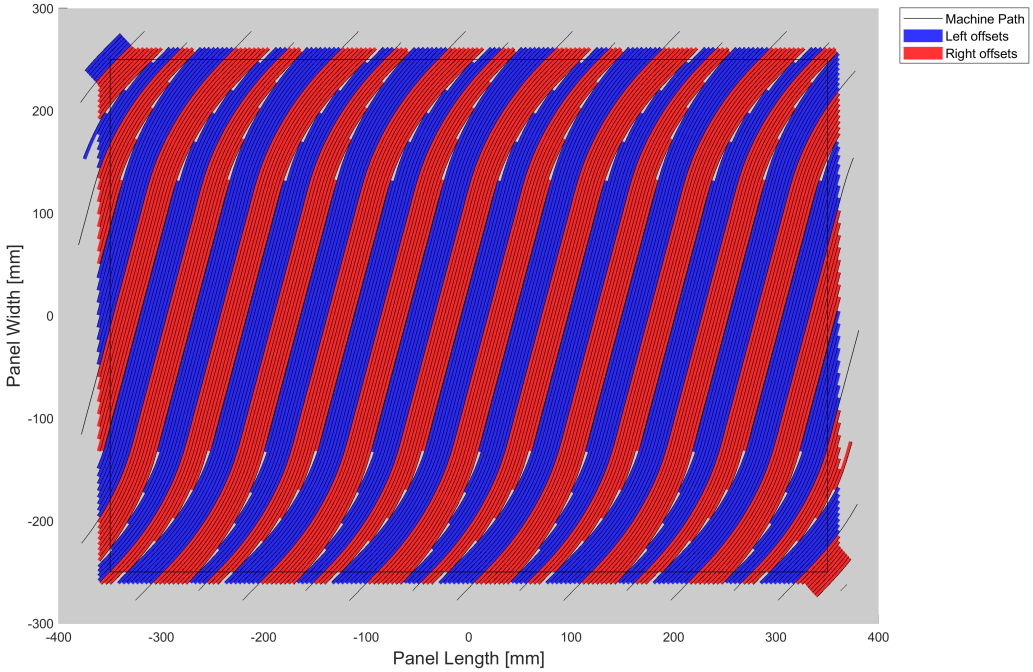


Figure 6.13: Halved course width design.

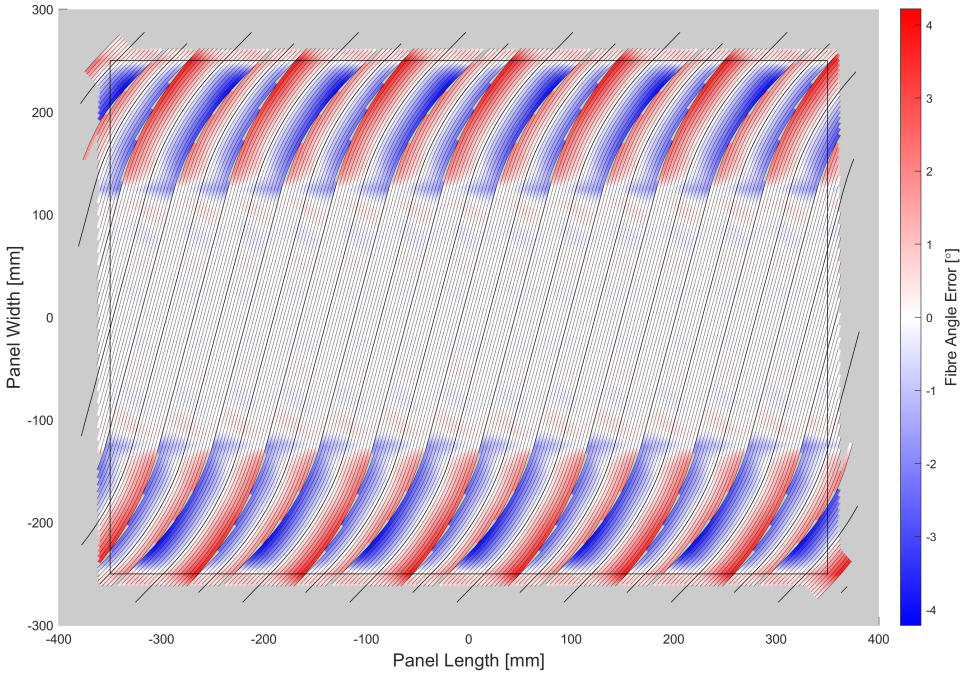


Figure 6.14: Fibre angle error of the halved course width design.

6.2.3 Effect of Tow Width

In this design, the number of courses is kept equal but the tows are twice the width (16 x 1/4 inch instead of 32 x 1/8 inch). Interestingly enough, when comparing the values of this design with the previous two

in Table 6.1, it has the worst parts of both. The gaps between courses are larger due the increased tow width, while the fibre angle error is identical to the baseline design. Lastly, it should be noted that 1/4 inch tows tend to kink at higher minimum turning radii than 1/8 inch tows.

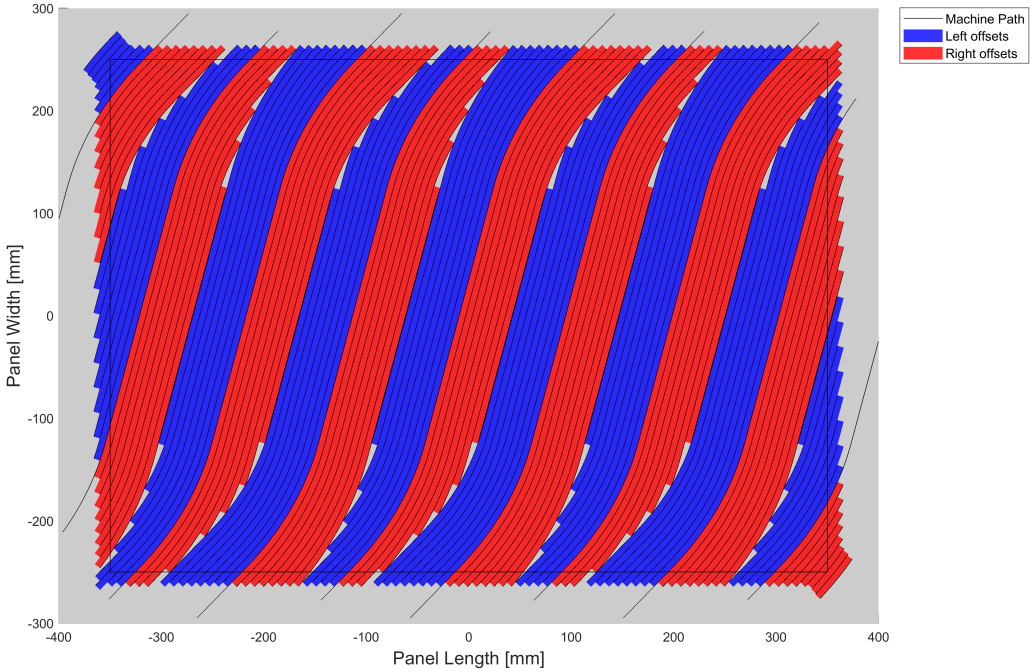


Figure 6.15: Doubled tow width design.

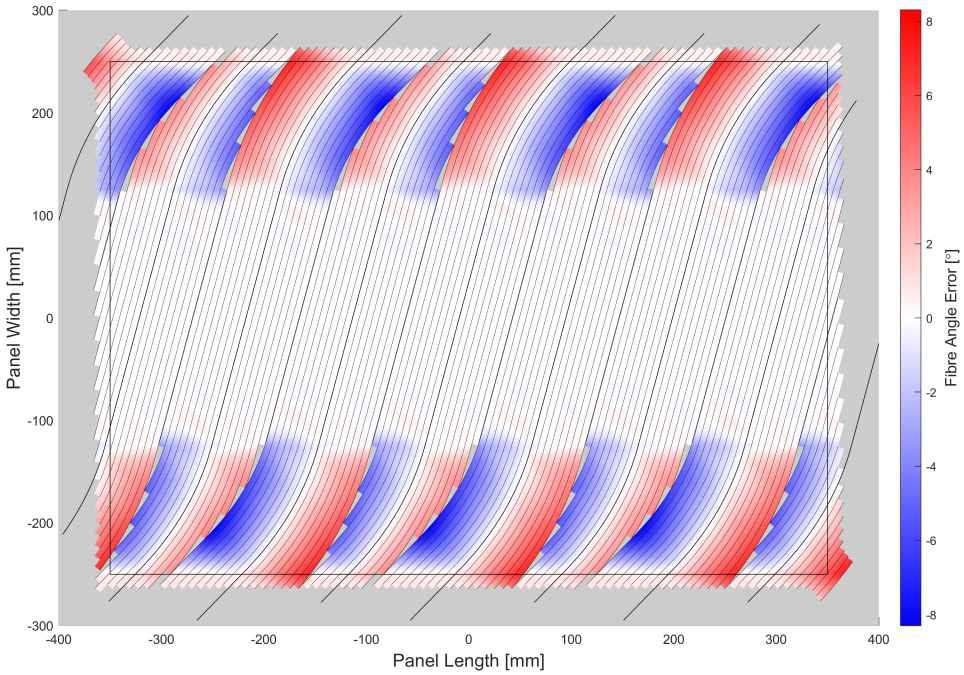


Figure 6.16: Fibre angle error of the doubled tow width design.

6.2.4 Comparison with Streamline Path Result

The streamline path design is a result from taking the output of the streamline code directly, as opposed to the altered approach from subsection 4.2.1 for the baseline design. Several differences can be noted.

The main difference between the streamline path result and the baseline is how the paths were constructed. For the streamline result, the progenitor paths are only defined within the design area of the mould, whereas the baseline design uses one progenitor curve, which is copied along the mould as described in subsection 4.2.1. In both cases, the progenitor curves are then extended from the endpoints to ensure that the outer tows also reach to the panel edges. However, as can be seen on the vertical edges of the panel in Figure 6.17, if the path is extended in a region where the course should be steered, it instead leaves a gap between two courses. Furthermore, the lack of steering may also lead to higher fibre angle errors, as can be noted in the bottom-left corner of Figure 6.18.

In addition, the incorrect spacing between the streamline paths, as already illustrated in Figure 4.5, also results in extra gaps between courses compared to the baseline. These are at most the width of one tow between two courses.

More interestingly though, the spacing also has an effect on the cutting algorithm, as can be seen in the less 'staggered' tow initiations in the top half of Figure 6.17. This is due to the cutting algorithm assuming that the outer edges of the course (i.e. the outer tows) act as the cutting barriers between courses. For the anti-straightening cutting as used here, the courses are trimmed alternately at the top and bottom by these outer edges, with the lower edges taking precedence over the top edges. This is not a flaw of the streamline path design as much as a limitation of the developed cutting algorithm.

Lastly, a note should be made of the fitting of the shorter paths from the streamline path design. Shorter paths, which consist of fewer data points, are more difficult to fit as the number of data points affects the number of control points that can be created. This can be mitigated for the most part by using a finer mesh for the streamline code, resulting in more data points, but this comes at the cost of severely increased run times.

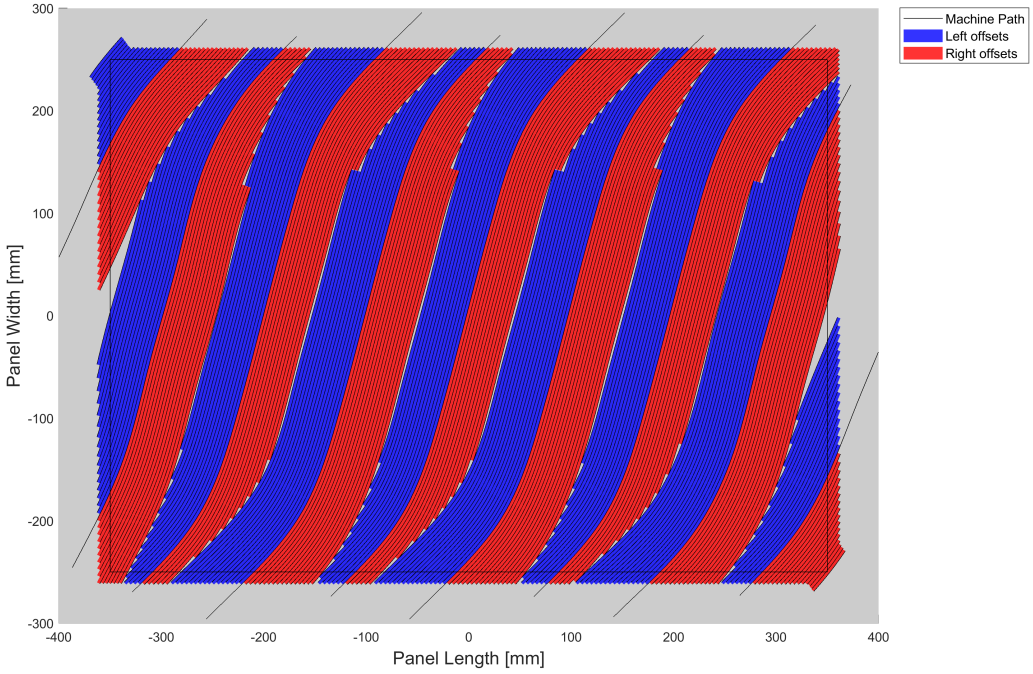


Figure 6.17: Streamline path design.

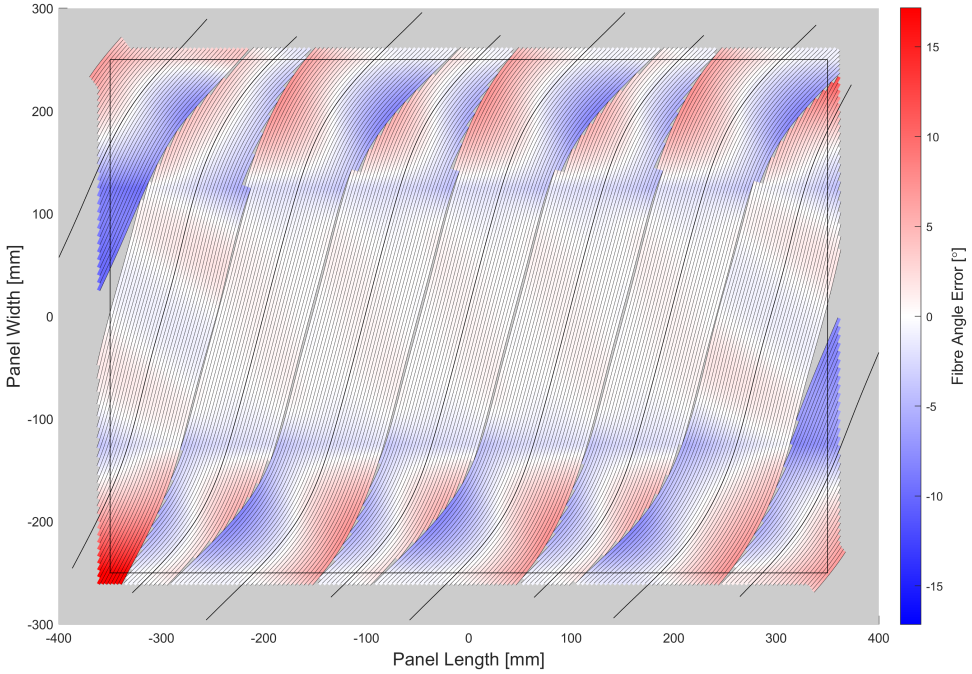


Figure 6.18: Fibre angle error of the streamline path design.

6.2.5 Cutting Comparison

In this design, the overlapping tows are only cut from the odd course numbers instead of with the 'anti-straightening' strategy. This cutting strategy is more widely applicable as it is less dependent on the fibre angle distribution to generate a valid layer. Furthermore, the number of unaffected, continuous tows is the highest for all designs. However, the number of cuts within the panel remains the same compared to the baseline design, which means that there could be an equal number of load discontinuities in addition to the higher chance of tow straightening.

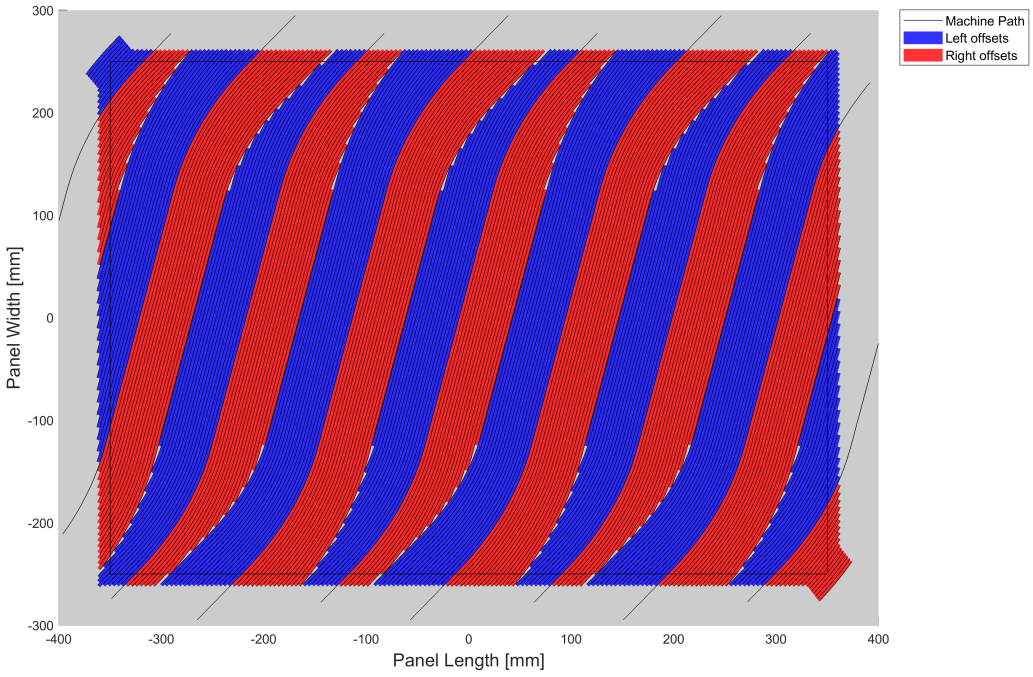


Figure 6.19: Alternated cutting design.

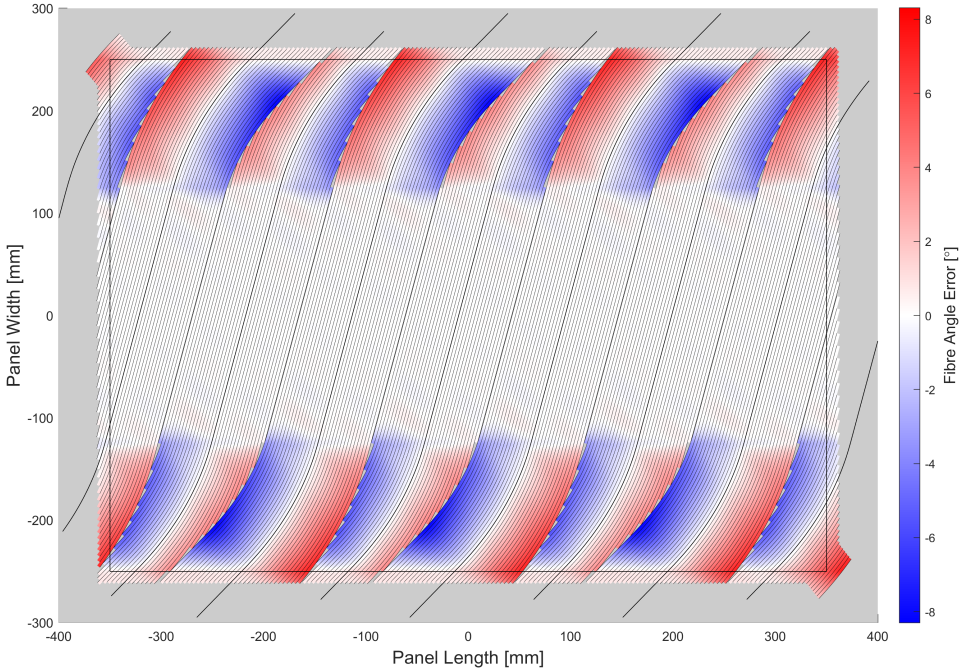


Figure 6.20: Fibre angle error of the alternated cutting design.

6.2.6 Constant Region Comparison

Compared to the baseline design, this layer lacks a region with a constant fibre angle. Therefore, it should be noted that the resulting panel should be intrinsically different from the previous ones, and that this comparison is made mainly to qualitatively assess the effects of the constant region.

First of all, the lack of a constant region results in more gaps between courses as there is a larger region where the courses are steered (and therefore overlap). However, the fibre angle error is considerably lower, as the steering is more gradual along a longer distance. It is also noteworthy that fibre angle error at the centre of the width very clearly illustrates the oscillations noted in Appendix C, as this is the location of the inflection point for the path.

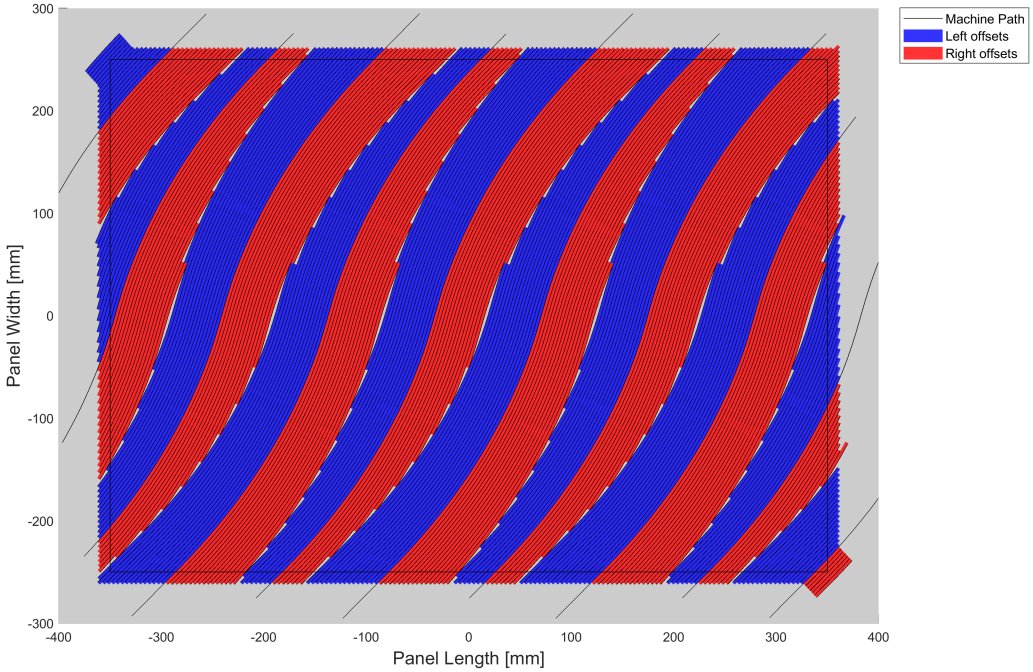


Figure 6.21: No constant region design.

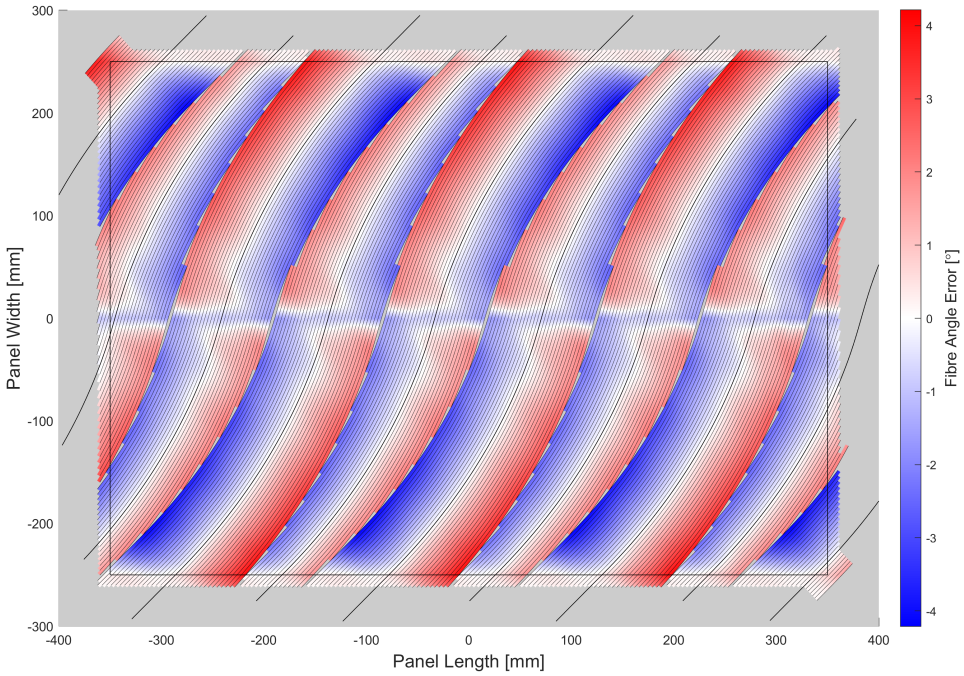


Figure 6.22: Fibre angle error of the no constant region design.

6.2.7 Inverted Fibre Angle Distribution

For this layer, the two fibre angles from the baseline design have been reversed, i.e. a fibre angle distribution of $<75,45,75>_y$ with a constant region between 1/4th and 3/4th of the width of the plate. For these kinds of designs, an extra step is required in determining the offset distance between courses. This has to do with the effective width of a tow in the direction of shift, i.e. the width of a tow along the length of the panel. Because there should be no gaps in the constant region, the offset distance is raised

from the smallest effective width of the course, i.e. at $\theta = 75°$ to the nearest whole number of tows at $\theta = 45°$.

Furthermore, this design also illustrates how 'regular' anti-straightening cuts are not always possible. As the overlaps occur in the middle of the course, it is not attainable to only place new tows within the panel dimensions.

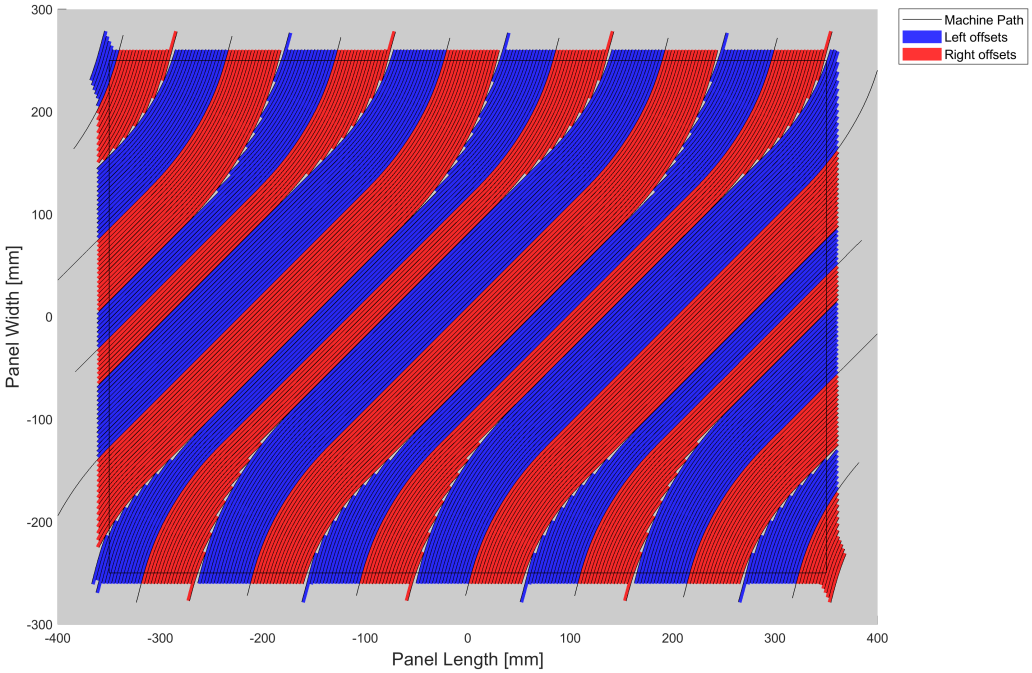


Figure 6.23: $<75,45,75>_y$ layer.

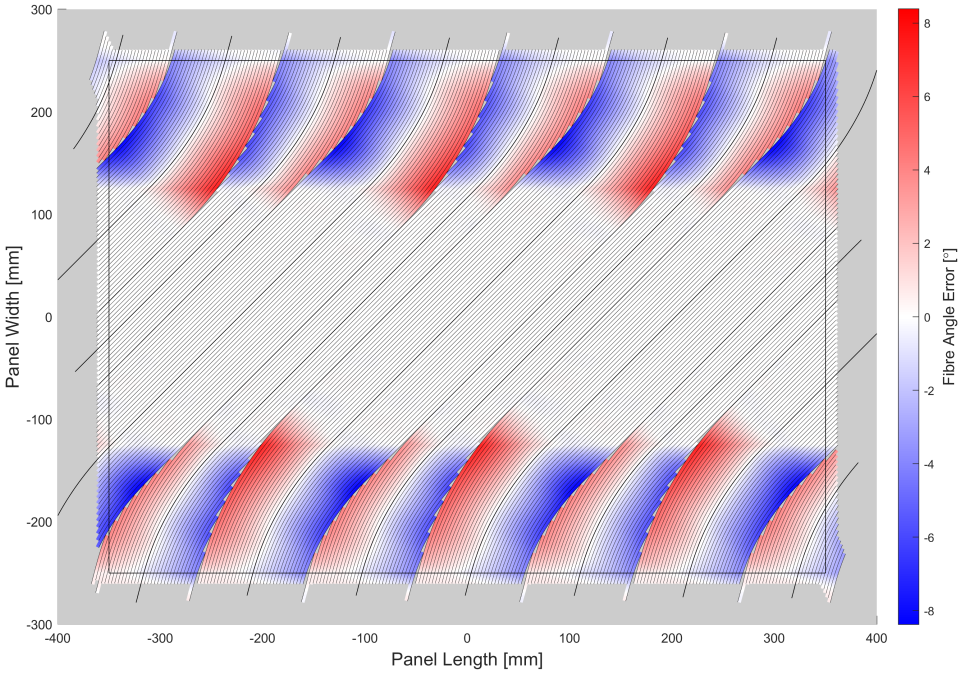


Figure 6.24: Fibre angle error of the $<75,45,75>_y$ layer.

6.2.8 Filler Course Layer

Lastly, a 'filler course' anti-straightening layer as described in subsection 4.8.2 is demonstrated for a $<75,45,75>_y$ fibre angle distribution. Contrary to the previous $<75,45,75>_y$ layer, the cutting occurs at the end of the panel again, enabling the anti-straightening cuts.

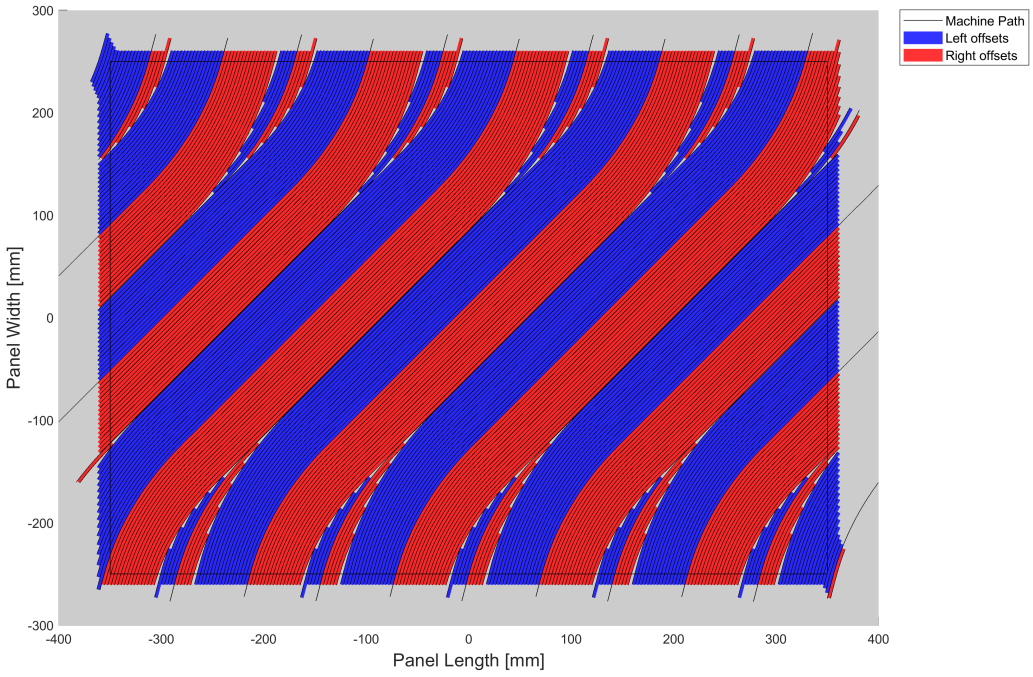


Figure 6.25: $<75,45,75>_y$ layer with filler courses.

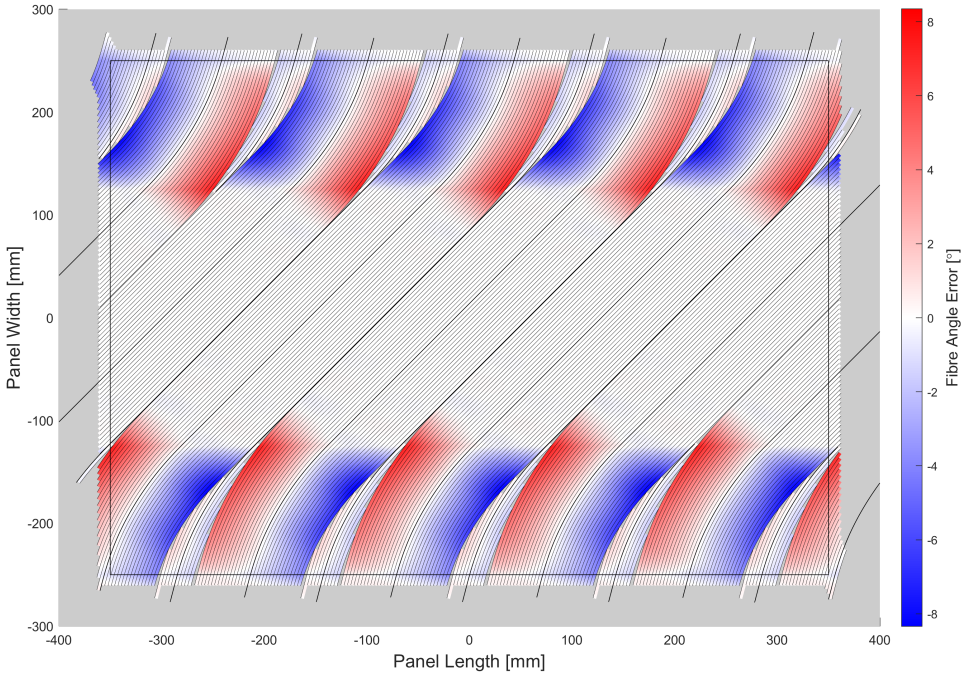


Figure 6.26: Fibre angle error of the $<75,45,75>_y$ layer with filler courses.

6.2.9 Results Table

The results of this section have been summarized in Table 6.1 and allows for a quantitative comparison between designs (where applicable), based on the fitness criteria defined in chapter 5.

Unsurprisingly, the best results in terms of coverage were achieved when the courses could be neatly placed adjacent to each other, while keeping the size and number of cutting regions to a minimum. Increasing the number of courses (Halved Course Width) or doubling the tow width (Doubled Tow Width) negatively affected the coverage by respectively increasing the number of cutting regions or increasing the size of the gaps between courses. Likewise, both the Streamline Path layer and the No Constant Region layer had lower coverage than the baseline (in part) due to a larger portion along the course length being subject to cutting. In contrast, the Inverted Fibre Angle layer has very similar coverage to the baseline despite only sharing the region where gaps occurred.

Three layers stand out in terms of the fibre angle error. The Halved Course Width has a lower fibre angle error than most, as the smaller course width incurs lower errors while being steered. The No Constant Region layer has a lower fibre angle error due to the lower steering rate. Both design choices had a negative effect on the coverage though, which makes it an interesting trade-off for design considerations. On the other side of the spectrum, the Streamline Path layer had the largest fibre angle error due the loss of steering of the fibre paths outside the dimensions of the layer.

The Continuous Tow Percentage (CTP) is largely the same for most layers, with outliers in the two layers that used the alternating cutting strategy instead of the anti-straightening cuts. However, the number of cuts is almost equal compared to most other designs. This is due to the cutting occurring multiple times on the same tow number. One could consider these two segments to be two individual tows, which would have lowered the CTP, but this convention was used instead.

Table 6.1: Fitness criteria results of the presented designs. The Maximum Fibre Angle Error (MFAE) and Continuous Tow Percentage (CTP) have been abbreviated for formatting reasons.

Design	Fibre Area	Uncovered Area	Coverage	MFAE	CTP	Cuts
Baseline	3.46e5 mm^2	4.0e3 mm^2	98.9 %	8.31°	58.9 %	113
Halved Course Width	3.42e5 mm^2	7.7e3 mm^2	97.8 %	4.22°	58.0 %	114
Doubled Tow Width	3.43e5 mm^2	7.4e3 mm^2	97.9 %	8.31°	56.9 %	59
Streamline Path	3.43e5 mm^2	7.1e3 mm^2	98.0 %	17.2°	53.0 %	134
Double-sided Cutting	3.46e5 mm^2	4.1e3 mm^2	98.9 %	8.31°	75.6 %	113
No Constant Region(*)	3.43e5 mm^2	6.6e3 mm^2	98.1 %	4.21°	60.4 %	116
Inverted Fibre Angle(*)	3.46e5 mm^2	4.2e3 mm^2	98.8 %	8.38°	76.4 %	108
Filler Course(*)	3.44e5 mm^2	6.0e3 mm^2	98.3 %	8.34°	72.3 %	96

(*) indicate different fibre angle distributions.

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

The main goal of this thesis was to extend the Three-Step Approach for fibre steering to include generic doubly-curved open shells that are more representative of aerospace structures. The parametric approach is suitable for this objective, by representing the surface and tows as a set of NURBS/B-spline surfaces and curves, as described in this thesis. It has demonstrated surface and course generation, tow cutting through intersection detections and accurate data gathering through the use of the parametric surface.

Furthermore, some of the manufacturing defects that would arise from design flaws are taken into consideration. The minimum cut length of tows can be accounted for by evaluating the length of each tow segment, the curvature is registered to prevent areas of tow kinking and a cutting strategy was proposed to counter tow straightening for a subset of designs. Fitness criteria have been proposed to evaluate fibre-steered layers on coverage / gap percentage, fibre angle mismatch and continuity. Although these values have little relevance to conventional fibre layers, it provides an objective grading between fibre-steered layers.

Additionally, the following observations were made during the creation of the fibre path algorithms and results generation:

- Fibre straightening remains an issue for most AFP produced fibre steered layers without overlaps, for both Linearly Varying Fibre Angle (LVFA) and generic distributions. The anti-straightening cuts proposed in subsection 4.8.2 can only be applied when the fibre angle either only gradually increases or only gradually decreases from the point of symmetry outwards, and there is a limit to the amount of steering that can be applied before too many courses overlap and anti-straightening cuts are impeded.

- While generic fibre angle distributions are possible with the current setup, there is still much room to improve. The initial path generation by the streamline algorithm is mostly unsatisfactory, with the spacing between courses being a limiting factor as noted in subsection 6.1.2. This can somewhat be alleviated with the single-progenitor approach for LVFA layers (subsection 4.2.2), but obviously cannot be applied to generic fibre angle distributions. The course spacing is a critical parameter in obtaining good coverage in a layer, as small deviations can lead to gaps between courses of at most one full tow width.

  Furthermore, cutting has been a more difficult task for generic fibre angle distributions than for LVFA layers. Generic fibre angle layers lend themselves worse for anti-straightening cuts due to the lack of a consistent repeated course 'pattern'. Also, alternating cuts may induce higher fibre angle errors than needed, as was shown in subsection 6.1.2. A re-evaluation of the 'zipper' cuts may be required, depending on the impact of the fibre angle error.

- For LVFA layers, the only viable method of placing multiple courses on a layer is through course shifting. This results in gaps or overlaps in the areas where the fibre angle is steered. A better coverage can be achieved by implementing constant regions within LVFA as shown in subsection 4.2.3,

but further research is needed to prove the viability of this design choice. Constant regions can also be beneficial when good surface smoothness is required without gaps, such as for joining. However, as the steering would need to occur over a smaller distance, the expected fibre angle error would be higher.

- Several drivers for the coverage and fibre angle error parameters of fibre steered layers have been identified in subsection 6.2.9, but optimizing for one generally comes at the cost of the other. For example, higher coverage is achieved when the number of courses and the fraction of the length of the course along which the steering occurs are minimized. This results in fewer overlap areas that are shorter in length along the course. In contrast, lower fibre angle errors are achieved when the course width is smaller and when the steering is more gradual, which would result in more gaps. Additionally, proper course spacing is critical for good layer coverage.

7.2 Recommendations

- A form of iterative curve fitting could be implemented to improve the curvature predictions. As mentioned in section 4.10 and Appendix C, the current fitting algorithm has difficulties with modelling inflection points in the fibre path. This can be alleviated by performing knot insertions at the inflection points to lower the continuity there to C^0 (positional continuity), as this better reflects the shape of the curve.

  Inflection points could be identified by examining the angle along the fibre path. Additionally, for LVFA layers the approximate location of the inflections points may be known beforehand. It is important that the knots are inserted near the inflection point or it will have an adverse effect. This is where an iterative approach may be required to find the best fit.

- The initial course centre line generation could be improved upon by creating a replacement for the streamline code. The difficulties of the streamline code have been documented in subsection 4.2.2, along with the proposal of the single-progenitor work-around for LVFA layers. Unfortunately, this is limited to a subset of LVFA distributions, is not applicable to generic fibre angle distributions and may produce fibre angle errors on curved surfaces as mentioned in subsection 6.1.1. A new algorithm, with a strong emphasis on proper course spacing would be highly beneficial for the further development of fibre steering layers. A promising direction could be to research the integration of isogeometric analysis within the Three-Step Approach for a more holistic solution for fibre steered variable stiffness laminates.

- Development of a new cutting strategy that prioritizes the placement of continuous tows near the course centre line could lower the fibre angle error for generic fibre layers. As was shown in subsection 6.1.2, the alternating cutting of courses resulted in relatively high fibre angle errors. Some courses overlapped so much that only a few tows could be placed. This is partially due to course spacing, but also due to the fibre angle distribution itself, where . Tows near the course centre line are closer to the design angle and should therefore be prioritized over tows near the edges.

- There are types of surfaces that are not possible yet with the current methods, but would still be of interest to be included. In particular, bodies of revolution and multiple connected B-spline/NURBS surfaces seem like the most impactful additions.

  Bodies of revolution, such as cylinders, were not considered due to several reasons. Most importantly, exact cylinders in NURBS are not possible without repeated knots, thereby making concessions in continuity at the points where the control polygon and control points coincide. This could pose problems for a program that requires surface derivatives for most operations. Additionally, a work-around would need to be implemented if the current intersection algorithm is used. The intersection algorithm checks for convex hull overlaps of the control polygons of NURBS/B-spline curves as part of detecting possible intersections, which can be tricky when the intersection algorithm does not consider the surface to be revolving.

  Combining multiple NURBS/B-spline surfaces are of interest as well, because multiple surfaces may be required to describe more complex geometries. As NURBS/B-spline curves only exist within

the domain of the surface itself, a global interface would be required to connect multiple (local) curves to one global course. The main challenge would lie in performing operations such as course offsets when the global curve lies on multiple surfaces, as the continuity cannot be guaranteed on the surface border. In a way, this is a similar problem as the one encountered with bodies of revolution, as the cylinder could be considered to be multiple connected surfaces with only C^0 continuity at specific points. A solution for one could therefore perhaps also be applied to the other.

- A more general method for the anti-straightening cutting than the one described in section B.2 may be required to make the algorithm more broadly applicable. Currently, a work-around is used to decide if a segment is cut depending on the position of the intersection on the surface, in addition to the 'regular' anti-straightening cutting. This ensures that tows are never cut but only added inside the panel dimensions. While this works for (D)LVFA layers, a generic layer may have different configurations where it would not be applicable to.

Bibliography

[1] P. Hollinger. Airlines bid to beat their weight problem. https://www.ft.com/content/6ce66d16-bd6a-11e6-8b45-b8b81dd5d080/, Visited on 2022-06-20.

[2] J. Sloan. Atl and afp: Signs of evolution in machine process control. https://www.compositesworld.com/articles/atl-and-afp-signs-of-evolution-in-machine-process-control/, Visited on 2022-06-20.

[3] R. Jones. *Mechanics of Composite Materials*. Taylor and Francis, Inc., Philadelphia, PA, 1999.

[4] Z. Gürdal and R. Olmedo. In-plane response of laminates with spatially varying fiber orientations - variable stiffness concept. *AIAA Journal*, 31:4:751–758, 1993.

[5] A.W. Blom. *Structural Performance of fibre-placed, variable stiffness composite conical and cylindrical shells*. PhD thesis, Delft University of Technology, 2010.

[6] S. IJsselmuiden. *Optimal design of variable stiffness composite structures using lamination parameters*. PhD thesis, Delft University of Technology, 2011.

[7] D.M.J. Peeters. *Design Optimisation of Practical Variable Stiffness and thickness laminates*. PhD thesis, Delft University of Technology, 2017.

[8] J.M.J.F. van Campen. *Optimum Lay-up Design of Variable Stiffness Composite Structures*. PhD thesis, Delft University of Technology, 2011.

[9] K. Wurpel. Modeling manufacturable towpaths. Master's thesis, Delft University of Technology, 2015.

[10] K.C. Wu, B.F. Tatting, and B. Smith. Design and manufacturing of tow-steered composite shells using fibre placement. Palm Springs, California, 2009. 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural dynamics and Materials conference.

[11] S. Setoodeh, M.M. Abdalla, S. IJsselmuiden, and Z. Gürdal. Design of variable-stiffness composite panels for maximum buckling load. *Composite Structures*, 87(1):109–117, 2009.

[12] Z. Gürdal and R. Olmedo. Composite laminates with spatially varying fiber orientations: Variable stiffness panel concept. pages 798–808. 33rd Structures, Structural Dynamics and Materials Conference, 1992.

[13] S.S. Yau and T.W. Chou. Strength of woven-fabric composites with drilled and molded holes. *Composite materials testing and design, ASTM STP-972*, pages 423–437, 1988.

[14] B.F. Tatting and Z. Gürdal. Design and manufacture of elastically tailored tow placed plates. *NASA/CR 2002-211919*, pages 1–14, 2002.

[15] H. Akhavan, P. Ribeiro, and M.F.S.F. de Moura. Composite laminates with linearly varying fiber angles under static and dynamic loads. 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 2013.

[16] Z. Gürdal, B.F. Tatting, and K.C. Wu. Variable stiffness composite panels; effects of stiffness variation on the in-plane buckling response. *Composites Part A: Applied Science and Manufacturing*, 39(5):911–922, 2008.

[17] Z. Zabinsky. Global optimization for composite structural design. pages 1406–1412. 35th Structures, Structural Dynamics, and Materials Conference, 1994.

[18] D.B. Adams, L.T. Watson, Z. Gürdal, and C.M. Anderson-Cook. Genetic algorithm optimization and blending of composite laminates by locally reducing laminate thickness. *Advances in Engineering Software*, 35:35–43, 2004.

[19] J.M.J.F. van Campen, O. Seresta, M.M. Abdalla, and Z. Gürdal. General blending definitions for stacking sequence design of composite laminate structures. Schaumburg, IL, 2008. 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference.

[20] A. Brasington, C. Sacco, J. Halbritter, R. Wehbe, and R. Harik. Automated fiber placement: A review of history, current technologies, and future paths forward. *Composites Part C: Open Access*, 6:1–18, 2021.

[21] V. Mishra. A methodology to predict the stiffness properties and buckling load of manufacturable variable stiffness panels. Master's thesis, Delft University of Technology, 2017.

[22] K. Croft, L. Lessard, D. Pasini, M. Hojjati, J. Chen, and A. Yousefpour. Experimental study of the effect of automated fibre placement induced defects on performance of composite laminates. *Composites: Part A*, 42:5:484–491, 2011.

[23] Trelleborg. Tape laying. `https://www.trelleborg.com/en/seals/products-and-solutions/advanced-composites/automation-equipment/tape-laying`, Visited on 2022-06-20.

[24] TFP Technology GmbH. Tfp - tailored fibre placement, carbon-preforms and smart textiles. `https://www.tfp-tech.de/`, Visited on 2022-06-20.

[25] G. Gardiner. Tailored fiber placement: Besting metal in volume production. `https://www.compositesworld.com/articles/tailored-fiber-placement-besting-metal-in-volume-production/`, Visited on 2022-06-20.

[26] P.J. Crothers, K. Drechsler, D. Feltin, I. Herszberg, and T. Kruckenberg. Tailored fibre placement to minimise stress concentrations. *Composites Part A: Applied Science and Manufacturing*, 28-7: 619–625, 1997.

[27] B. Kim, K. Hazra, P. Weaver, and K. Potter. Limitations of fibre-placed techniques for variable angle tow composites and their process-induced effects. ICCM International Conferences on Composite Materials, 2011.

[28] B. Kim, K. Potter, and P. Weaver. Continuous tow shearing for manufacturing variable angle tow composites. *Composites Part A: Applied Science and Manufacturing*, 43(8):1347–1359, 2012.

[29] T. Dodwell, R. Butler, and A. Rhead. Optimum fiber steering of composite plates for buckling and manufacturability. *AIAA*, 54(3):1146–1149, 2016.

[30] D.M.J. Peeters, D. van Baalen, and M.M. Abdalla. Combining topology and lamination parameter optimisation. National Harbor, Maryland, 2014. 55th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference.

[31] D.M.J. Peeters, S. Hesse, and M.M. Abdalla. Stacking sequence optimisation of variable stiffness laminates with manufacturing constraints. *Composite Structures*, 125:596–604, 2015.

[32] A.W. Blom, M.M. Abdalla, and Z. Gürdal. Optimization of course locations in fiber-placed panels for general fiber angle distributions. *Composites Science and Technology*, 70:564–570, 2010.

[33] N.M. Patrikalakis, T. Maekawa, and W. Cho. Shape interrogation for computer aided design and manufacturing (hyperbook edition). `https://web.mit.edu/hyperbook/Patrikalakis-Maekawa-Cho/mathe.html`, 2009. Visited on 2022-06-20.

[34] G. Farin. A history of curves and surfaces in cagd. *Handbook of Computer Aided Geometric Design*, pages 1–21, 2002.

[35] L. Piegl and W. Tiller. *The NURBS Book Second Edition*. Springer-Verlag, Berlin, 1997.

[36] C. De Boor. *A Practical Guide to Splines*. Springer, New York, 1978.

[37] C.-K. Shene. Cs3621 introduction to computing with geometry notes. `https://pages.mtu.edu/~shene/COURSES/cs3621/NOTES/`, Visited on 2022-06-20.

[38] TouchDesigner. Touchdesign documentation - spline. `https://docs.derivative.ca/Spline`, Visited on 2022-06-20.

[39] C. De Boor. On calculating with b-splines. *Journal of Approximation Theory*, 6:50–62, 1972.

[40] M.G. Cox. The numerical evaluation of b-splines. *Journal of the institute for Mathematics Applications*, 10:134–149, 1972.

[41] E.W. Weisstein. Leibniz identity. `https://mathworld.wolfram.com/LeibnizIdentity.html`, Visited on 2022-06-20.

[42] M. Estratat and R. Greca. An interfacing module using configuration for declarative design of nurbs surfaces. 2019.

[43] Dassault Systèmes. Creating multi-sections surfaces. `http://catiadoc.free.fr/online/cfyug_C2/cfyugloft.htm`, Visited on 2022-06-20.

[44] E.W. Weisstein. Fundamental forms. `https://mathworld.wolfram.com/FundamentalForms.html`, Visited on 2022-06-20.

[45] E.W. Weisstein. Frenet formulas. `https://mathworld.wolfram.com/FrenetFormulas.html`, Visited on 2022-06-20.

[46] N.M. Patrikalakis and L. Bardis. Offsets of curves on rational b-spline surfaces. *Engineering with Computers*, 5:39–46, 1989.

[47] The MathWorks Inc. Solve nonstiff differential equations - medium order method. `https://www.mathworks.com/help/matlab/ref/ode45.html`, Visited on 2022-06-20.

[48] T.W. Sederberg and S.R. Parry. Comparison of three curve intersection algorithms. *Computer-Aided Design*, 18:58–63, 1986.

[49] T.W. Sederberg and T. Nishita. Curve intersection using bézier clipping. *Computer-Aided Design*, 22:9:538–549, 1990.

[50] J.M. Lane and R. Riesenfeld. A theoretical development for the computer generation and display of piecewise polynomial surfaces. *IEEE Trans, RAMI*, 2:35–46, 1980.

[51] P.A. Koparkar and S.P. Mudur. A new class of algorithms for the processing of parametric curves. *Computer-Aided Design*, 15(1):41–45, 1983.

[52] G. Wang. The subdivision method for finding the intersection between two bézier curves or surfaces. *Zheijang University Journal*, Special issue on computational geometry, 1984.

[53] K. Mørken, M. Reimers, and C. Schulz. Computing intersections of planar spline curves using knot insertion. *Computer Aided Geometric Design*, 26:3:351–366, 2009.

[54] K. Rajab and L.A. Piegl. A knowledge-guided approach to line nurbs curve intersection. *Computer-Aided Design and Applications*, 11:1:1–9, 2014.

[55] The GSL Team. Basis splines. `https://www.gnu.org/software/gsl/doc/html/bspline.html`, Visited on 2022-06-20.

[56] E.W. Weisstein. Triangle area. https://mathworld.wolfram.com/TriangleArea.html, Visited on 2022-06-20.

[57] The MathWorks Inc. Interpolation for 1-d, 2-d, 3-d, and n-d gridded data in ndgrid format. https://www.mathworks.com/help/matlab/ref/interpn.html, Visited on 2022-06-20.

[58] O. Falcó, C.S. Lopes, F. Naya, F. Sket, P. Maimi, and J.A. Mayugo. Modelling and simulation of tow-drop effects arising from the manufacturing of steered-fibre composites. *Composites Part A: Applied Science and Manufacturing*, 93:59–71, 2017.

[59] B.D. Marx P.H.C. Eilers. Flexible smoothing with b-splines and penalties. *Statist. Sci.*, 11(2): 89–121, 1996.

Appendix A

Mock-up Intersection and Segments Table

This appendix demonstrates the implementation of the cutting of tows in this thesis through the use of the intersection and segments table, according to the cutting strategy.

Figure A.1 shows a full course (blue and red) and a tow from the neighbouring course (dashed line) of a mock design. Every white dot is an intersection between the left (outer) edge of the tow and any part of the neighbouring course or the panel edge.

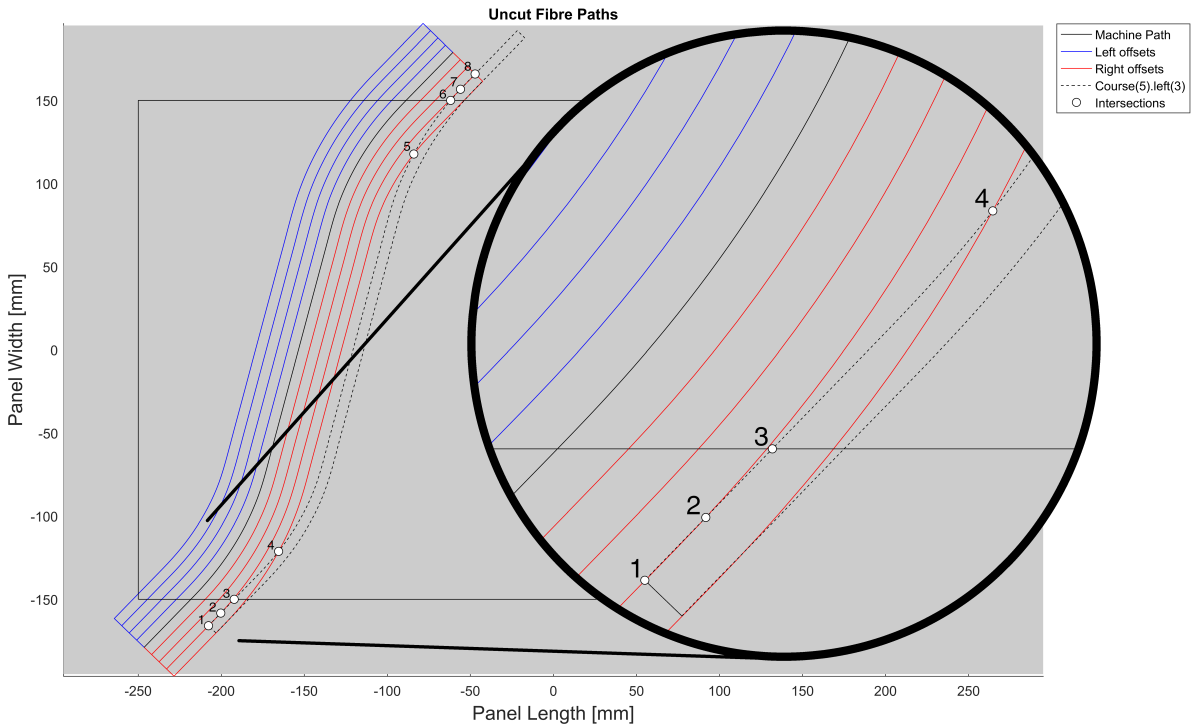


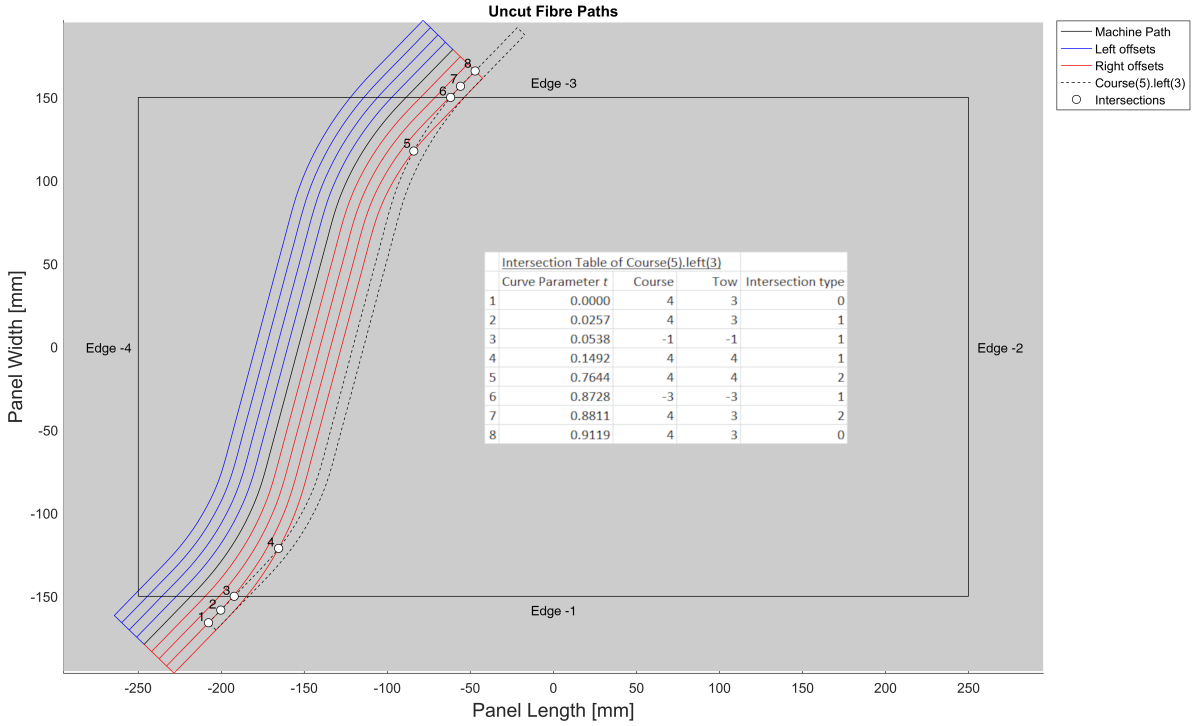
Figure A.1: The intersections of the tow with the panel edge and the neighbouring course. Intersections are only checked between the outer, nearest edges of the tows.

For the sake of this example, assume that the dashed line tow should be cut at any point where it overlaps with any part of the neighbouring course, or if crosses the panel edges. Visually, it can be seen that there is a segment between intersections #4 and #5 that does not overlap with any tows and therefore should still be placed, which should be deduced by the algorithm through the intersection table.

Figure A.2 shows the same course and tow, but with the intersection table of the tow and the

convention of the panel edges. Each intersection corresponds with the numbering in the table. The table columns display the following information:

- Curve parameter - location of intersection in curve parameter t [0,1].
- Course - the course or panel edge that the curve intersected with. A positive number represents a course, a negative number is one of the four panel edges. In this case, the full course is the fourth course of the mock design.
- Tow - the tow that the curve intersected with, if applicable. Tows are counted from the centre outwards, and a distinction is made between the tows on the left and the tows on the right of the machine path. For example, the rightmost tow of the course is named as Course(4).right(4).
- Intersection type – A number that acts as an extra identifier or as a counter. A zero is an 'Overlap' intersection, which means that while the lines did not intersect at that point, the tows are still overlapping. Any number larger than zero represents the number of times the tow has intersected with that particular tow or edge.



Intersection Table of Course(5).left(3)

	Curve Parameter t	Course	Tow	Intersection type
1	0.0000	4	3	0
2	0.0257	4	3	1
3	0.0538	-1	-1	1
4	0.1492	4	4	1
5	0.7644	4	4	2
6	0.8728	-3	-3	1
7	0.8811	4	3	2
8	0.9119	4	3	0

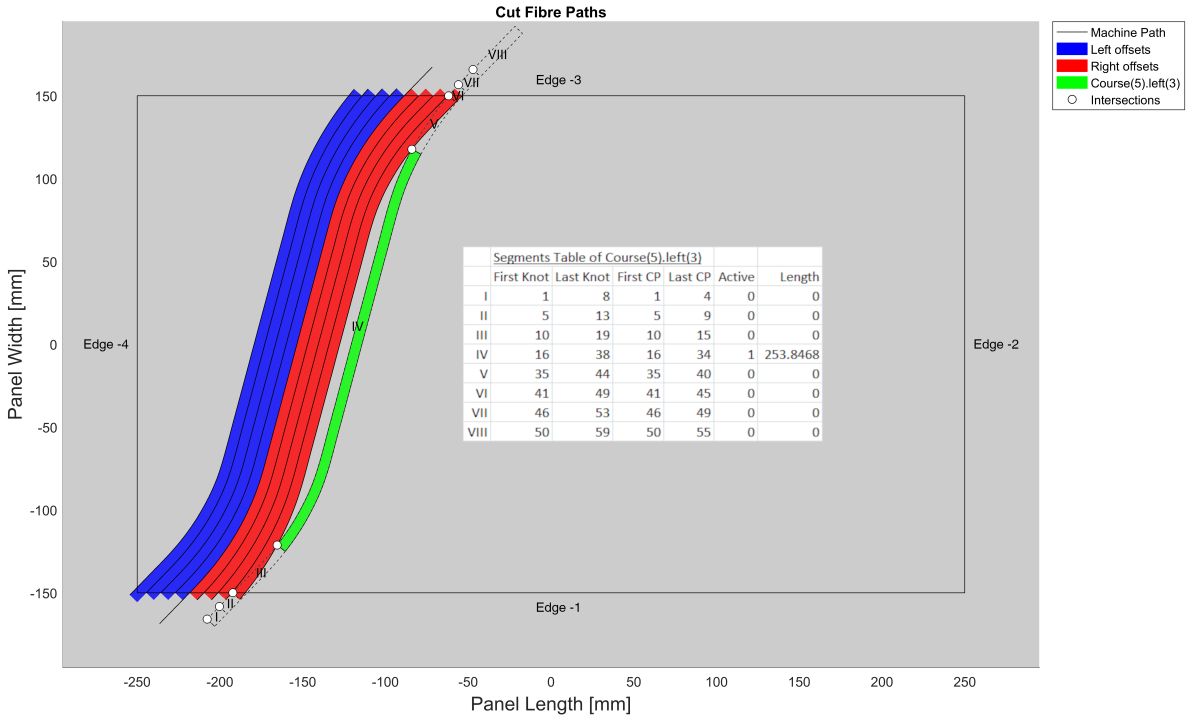
Figure A.2: The intersection table for the tow and their locations.

The registrations in the intersection table are used for two applications:

1. Tow, course and intersection type are registered to identify when a tow can be restarted, i.e. outside the bounds of the outer tow of Course 4. In this case, the first identified intersection (intersection #1) was with tow 3 of Course 4. This means that at the beginning of the dashed line, the tow cannot be placed yet. Only when it intersects with the outer tow (intersection #4) will it be free to be placed. Then, when it intersects with the outer tow again (intersection #5, the intersection 'type' counter is increased by one), it is again terminated.
2. Panel edge intersections are used to trim the panel to the minimum length to cover the design area, if not cut already by other tows.

Intersections #2 to #8 divide the curve into eight segments. For each segment, the algorithm should register if it is placed in the final design. This is done in the segments table, shown in Figure A.3. The table columns display the following information:

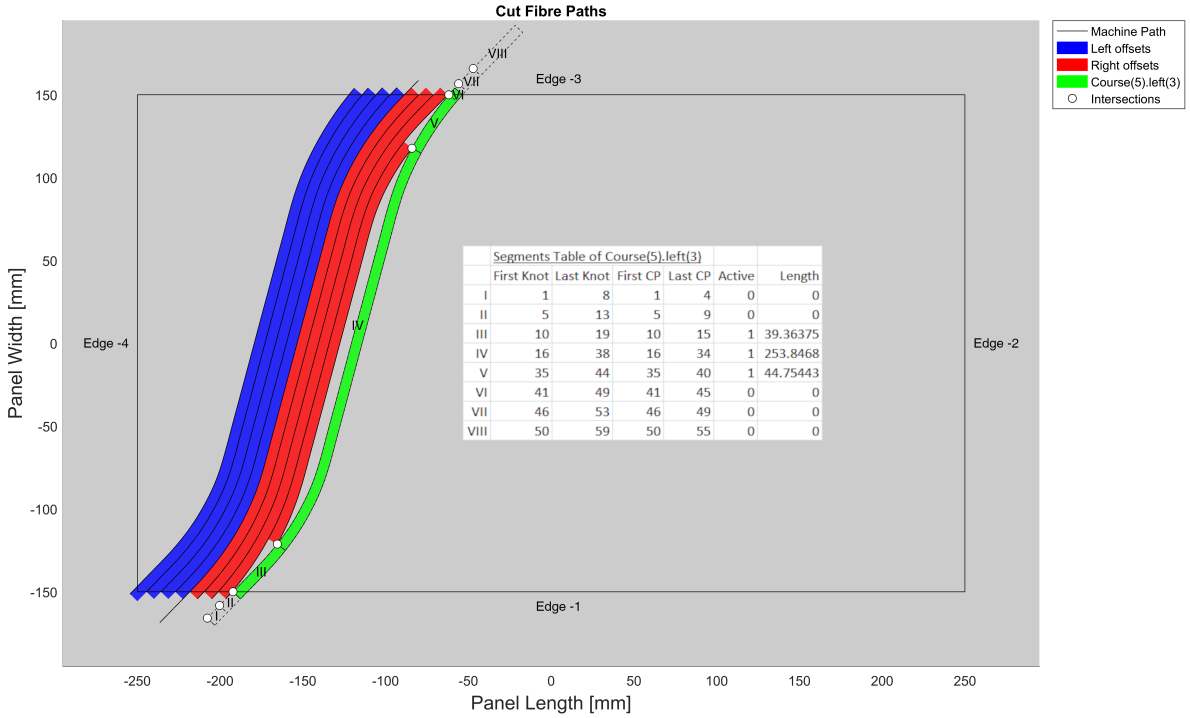
- First Knot - First index of the knot vector for this segment
- Last Knot - Last index of the knot vector for this segment
- First CP - First index of the control point vector
- Last CP - Last index of the control point vector
- Active - A boolean that determines if a tow segment is placed
- Length - Length of the segment. The sum of adjacent non-zero segments should be larger than the minimum cut length.



	First Knot	Last Knot	First CP	Last CP	Active	Length
I	1	8	1	4	0	0
II	5	13	5	9	0	0
III	10	19	10	15	0	0
IV	16	38	16	34	1	253.8468
V	35	44	35	40	0	0
VI	41	49	41	45	0	0
VII	46	53	46	49	0	0
VIII	50	59	50	55	0	0

Figure A.3: Segments Table and active segments of the tow.

It can be seen that indices in the knot and control point vectors are shared between segments. The vectors have been modified through knot insertions at the intersection locations such that the segments are independent of each other. By using this method, only one knot vector and one control point vector are required for all segments. Moreover, this setup allows for different cutting strategies to be used while using the same dataset. For example, if the course was to be cut instead of the tow, one would only need to alter the 'Active' column in the segment tables of the tows according to the cutting strategy. This is illustrated in Figure A.4. Here, segments III and V of the tow are also placed, and it is only cut by the panel edges.



Segments Table of Course(5).left(3)

	First Knot	Last Knot	First CP	Last CP	Active	Length
I	1	8	1	4	0	0
II	5	13	5	9	0	0
III	10	19	10	15	1	39.36375
IV	16	38	16	34	1	253.8468
V	35	44	35	40	1	44.75443
VI	41	49	41	45	0	0
VII	46	53	46	49	0	0
VIII	50	59	50	55	0	0

Figure A.4: Segments Table and active segments of the tow, course cut instead of tow

Appendix B

Cutting Algorithms

B.1 Alternating Course Cutting

Alternating course cutting is applied to either the odd or even course numbers, cutting them at any point when they overlap with any part of the neighbouring courses. As such, only segments outside the boundaries of the outermost tow can be placed. The flow diagram of the algorithm is shown in Figure B.1.

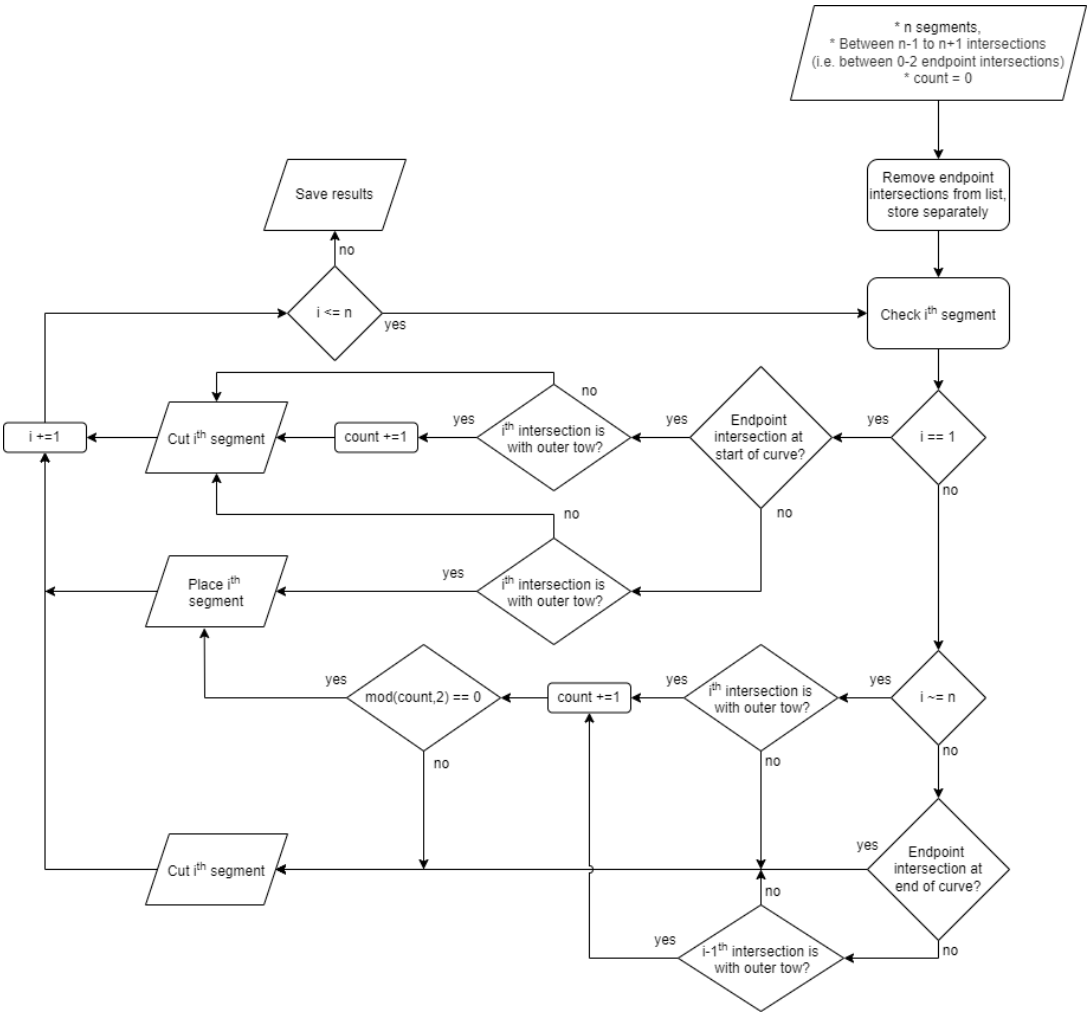


Figure B.1: Flow diagram describing the alternating course cutting logic.

To determine if the tow is free to be placed, the intersection table is used. The intersection and segment tables are explained in Appendix A. Endpoint intersections, i.e. intersections at the start- or endpoint of the curve, are temporarily removed from the intersection table to ensure that there are always $n-1$ intersections for n segments. This way, the i^{th} intersection can be used to determine placement of the i^{th} segment.

The first intersection ($i == 1$) and the endpoint intersection (or lack thereof) are used to deduce the initial position and direction of the tow. If there is an endpoint intersection, or if the first intersection is not with the outermost tow, then the first segment is cut, as the tow started on the inside of a neighbouring course. If first intersection is with the outer tow and there is no endpoint intersection, then it can be assumed that the tow started outside of any boundaries and the first segment may be placed.

From this point onwards, a count is kept of how many times a tow intersects with the outermost tow. The count can be used to alternatingly place or cut a segment. This is continued until the last segment ($i == n$), as there are only $n-1$ intersections. Here instead, the last intersection is examined in addition to the potential endpoint intersection to determine if the final segment can be placed.

B.2 Anti-Straightening Cutting

Anti-straightening cutting can be divided into two types: 'regular' anti-straightening cutting, and 'filler course' anti-straightening, as shown in Figure 4.14. This section is about the first type, since the latter is performed by using the alternating course cutting strategy.

The anti-straightening cutting logic, shown in Figure B.2, is very similar to the previous section, with regions of the course being cut whenever they overlap with any part of neighbouring courses. The main difference lies in the fact that all courses are 'cut' as opposed to only half of them, and that the cutting only occurs near the start of the course, as is illustrated in Figure 4.13. This ensures that the tows are only ever placed/added within the layer dimensions, which prevents tow straightening in the design area.

Since the anti-straightening cuts imply the direction of course by design, the direction in which the segments are checked should account for that; the intersection and segment tables are flipped depending on the course direction. Additionally, a breakpoint is added in the cutting logic, which checks the position of the i^{th} intersection on the surface against some value (in this example the half-way point), after which every remaining segment is placed. The addition of the breakpoint is a work-around at best, as it only works reliably for neatly organized (Discrete) Linearly Varying Fibre Angle layers. A more widely applicable anti-straightening cutting logic algorithm may need to be developed if it is to be applied for generic fibre angle distributions.

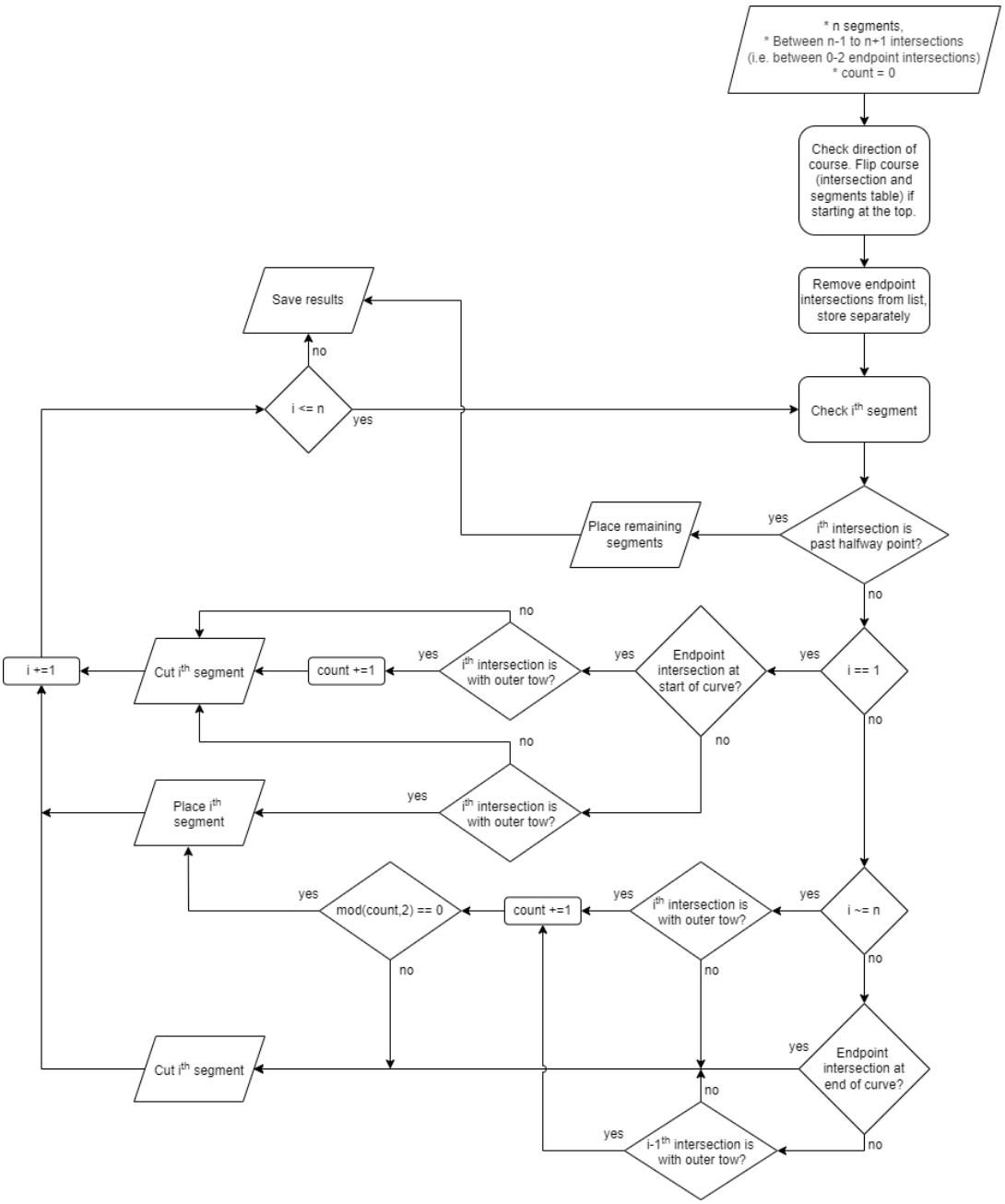


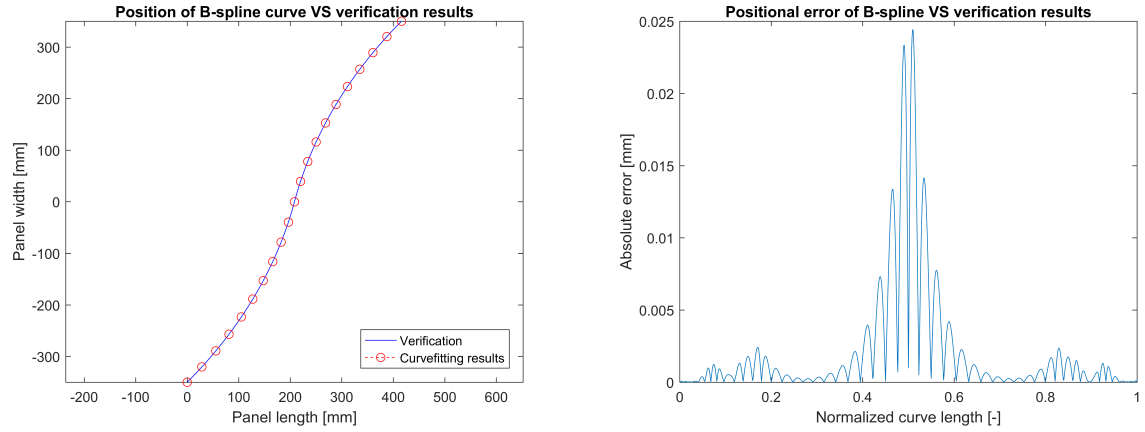
Figure B.2: Flow diagram describing the anti-straightening course cutting logic.

Appendix C

Curve Fitting LVFA Designs

This appendix was made to report the findings of fitting Linearly Varying Fibre Angle (LVFA) designs by the curve fitting algorithm. Two recommendations are made for future work to address the problems that were encountered.

As part of the verification of the curvature code, comparisons were made between the results from the code to a known result from an external MATLAB module. Both methods used the same dataset, consisting of a set of points (x,y,z) prescribing a typical LVFA fibre path. The MATLAB module converted the values to curvatures directly, whereas the curvature code used the results from the curve fitting algorithm from section 4.4.

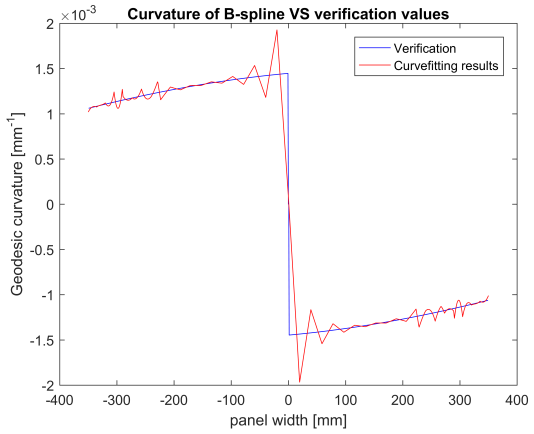


(a) Position graph

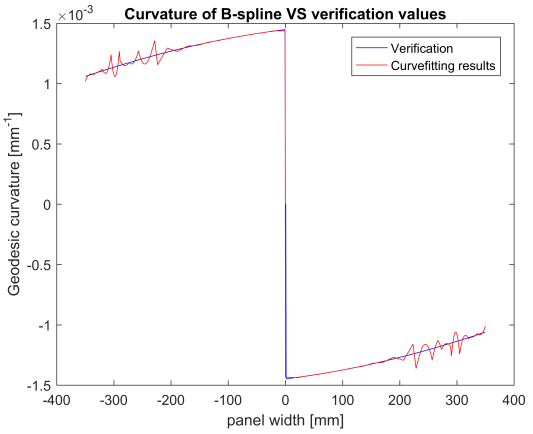
(b) Error graph

Figure C.1: Position and Error graphs of the B-spline fit

Although the curve fitting algorithm reproduced the curve quite well in terms of position as shown in Figure C.1, the curvature results were unsatisfactory. Oscillations were found around the discontinuity in the curvature graph, which should not be present according to the verification results, as seen in Figure C.2a. The discontinuity is a result of the change of direction (or inflection point) of the fibre angle at the halfway point, which is an inherent feature of LVFA paths. The oscillations were deemed problematic because it limited how well the curvature could be predicted. In turn, this would limit the ability of the algorithm to pre-emptively filter bad designs with excessive curvature.



(a) Oscillations in the curvature plot due to the curve fitting algorithm



(b) Oscillations at the discontinuity have disappeared due to knot additions.

Figure C.2: Curvature graphs of the B-spline fit. Figure C.2b has extra knots at the location of the discontinuity, resulting in a better fit.

Further testing revealed that the curvature code produced identical results to the external module when the curve fitting results were used as inputs for both. This led to the conclusion that the curve fitting algorithm could not fit the current curve well enough to obtain acceptable curvature values. Furthermore, it was noted that the oscillations disappeared when the fibre path had no discontinuities, which suggested that the discontinuity induced the issues in the curve fitting algorithm.

Wurpel (2015) also described difficulties with the curvature results, for which he decided to use P-splines [9]. P-splines are "Penalized B-splines", which includes a penalty function in addition to the regular B-spline expression to induce smoothness to the curve [59]. However, the inclusion of the penalty function in the current curve fitting algorithm had adverse effects with regards to the positional fit of the curve, and the concept was therefore discarded.

Instead, the solution lay in lowering the continuity of the B-spline curve at the location of the inflection point / curvature discontinuity to C^0 (positional continuity), as this more accurately represents the geometry of the curve. This was done by adding extra knots to the knot vector when constructing the B-spline in the curve fitting algorithm from Equations (4.17) to (4.19) at the location of the inflection point. These knots are a single repeated value such that only the positional continuity is preserved. The value of the knots is determined by taking the normalized arc length at the estimated location of the inflection point. The resulting fit is shown in Figure C.2b.

It is paramount that the knot additions are close to the inflection point, or the effect of the knot additions is adverse to the fit. This can be difficult in the case of a generalized result, such as from the streamline code due to the discrete nature of the finite element model. For this application, preferably an iterative method is used for the curve fitting, where the locations of knot additions are varied and then compared for fitness.

On the other hand, this has been one of many difficulties experienced from the streamline code and the finite element method approach. The problems from sections 4.2.2 and 4.2.3 could probably be avoided all together if a different approach was used to generate the fibre paths. One such method would be to use an isogeometric analysis approach, which integrates the finite element analysis into the desired B-spline/NURBS format more smoothly. This would probably negate the need for adaptive fitting, as the inflection points are easier to determine on continuous curves.

However, both options were decided to be outside the scope of the current thesis. Therefore, two recommendations will be made regarding this topic. The implementation of an adaptive curve fitting method will be included as a recommendation for future work if one decided to continue with the streamline algorithm. Secondly, a suggestion is made to redo the streamline code by isogeometric analysis to better suit the B-spline design of the rest of the thesis.